Acknowledgment of Support

The International Zurich Seminar on Communications (IZS) is organized jointly by the IEEE Switzerland Chapter on Digital Communication Systems and the Department of Information Technology and Electrical Engineering of ETH Zurich.
Conference Organization

General Co-Chairs
Helmut Bölcskei and Amos Lapidoth

Technical Program Committee
Ezio Biglieri Nikolai Nefedov
Pierre Chevillat Bernhard Plattner
Bernard Fleury Igal Sason
Martin Hänggi Jossy Sayir
Walter Hirt Giorgio Taricco
Gerhard Kramer Emre Telatar
Frank Kschischang Emanuele Viterbo
Hans-Andrea Loeliger Pascal Vontobel
Stefan Mangold Michèle Wigger
Thomas Mittelholzer Armin Wittneben
Stefan Moser Raymond Yeung

Organizers of Invited Sessions
Emmanuel Abbe Emanuele Viterbo
Uri Erez Michèle Wigger
Ioannis Kontoyiannis

Local Organization
Conference Secretaries Web and Publications
Rita Hildebrand Michael Lerjen
Silvia Tempel
Table of Contents

Keynote Talks

Wed 09:00 – 10:00
Gerhard Kramer, TU München
Progress on Relaying and Noisy Network Coding

Thu 09:00 – 10:00
Peter Winzer, Alcatel-Lucent
Spatial Multiplexing and MIMO for Fiber Capacities beyond 100 Tb/s

Fri 09:00 – 10:00
Mitchell Trott, HP Labs
Why is Wyner-Ziv coding for video so difficult?

Session 1
Invited session organizer: Michèle Wigger

*The Multiple Access Channel with Correlated Sources and Cribbing Encoders… 8
E. Amir and Y. Steinberg

*The Source Coding Side of Secrecy......................................................... 11
P. Cuff

*A Comparative Study of Cooperation for the Two-User Interference Channel . 12
D. Tuninetti

*Refinement of the Random Coding Bound . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15
Y. Altuğ and A. B. Wagner

*Source Coding with Side-Information at the Receivers and an Application . . 19
M. Wigger

Session 2: Shannon Theory
Chaired by Michael Gastpar

On Gaussian Channels with MMSE Interference . . . . . . . . . . . . . . . . . . . . . . . . 20
R. Bustin and S. Shamai (Shitz)

On the Gaussian Listening-Helper Source-Coding Problem . . . . . . . . . . . . . . . . 24
S. I. Bross and H. Permuter

*Invited papers are marked by an asterisk.
Upper-Bounding Rate Distortion Functions based on the Minimum Mean-Square Error ................................. 28
M. Dörpinghaus, M. Rüngeler, and R. Mathar

On the Information Loss in Memoryless Systems: The Multivariate Case ........ 32
B. C. Geiger and G. Kubin

One-Bit Quantizers for Fading Channels .................................................. 36
T. Koch and A. Lapidoth

Session 3  Wed 15:30 – 16:50
Invited session organizer: Ioannis Kontoyiannis

*Model Switching and other Unconventional Fun with Experts ................. 40
S. de Rooij

*Some Old and New Relations between Information and Estimation ............ 43
K. Venkat and T. Weissman

*Polar Martingales of Maximal Spread .................................................. 44
E. Abbe

*Sumset Differential Entropy Bounds .................................................... 48
I. Kontoyiannis

Session 4  Thu 10:20 – 12:00
Invited session organizer: Emmanuel Abbe

*Latest and Greatest on Spatially Coupled Codes
R. Urbanke

*Coding Theory and Compressed Sensing through Convex Relaxations .......... 49
A. Saber Tehrani and A. G. Dimakis

*Hybrid Coding
Y.-H. Kim

*Polar Coding
E. Telatar

*Competitive Classification
A. Orlitsky

Session 5: Coding I  Thu 13:20 – 15:00
Chaired by Hans-Andrea Loeliger

Polar Codes for the Deterministic Broadcast Channel ............................. 51
N. Goela, E. Abbe, and M. Gastpar
Session 6: Coding II
Thu 15:30 – 17:10
Chaired by Alex Grant

The Relation between Block Length and Reliability for a Cascade of AWGN Links ................................................................. 71
R. Subramanian

Secure Multiplex Coding over Interference Channel with Confidential Messages ................................................................. 75
X. Li and R. Matsumoto

Polymer Expansions for Cycle LDPC Codes ........................................ 79
N. Macris and M. Vuffray

Factoring Linear Trellises ................................................................. 83
D. Conti and N. Boston

Information Theoretic Authentication and Secrecy Codes in the Splitting Model . . . 87
M. Huber

Session 7
Fri 10:20 – 12:00
Invited session organizer: Uri Erez

*Anti-Structure Problems ................................................................. 91
R. Zamir

*Real-Time Coding of Markov Sources over Erasure Channels: When is Binning Optimal? ............................................................. 95
A. Khisti, F. Etezadi, and M. Trott

*Spinal Codes
D. Shah

*SuperNyquist Rateless Coding for Intersymbol Interference Channels .......... 99
U. Erez and G. W. Wornell

*Successive Compute-and-Forward ................................................... 103
B. Nazer
Session 8  Fri 13:20 – 15:00
Invited session organizer: Emanuele Viterbo

*What are Good Coding Schemes for Multicast in Heterogeneous Wireless Networks?
E. Soljanin

*Tunable Sparse Network Coding ......................................................... 107
S. Feizi, D. E. Lucani, and M. Médard

*Coding for Wireless Communications
M. Bossert

*The Flatness Factor in Lattice Network Coding: Design Criterion and Decoding Algorithm ......................................................... 111
J.-C. Belfiore and C. Ling

*Wideband non-coherent Wireless Communications
E. Viterbo

Session 9: Communication Theory  Fri 15:30 – 17:10
Chaired by Tobias Koch

Analysis of the Stochastic Channel Model by Saleh & Valenzuela via the Theory of Point Processes ................................................................. 115
M. L. Jakobsen, T. Pedersen, and B. H. Fleury

A. Schenk and R. F. H. Fischer

G. Levin and S. Loyka

Complexity Analysis for ML-based Sphere Decoder Achieving a Vanishing Performance-Gap to Brute-Force ML Decoding ............................. 127
A. Singh, P. Elia, and J. Jaldén

Symbol-by-Symbol APP Decoding based on Supercode Decoding .................. 131
J. Freudenberger and U. Kaiser
The Multiple Access Channel with Correlated Sources and Cribbing Encoders

Eliron Amir and Yossef Steinberg
Dept. of EE, Technion, Israel

I. NOTATION

A discrete memoryless multiple access channel (MAC) is a quadruple \( \{X_1, X_2, P_{Y|X_1, X_2}, Y\} \) where \( X_1, X_2 \) are the input alphabets, \( Y \) the output alphabet, and \( P_{Y|X_1, X_2} \) a transition probability matrix from \( X_1 \times X_2 \) to \( Y \). For shorthand notation, we will refer to the MAC by \( P_{Y|X_1, X_2} \). A source pair is a triplet \( \{U, V, P_{Y|U, V}\} \) where \( P_{Y|U, V} \) is a distribution on \( U \times V \). We use the following notation: random variables are denoted by uppercase letters. Sequence of \( n \) letters from alphabet \( X \) is denoted by \( x^n \), and the substring \( x_i, x_{i+1}, \ldots, x_j \) as \( x^j_i \). When the dimension \( n \) is understood from the context, \( n \)-vectors are denoted by boldface letters: \( x^n = x \). Similar notation holds for random variables and random vectors, e.g., \( V, U, U^n, \) etc. The channel and source are assumed memoryless, thus probabilities of \( n \)-sequences are given by

\[
P_{Y|X_1, X_2}(y|x_1, x_2) = \prod_{i=1}^{n} P_{Y|X_1, X_2}(y_i|x_1, x_2, i)
\]

\[
P_{U|V}(u, v) = \prod_{i=1}^{n} P_{U|V}(u_i, v_i)
\]

II. PROBLEM DEFINITION AND PREVIOUS RESULTS

We investigate a joint source-channel coding schemes for the discrete memoryless MAC with cribbing, the sources, \( U \) and \( V \), deliver their output to two separate encoders 1 and 2. An \( (n, m, \epsilon) \) joint source-channel code for the source \( P_{U|V} \) and the MAC \( P_{Y|X_1, X_2} \) with strictly causal cribbing encoder consists of \( m + 1 \) encoding functions

\[
f_1 : U^n \to X_1^n \tag{1a}
\]

\[
f_{2,i} : V^n \times X_1^{i-1} \to X_{2,i} \quad i = 1, \ldots, m \tag{1b}
\]

and a decoding function

\[
d^m : Y^n \to U^n \times V^n
\]

such that the probability of error is bounded from above by \( \epsilon \):

\[
P_e = Pr\left\{ (u^n, v^n) \neq d^m(y^n) \right\} = \sum_{(u, v) \in U^n \times V^n} Pr\left\{ (u, v) \neq (u^n, v^n) \right\} \leq \epsilon
\]

We say that encoder 2 is the cribbing encoder, as reflected in (1b). The rate of the code is defined as \( \rho = n/m \). A source pair \( (U, V) \) is said to be transmissible via the MAC \( P_{Y|X_1, X_2} \) with strictly causal cribbing at rate \( \rho \) if for every \( \epsilon > 0, \delta > 0 \), and sufficiently large \( n \) there exists an \( (n, n/(\rho - \delta), \epsilon) \) code for the source pair and the channel.

We will examine also the model of causal cribbing, where the cribbing encoder sees the output of the other encoder without the delay of (1b), that is, when the sequence of \( m \) encoders at channel input 2 is of the form

\[
f_{2,i} : V^n \times X_1^{i-1} \to X_{2,i} \quad i = 1, \ldots, m
\]

The definition of transmissibility of a source pair via the channel remains as in the case of strictly causal cribbing, with the only difference that (2) replaces (1b). The joint probability mass function for strictly causal cribbing is given by

\[
P(u, v, x_1, x_2, y) = \prod_{i=1}^{n} P_{U_i, V_i}(u_i, v_i) \prod_{i=1}^{n} P_{Y_i|X_1, X_2}(y_i|x_{1i}, (u^n)_{\neq i}, x_{2i}, (v^n)_{\neq i})
\]

and for causal cribbing by

\[
P(u, v, x_1, x_2, y) = \prod_{i=1}^{n} P_{U_i, V_i}(u_i, v_i) \prod_{i=1}^{n} P_{Y_i|X_1, X_2}(y_i|x_{1i}, (u^n)_{\neq i}, x_{2i}, (v^n)_{\neq i})
\]

We are interested in finding conditions for transmissibility of a source via the MAC with causal and strictly causal cribbing. We start by stating previous relevant results, on capacity regions of MAC with cribbing, and on distributed source coding.

**Theorem 1.** F. Willsens & E. C. van der Meulen [4]: The capacity region for the strictly causal cribbing communication model is the collection of all pairs \((R_1, R_2)\) such that

\[
R_1 \leq H(X_1 | W) \tag{3}
\]

\[
R_2 \leq I(X_2; Y | X_1, W) \tag{4}
\]

\[
R_1 + R_2 \leq I(X_1, X_2; Y) \tag{5}
\]

for some distribution of the form

\[
P_{W,X_1,X_2,Y}(w, x_1, x_2, y) =
\]

\[
P_W(w) P_{X_1|W}(x_1 | w) \cdot P_{X_2|W}(x_2 | w) P_{Y|X_1, X_2}(y | x_1, x_2) \tag{6}
\]

and \( |W| \leq \min \{|X_1| \cdot |X_2| + 1, |Y| + 2\} \)

The capacity region for the causal cribbing communication model is given by the collection of all rate pairs \((R_1, R_2)\) such that

\[
R_1 \leq H(X_1) \tag{7}
\]

\[
R_1 \leq I(X_2; Y | X_1) \tag{8}
\]

\[
R_1 + R_2 \leq I(X_1, X_2; Y) \tag{9}
\]

This research was supported by the ISAREL SCIENCE FOUNDATION (grant No. 684/11).
for
\[
\begin{align*}
& P_{x_1,x_2,y}(x_1,x_2,y) = \\
& P_{x_1,x_2}(x_1,x_2) P_{y|x_1,x_2}(y|x_1,x_2)
\end{align*}
\] (10)

**Theorem 2.** Slepian-Wolf [3]: For the distributed source coding problem with source \((U,V)\) drawn i.i.d \(\sim P_{U,V}\), the achievable rate region is given by:

\[
\begin{align*}
& H(U|V) \leq R_1 \\
& H(V|U) \leq R_2 \\
& H(U,V) \leq R_1 + R_2
\end{align*}
\] (11) (12) (13)

**III. STATEMENT OF RESULTS**

**Theorem 3.** A source \((U,V) \sim P_{U,V}\) is transmissible via the MAC \(P_{Y|X_1,X_2}\) with strictly causal cribbing at rate \(\rho \geq 0\) if

\[
\begin{align*}
& H(U|V) \leq H(X_1 | W,V) \\
& H(V|U) \leq I(Y;X_2 | W,U,X_1) \\
& H(U,V) \leq I(Y;X_1,X_2 | U)
\end{align*}
\] (14) (15) (16)

for some

\[
\begin{align*}
P_{U,V|X_1,X_2,y}(w,u,v,x_1,x_2,y) &= P_{W}(w) P_{U,V}(u,v) P_{X_1|W,U}(x_1 | w,u) \cdot \\
P_{X_2|W,V}(x_2 | w,v) P_{Y|X_1,X_2}(y | x_1,x_2)
\end{align*}
\] (17) (18) (19)

where \(P_{W}(\omega) = P_{U}(\omega)\) and \(W = U\).

The proof of Theorem 3 is omitted due to space considerations.

**Theorem 4.** A source \((U,V) \sim P_{U,V}\) is transmissible via the MAC \(P_{Y|X_1,X_2}\) with causal cribbing at rate \(\rho \geq 0\) if and only if there exists a joint input distribution \(P_{X_1,X_2}\) such that:

\[
\begin{align*}
& \rho H(U|V) \leq H(X_1) \\
& \rho H(V|U) \leq I(Y;X_2 | X_1) \\
& \rho H(U,V) \leq I(Y;X_1,X_2)
\end{align*}
\] (20) (21) (22)

The proof of the converse part of Theorem 4 is omitted. Two proofs of the achievability part of Theorem 4, based on Theorem 3 and based on a separation argument, are given in Section V.

Further results on transmission with distortion, and different models of cribbing, are reported elsewhere.

**IV. DISCUSSION**

As shown in Section V, Theorem 4 implies that a source \(P_{U,V}\) is transmissible via the MAC \(P_{Y|X_1,X_2}\) with causal cribbing at rate \(\rho \geq 0\) if and only if the Slepian-Wolf region scaled by \(\rho\) intersects the capacity region of the MAC with causal cribbing. This implies that with causal cribbing, a separation strategy, where we use a Slepian-Wolf code to compress the source and a channel code to transmit the source codewords via the MAC, is optimal. In view of this, let us examine the achievability result for strictly causal cribbing. By Theorem 3, the following inequalities are sufficient conditions for the source \(P_{U,V}\) to be transmissible via the channel \(P_{Y|X_1,X_2}\) with strictly causal cribbing at rate \(\rho = 1\):

\[
\begin{align*}
& H(U|V) \leq H(X_1 | W) \\
& H(V|U) \leq I(Y;X_2 | W,X_1) \\
& H(U,V) \leq I(Y;X_1,X_2)
\end{align*}
\] (23) (24) (25)

for some

\[
\begin{align*}
P_{W,U,V,X_1,X_2,Y}(w,u,v,x_1,x_2,y) &= P_{W}(w) P_{U,V}(u,v) P_{X_1|W}(x_1 | w) \cdot \\
P_{X_2|W,V}(x_2 | w,v) P_{Y|X_1,X_2}(y | x_1,x_2)
\end{align*}
\]

where \(P_{W}(\omega) = P_{U}(\omega)\) and \(U = W\). Note that the right hand sides of (20)-(22) do not depend on the source, and coincide with the right hand sides of (3)-(5). The left hand sides of (20)-(22) coincide with the left hand sides of (11)-(13). This observation may suggest that if \(P_{U,V}\) and \(P_{Y|X_1,X_2}\) satisfy (20)-(22), then the source can be transmitted via the channel with a separation strategy, namely, by concatenating a Slepian-Wolf code that compresses the source, with a channel code for the MAC with strictly causal cribbing. Unfortunately, this is not the case - conditions (20)-(22) do not imply separation. This is demonstrated in the following example.

**Example 1.** Consider the joint source \((U,V)\) where \(U \sim \text{Bernoulli}(1/2), V = U\), and the binary MAC given by:

\[
P_{Y|X_1,X_2}(y|x_1,x_2) = \begin{cases} 1 & \text{if } x_1 = x_2 \\ \text{Bernoulli}(1/2) & \text{otherwise} \end{cases}
\]

where \(1[\cdot]\) is the indicator function. Thus, if \(x_1 = x_2\) the inputs are connected directly to the output. If \(x_1 \neq x_2\) the output is uniformly distributed on \([0,1]\), independently of the inputs.

The Slepian-Wolf region for this source is given by

\[
R_1 + R_2 \geq 1
\]

(24)

We claim that the capacity region of this MAC with strictly causal cribbing does not contain the line \(R_1 + R_2 = 1\). Indeed, by (5)

\[
R_1 + R_2 \leq I(X_1,X_2;Y) \leq 1 - H(Y|X_1,X_2) \leq 1
\]

where the last inequality holds with equality only if \(Y\) is a deterministic function of \(X_1\) and \(X_2\). Since by (6) we have the Markov relations \(X_1 \rightarrow W \rightarrow X_2\), \(H(Y|X_1,X_2) = 0\) only if \(X_1\) and \(X_2\) are determined by \(W\). But this, in turn, implies that \(R_1 = R_2 = 0\), by (3) and (4). Therefore the capacity region of this MAC with strictly causal cribbing does not contain the line \(R_1 + R_2 = 1\).

It is easy to verify that the source \((U,V)\) is transmissible via the channel at rate \(\rho = 1\). We can employ single letter code: connect \(U\) and \(V\) directly to \(X_1\) and \(X_2\), respectively. Moreover, transmissibility can be verified directly from conditions (20)-(22): the conditional entropies on the left hand side there are 0, so we can choose \(P_W = P_U\), \(X_1 = X_2 = W\), and the transmissibility conditions are satisfied. Hence, (20)-(22) (and therefore also Theorem 3) predict that the source can be transmitted via the MAC with strictly causal cribbing, yet it cannot be transmitted by a separation strategy.

**V. PROOF OF THE ACHIEVABILITY PART OF THEOREM 4**

**Proof:** The achievability part of Theorem 4 can be proved based on Theorem 3, or based on a separation argument. We first give a proof based on separation.

Given three nonempty intervals \(I_1 = [a,b], I_2 = [c,d]\), and \(I_3 = [e,f]\), a necessary and sufficient condition for the existence of a pair \((R_1, R_2)\) such that \(R_1 \in I_1, R_2 \in I_2, \) and \(R_1 + R_2 \in I_3\), is that the endpoints satisfy

\[
a + c \leq f \quad \text{and} \quad b + d \geq e
\]

(25)
If the source $P_{U,V}$ and the channel and input distribution $P_{Y|X_1,X_2}$, $P_{X_1,X_2}$ satisfy (17), (18), and (19), then necessarily the intervals

\[ I_1 = [\rho H(U|V), H(X_1)] \quad (26) \]
\[ I_2 = [\rho H(V|U), I(Y;X_2|X_1)] \quad (27) \]
\[ I_3 = [\rho H(U,V), I(Y;X_1,X_2)] \quad (28) \]

are nonempty. By basic properties of information functions we have

\[ \rho H(U|V) + \rho H(V|U) \leq \rho H(U, V) \]
\[ \leq I(X_1;X_2;Y) \]
\[ = I(X_1;Y) + I(X_2;Y|X_1) \]
\[ \leq H(X_1) + I(X_2;Y|X_1) \quad (29) \]

where the second inequality above holds since $I_3$ is nonempty. It is easy to see that (29) implies (25). Therefore, we can find a pair $(R_1, R_2)$ satisfying

\[ \rho H(U|V) \leq R_1 \leq H(X_1) \quad (30) \]
\[ \rho H(V|U) \leq R_2 \leq I(Y;X_2|X_1) \quad (31) \]
\[ \rho H(U,V) \leq R_1 + R_2 \leq I(Y;X_1,X_2) \quad (32) \]

In other words, the scaled (by $\rho$) Slepian-Wolf rate region and the capacity region of the MAC with causal cribbing intersect. We can now construct a source-channel code by concatenating a Slepian-Wolf source code and a channel code for the MAC with causal cribbing. The details are omitted.

We give now a proof of the direct part of Theorem 4 for the special case of $\rho = 1$, based on Theorem 3. We will use Shannon strategies, as in [4]. Consider all different strategies $(R_1, R_2)$ that map inputs $x_1 \in X_1$ into inputs $x_2 \in X_2$. A distribution on this set induces a random variable, which we denote by $T$. For the DMMAC $(X_1, X_2, P_{Y|X_1,X_2}, Y)$, the DM derived MAC is denoted by $(X_1, T, P^*_Y|X_1,T,Y)$, where

\[ P^*_Y|X_1,T(y | x_1,t) \triangleq P_{Y|X_1,X_2}(y | x_1,t(x_1)) \]

Define a set of conditions as follows:

\[ H(U|V) \leq H(X_1|W,V) \]
\[ H(V|U) \leq I(T;Y|X_1,W,U) \]
\[ H(U,V) \leq I(X_1,T;Y|U) \]

for

\[ P_{U,V,X_1,T,Y}(w,u,v,x_1,t,y) = P_{U}(w) P_{V|U}(u | w) P_{X_1|W,U}(x_1 | w,u) \]
\[ \cdot P_{T|W,V}(t | w,v) P^*_Y|X_1,T(y | x_1,t) \]

By Theorem 3, a source $P_{UV}(u,v)$ for which the conditions above hold, is transmissible by the derived MAC with strictly causal cribbing. If we now restrict the distribution in (33) to satisfy

\[ P_{U,V,X_1,T,Y}(w,u,v,x_1,t,y) = P_{W}(w) P_{U,V}(u,v) P_{X_1}(x_1) \]
\[ \cdot P_{T}(t) P^*_Y|X_1,T(y | x_1,t) \]

then

\[ H(X_1|W,V) = H(X_1) \]
\[ I(T;Y|X_1,W,U) = I(T;Y|X_1) \]
\[ = I(T;X_2;Y|X_1) \]
\[ = I(X_2;Y|X_1) \]
\[ = I(X_2;Y|X_1) \]
\[ = I(X_1,T;Y|U) \]

and

\[ P_{X_1,X_2,Y}(x_1,x_2,y) = P_{X_1}(x_1) \cdot \sum_{t, I(x_1) = x_2} P_{T}(t) \quad (34) \]

\[ P_{T}(t) = \prod_{x_1} \frac{P_{X_1,X_2,Y}(x_1,x_2=t(x_1))}{P_{X_1}(x_1)} \]

Note that any joint distribution $P_{X_1,X_2}$ can be expressed with strategies, as indicated in (10) above, and in [4], (44), (45)]. Therefore the joint distribution of $X_1$ and $X_2$ can be arbitrary. This completes the proof of achievability, based on Theorem 3.

**References**

The Source Coding Side of Secrecy

Paul Cuff – Princeton University

I. INTRODUCTION

Here we discuss achievability schemes and analysis techniques useful for source coding in adversarial settings. First we mention the equivalence between defining a “valid” joint distribution over all signals in a communication system and defining the behavior of the encoders and decoders. Then we show how to construct a joint distribution in reverse of the operation direction of the encoder, resulting in a communication system that is easy to analyze from the point of view of an optimal adversary. This relates to the work in [1] and [2].

II. CODING DEFINED BY A JOINT DISTRIBUTION

Questions of interest in information theory are usually defined in an operational context, involving encoders, decoders, messages, error probabilities, and distortions. All of these questions could be translated into statement and inquiries involving only probably distributions and their properties. For example, consider the definition of the rate-distortion function.

The standard definition of the rate-distortion function asks for the lowest communication rate such that there exists encoding functions and decoding functions that satisfy certain properties concerning the range of the encoding function and the resulting average distortion. We can alternatively state an equivalent definition for the rate-distortion function as a question about existence of probability distributions that satisfy certain conditions, as follows.

Rate-distortion Function: Given a source distribution \( Q(x^n) \) and a distortion function \( d(\cdot, \cdot) \), what is the minimum rate \( R \) such that for all \( \epsilon > 0 \) and \( n \) sufficiently large there exists a joint distribution \( p(x^n, m, \hat{x}^n) \) such that the cardinality of \( M \) is less than \( 2^{nR} \), \( X^n - M - \hat{X}^n \) form a Markov chain, \( p(x^n) = Q(x^n) \), and \( E_{\hat{x}} \sum_{i=1}^{n} d(x_i, \hat{x}_i) < D + \epsilon \)? Let this be the value of the rate-distortion function \( R(D) \).

The above restatement of the rate-distortion function is trivial and not very helpful. However, in some problems, particularly those involving an adversary, we find this form of the problem statement to be revealing. For one thing, the operation of the encoders and decoders are implicitly specified in the joint distribution of the signals. For example, the encoder in the rate-distortion function statement above operates according to the induced conditional distribution \( p(m|x^n) \). But by changing the emphasis from defining how the encoder and decoder behave to instead defining a joint distribution to satisfy the constraints of the problem, a reverse-encoding construction is easily conceived. That is, we might find it useful to construct the inverse of the encoder, \( p(x^n|m) \), to have certain properties rather than design the actual working of the encoder explicitly.

III. REVERSE-CHANNEL ENCODER

In [1] we characterize the best lossless source coder that makes use of limited secret key to cause maximal distortion to an eavesdropper. An asymptotically optimal construction for the encoder is derived using two key concepts, discussed in the following two paragraphs. The communication is split into two parts, one message which has the secret key applied to it as a one-time pad, and the other non-secure message which is the crux of the analysis. The eavesdropper obtains the non-secure message, and we must be able to analyze the lowest distortion the eavesdropper can achieve given the received message.

The first key concept for constructing an optimal encoder is to design the encoder in reverse. Let \( M \) be the non-secure message obtained by the eavesdropper, and let \( X^n \) be the source sequence. The eavesdropper’s minimum average distortion when a particular message \( m \) is observed depends on the distribution \( p(x^n|m) \), which is the inverse of the encoder. Consider the following design. Using the standard technique of random codebook construction, let \( U \) be an auxiliary random variable correlated with \( X \) and construct a codebook of i.i.d. \( U^n \) sequences indexed by \( M \). Let \( X^n \) be conditionally distributed as the output of a memoryless channel \( p(x^n|u) \) when the input to the channel is \( U^n(M) \). The memoryless nature of this constructed conditional distribution \( p(x^n|m) \) makes it trivial to characterize the minimum distortion incurred by the eavesdropper.

The last concept needed for the analysis is that of resolvability. In the construction above, if \( M \) is uniformly distributed over a set of size \( 2^{nR} \), with \( R > I(X; U) \), then the resulting distribution \( p(x^n) \) will be close to the i.i.d. source distribution \( Q(x^n) \) in total variation. Consequently, the constructed joint distribution is close in total variation to a valid joint distribution. We close this gap with some lemmas about total variation distance.

REFERENCES

A Comparative Study of Cooperation for the Two-user Interference Channel

Daniela Tuninetti
University of Illinois at Chicago
Department of Electrical and Computer Engineering
851 S. Morgan St. (M/C 154)
Chicago, IL 60607, USA
Email: danielat@uic.edu

Abstract—The interference channel is a model for a wireless network where several source-destination pairs compete for the same resources. When the sources transmit simultaneously the destinations experience unwanted interference. Determining how to optimally manage the interference is a long standing open problem in network information theory. In this talk we review one emerging interference management technique: cooperation. Cooperation can occur among source nodes only (Source Cooperation or Generalized Feedback), or among destination nodes only (Destination Cooperation), or among all nodes (General Cooperation). For the special case of Gaussian noise channels, we compare the different forms of cooperation in terms of symmetric generalized degrees of freedom. When known, we also discuss capacity results to within a constant number of bits.

I. CHANNEL MODEL

A general full-duplex cooperative interference channel is a $2K$-node network with inputs $X_0, \ldots, X_{2K-1}$, outputs $Y_0, \ldots, Y_{2K-1}$ and a memoryless channel with transition probability $P := P[X_0, \ldots, X_{2K-1}][Y_0, \ldots, Y_{2K-1}]$. Sources have index $0$ to $K-1$ and destinations $K$ to $2K-1$. Message $W_i$ is independent of everything else and uniformly distributed over $[1 : 2^{NR_i}]$, with $N \in \mathbb{N}$ denoting the block-length and $R_i \in \mathbb{R}^+$ the rate in bits per channel use, for $i \in [0 : K-1]$. Encoding at node $i \in [0 : 2K-1]$ at channel use $t \in [1 : N]$ is a deterministic function $X_i(W_i, Y_{i-1}^{t-1})$, with the convention $W_K = \ldots = W_{2K-1} = \emptyset$. Decoding at node $i \in [K : 2K-1]$ after the $N$-th channel use is a deterministic function $W_{i-K}(Y_i^N)$. A rate vector $(R_0, \ldots, R_{K-1})$ is $\epsilon$-achievable if $\max_{i \in [0 : K-1]} P[W_i \neq W_i] \leq \epsilon$ for some $\epsilon \in (0, 1]$. The capacity region is the convex closure of the set of rate vectors that are $\epsilon$-achievable for all $\epsilon > 0 [1]$. In this work we consider the case $K = 2$ only, which is depicted in Fig. 1.

II. MODELS OF COOPERATION

Several cooperation models have been analyzed in literature such as: cooperation only occurs among sources (Source Cooperation, also known as Generalized Feedback), or only among destinations (Destination Cooperation), or among all nodes (General Cooperation). In each case, cooperation can be in-band or out-of-band. For out-of-band cooperation, the network is effectively composed of two parallel networks: the underlying interference channel (from the source inputs to the destination outputs) and the cooperation channel (which is usually assumed to be deterministic/noiseless).

In the following we list some recent journal papers that deal with cooperation for the two-user interference channel and that we will compare in this talk. The interested reader can find the technical details in these works and the list of related preceding publications in the references therein. Some extensions to networks with more than two source-destinations pairs can be found for example in [14], [15].

A. Source Cooperation/Generalized Feedback

The destinations do not have an input to the channel, i.e., $X_K = \ldots = X_{2K-1} = \emptyset$.

A.1) Out-of-band. The channel input at source $i, i \in [0 : K-1]$, is a two-dimensional vector $X_i := [X_{i,d}, X_{i,s}]$, where $X_{i,d}$ affects the channel outputs at the destinations and $X_{i,s}$ the channel outputs at the sources. Effectively, there are two parallel channels: the underlying interference channel with transition probability $P[Y_K, \ldots, Y_{K-1}, X_{0,d}, \ldots, X_{K-1,d}]$ and the cooperation channel with transition probability $P[Y_0, \ldots, Y_{K-1}, X_{0,s}, \ldots, X_{K-1,s}]$. 

Fig. 1. The memoryless cooperative interference channel (channel inputs in green, channel outputs in pink, transmitted messages in black, and decoded messages in blue).
Conferencing Encoders is a special case of out-of-band source cooperation where the cooperation channel is deterministic [2].

Cognition is a special case of conferencing encoders where the channel output at some sources is constant (these sources are known as “primary users”) and the channel output at the other sources is such that all messages can be conveyed in a single channel use (these sources are known as the “cognitive users”) [4], [5].

A.2) In-band. The same inputs affect both the underlying interference channel and the cooperation channel. The cooperation channel is noisy [6], [7], [8], [9].

Noisy Output Feedback is a special case of in-band source cooperation where the channel output at a source is a degraded version of the channel output at its intended destination, i.e., \( Y_i = g_i(Y_{i+K}, Z_i) \) for some deterministic function \( g_i \) and some random process \( Z_i \) representing the “extra noise”, for all \( i \in [0 : K−1] \) [11]. If the “extra noise” process is deterministic then the channel is known simply as interference channel with feedback [12].

Ultimate Limit of Source Cooperation. By sharing all messages among the sources we obtain an equivalent memoryless broadcast channel with input \( X_\text{eq} := (X_0, \ldots, X_{K−1}) \) and outputs \( Y_K, \ldots, Y_{2K−1} \), whose capacity (which is not known in general [1]) gives the ultimate gain that can be obtained with source cooperation.

B. Destination Cooperation

The sources do not have an output from the channel, i.e.,
\[ Y_0 = \ldots = Y_{K−1} = 0. \]

B.1) Out-of-band. The channel output at destination \( i, i \in [K : 2K−1] \), is a two-dimensional vector \( Y_i := [Y_{i,d}, Y_{i,a}] \), where \( Y_{i,d} \) is affected by the channel inputs from the destinations and \( Y_{i,a} \) by the channel inputs from the sources. Effectively, there are two parallel channels: the underlying interference channel with transition probability \( P[Y_0, \ldots, Y_{K−1}, X_0, \ldots, X_{K−1}] \) and the cooperation channel with transition probability \( P[Y_0, \ldots, Y_{K−1}, X_K, \ldots, X_{2K−1}]. \)

Conferencing Decoders is a special case of out-of-band destination cooperation where the cooperation channel is deterministic [3].

B.2) In-band. The channel outputs are affected by all channel inputs [6], [10].

Ultimate Limit of Destination Cooperation. By sharing all channel outputs at the destinations among the destinations we obtain an equivalent memoryless multiple access channel with inputs \( X_0, \ldots, X_{K−1} \) and output \( Y_\text{eq} := (Y_K, \ldots, Y_{2K−1}) \), whose capacity region [1] gives the ultimate gain that can be obtained with destination cooperation.

C. General Cooperation

This is the most general case where all nodes cooperate.

C.1) Out-of-band. Channel outputs at the destinations are affected by the channel inputs from the sources only and channel outputs at the sources are affected by the channel inputs from the destinations only. Effectively, there are two parallel channels: the underlying interference channel with transition probability \( P[Y_K, \ldots, Y_{2K−1}, X_0, \ldots, X_{K−1}] \) and a cooperation channel with transition probability \( P[Y_0, \ldots, Y_{K−1}, X_K, \ldots, X_{2K−1}] \).

Rate-limited Feedback is a special case of out-of-band general cooperation where the cooperation channel is deterministic [13].

C.2) In-band. All channel inputs affect all channel outputs.

To the best of the author’s knowledge the case of in-band general cooperation has not been studied so far.

Ultimate Limit of General Cooperation. By sharing all messages among the sources and all channel outputs at the destinations among the destinations we obtain an equivalent memoryless point-to-point channel with input \( X_\text{eq} := (X_0, \ldots, X_{K−1}) \) and output \( Y_\text{eq} := (Y_K, \ldots, Y_{2K−1}) \), whose capacity gives the following sum-rate upper bound
\[ R_0 + \ldots + R_{K−1} \leq C_{\text{perfect coop.}} := \max_{I(X_\text{eq}; Y_\text{eq})} I(X_\text{eq}; Y_\text{eq}). \]
which cannot be improved by feedback, i.e., \( C_{\text{perfect coop.}} \) does not depend on \( (Y_0, \ldots, Y_{K−1}) \).

ACKNOWLEDGMENT

The author’s work on the Interference Channel with Generalized Feedback [7], [8] is part of Dr. S. Yang’s doctoral dissertation, while that on the Cognitive Interference Channel [4], [5] is part of Dr. S. Rini’s doctoral dissertation and was done in collaboration with Dr. N. Devroye.

REFERENCES


Refinement of the Random Coding Bound

Yücel Altuğ Student Member and Aaron B. Wagner Member

Abstract—We provide an improved random coding bound for a class of discrete memoryless channels that improves the pre-factor in front of the exponent. Specifically, while the tightest known bounds have a constant pre-factor, the presented bound has a pre-factor of order $O(1/\sqrt{N})$, where $N$ is the blocklength.

I. INTRODUCTION

Characterizing the interplay between the data rate, error probability, and blocklength of the best codes for a given discrete memoryless channel (D.M.C.) is a central problem in information theory. Although this interplay has been investigated starting from the early days of the field [1], [2], [3], [4], [5], [6], [7], [8], [9], it is still an active research topic [10], [11], [12], [13], [14]. Due to the difficulty of the problem, it is customary to consider formulations that are asymptotic in the blocklength to characterize this interplay.

One of the best-known formulations is “error exponents,” in which the rate is held fixed below the channel capacity and one characterizes the rate of decay of the error probability, which is exponential in $N$ for most channels [3], [4], [5], [6]. Another asymptotic characterization is the “normal approximation” in which the error probability is fixed at a positive constant in $(0, 1/2)$ and one characterizes the speed with which the data rate converges to capacity [2], [13]. Between these two characterizations, one can consider the setup in which the error probability tends to zero and the rate simultaneously approaches capacity, with the goal of characterizing the speed of one convergence as a function of the other [14]. We call these three asymptotics the “small error probability,” “large error probability,” and “medium error probability” regimes, respectively. It should be noted that although there is an underlying relation between these three asymptotics via the minimum error probability of the $(N, R)$ codes, it is not possible to directly derive one regime’s conclusion from any of the other two.

In this paper, we consider upper bounds on the error probability of the best $(N, R)$ code in the small error probability regime. We prove a refinement of the random coding bound that has a pre-factor of $O(1/\sqrt{N})$ for a broad class of D.M.C.s, which includes positive channels with positive dispersion. This improves the best known pre-factor of $O(1)$ due to Fano [3] and Gallager [4]. It should be remarked that the improvement in the pre-factor is significant for rates close to the capacity, where the optimal error exponent itself is small and the pre-factors are therefore important. Although some improved finite-$N$ bounds could easily be extracted from the proofs in this paper, we defer the task of optimizing these bounds and numerically comparing them to the existing bounds for future work.

To better understand the asymptotic regimes mentioned above, noting the analogy between channel coding and the analysis of sums of i.i.d. random variables in probability theory is instructive. In particular, the small, medium, and large error probability regimes of channel coding correspond to large deviations, moderate deviations, and central limit theory, respectively, for i.i.d. sums. With this analogy in mind, the problem we consider in this paper is reminiscent of the “exact asymptotics” problem [18], [21, Theorem 3.7.4] in large deviations. The aim of that problem is to determine the pre-factor of the exponentially vanishing term in the large deviations theorem. Bahadur and Rao [18] characterized this pre-factor, including the constant, under some regularity conditions. Their result gives some hope that it may be possible to determine the optimal pre-factor, or at least its order, for channel coding in the small error probability regime. The current best pre-factors in the sphere packing bound [22] and random coding bound (this paper) do not coincide, however, and the order of the optimal pre-factor is only known for special cases such as the binary symmetric [15], binary erasure [15], and power-constrained A.W.G.N. [16] channels.

II. NOTATION, DEFINITIONS AND STATEMENT OF THE RESULT

A. Notation

Boldface letters denote vectors, regular letters with subscripts denote individual elements of vectors. Furthermore, capital letters represent random variables and lowercase letters denote individual realizations of the corresponding random variable. Throughout the paper, all logarithms are base-$e$. For a finite set $\mathcal{X}$, $P(\mathcal{X})$ denotes the set of all probability measures on $\mathcal{X}$. Similarly, for two finite sets $\mathcal{X}$ and $\mathcal{Y}$, $P(\mathcal{Y}|\mathcal{X})$ denotes the set of all stochastic matrices from $\mathcal{X}$ to $\mathcal{Y}$. $\mathbb{R}$, $\mathbb{R}^+$ and $\mathbb{R}_+$ denote the set of real, positive real and non-negative real numbers, respectively. $\mathbb{Z}^+$ denotes the set of positive integers. We follow the notation of Csiszár-Körner [9] for fundamental information theoretic quantities.

B. Definitions

Throughout the paper, let $W$ be a D.M.C. from $\mathcal{X}$ to $\mathcal{Y}$, $E_{R}(R, W)$ and $E_{\rho}(\rho, Q, W)$ denotes the random coding exponent (cf. [7, eq. (5.6.16)]) and the Gallager’s function (cf. [7, eq. (5.6.14)]), respectively. For any $W \in P(\mathcal{Y}|\mathcal{X})$ and $Q \in P(\mathcal{X})$ the ensemble average error probability (resp. ensemble average error probability conditioned on the
message \( m \) of a random code with codewords generated by using \( Q \) along with a maximum likelihood decoder\(^1\) is denoted by \( \hat{P}_e(Q) \). \( \sigma^2(W) \) denotes the dispersion (cf. [2], [13]) of \( W \in \mathcal{P}(\mathcal{Y}|\mathcal{X}) \).

Further,
\[
S_{Q,W} := \{(x, y) \in \mathcal{X} \times \mathcal{Y} : Q(x)W(y|x) > 0 \},
\]
\[
\hat{S}_{Q,W} := \{(x, y, z) : Q(x)W(y|z)Q(z)W(y|z) > 0 \},
\]
\[
\Lambda_q(W) := \{x \in \mathcal{X} : W(y|x) > 0 \}.
\]

Given a \((Q, W) \in \mathcal{P}(\mathcal{X}) \times \mathcal{P}(\mathcal{Y}|\mathcal{X})\) pair, the following property amounts to saying that “feasibility decoding is optimal” (F.D.I.O.) when a random code with distribution \( Q \) is used for transmission over the D.M.C. \( W \).

Property 1: (F.D.I.O.) For all \((x, y, z) \in \hat{S}_{Q,W}, W(y|x) = W(y|z)\).

C. Statement of the result

Theorem 1: Let \( W \in \mathcal{P}(\mathcal{Y}|\mathcal{X}) \) be arbitrary with \( \sigma^2(W) > 0 \) and\(^2\) \( R \in [R_{\min}(W), C(W)] \). If there exists \( Q \in \mathcal{P}(\mathcal{X}) \) achieving \( E_e(R, W) \) such that the F.D.I.O. property does not hold, then for all \( N \in \mathbb{Z}^+ \)
\[
\hat{P}_e(Q) \leq \frac{K_1}{\sqrt{N}} e^{-\frac{N E_e(R,W)}{2}},
\]
where \( K_1 \) is a positive constant that only depends on \( W \) and \( R \).

Remark 1: Using the standard expurgation arguments, Theorem 1 ensures the existence of an \((N, R)\) code with maximal error probability upper bounded by the right side of (4), with a different constant.

Remark 2: The proof of Theorem 1 is based on Fano’s proof of the random coding bound [3, pp. 324–331] with a refinement to prove the smaller pre-factor. In the literature, Fano’s argument has been superceded by Gallager’s, which is simpler and gives the same exponent. Fano’s argument, however, lends itself more readily to obtaining improved pre-factors.

Remark 3: Note that the assumption related to the F.D.I.O. property in Theorem 1 holds for a reasonably large class of channels, such as the class of positive channels with positive dispersion. Yet this assumption does not exclude only degenerate cases; note that the binary erasure channel (B.E.C.) is excluded. Moreover, the assumption is not necessary in order to have an upper bound on the error probability with a pre-factor of \( O(1/\sqrt{N}) \). Indeed, Elias [15] proves such an upper bound for B.E.C.

Remark 4: The above theorem applies to the ensemble average error probability at rates above the critical rate. For rates below the critical rate, the order of the pre-factor turns out to hinge on the F.D.I.O property, a fact that was heretofore overlooked in the literature. The next result, Theorem 2, corrects a small oversight by Gallager [17], who claims that for any D.M.C. \( W \) and rate \( R < R_{\min}(W) \)
\[
\min_{Q \in \mathcal{P}(\mathcal{X})} \hat{P}_e(Q) \sim \frac{g}{\sqrt{N}} e^{-\frac{N E_e(R,W)}{2}},
\]
for some constant \( g \). In fact, his argument is valid only for those \( W \in \mathcal{P}(\mathcal{Y}|\mathcal{X}) \) such that there exists \( Q \in \mathcal{P}(\mathcal{X}) \) with \( E_e(1, Q, W) = \max_{P \in \mathcal{P}(\mathcal{X})} E_e(1, P, W) \) and the F.D.I.O. property is not true. For the remaining channels, we have the following result.

Theorem 2: Consider an arbitrary \( W \in \mathcal{P}(\mathcal{Y}|\mathcal{X}) \) with \( C(W) > 0 \) and \( R < R_{\min}(W) \). If the F.D.I.O. property holds for all \( Q \in \mathcal{P}(\mathcal{X}) \) with \( E_e(1, Q, W) = \max_{P \in \mathcal{P}(\mathcal{X})} E_e(1, P, W) \), then for all \( N \in \mathbb{Z}^+ \)
\[
K_2 e^{-\frac{N E_e(R,W)}{2}} \leq \hat{P}_e(Q) \leq e^{-\frac{N E_e(R,W)}{2}},
\]
for some \( 0 < K_2 < 1 \) that depends on \( W \) and \( R \).

III. PROOF OF THEOREM 1

Let \( W \in \mathcal{P}(\mathcal{Y}|\mathcal{X}) \) with \( \sigma^2(W) > 0 \) be arbitrary. Fix an arbitrary \( R \in [R_{\min}(W), C(W)] \). Consider an arbitrary \( Q \in \mathcal{P}(\mathcal{X}) \) with \( E_e(R, Q, W) = E_e(R, W) \), such that the F.D.I.O. property does not hold, whose existence is guaranteed by the assumption of the theorem. For the sake of notational convenience, let \( S_Q, \hat{S}_Q \) and \( \Lambda_q \) denote \( S_{Q,W}, \hat{S}_{Q,W}, \Lambda_q(W) \) as given in (1), (2) and (3), respectively. Define
\[
P_{X,Y,Z}(x, y, z) := Q(x)W(y|x)Q(z),
\]
for all \((x, y, z) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{X} \). Also, let
\[
\hat{P}_{X,Y,Z}(x, y, z) := \left\{ \begin{array}{ll}
P_{X,Y,Z}(x, y, z) & \text{if } (x, y, z) \in \hat{S}_Q \vspace{1mm} \\bar{0} & \text{else} \end{array} \right.
\]
\[
\tilde{P}_{X,Y,Z}^N(x, y, z) := \sum_{n=1}^{N} P_{X,Y,Z}(x_n, y_n, z_n) = \sum_{n=1}^{N} \hat{P}_{X,Y,Z}(x_n, y_n, z_n).
\]

One can show that
\[
\exists \rho^* \in (0, 1], \text{s.t.} \quad \frac{\partial E_e(\rho, Q, W)}{\partial \rho} \bigg|_{\rho=\rho^*} = R.
\]

Define
\[
D_0 := \Lambda^t(\frac{\rho^*}{1 + \rho^*}),
\]
\[
\Lambda_1(s, t) := \log E_{\hat{P}_{X,Y,Z}} \left[ e^{s \log \frac{Q(Y|X)}{Q(Y|X)} + t \log \frac{W(Y|X)}{Q(Y|X)}} \right],
\]
for any \((s, t) \in \mathbb{R}^2\). Observe that owing to the fact that \( \log \frac{Q(Y|X)}{Q(Y|X)} \) is a convex function, \( \Lambda_1(s, t) \) is infinitely differentiable on \( \mathbb{R}^2 \).

\[1\]We assume that the ties are broken in such a way that always results in an error. However, this assumption increases the error probability by at most a factor of 2.

\[2\]Since \( \sigma^2(W) > 0 \) is equivalent to \( R_e(W) < C(W) \) (e.g. [7, pg. 160]), \( [R_e(W), C(W)] \) is well-defined.
Finally, for any \( N \in \mathbb{Z}^+ \), let

\[
D_N := \left\{ \frac{1}{N} \sum_{n=1}^{N} \log \frac{f(Y_n)}{W(Y_n|X_n)} > D_0 \right\}.
\]

From the standard random coding arguments, one can show that

\[
P_{e,1}(Q) \leq P_{X,Y}^N \left\{ \frac{1}{N} \sum_{n=1}^{N} \log \frac{f(Y_n)}{W(Y_n|X_n)} \leq D_0, \right. \\
\left. \frac{1}{N} \sum_{n=1}^{N} \log \frac{W(Y_n|X_n)}{W(Y_n|Z_n)} \leq 0 \right\} e^{NR} + P_{X,Y}^N \{D_N\}.
\]

(12)

In order to conclude the proof, we need to upper bound the two terms on the right side of (12).

We begin with the second term. To this end, define

\[
\Lambda^*(a) := \sup_{\lambda \in \mathbb{R}} (a\lambda - \Lambda(\lambda)),
\]

(13)

for any \( a \in \mathbb{R} \). By noting the convexity of \( \Lambda(\cdot) \); (10), (13) and some algebra imply that

\[
\Lambda^*(D_0) = E_r(R,W).
\]

(14)

Owing\(^3\) to a result of Ney [19, eq. (2)], we deduce\(^4\) that

\[
P_{X,Y}^N \{D_N\} \leq \frac{k_1}{\sqrt{N}} e^{-N\Lambda^*(D_0)} = \frac{k_1}{\sqrt{N}} e^{-N E_r(R,W)},
\]

(15)

for some \( k_1 \in \mathbb{R}^+ \) and for all sufficiently large \( N \), where the equality follows from (14).

Next, we upper bound the first term on the right side of (12). First, since for any \((x, y, z) \notin S_Q\),

\[
\log \frac{W(y|x)}{W(y|z)} = \infty, \quad Q \times W \times Q - (a.s.)
\]

we have

\[
\alpha_N := P_{X,Y}^N \left\{ \frac{1}{N} \sum_{n=1}^{N} \log \frac{f(Y_n)}{W(Y_n|X_n)} \leq D_0, \right. \\
\left. \frac{1}{N} \sum_{n=1}^{N} \log \frac{W(Y_n|X_n)}{W(Y_n|Z_n)} \leq 0 \right\}
\]

\[
= P_{X,Y}^N \{S_Q\} \tilde{\alpha}_N,
\]

(16)

where, in (16) we define

\[
\tilde{\alpha}_N := \tilde{P}_{X,Y,Z}^N \left\{ \frac{1}{N} \sum_{n=1}^{N} \log \frac{f(Y_n)}{W(Y_n|X_n)} \leq D_0, \right. \\
\left. \frac{1}{N} \sum_{n=1}^{N} \log \frac{W(Y_n|X_n)}{W(Y_n|Z_n)} \leq 0 \right\}.
\]

(17)

Define

\[
t^* := -1/(1 + \rho^*), \quad s^* := 1 + 2t^*.
\]

(18)

One can show that

\[
\frac{\partial \Lambda_1(s,t)}{\partial s} \bigg|_{s=s^*} = \frac{\partial \Lambda_1(s,t)}{\partial t} \bigg|_{t=t^*} = 0.
\]

(19)

Before proceeding further, we define

\[
B := (-\infty, D_0) \times (-\infty, 0), \quad a_B := [D_0, 0]^T,
\]

(20)

\[
\Lambda_1^*(b) := \sup_{a \in \mathbb{R}^2} \{\langle b, a \rangle - \Lambda_1(a)\},
\]

(21)

for any \( b \in \mathbb{R}^2 \). Note that owing to the convexity of \( \Lambda_1(\cdot) \), one can show that

\[
\Lambda_1^*(a_B) = s^* D_0 - \Lambda_1(s^*, t^*),
\]

(22)

where \( s^* \) and \( t^* \) are as defined in (18).

Moreover,

\[
\Lambda_0(\lambda) := \log E_{P_{X,Y,Z}} \left[ e^{\lambda \log \frac{W(Y|X)}{W(Y|Z)}} \right] > 0, \quad \lambda \in \mathbb{R}.
\]

(23)

A result of Ney [19, Theorem] implies that\(^5\) there exists a unique dominating point (cf. [19, Definition]) \( b_B \in B \), which also satisfies (cf. [20, eq. (3.3)])

\[
\Lambda_1^*(b_B) = \inf_{b \in B} \sup_{a \in \mathbb{R}^2} \{\langle b, a \rangle - \Lambda_1(a)\}.
\]

(25)

Moreover, [19, eq. (2)] implies that

\[
\frac{c_1}{N} e^{-N \Lambda_1^*(d_n)} \leq \hat{\alpha}_N \leq \frac{c_2}{\sqrt{N}} e^{-N \Lambda_1^*(d_\infty)},
\]

(26)

for some \( c_1, c_2 \in \mathbb{R}^+ \) and all sufficiently large \( N \). Next, one can prove that

\[
\Lambda_1^*(a_B) = \Lambda_1^*(b_B).
\]

(27)

Plugging (27) into (26) yields

\[
\hat{\alpha}_N \leq \frac{c_2}{\sqrt{N}} e^{-N \Lambda_1^*(a_B)},
\]

(28)

for some \( c_2 \in \mathbb{R}^+ \) and all sufficiently large \( N \). Moreover, it is possible to check that

\[
\Lambda_1(s^*, t^*) = -\log P_{X,Y,Z} \{S_Q\} + 2\Lambda \left( \frac{\rho^*}{1+\rho^*} \right).
\]

(29)

\(^3\)Because of the fact that the random variables in \( D_N \) are i.i.d., we might have used the stronger ‘exact asymptotics’ result [18] by considering lattice and non-lattice cases separately. However, the application of such a result has little effect on the ultimate bound since the second term of (12) cannot be handled by an analogous theorem.

\(^4\)To justify the application of the aforementioned result, one can check that \( \Lambda(\cdot) \) satisfies the regularity conditions stated in [19, Theorem]. Moreover, one can verify that \( \Lambda^* \left( \tilde{P}_{X,Y,Z}(Q,B) \right) > \Lambda^*(0) \geq E_{P_{X,Y}} \left[ \log \frac{W(Y|X)}{W(Y)} \right] \). This guarantees the ‘mean does not belong to the set’ regularity condition that \( D_N \) needs to satisfy. The remaining regularity conditions on \( D_N \) are easy to check. Therefore the existence of a unique ‘dominating point’ (cf. [19, Definition]) is guaranteed owing to [19, Theorem]. However, one can check that this dominating point is \( D_0 \) (e.g. [20, pg. 159]). Lastly, [20, eq. (3.3)] gives the exponential factor in (15).

\(^5\)Similar to the previous footnote, (24) ensures the fulfillment of the ‘mean does not belong to the set’ condition of the theorem. The remaining conditions are easy to check.
Finally, plugging (18), (22) and (29) into (28) along with (10), (14) and some algebra yields
\[
\tilde{\alpha}_N \leq P_{X,Y,Z} \left\{ \tilde{S}_Q \right\}^{-N} \frac{c_2}{\sqrt{N}} e^{-N|E_o(R,W)+R|}.
\] (30)

By plugging (30) into (16) and noting \(P_{X,Y,Z}^N \left\{ \tilde{S}_Q \right\} = P_{X,Y,Z} \left\{ \tilde{S}_Q \right\}^N\), we deduce that
\[
\alpha_N \leq \frac{c_2}{\sqrt{N}} e^{-N|E_o(R,W)+R|}.
\] (31)

Lastly, (12), (15) and (31) imply that
\[
P_e(Q) \leq \frac{K_1}{\sqrt{N}} e^{-NE_o(R,W)},
\] (32)
for some \(K_1 \in \mathbb{R}^+\) and all sufficiently large \(N\). \(\square\)

IV. PROOF OF THEOREM 2

Let \(W \in \mathcal{P}(\mathcal{Y}|\mathcal{X})\) with \(|C(W)| > 0\) and \(R < R_c(W)\) be arbitrary. Assume that for all \(Q \in \mathcal{P}(\mathcal{X})\) with \(E_o(1,Q,W) = \max_{P \in \mathcal{P}(\mathcal{X})} E_o(1,P,W)\), the F.D.I.O. property holds. Consider any such \(Q \in \mathcal{P}(\mathcal{X})\). For this \((Q,W)\) pair, let \(P_{X,Y,Z}\) be as given in (7) and (8), respectively. For the sake of notational convenience, let \(S_Q\) and \(X_q^\ell\) denote \(S_Q,W\) and \(X_q^{\ell\ell}\) as given in (2) and (3), respectively.

Owing to the fact that the F.D.I.O. property holds, one can show that
\[
\log P_{X,Y,Z} \left\{ \tilde{S}_Q \right\} = -E_o(1,Q,W).
\] (33)

By using the standard random coding arguments, we have
\[
P_e(1,Q) \leq e^{-NR} P_{X,Y,Z} \left\{ \tilde{S}_Q \right\}^N
\] (34)
\[
= e^{-N|E_o(1,Q,W)|}
\] (35)
\[
= e^{-NE_o(R,W)},
\] (36)
where (34) follows from the fact that for any \((x,y,z) \notin \tilde{S}_Q\), \(\log \frac{W(y|x)}{W(y|x)} = \infty\). \(Q \times W \times Q \sim (a,s)\), and the F.D.I.O. property is satisfied, (35) follows from (33) and (36) is true owing to the choice of \(Q \in \mathcal{P}(\mathcal{X})\) and the fact that \(R < R_c(W)\) (e.g. [17, pg. 245]). Hence, the upper bound of (6) follows.

In order to establish the lower bound of (6), one can use Gallager’s arguments [17, pg. 245-246] by noting
\[
P_{X,Y,Z}^N \left\{ \frac{1}{N} \sum_{n=1}^N \log \frac{W(Y_n|X_n)}{W(Y_n|Z_n)} \leq 0 \right\} = e^{-N E_o(1,Q,W)}.
\]

V. COMPARISON WITH THE SPHERE-PACKING BOUND

Our main result is an improvement of the random coding bound for rates greater than the critical rate, obtained by improving the pre-factor from \(O(1)\) to \(O(1/\sqrt{N})\) for a broad class of channels.

Even for the symmetric channels, however, this pre-factor does not match its lower bound counterpart [22], namely \(O(N^{1/2} \log^{1/2} R)\), where \(E_o(1,R)\) is the slope of \(E_o\). It is worth noting, however, that for rates close to capacity, \(E_o(1,R)\) is nearly zero, so the order of the pre-factor is nearly determined in this regime.

REFERENCES

Source Coding with Side-Information at the Receivers and an Application

Michèle Wigger
Telecom ParisTech, Paris, France
Email: michele.wigger@telecom-paristech.fr

Abstract—We review results on the source-coding problem in Figure 1, i.e., on the simple network proposed by Gray and Wyner where additionally the receivers have side-information about the source. We also describe how this problem is relevant for coding over broadcast channels (BC) with feedback.

I. THE SOURCE CODING PROBLEM

We consider the source-coding problem in Figure 1, where a transmitter describes the source sequence $X^n = (X_1, \ldots, X_n)$ to Receivers 1 and 2, who observe the side-information $Y_{i,n} = (Y_{1,1}, \ldots, Y_{1,n})$ and $Y_{2,n} = (Y_{2,1}, \ldots, Y_{2,n})$, respectively. Here, $\{ (X_t, Y_{1,t}, Y_{2,t}) \}_{n=1}^{\infty}$ is a sequence of independent and identically distributed (IID) triples of joint law $P_{X,Y_{1},Y_{2}}$.

The transmitter can send a common message $M_0$ of rate $R_0$ to both receivers and private messages $M_1$ and $M_2$ of rates $R_1$ and $R_2$ to Receivers 1 and 2, respectively. The goal is that Receiver $i$, for $i \in \{1,2\}$, produces a reconstruction sequence $\hat{X}_i^n = (\hat{X}_{i,1}, \ldots, \hat{X}_{i,n})$ that satisfies

$$\mathbb{E} \left[ \sum_{t=1}^{n} d_i(X_t, \hat{X}_{i,t}) \right] \leq D_i,$$

where $d_i$ is the symbol-wise distortion function and $D_i \geq 0$ the distortion constraint. The described setup includes the case where $X^n = (X_1^n, X_2^n)$ and Receiver $i$ only wishes to reconstruct a lossy version of $X_i^n$.

Theorem 1 ([5]): A triple $(R_0, R_1, R_2)$ is achievable if

$$R_0 + R_1 \geq I(X; X_1, V_0|Y_1), \quad i \in \{1,2\},$$

$$R_0 + R_1 + R_2 \geq I(X; X_1|Y_0)V_0 + I(X; X_2|Y_2, V_0) + \max_{i \in \{1,2\}} I(X; V_0|Y_i)$$

for some $X_1, X_2, V_0$ such that $X_1, X_2, V_0 \rightarrow X_1 \rightarrow X_2 \rightarrow Y_1, Y_2$ form a Markov chain and $\mathbb{E} [d(X, \hat{X}_i)] \leq D_i$, for $i \in \{1,2\}$.

Though the region is not known to be optimal in general, it is optimal for all special cases where the rates-distortions region is known. In the following we review these special cases. Gray and Wyner characterized the rates-distortions region when there is no side-information [1]. Heegard and Berger [2] proposed inner and outer bounds on the rates-distortions region when the transmitter can send only a common message, but no private messages, i.e., when $R_1 = R_2 = 0$. The bounds coincide when $X \rightarrow Y_1 \rightarrow Y_2$ or $X \rightarrow Y_2 \rightarrow Y_1$, Steinberg and Merhav [3], and Tian and Diggavi [4] considered the case when the transmitter can send a common message and a private message to Receiver 2 but not to Receiver 1, i.e., when $R_1 = 0$. More specifically, [3] characterizes the rates-distortions region for the successive Wyner-Ziv refinement problem where Receiver 2 (who receives the additional private message)

II. APPLICATION TO THE BC WITH FEEDBACK

Our lossy source-coding problem has application to coding over BCs with feedback, see especially [5] but also [6], [7]. The core idea of these schemes is the following. In a first stage the transmitter sends the desired messages at rates that do not allow for decoding at the receivers. Then, using the feedback, it identifies resolution information which it sends during the second stage. To this end, it first compresses the resolution information using a good lossy source-code for the setup in Figure 1, where the side-information at the receivers is given by their channel outputs in the first stage. It then sends the compression indices over the BC so that the receivers can decode them. With the lossy reconstructions of the resolution information learned in the second stage and the outputs from the first stage the receivers then decode the desired messages.

REFERENCES

Abstract—We examine codes, over the additive Gaussian noise channel, designed for for reliable communication at some specific signal-to-noise ratio (SNR) and constrained by the permitted MMSE at some lower SNR. We show that the maximum possible rate is the one attained by superposition codebooks. Moreover, the MMSE function of codes attaining this maximum rate under the MMSE constraint is completely defined for all SNR. The problem is also extended to the maximization of the rate under two MMSE constraints. The optimal rate is again achieved by three-layers superposition codes.

I. INTRODUCTION

Capacity and capacity achieving codes have been the main concern of Information Theory from the very beginning. Trying to design capacity achieving codes is a central goal of many researchers in this field. Recently some emphasis has been given to the research of non-capacity achieving point-to-point codes [1], [2]. These codes, referred to as “bad” point-to-point codes [2], are heavily used in many multi-terminal wireless networks. In [3] it was shown that the mutual information and thus also the minimum mean square error (MMSE) of “good” (capacity achieving) point-to-point codes is known exactly, no matter the specific structure of the code. Furthermore, it is known that “bad” codes can obtain lower MMSE at low signal-to-noise ratios (SNRs) [2]. This advantage is meaningless in point-to-point communication, where all that matters is the performance at the receiver. However, in multi-terminal wireless networks, such as a cellular network, the case is different. In such networks there are two fundamental phenomena: interference from one node to another (an interference channel), and the potential cooperation between nodes (a relay channel). In the interference channel, where a message sent to an intended receiver acts as interference to other receivers in the network, a lower MMSE implies better interference cancelation, and thus improved rates for the interfered user. The performance of optimal point-to-point codes in the interference setting was the investigation in [4].

The best known achievable region for the two-user interference channel is given by the Han and Kobayashi (HK) scheme [5]. This scheme uses partial decoding of the interfering message at the receiver. Rate splitting (that is, superposition coding) is a special case of the HK scheme, and is also point-to-point “bad” (see [1, Appendix VIII-C]). It was shown in [6] that these codes are close to optimal for the Gaussian interference channel, and in fact are within one bit from capacity. In this work we show that these codes are in fact optimal MMSE-wise.

Although it is known that we can obtain an advantage, MMSE-wise, when using “bad” codes, the question “how much better can we do, given a looser requirement on the rate?” is still open. In this work we answer this question for the additive Gaussian noise channel showing that superposition codebooks, optimal for a specific Gaussian broadcast channel (BC) are optimal in this sense. We further extend the question to the case of two MMSE constraints in lower SNRs, and show that three-layers superposition codebooks attain the maximum possible rate. This result can also be extended to the general K receiver case.

A similar question has been raised in the work of Bandemer and El Gamal [7], where they provide the rate-disturbance region: for any given rate that can be transmitted reliably to the intended receiver, what is the minimum possible disturbance that can be attained at some interfered user. In [7] the authors measure the disturbance using the mutual information between the codeword and the output at the interfered user, rather than the minimum possible MMSE, as done here. We further discuss and compare the two measures in our concluding remarks (section IV).

II. PROBLEM FORMULATION AND PRELIMINARY RESULTS

In this work we are looking at the transmission of codewords, of length n, through a discrete memoryless standard Gaussian channel:

\[ Y = \sqrt{\gamma} X + N \]  

where \( N \) is standard additive Gaussian noise. The codewords are constrained by the standard average power constraint: \( \forall x \in C_n \quad \frac{1}{n} \sum_{i=1}^{n} x_i^2 \leq 1 \), where \( C_n \) stands for a code of n-dimensional codewords. We examine codebooks designed for reliable transmission at \( \gamma = \text{snr}_2 \) (reliable decoding of the codeword from \( Y(\gamma = \text{snr}_2) \)). Our main interest will be in examining non-optimal codes, alternatively known as “bad” codes [2], defined using code-sequences, as follows:

Definition 1: A non-optimal code-sequence \( C = \{C_n\}_{n=1}^\infty \), for a channel with capacity \( C \), is a code-sequence with vanishing error probability and rate satisfying \( \lim_{n \to \infty} \frac{1}{n} \log M_n < C \) where \( M_n \) is the size of code \( C_n \).

Their associated MMSE defined as:

\[ \text{MMSE}^C(\gamma) = \lim_{n \to \infty} \text{MMSE}^C_n(\gamma) = \lim_{n \to \infty} \frac{1}{n} \text{Tr}(E_X(\gamma)) \]  

where \( E_X(\gamma) \) is the MMSE matrix when estimating the codeword \( X \) from the output of the channel \( Y = \sqrt{\gamma} X + N \). We further define the following for abbreviation:

\[ I(\gamma) = \lim_{n \to \infty} I_n(\gamma) = \lim_{n \to \infty} \frac{1}{n} I(X;Y(\gamma)) \]
Surely, for any such codes, the error probability for any \( \gamma > \text{snr}^2 \) is zero, when \( n \to \infty \), since reliable transmission is guaranteed at \( \text{snr} \). As a result MMSE\( ^E(\gamma) \) for these \( n \) is also zero. On the other hand, for \( \gamma < \text{snr}^2 \) the value of the error probability is not guaranteed to be any specific value. For an optimal code, it was shown in [3], that \( I(\gamma) \) for \( \gamma < \text{snr}^2 \) follows that of the Gaussian i.i.d. input and thus MMSE\( ^E(\gamma) \) is also known exactly and descends gradually according to \( \frac{1}{\sqrt{n}} \).

Our goal is to find a code-sequence which both attains a minimum required rate at \( \text{snr}^2 \), denoted as \( I(\text{snr}^2) \geq \frac{1}{2} \log (1 + \text{snr}^2) \), for some predetermined \( \alpha \in (0,1] \), and secondly, obtains the minimum possible MMSE at some \( \text{snr}^1 < \text{snr}^2 \). Clearly, if \( \alpha = 1 \) we have the maximum rate at \( \text{snr}^2 \), attainable only by an optimal code-sequence.

As mentioned above, this is a well understood case [3]. For \( \alpha \leq \text{snr}^2 \) we also have an obvious solution, since for these low rates we can obtain zero MMSE\( ^E(\text{snr}^1) \) by simply choosing an optimal code-sequence of rate \( \frac{1}{2} \log (1 + \text{snr}^1) \). Thus, the interesting range of parameters is \( \text{snr}^2 > \alpha < 1 \), for which the minimum attainable MMSE\( ^E(\text{snr}^1) \) is unknown. This problem is equivalent to maximizing the rate at \( \text{snr}^2 \), given some constraint on the maximum allowed MMSE at \( \text{snr}^1 \). In this formulation the obvious extension, motivated by multi-user interference channels, is to maximize the rate at \( \text{snr}^2 \), given two MMSE constraints. In this work we proceed with this view of the problem.

A. The I-MMSE approach

The approach used in order to provide insight into the above mentioned problem is the I-MMSE approach, this to say that we make use of the fundamental relationship between the mutual information and the MMSE in the Gaussian channel and its generalizations [8], [9]. Even though we are examining a scalar Gaussian channel, the \( n \)-dimensional version of this relationship is required since we are looking at the transmission of \( n \)-dimensional codewords through the channel. In our setting the relationship is as follows:

\[
I_n(\text{snr}) = \frac{1}{2} \int_0^{\text{snr}} \text{MMSE}^E(\gamma) \, d\gamma.
\]

(4)

Taking the limit of \( n \to \infty \) on both sides results with:

\[
I(\text{snr}) = \frac{1}{2} \int_0^{\text{snr}} \text{MMSE}^E(\gamma) \, d\gamma.
\]

(5)

The main property of the I-MMSE used for these proofs is an \( n \)-dimensional “single crossing point” property derived in [10] given here for completeness. This property is an extension of the scalar “single crossing point” property shown in [11]. In [10] the following function is defined for an arbitrary random vector \( X \):

\[
q_A(\mathbf{X}, \sigma^2, \gamma) = \sigma^2 \text{Tr} (A) - \text{Tr} (AEX(\gamma))
\]

(6)

where \( A \) is some \( n \times n \) general weighting matrix. The following theorem is proved in [10].

**Theorem 1 ([10]):** Let \( A \in \mathbb{S}_n^+ \) be a positive semidefinite matrix. Then, the function \( \gamma \mapsto q_A(\mathbf{X}, \sigma^2, \gamma) \), defined in (6), has no nonnegative-to-negative zero crossings and, at most, a single negative-to-nonnegative zero crossing in the range \( \gamma \in [0, \infty) \). Moreover, let \( \text{snr}^0 \in [0, \infty) \) be that negative-to-nonnegative crossing point. Then,

1. \( q_A(\mathbf{X}, \sigma^2, 0) \leq 0 \).
2. \( q_A(\mathbf{X}, \sigma^2, \gamma) \) is a strictly increasing function in the range \( \gamma \in [0, \text{snr}^0] \).
3. \( q_A(\mathbf{X}, \sigma^2, \gamma) > 0 \) for all \( \gamma \in [\text{snr}^0, \infty) \).
4. \( \lim_{\gamma \to \infty} q_A(\mathbf{X}, \sigma^2, \gamma) = 0 \).

In this work, the matrix \( A \) can be set to the identity matrix.

The above property is valid for all natural \( n \), thus we may also take \( n \to \infty \).

B. Superposition Coding

An important family of non-optimal codes, that is, a family of codes that do not attain the point-to-point capacity at \( \text{snr}^2 \), is that of Gaussian superposition codes which are optimal for a degraded Gaussian BC [12]. As will be shown in the sequel these codes are optimal MMSE-wise. The analysis of this family was done by Merhav et. al. in [13, section 5.3] from a statistical physics perspective. As noted in [13], the MMSE of this family of codebooks undergoes phase transitions, that is, it is a discontinuous function of \( \gamma \). The mutual information, \( I(\gamma) \), and MMSE\( ^E(\gamma) \) of this family of codebooks is known exactly and given in the next theorem. An example is depicted in Figure 1.

**Theorem 2 ([13] section 5.3):** A superposition codebook designed for \( (\text{snr}_0, \text{snr}_1, \ldots, \text{snr}_K) \) with the rate-splitting coefficients \((\beta_0, \ldots, \beta_{K-1})\) has the following \( I(\gamma) \):

\[
\begin{cases}
\frac{1}{2} \log (1 + \gamma), & \text{if } 0 \leq \gamma < \text{snr}_0 \\
\frac{1}{2} \log \left( \frac{1 + \text{snr}_0}{1 + \text{snr}_0 \beta_0} \prod_{j=1}^K \frac{1 + \text{snr}_j}{1 + \text{snr}_j \beta_j} \right) + \frac{1}{2} \log (1 + \beta_1 \gamma), & \text{if } \text{snr}_j \leq \gamma \leq \text{snr}_{j+1} \\
\frac{1}{2} \log \left( \frac{1 + \text{snr}_0}{1 + \text{snr}_0 \beta_0} \prod_{j=1}^{K-1} \frac{1 + \text{snr}_j}{1 + \text{snr}_j \beta_j} \right) + \frac{1}{2} \log (1 + \beta_{K-1} \text{snr}_K), & \text{if } \text{snr}_K \leq \gamma
\end{cases}
\]

and the following MMSE\( ^E(\gamma) \):

\[
\text{MMSE}^E(\gamma) = \begin{cases}
\frac{1}{1 + \beta_0 \gamma}, & 0 \leq \gamma < \text{snr}_0 \\
\frac{1}{1 + \beta_j \gamma}, & \text{snr}_j \leq \gamma \leq \text{snr}_{j+1} \\
0, & \text{snr}_2 < \gamma
\end{cases}
\]

(7)

III. MAIN RESULTS

The main result is given in the next theorem.

**Theorem 3:** Assuming \( \text{snr}_1 < \text{snr}_2 \) the solution of the following optimization problem,

\[
\max \; I(\text{snr}_2) \quad \text{s.t.} \quad \text{MMSE}^E(\text{snr}_1) \leq \frac{\beta}{1 + \beta \text{snr}_1}
\]

(8)

for some \( \beta \in [0,1] \), is the following

\[
\begin{align*}
I(\text{snr}_2) &= \frac{1}{2} \log (1 + \beta \text{snr}_2) + \frac{1}{2} \log \left( \frac{1 + \text{snr}_1}{1 + \beta \text{snr}_1} \right) \\
\end{align*}
\]

(9)
and is attainable when using the optimal Gaussian superposition codebook designed for \((\text{snr}_1, \text{snr}_2)\) with a rate-splitting coefficient \(\beta\).

**Proof:** It is simple to verify that the optimal Gaussian superposition codebook complies with the above MMSE constraint and attains the maximum rate. Thus, the focus of the remainder of the proof is on deriving a tight upper bound on the rate. We prove an equivalent claim, by assuming a code of rate \(R_c = \frac{1}{2} \log (1 + \alpha \text{snr}_2)\), designed for reliable transmission at \(\text{snr}_2\), and deriving a lower bound on \(\text{MMSE}^c(\text{snr}_1)\).

If \(\alpha \text{snr}_2 \leq \gamma \leq 1\) the lower bound is trivially zero using the optimal Gaussian codebook designed for \(\alpha \text{snr}_2\). This is equivalent to setting \(\beta = 0\) in which case we obtain the rate \(I(\text{snr}_2) = \frac{1}{2} \log (1 + \alpha \text{snr}_2)\) using an optimal Gaussian codebook (no superposition). Thus, we assume \(\gamma \leq \alpha \text{snr}_2\).

Using the trivial upper bound on \(I(\gamma) \leq \frac{1}{2} \log (1 + \gamma)\) (due to maximum entropy), we can lower bound the following difference, for any \(\gamma < \alpha \text{snr}_2\):

\[
I(\text{snr}_2) - I(\gamma) \geq I(\text{snr}_2) - \frac{1}{2} \log (1 + \gamma). \tag{10}
\]

Using the I-MMSE relationship (5), the above translates to the following inequality:

\[
\frac{1}{2} \int_\gamma^{\text{snr}_2} \text{MMSE}^c(\tau) \, d\tau \geq R_c - \frac{1}{2} \log (1 + \gamma) \geq \frac{1}{2} \log (1 + \alpha \text{snr}_2) - \frac{1}{2} \log (1 + \gamma). \tag{11}
\]

Defining \(d\) through the following equality:

\[
\frac{1}{2} \log (1 + \alpha \text{snr}_2) - \frac{1}{2} \log (1 + \gamma) = \frac{1}{2} \log (1 + d\text{snr}_2) - \frac{1}{2} \log (1 + d\gamma). \tag{12}
\]

it is simple to check that for \(\gamma < \alpha \text{snr}_2\), \(d\) is in the range of \((0, 1)\). Now we can continue with equation (11):

\[
\frac{1}{2} \int_\gamma^{\text{snr}_2} \text{MMSE}^c(\tau) \, d\tau \geq \frac{1}{2} \log (1 + d\text{snr}_2) - \frac{1}{2} \log (1 + d\gamma) = \frac{1}{2} \int_\gamma^{\text{snr}_2} \text{MMSE}_G(\tau) \, d\tau. \tag{13}
\]

where \(\text{MMSE}_G(\tau)\) is the MMSE assuming a Gaussian random variable with variance \(d\) through the additive Gaussian channel at \(\text{snr} = \tau\). The single crossing point property (Theorem 1, with \(A = I\)) tells us that \(\text{MMSE}^c(\tau)\) and \(\text{MMSE}_G(\tau)\) cross each other at most once, and after that crossing point \(\text{MMSE}_G(\tau)\) remains an upper bound. From the inequality in (13) we can thus conclude that the single crossing point, if exists, will occur in the region \((\gamma, \infty)\). Thus, for \(\gamma\) we have the following lower bound:

\[
\text{MMSE}^c(\gamma) \geq \frac{d(\gamma)}{1 + d(\gamma)\gamma} = \frac{\alpha \text{snr}_2 - \gamma}{\text{snr}_2 - \gamma} \frac{1}{1 + \gamma}. \tag{14}
\]

A similar derivation can be done for any \(\gamma < \alpha \text{snr}_2\), and will result with a different \(d(\gamma)\). Specifically for \(\gamma = \text{snr}_1\) we obtain

\[
\text{MMSE}^c(\text{snr}_1) \geq \frac{\alpha \text{snr}_2 - \text{snr}_1}{\text{snr}_2 - \text{snr}_1} \frac{1}{1 + \text{snr}_1}. \tag{15}
\]

Deriving \(\alpha\) as a function of the constraining \(\beta\), and substituting it in \(R_c = \frac{1}{2} \log (1 + \alpha \text{snr}_2)\) results with the superposition rate given in (9).

An interesting question to ask is whether there could be a different code that can attain maximum rate under the MMSE constraint at \(\text{snr}_1\) and also provide better MMSE for other values of \(\text{snr}\). The answer is to the negative, and is given in the next theorem, proof of which is omitted and can be found in [14].

**Theorem 4:** From the set of reliable codes of rate \(R_c = \frac{1}{2} \log (1 + \alpha \text{snr}_2)\), complying with the MMSE constraint at \(\text{snr}_1\), the superposition codebook provides the minimum MMSE for all \(\text{snr}\).

Two MMSE Constraints: We now wish to extend the result of Theorem 3 to the case of two constraints:

**Theorem 5:** Assuming \(\text{snr}_0 < \text{snr}_1 < \text{snr}_2\) the solution of the following optimization problem,

\[
\max I(\text{snr}_2) \quad \text{s.t.} \quad \text{MMSE}^c(\text{snr}_1) \leq \frac{\beta_1}{1 + \beta_1 \text{snr}_1}, \quad \text{MMSE}^c(\text{snr}_0) \leq \frac{\beta_0}{1 + \beta_0 \text{snr}_0} \tag{14}
\]

for some positive \(\beta_1, \beta_0\) such that \(\beta_1 + \beta_0 \leq 1\) and \(\beta_1 < \beta_0\), is the following

\[
I(\text{snr}_2) = \frac{1}{2} \log \left(1 + \beta_1 \text{snr}_2 \frac{1}{1 + \beta_1 \text{snr}_1 + \beta_0 \text{snr}_0} \frac{1 + \text{snr}_0}{1 + \beta_1 \text{snr}_1 + \beta_0 \text{snr}_0} \right)
\]

and is attainable when using the optimal three-layers Gaussian superposition codebook designed for \((\text{snr}_0, \text{snr}_1, \text{snr}_2)\) with rate-splitting coefficients \((\beta_0, \beta_1)\).
When $\beta_0 < \beta_1$ the first constraint can be removed and we return to the case of a single constraint given in Theorem 3.

Proof: It is simple to verify that the optimal Gaussian three-layers superposition codebook complies with the above MMSE constraints and attains the maximum rate. Thus, we need to derive a tight upper bound on the rate. Deriving the upper bound begins with the usage of Theorem 3 on $I(snr_1)$, when we take into consideration only the constraint on $\text{MMSE}_c(snr_0)$. This provides the following upper bound:

$$I(snr_1) \leq \frac{1}{2} \log \left( 1 + \beta_0 snr_1 \right) + \frac{1}{2} \log \left( 1 + \frac{snr_0}{1 + \beta_0 snr_0} \right)$$

(15)

On the other hand, using the I-MMSE approach we have:

$$I(snr_2) - I(snr_1) = \frac{1}{2} \int_{snr_1}^{snr_2} \text{MMSE}_c(\tau) d\tau \leq \frac{1}{2} \int_{snr_1}^{snr_2} \text{mmse}_C(\tau) d\tau$$

where $\text{mmse}_C(\tau)$ is the MMSE assuming a Gaussian random variable with variance $\beta_1$. This inequality is valid since according to the first constraint we have

$$\text{MMSE}_c(snr_1) \leq \frac{\beta_1}{1 + \beta_1 snr_1} = \text{mmse}_C(snr_1)$$

(16)

thus, according to the single crossing point property

$$\text{MMSE}_c(\tau) \leq \text{mmse}_C(\tau), \quad \forall \tau \geq \text{snr}_1$$

(17)

leading to the inequality in equation (16). Putting together (15) and (16) we obtain the desired upper bound.

IV. DISCUSSION AND CONCLUSIONS

These results provide the engineering insight to the good performance of the HK superposition scheme on the two-user interference channel, as shown in [6]. From our results, that show that the HK superposition scheme, is optimal MMSE-wise, we can conclude that one cannot construct better codes of the type defined in [1], [2] that will beat HK, through the use of estimation. Note that, as mentioned in [1, section V], the codes constructed there have an important complexity advantage over HK codes. Furthermore, keep in mind that the HK scheme is efficient in the two-user interference channel and a simple approach in the general $K$-user interference channel. Finally, we expect that the I-MMSE approach to shed further light on the capacity region of the two-user interference channel itself.

As mentioned in the introduction, Corollary 2 in [7] can also be derived directly from the I-MMSE formulation. Using our notation, we begin with the disturbance rate. Since, $0 \leq I(snr_1) \leq \frac{1}{2} \log \left( 1 + \alpha snr_1 \right)$ there exists an $\alpha^* \in [0, 1]$ such that $R_d = \frac{1}{2} \log \left( 1 + \alpha^* snr_1 \right)$. Thus, the averaged MMSE of the code crosses the MMSE of a scalar Gaussian input with power $\alpha^*$ in the range $[0, snr_1]$. Now, using the I-MMSE,

$$I(snr_2) = \frac{1}{2} \log \left( 1 + \alpha^* snr_1 \right) + \int_{snr_1}^{snr_2} \text{MMSE}_c(\gamma) d\gamma$$

$$\leq \frac{1}{2} \log \left( 1 + \alpha^* snr_2 \right)$$

(18)

where the last transition is due to the “single crossing point” property which ensures us that the MMSE of the scalar Gaussian input with power $\alpha^*$ will remain an upper bound on $\text{MMSE}_c(\gamma)$ in the range $[snr_1, \infty)$. Moreover, the above derivation does not indicate a superposition coding scheme, but rather a Gaussian code with reduced power of $\alpha^*$. Such a scheme, which attains the required minimum rate at $snr_2$, does not attain the minimum MMSE at $snr_1$. Extending to two mutual information disturbance constraint is trivial, as only the most constraining one determines the result. Thus, the different measurement of “disturbance” suggested here is conceptually different than the one suggested in [7].

Finally, the two MMSE constraints solution, presented here, can be directly extended to the $K$ MMSE constraint, as shown in [15].

ACKNOWLEDGMENT

This work has been supported by the Israel Science Foundation (ISF) and the CORNET Consortium. Ronit Bustin is supported by the Adams Fellowship Program of the Israel Academy of Sciences and Humanities.

REFERENCES


On the Gaussian Listening-Helper Source-Coding Problem

Shraga I. Bross*, and Haim Permuter†

* Engineering Department, Bar-Ilan University, Ramat Gan, 52900, Israel
Email: brosss@biu.ac.il
† Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Beer-Sheva
Email: haimp@bgu.ac.il

Abstract—In the Gaussian listening-helper source-coding problem Encoder 1 and Decoder 1 observe respectively a pair of i.i.d. correlated Gaussian sources and both wish to communicate the first source to Decoder 2 subject to a distortion constraint. Encoder 1 sends a message to both decoders and then Decoder 1 — the listening-helper — sends a message just to Decoder 2.

We derive an inner bound on the rate region of this problem by proposing a Gaussian achievable scheme. This inner bound coincides with the solution to the Gaussian optimization problem associated with a single-letter outer bound corresponding to this problem.

Index Terms—multi-terminal source coding, listening-helper problem.

I. INTRODUCTION

The Gaussian listening-helper source-coding problem that we consider is depicted in Fig. 1. Encoder 1 observes a scalar i.i.d. Gaussian source $X$ that is correlated with a scalar i.i.d. source $Y$ observed by Decoder 1. Encoder 1 wishes to communicate the source $X$ to within certain distortion constraint to Decoder 2 by sending a message at rate $R$ to both decoders. Decoder 1 (the listening-helper) is allowed to communicate to Decoder 2 by sending a message at rate $R_1$. Decoder 2 uses both messages to form its estimate of the source $X$. The goal is to determine the set of all rate pairs $(R, R_1)$ that satisfy the distortion constraint.

We derive an inner bound on the rate region for this source-coding problem by proposing a Gaussian scheme in which Encoder 1 vectorizes its observation using a Gaussian test channel and then compresses the quantized values using proper binning. Decoder 1 quantizes its observation using another test channel and finally, both Encoder 1 and Decoder 1 cooperate to provide Decoder 2 with the enumerated information it needs in order to resolve its ambiguity regarding the compressed information Encoder 1 had generated. Furthermore, we show that the Gaussian solution associated with an upper bound for our problem coincides with the proposed Gaussian achievable scheme.

The paper is organized as follows. In Section II we provide a formal definition of the Gaussian listening-helper source-coding model. In Section III we present our main result, while Section IV is devoted to the description of the main steps of the proof.

The work of S. Bross was supported by the Israel Science Foundation under Grant 497/09.

II. PROBLEM FORMULATION

The core problem we consider, the listening-helper source-coding problem, is described as follows. Let $\{(X_i, Y_i)\}_{i=1}^n$ be a pair of zero-mean Gaussian random vectors generated by i.i.d. drawings of a pair of jointly Gaussian random variables with variance $\sigma_X^2$ and $\sigma_Y^2 = a_i^2 \sigma_X^2 + \sigma_{W_i}^2$, respectively. With no loss of generality, we can write

$$Y_i = a_i X_i + N_i^{(y)},$$

where $N_i^{(y)} \sim N(0, \sigma_{W_i}^2)$ is independent of $X_i$.

Encoder 1 which observes $X^n$ sends a message to both decoders using an encoding function

$$f^{(n)}: X^n \rightarrow \{1, 2, \ldots, M_f^{(n)}\},$$

where $X^n$ denotes the set of $n$-vectors over $X$. Based on $f^{(n)}(X^n)$ and its observation $Y^n$ Decoder 1 sends a message to Decoder 2 using an encoding function

$$f_1^{(n)}: \{1, 2, \ldots, M_f^{(n)}\} \times Y^n \rightarrow \{1, 2, \ldots, M_1^{(n)}\}. \quad (3)$$

Decoder 2 then uses both received messages to estimate $X^n$ using a reproduction function

$$g^{(n)}: \{1, 2, \ldots, M_1^{(n)}\} \times \{1, 2, \ldots, M_1^{(n)}\} \rightarrow \hat{X}_2^n. \quad (4)$$

Definition 1: A rate-distortion triple $(R, R_1, D_2)$ is achievable for the Gaussian listening-helper source-coding problem if there exists a block length $n$, encoders $f^{(n)}$ and $f_1^{(n)}$, and a decoder $g^{(n)}$ such that,

$$R \geq \frac{1}{n} \log M_f^{(n)}$$

$$R_1 \geq \frac{1}{n} \log M_1^{(n)}$$

$$D_2 \geq \frac{1}{n} \sum_{i=1}^n \mathbb{E} [(X_i - \hat{X}_2,i)^2]. \quad (5)$$

Let $\hat{R}(D_2)$ be the closure of the set of achievable rate-distortion triples. The rate region $R(D_2)$ for the Gaussian listening-helper source-coding problem is defined by

$$R(D_2) = \{(R, R_1): (R, R_1, D_2) \in \hat{R}(D_2)\}. \quad (6)$$

24
Since we are interested in the Mean-Square-Error (MSE) distortion constraint we can restrict the reproduction function to be the MMSE estimate of $X^n$ given $f^{(n)}(X^n)$ and $f^{(n)}_1(f^{(n)}(X^n), Y^n)$, that is
\[
\hat{X}_n = \mathbb{E}[X^n | f^{(n)}(X^n), f^{(n)}_1(f^{(n)}(X^n), Y^n)].
\]

III. MAIN RESULTS

Our main result is a characterization of an inner bound on the rate-region for the Gaussian listening-helper source-coding problem.

**Proposition 1:** For a positive number $D_2 \leq \tilde{\sigma}_X^2$, the rate-region $R(D_2)$ contains the rate region $R_G(D_2)$ defined by
\[
R_G(D_2) = \{ (R, R_1) : R \geq \frac{1}{2} \log \left[ \frac{\sigma^2_X}{\tilde{\sigma}_W^2} \right] + \frac{\sigma^2 \sigma^2_W + \tilde{\sigma}_W^2}{\sigma^2_X + \tilde{\sigma}_W^2} \}
\]

\[R_1 \geq \frac{1}{2} \log \left[ \frac{\sigma^2 \sigma^2_W + \tilde{\sigma}_W^2}{\sigma^2_X + \tilde{\sigma}_W^2} \right],\]
\[R + R_1 \geq \frac{1}{2} \log \left[ \frac{\sigma^2_X}{\tilde{\sigma}_W^2} \frac{\sigma^2 \sigma^2_W + \tilde{\sigma}_W^2}{\sigma^2_X + \tilde{\sigma}_W^2} \right],\]
\[D_2 = \frac{\sigma^2 \sigma^2_W + \tilde{\sigma}_W^2}{\sigma^2_X + \tilde{\sigma}_W^2}.\]

A. A Gaussian achievable scheme

A natural Gaussian achievable scheme for the listening-helper problem is described as follows. Let $S(U, V)$ be the set of zero-mean jointly Gaussian random variables $U$ and $V$ such that

1) $U \circlearrowleft X \circlearrowleft Y$ and $X \circlearrowleft UY \circlearrowleft V$ are Markov chains.

2) $\sigma^2_{X|U,V} \leq D_2$.

For any $(U, V) \in S(U, V)$ and a large block-length $n$:

- Generate $\epsilon_{n[U,X,U]+\epsilon}$, $\epsilon > 0$ independent codewords $u(1), \ldots, u(\epsilon_{n[U,X,U]+\epsilon})$ randomly according to $p_U$. Assign these codewords universally into $e^{nR}$ bins. Label them $u(j, l)$, $j \in \{1, \ldots, e^{nR}\}$, $l \in \{1, \ldots, e^{nR}\}$. Share the codebook and its partition between Encoder 1 and both decoders.

- For each $u(j, l)$ generate $e^{n[R_1+\epsilon]}$, $\epsilon_1 > 0$ independent codewords randomly according to $p_{U|V}$. Label the codewords by $v(j, l, k)$, $k \in \{1, \ldots, e^{n[R_1+\epsilon]}\}$ and share this codebook between Decoder 1 and Decoder 2.

Given a source and a side-information sequence pair $(x, y)$:

1) Encoder 1 looks for the codeword $u(j, l)$ that is jointly typical with $x$, and sends its index $l$ to both decoders.

2) Encoder 1 and Decoder 1 send together the enumeration index $j$ to Decoder 2.

3) Decoder 1 looks for the codeword $v(j, l, k)$ that is jointly typical with $(u(j, l), y)$, and sends the index $k$ of that codeword to Decoder 2.

Decoder 1 can recover $u$ with high probability provided that
\[R \geq I(X; U|Y).\]

Decoder 1 can compute $v$ with high probability provided that
\[R_1 \geq I(Y; V|U),\]

and taking into account that Encoder 1 and Decoder 1 join efforts in sending the index $j$ to Decoder 2 we obtain
\[R + R_1 \geq I(X; U) + I(Y; V|U).\]

Decoder 2 computes the MMSE estimate of the source sequence $X^n$ given $(U^n, V^n)$ and condition 2) above guarantees that the estimate meets the desired distortion.

The following lemma characterizes an achievable rate-region based on this scheme.

**Lemma 1:** A Gaussian achievable scheme as above which satisfies the long Markov chain $U \circlearrowleft X \circlearrowleft Y \circlearrowleft V$ achieves $R_G(D_2)$. Moreover, the Gaussian solution associated with an upper bound for our main problem coincides with $R_G(D_2)$.

In Section IV.B we show that for Gaussian $(U, V) \in S(U, V)$ such that $U \circlearrowleft X \circlearrowleft Y \circlearrowleft V$ the rate region associated with (7)-(9) equals $R_G(D_2)$.

IV. PROOFS

The source-coding problem considered next will be used to show that the Gaussian solution associated with an upper bound for our main problem equals $R_G(D_2)$.

**A. A source-coding sub-problem**

Let $(X_i, Y_i, Z_i)_{i=1}^n$ be a sequence of independent drawings of a triplet of zero-mean jointly Gaussian random variables with covariance $K_{(X,Y,Z)}$. An encoder observes $Y^n$ while both the encoder and the decoder have access to the side-information $Z^n$. The encoder sends a message to the decoder using an encoding function
\[f^{(n)}_E : Y^n \times Z^n \to \{1, 2, \ldots, M^{(n)}\}.\]

Based on $f^{(n)}_E(Y^n, Z^n)$ and its side-information $Z^n$ the decoder forms an estimate for $X^n$ using a reproduction function
\[g^{(n)}_D : \{1, 2, \ldots, M^{(n)}\} \times Z^n \to \hat{X}^n.\]
This system is shown in Fig. 2.

Definition 2: A rate-distortion pair \((R_E, D)\) is achievable for the Gaussian source-coding sub-problem if there exists a block length \(n\), an encoder \(f_E^{(n)}\) and a decoder \(\hat{g}_D^{(n)}\) such that,

\[
R_E \geq \frac{1}{n} \log M^{(n)}
\]

\[
D \geq \frac{1}{n} \sum_{i=1}^{n} \mathbb{E} \left[ (X_i - \hat{X}_i)^2 \right].
\]

(12)

Here also since we are interested in the MSE distortion constraint we can restrict the reproduction function to be the MMSE estimate of \(X^n\) given \(f_E^{(n)}(Y^n, Z^n)\) and \(Z^n\), that is,

\[
\hat{X}_n = \mathbb{E} \left[ X^n | f_E^{(n)}(Y^n, Z^n), Z^n \right].
\]

For a given distortion \(D > 0\), define \(P_0(D)\) as the set of laws \(p_{XYZV}(x, y, z, v)\), \(x, y, z, v \in \mathbb{R}\), such that the marginal law \(f_{XYZV}dv\) identifies with \(p_{XYZ}\), and

\[
\mathbb{E} \left[ (X - \mathbb{E}(X|V, Z))^2 \right] \leq D.
\]

Then, by a modification of [4, sec. 6.1.1], we have that

\[
R_E \geq \min_{p_{XYZV} \in P_0(D)} I(Y; V|Z). \quad (13)
\]

Consequently, define an optimization problem \(P\) as follows

\[
\max \quad h(Y|VZ)
\]

subject to

\[
D \geq \sigma^2_{Y|VZ}, \quad \sigma^2_{Y|VZ}
\]

then let \(P_G[D, (X, Y, Z), \sigma^2_{Y|VZ}]\) be the corresponding optimization problem assuming \(V\) is Gaussian and denote by \(\varphi\left(P_G[D, (X, Y, Z), \sigma^2_{Y|VZ}]\right)\) the conditional variance \(\sigma^2_{Y|VZ}\) associated with the optimal value of \(P_G\).

Consequently, \(P_G\) may be expressed as

\[
R_{Y|Z}(D) = \min_{p_{XYZV} \in P_0(D)} \frac{1}{2} \log \frac{\sigma^2_{Y|VZ}}{\sigma^2_{Y|VZ}}.
\]

(15)

On the other hand, when \(Z^n\) is available just at the decoder, i.e. \(f_E^{(n)} = f_E^{(n)}(Y^n)\), by a modification of [2], we have that

\[
\hat{R}_E \geq R_{Y|Z}^{WZ}(D) = \min I(Y; V|Z),
\]

(16)

where the minimization is over the set of laws \(P(D)\) defined by

\[
p_{XYZV}(x, y, z, v) = p_{XYZ}(x, y, z)p_{V|Y}(v|y).
\]

(17)

Consequently,

\[
R_{Y|Z}^{WZ}(D) = \min_{p_{XYZV} \in P(D)} \frac{1}{2} \log \frac{\sigma^2_{Y|Z}}{\sigma^2_{Y|VZ}}
\]

subject to

\[
D \geq \sigma^2_{Y|VZ}, \quad V^q \circ Y \circ (X, Z).
\]

(18)

In [3, Section 3] the author shows that, when the encoder observes the source sequence \(X^n\) – i.e. \(Y = X\), both optimization problems (15) and (18) have the same solution and, in fact, the additional Markov relation \(Z \circ X \circ V^q\) in (18) characterizes the optimal solution of (15). The following lemma asserts that the same is true when the encoder observes a correlated random variable \(Y\) which satisfies a Markov relation.

Lemma 2: For jointly Gaussian random variables \((X, Y, Z)\) satisfying \(Z \circ X \circ Y\) both optimization problems (15) and (18) have the same solution and, in fact, the additional Markov relation in (18) characterizes the optimal solution of (15).

Proof: Omitted.

B. The Gaussian solution associated with an outer bound

Lemma 3: (Single-letter outer bound) If the rate-distortion triple \((R, R_1, D_2)\) is achievable then there exist random variables \((U, V)\) satisfying the Markov relation \(U \circ X \circ Y\), such that

\[
R + R_1 \geq I(X; U) + I(Y; V|U)
\]

\[
R \geq I(X; U|Y), \quad R_1 \geq I(Y; V|U)
\]

\[
D_2 \geq \sigma^2_{Y|UV}.
\]

(19)

Proof: Assume that \((R, R_1, D_2)\) is achievable. Let \(T = f^{(n)}(X^n)\) and \(T_1 = f_1^{(n)}(f^{(n)}(X^n), Y^n)\), then the rate \(R\) can be lower bounded as follows

\[
nR \geq \log M^{(n)} \geq H(T) \geq H(T|Y^n)
\]

\[
= I(X^n; T|Y^n) = \sum_{k=1}^{n} I(X_k; T|Y^n, X^{k-1})
\]

\[
(q) \quad \sum_{k=1}^{n} I(X_k; T|Y^n, X^{k-1}|Y_k)
\]

\[
\geq \sum_{k=1}^{n} I(X_k; U_k|Y_k),
\]

(20)

where \(X^{k-1} \triangleq (X_1, \ldots, X_{k-1}), U_k \triangleq (T, Y^{k-1}), Y^n \triangleq (Y_1, \ldots, Y_{k-1}, Y_{k+1}, \ldots, Y_n)\), and the equality \((q)\) follows by the Markov relation \(X_k \circ Y_k \circ Y^n \circ X^{k-1}\).

Let \(V_k \triangleq T_k\), then the rate \(R_1\) can be lower bounded as follows

\[
nR_1 \geq \log M^{(1)} \geq H(T_1) \geq H(T_1|T)
\]

\[
= I(Y^n; T_1|T) = \sum_{k=1}^{n} I(Y_k; T_1|T, Y^{k-1})
\]

\[
= \sum_{k=1}^{n} I(Y_k; V_k|U_k).
\]

(21)
The sum-rate can be lower bounded as follows
\[
n(R + R_2) \geq \log M_n + \log M_1^{(n)}
\]
\[
\geq H(T) + H(T_1) \geq H(T; T_1)
\]
\[
= H(T) + H(T_1|T)
\]
\[
= H(T|Y^n) + I(T; Y^n) + H(T_1|T)
\]
\[
= H(T|Y^n) + H(T_1|T) + \sum_{k=1}^n I(Y_k; T|Y^{k-1})
\]
\[(b) \sum_{k=1}^n [I(X_k; U_k|Y_k) + I(Y_k; V_k|U_k)]
\]
\[= \sum_{k=1}^n [I(X_kY_k; U_k) + I(Y_k; V_k|U_k)]
\]
\[= \sum_{k=1}^n [I(X_k; U_k) + I(Y_k; V_k|U_k)].
\]

Here

(b) follows since \(Y^n\) is a memoryless sequence; and

(c) follows since \(Y_k \| X_k \| U_k\) is a Markov chain.

By (2) and the memoryless property of the sequence \((X_k, Y_k), k = 1, \ldots, n\) one can verify the Markov relation \(U_k \| X_k \| Y_k\) which implies the Markov relation \(U \| X \| Y\).

The combination of (20), (21) and (22) together with the latter Markov relation establish the desired single-letter lower bound.

Define next an optimization problem \(P_G\) over the conditional variance \(\sigma_{XYUV}^2\) and \(V^G\) as follows

\[
\min_{V^G, \sigma_{XYUV}^2} \left\{ -\frac{1}{2} \log \frac{\sigma_X^2}{\sigma_{XYUV}^2} - \frac{\sigma_{YUV}^2}{\sigma_{XYUV}^2} \right\}
\]

subject to

\[
R \geq \frac{1}{2} \log \frac{\sigma_X^2}{\sigma_{XYUV}^2}, \quad R_1 \geq \frac{1}{2} \log \frac{\sigma_{YUV}^2}{\sigma_{XYUV}^2}
\]

\[
D_2 \geq \sigma_{XYUV}^2 \quad \text{and} \quad U^G \| X \| Y.
\]

\(P_G\) is the restriction of the outer bound (19) to a Gaussian law and on account of the Markov relation \(U^G \| X \| Y\) and (1) it may be expressed as

\[
\min_{V^G, \sigma_{XYUV}^2} \left\{ -\frac{1}{2} \log \frac{\sigma_X^2}{\sigma_{XYUV}^2} - \frac{\sigma_{YUV}^2}{\sigma_{XYUV}^2} \right\}
\]

subject to

\[
R \geq \frac{1}{2} \log \frac{\sigma_X^2}{\sigma_{XYUV}^2}, \quad R_1 \geq \frac{1}{2} \log \frac{\sigma_{YUV}^2}{\sigma_{XYUV}^2}
\]

\[
D_2 \geq \sigma_{XYUV}^2 \quad \text{and} \quad U^G \| X \| Y.
\]

Lemma 4: In the optimization problem \(P_G\) the conditional variance \(\sigma_{XYUV}^2\) is monotonically increasing with \(\sigma_{XYUV}^2\).

Proof: By the Markov relation \(U^G \| X \| Y\) and (1) we may assume that

\[
U^G = X + N^{(u)},
\]

where \(N^{(u)} \sim N(0, \sigma_{W^u}^2)\) is independent of \(X\). Then a straightforward computation yields the relation

\[
\sigma_{XYUV}^2 = \frac{1}{\bar{p}_{W^u}^2} \left[ \frac{\sigma_{W^u}^2}{\sigma_{XYUV}^2} + \sigma_{YUV}^2 \right]^{-1}.
\]

We may express \(P_G\) as a double optimization problem

\[
\min_{\sigma_{XYUV}^2, v^G} \left\{ -\frac{1}{2} \log \frac{\sigma_X^2}{\sigma_{XYUV}^2} \right\}
\]

subject to

\[
R \geq \frac{1}{2} \log \frac{\sigma_X^2}{\sigma_{XYUV}^2}, \quad R_1 \geq \frac{1}{2} \log \frac{\sigma_{YUV}^2}{\sigma_{XYUV}^2}
\]

\[
D_2 \geq \sigma_{XYUV}^2 \quad \text{and} \quad U^G \| X \| Y \| V^G.
\]

Let \((X, Y, U^G)\) be jointly Gaussian as per (1) and (23) then the choice of \(V^G = Y + N^{(v)}\) where \(N^{(v)} \sim N(0, \sigma_{W^v}^2)\) is independent of \(X, N^{(u)}, N^{(w)}\) establishes that the optimal solution to \(P_G\) achieves \(R_G(D_2)\).

References


Upper-Bounding Rate Distortion Functions based on the Minimum Mean-Square Error

Meik Dörpinghaus*, Matthias Rüngeler**, and Rudolf Mathar*

*Institute for Theoretical Information Technology, RWTH Aachen University,
52056 Aachen, Germany, {doerpinghaus, mathar}@ti.rwth-aachen.de
**Institute for Communication Systems and Data Processing, RWTH Aachen University,
52056 Aachen, Germany, ruengeler@ind.rwth-aachen.de

Abstract—We derive a new upper bound on the rate distortion function for arbitrary memoryless sources, which is based on the relation between mutual information and minimum mean-square error discovered by Guo et al. This upper bound is in general tighter than the well known upper bound given by the rate distortion function of a Gaussian source with an equal variance found by Shannon and becomes tight for Gaussian sources. We evaluate the new upper bound for various source distributions and compare it to the Shannon lower bound and upper bound and to the rate distortion function calculated with the Blahut-Arimoto algorithm. This shows that the new upper bound is quite tight.

I. INTRODUCTION AND SETUP

It is well known that the rate distortion function states which rate R is at least required to describe a source X, such that it can be reconstructed up to a given distortion D for a specific distortion measure.

Let X be a random source that generates independent identically distributed (i.i.d.) symbols with respect to the probability distribution function (PDF) p(x), i.e., X ∼ p(x). When encoding, source sequences X^n consisting of n source symbols are mapped onto indices

f_n : X^n → {1, 2, . . . , 2^{nR}}.

Here, R is the rate of the encoded sequence. By this mapping the source sequences are represented by nR bits. Based on this index representation the decoder is able to generate an estimate \( \hat{X}^n \) of the source sequence X^n, i.e., the decoding function is

g_n : {1, 2, . . . , 2^{nR}} → \hat{X}^n.

The encoding and decoding given by the functions f_n and g_n is often referred to as a (2^{nR}, n)-rate distortion code.

The question is: How close can the source sequences X^n be reconstructed when they are encoded with rate R? I.e., how large is the expected distortion D between the source sequences X^n and their reconstruction \( \hat{X}^n \) with

\[ D = \mathbb{E}[d(X^n, g_n(f_n(X^n)))] \]

where

\[ d(x^n, \hat{x}^n) = \|x^n - g_n(f_n(x^n))\| = \frac{1}{n} \sum_{i=1}^{n} d(x_i, \hat{x}_i) \]

is the distortion between the source sequence x^n and its reconstruction \( \hat{x}^n \). In addition, d(x_i, \hat{x}_i) is the distortion between the individual source symbols x_i and their estimates \( \hat{x}_i \).

The distortion D and the rate R form a rate distortion pair (R, D). A rate distortion pair is achievable if a sequence of (2^{nR}, n)-rate distortion codes \( (f_n, g_n) \) exists such that

\[ \lim_{n \to \infty} \mathbb{E}[d(X^n, g_n(f_n(X^n)))] \leq D. \]

The rate distortion function R(D) is defined as the minimum rate R that is required to encode the source X for a given distortion D. In [1], [2] Shannon has shown that for an i.i.d. source X with distribution p(x) and bounded distortion function \( d(x^n, \hat{x}^n) \) the rate distortion function is given by

\[ R(D) = \min_{p(\hat{x}|x) \in S} I(X; \hat{X}) \]

where \( I(X; \hat{X}) \) is the mutual information of X and \( \hat{X} \) and

\[ S = \left\{ p(\hat{x}|x) \mid \int \int p(\hat{x}|x)p(x)d(x, \hat{x})dx d\hat{x} \leq D \right\} \]

see also [3, Th. 10.2.1]. Minimization in (5) is hence over all p(\( \hat{x}|x) \) for which p(x, \( \hat{x} \)) fulfills the distortion constraint D.

For certain distortion measures \( d(x, \hat{x}) \) and source distributions p(.), R(D) is known. E.g., it is a well known result that for a Gaussian source X ∼ N(0, \( \sigma_X^2 \)) and the mean-square error as distortion measure, i.e., \( E[|X - \hat{X}|^2] \leq D \), the rate distortion function is given by\(^1\), see, e.g., [3, Theorem 10.3.2]

\[ R(D) = \left\{ \begin{array}{ll}
\frac{1}{2} \log \left( \frac{\sigma_X^2}{D} \right) & \text{for } 0 \leq D \leq \sigma_X^2, \\
0 & \text{for } D > \sigma_X^2.
\end{array} \right. \]

While for a few combinations of distortion measures and source distributions the rate distortion functions are known, they are in general unknown. Existing bounds like the Shannon lower bound [2] and the upper bound given by the fact that the rate distortion function for a Gaussian source in (7) is an upper bound to the rate distortion function of arbitrarily distributed sources X with the same variance \( \sigma_X^2 \), see [1], are in general not tight. This is the motivation for the present work. We derive a new upper bound on the rate distortion function for the mean-square error as distortion measure and arbitrary source distributions. This upper bound is based on the relation between the minimum mean-square error (MMSE) and the mutual information given in [4]. Recently, in [5] this relation has already been applied in the context of rate distortion theory. However, in contrast to the present work [5] does not consider minimization over the reproduction distribution for the study of R(D).

\(^1\)All logarithms are to the base e and, thus, all rates are in nats.
II. NEW UPPER BOUND ON R(D)

Theorem 1. Let X be a random source that generates i.i.d. symbols according to $X \sim p(x)$ with $\text{var}(X) = \sigma_X^2$. The minimal required rate $R$ such that $\mathbb{E}[(X - \hat{X})^2] \leq D$ is upper-bounded by

$$R(D) \leq \frac{1}{2} \log \left( 1 + \frac{\sigma_X^2}{\sigma_N^2} \right) - \frac{1}{2} \int_0^{\sigma_N^2} \left( \frac{\sigma_X^2}{1 + \sigma_X^2 \gamma} - \text{mmse}(\gamma) \right) d\gamma$$

where $\text{mmse}(\gamma)$ is the minimum mean-squared error when estimating $X$ disturbed by additive Gaussian noise $N$, i.e.

$$\text{mmse}(\gamma) = \mathbb{E} \left[ (X - \mathbb{E}[X] + N)^2 \right]$$

and where $\gamma$ is the normalized SNR, i.e., $\gamma = \frac{\text{var}(N)}{\sigma_X^2}$. Furthermore, $\sigma_{X|N}^2$ is related to the distortion $D$ by

$$D = \text{mmse}(1/\sigma_{X|N}^2)$$

$$= \frac{\sigma_X^2 \sigma_N^2}{\sigma_X^2 + \sigma_N^2} - \frac{d}{d(1/\sigma_{X|N}^2)} D(p_{X+N} \| p_{X'+N}).$$

Here $D(p_{X+N} \| p_{X'+N})$ is the Kullback-Leibler divergence between the PDF of $X+N$ and of $X'+N$ where $X'$ is a Gaussian random variable with the same mean and variance as $X$.

Theorem 1 also has the following interpretation. The rate distortion function $R(D)$ for an arbitrarily distributed source $X$ with variance $\sigma_X^2$ is upper-bounded by the mutual information of an AWGN channel with the input $X - \mathbb{E}[X]$ and a noise variance $\sigma_N^2$, which is chosen such that the distortion is equal to $D = \text{mmse}(1/\sigma_{X|N}^2)$. As (9), i.e., the upper bound on $R(D)$, corresponds to the mutual information of an AWGN channel, $\text{mmse}(1/\sigma_{X|N}^2)$ and hence $D$ is given by two times the derivative of (9) with respect to $1/\sigma_{X|N}^2$ yielding (12), see [4] for the relation between mutual information and MMSE.

The representations in (9) and (12) using the Kullback-Leibler divergence allow an easy numerical evaluation of the upper bound on the rate distortion function. To calculate $D(p_{X+N} \| p_{X'+N})$ the PDF $p_{X+N}$ is required. As $X$ and the additive noise $N$ are independent, $p_{X+N}$ is given by the convolution of $p_X$ and $p_N = N(0, \sigma_N^2)$. In case $p_{X+N}$ cannot be obtained in closed-form, a numerical convolution is required to calculate the upper bound on $R(D)$. Thus, evaluation of (9) gives an upper bound on the rate $R$ for a given $\sigma_{X|N}^2$. The corresponding distortion $D$ can be calculated from (12). Hence, to evaluate the upper bound on the rate distortion function pairs of the upper bound on $R$ given by (9) and $D$ given by (12) have to be calculated by varying the parameter $\sigma_{X|N}^2$.

III. PROOF OF THEOREM 1

As the rate distortion function is independent of the mean $\mathbb{E}[X]$ of the source, i.e., the rate distortion functions $R(D)$ for the sources $X$ and $X - \mathbb{E}[X]$ are equal, in the following proof we assume that $\mathbb{E}[X] = 0$ without loss of generality.

With (5) the rate distortion function is given by

$$R(D) = \min_{p(\hat{x}|x) \in S} \mathcal{I}(X; \hat{X})$$

where the minimization is over all $p(\hat{x}|x)$ such that

$$\mathbb{E}[(X - \hat{X})^2] \leq D$$

i.e., all $p(\hat{x}|x)$ in the set $S$ in (6) with $d(x, \hat{x}) = (x - \hat{x})^2$. As $R(D)$ is monotonically decreasing in $D$, all $p(\hat{x}|x)$ being solutions of (13) fulfill (14) with equality. Obviously, any choice $p(\hat{x}|x) \in S$ different from the optimal $p(\hat{x}|x)$ which minimizes $\mathcal{I}(X; \hat{X})$ yields an upper bound on $R(D)$:

$$R(D) \leq \mathcal{I}(X; \hat{X}).$$

Thus, we construct an upper bound on the rate distortion function $R(D)$ by choosing some $p(\hat{x}|x)$ satisfying

$$\mathbb{E}[(X - \hat{X})^2] = D$$

for which $\mathcal{I}(X; \hat{X})$ can be easily evaluated and at the same time is a sufficiently tight upper bound. Therefor, we define

$$\hat{X} = \mathbb{E}[X|Y]$$

where $N$ is zero-mean additive white Gaussian noise with variance $\sigma_N^2$. Furthermore, let

$$\hat{X} = \mathbb{E}[X|Y]$$

be the MMSE estimate of $X$ from $Y$. Now, the variance $\sigma_{\hat{X}|Y}^2$ has to be chosen such that (16) is fulfilled. As the MMSE estimate $\hat{X}$ is a sufficient statistic of $Y$, see [6], it does not change the mutual information [3, Sect. 2.9] and it holds that

$$\mathcal{I}(X; \hat{X}) = \mathcal{I}(X; Y).$$

Thus, an upper bound on $R(D)$ is given by the mutual information of an AWGN channel with the source $X$ at its input and the variance of the additive Gaussian noise $\sigma_N^2$ depending on the distortion $D$ and implicitly given by (10) and (11).

Hence, to prove Theorem 1 it remains to show that (9) is equal to $\mathcal{I}(X; Y)$, i.e., the mutual information of the AWGN channel in (17) with the source $X$ at its input and noise variance $\sigma_N^2$. Therefore, we rewrite $\mathcal{I}(X; Y)$ as follows, see [7]:

$$\mathcal{I}(Y; X) = h(Y) - h(Y|X) = h(Y) - h(N)$$

$$- = \int p_Y(y) \log \left( \frac{p_Y(y)}{p_{Y|X}(y)} \right) dy - h(N)$$

$$- = \int p_Y(y) \log \left( \frac{p_Y(y)}{p_{Y|X}(y)} \right) dy - \int p_Y(y) \log \left( \frac{p_Y(y)}{p_{Y|X}(y)} \right) dy$$

$$- h(N)$$

$$= \int p_Y(y) \log \left( \frac{p_Y(y)}{p_{Y|X}(y)} \right) dy - \int p_Y(y) \log \left( \frac{p_Y(y)}{p_{Y|X}(y)} \right) dy$$

$$- h(N)$$

$$= h(Y) - D(p_Y \| p_{Y|X}) - h(N)$$

$^2$Note that [7] states that inequality (8) in Theorem 1 holds with equality. However, this statement is wrong in general. The erroneous assumption in [7] is that the information rate of a system with optimal source and channel coding is equal to the information rate of an AWGN channel where the source symbols are transmitted without coding. It can be shown that this statement does not hold in general and, therefore, (8) does not hold with equality in general.
where $h(\cdot)$ denotes the differential entropy, $p_Y(y)$ is the PDF of $Y$, and $p_Y(y)$ is the PDF of a corresponding zero-mean Gaussian variable with the same variance
\begin{equation}
\sigma_Y^2 = \sigma_X^2 + \sigma_N^2.
\end{equation}
Moreover, $D(p_Y \parallel p_Y')$ is the Kullback-Leibler divergence between $p_Y(y)$ and $p_Y'(y)$. Furthermore, (20) holds because
\begin{equation}
- \int p_Y(y) \log(p_Y(y)) \, dy = - \int p_Y(y) \log \left( \frac{\exp \left( \frac{-y^2}{2\sigma_Y^2} \right)}{\sqrt{2\pi\sigma_Y^2}} \right) \, dy
= \frac{1}{2} \log \left( 2\pi e \sigma_Y^2 \right) = - \int p_Y(y) \log(p_Y(y)) \, dy.
\end{equation}
As $Y$ and $N$ are Gaussian, (21) is given by
\begin{equation}
\mathcal{I}(Y; X) = \frac{1}{2} \log \left( 2\pi e \sigma_Y^2 \right) - \frac{1}{2} \log \left( 2\pi e \sigma_X^2 \right) - D(p_X \parallel p_Y')
= \frac{1}{2} \log \left( 1 + \frac{\sigma_Y^2}{\sigma_X^2} \right) - D(p_X \parallel p_Y')
\end{equation}
where for (24) we have used (22). This shows that (9) is the mutual information of an AWGN channel with noise variance $\sigma_N^2$ and an arbitrarily distributed source $X$ with zero-mean and variance $\sigma_X^2$ at its input.

The alternative representation of the upper bound on $R(D)$ in (8) holds, as $D(p_Y \parallel p_Y')$ in (24) is given by
\begin{equation}
D(p_Y \parallel p_Y') = h(Y') - h(Y)
= \mathcal{I}(Y'; X) + h(Y'|X') - \mathcal{I}(Y; X) - h(Y|X)
= \mathcal{I}(Y'; X') - \mathcal{I}(Y; X) + h(X' + N|X') - h(X + N|X)
\end{equation}
\begin{equation}
= \frac{1}{2} \int_0^{\infty} \left( \frac{\sigma_Y^2}{1 + \sigma_X^2} - \text{mmse}(\gamma) \right) \, d\gamma
\end{equation}
where $X'$ is a zero-mean Gaussian random variable with variance $\sigma_X^2$, and $X$ is the input to the AWGN channel in (17), this results in the zero-mean Gaussian output $Y'$, yielding (26). Moreover, (25) and (27) follow from the results given in [8].

IV. PROPERTIES OF THE UPPER BOUND

A. Gaussian Sources

For a Gaussian source $X$, $\text{mmse}(\gamma)$ in the upper bound (8) becomes $\sigma_X^2/(1 + \sigma_X^2 \gamma)$ and, thus, the integral in (8) is zero. Hence, the upper bound of the AWGN channel. However, for a Gaussian source the capacity of the AWGN channel and the rate distortion function $R(D)$ in (7) are equal, as (11) becomes equal to
\begin{equation}
D = \frac{\sigma_X^2 \sigma_N^2}{\sigma_X^2 + \sigma_N^2}.
\end{equation}
which follows from (12) as the Kullback-Leibler divergence in (12) is zero for Gaussian inputs. Thus, for a Gaussian source the upper bound on $R(D)$ in (8) and (9) is tight.

This discussion also shows another well known fact. In case we estimate the Gaussian source $X$ after transmission over an AWGN channel based on $Y$ using an MMSE estimator, the MMSE is given by $D$ in (28), see (10) to (12). To achieve this distortion no code at all is required. On the other hand, we can source encode the source $X$ using a rate-distortion code, then use a channel code to transmit it reliably over the AWGN channel. Following Shannon’s source channel coding separation theorem [1], this separation of source and channel coding is optimal. At the receiver the signal can be channel and source decoded. To achieve a maximum distortion $D$, we source encoder has to encode the source with rate $R(D)$ given in (7). For reliable transmission over the AWGN channel its capacity has to correspond to $R(D)$, i.e., for $0 \leq D \leq \sigma_X^2$,
\begin{equation}
\frac{1}{2} \log \left( 1 + \frac{P}{\sigma_N^2} \right) \geq R(D) \geq \frac{1}{2} \log \left( 1 + \frac{\sigma_X^2}{\sigma_N^2} \right)
\end{equation}
where $P$ is the required transmit power. Obviously $P = \sigma_X^2$. However, this means that for a given transmit power, there is no advantage of the use of source and channel coding when a Gaussian source has to be transmitted over an AWGN channel. Directly transmitting uncoded source symbols over the AWGN channel yields the same distortion $D$ as perfect source and channel coding. In addition, the latter one implies an infinite delay. This is a well known result, see e.g. [9], [10].

B. Gap to Shannon’s Lower Bound

Shannon gave a lower bound on $R(D)$, which for the mean-square error distortion measure becomes [2]
\begin{equation}
R(D) \geq h(X) - \frac{1}{2} \log (2\pi e D) = \text{LB}_{\text{Sha}}.
\end{equation}
The gap $\Delta$ between the new upper bound on $R(D)$ given in Theorem 1 and Shannon’s lower bound in (30) is given by
\begin{equation}
\Delta = \text{UB}_{\text{new}} - \text{LB}_{\text{Sha}} = -h(X|Y) + \frac{1}{2} \log \left( 2\pi e \text{mmse} \left( \frac{1}{\sigma_N^2} \right) \right).
\end{equation}
where we have used (9), (30), (11), and (24). The variance of the MMSE estimate $E[X|Y]$ is equal to $\text{mmse}^2(1/\sigma_N^2)$. Furthermore, for a given variance Gaussian random variables are entropy maximizers and the second term on the RHS of (31) corresponds to the differential entropy of a Gaussian random variable with variance $\text{mmse}^2(1/\sigma_N^2)$. In general, the estimation error of the MMSE estimate $E[X|Y]$ is non-Gaussian. Thus, $h(X|Y)$ in (31) is upper-bounded by the second term on the RHS of (31). $\Delta$ corresponds to the negative difference of the entropy of $X$ conditioned on $Y$ and the corresponding entropy in case $X$ is Gaussian with the same variance. Hence, $\Delta$ becomes zero for $X$ being Gaussian.

C. Comparison of the New Upper Bound on $R(D)$ with Shannon’s Upper Bound on the Rate Distortion Function

Shannon has shown that for an arbitrary distributed source $X$ with variance $\sigma_X^2$, the rate distortion function is upper-bounded by the rate distortion function of a Gaussian source with the same variance, which is given in (7). I.e., Gaussian sources are hardest to encode, cf. [11].

The difference between this upper bound in (7) given by Shannon and the new upper bound on $R(D)$ stated in
Theorem 1 is for $D \leq \sigma_X^2$ given by

$$UB_{Sha} - UB_{new} = \frac{1}{2} \log \left( \frac{\sigma_X^2}{\text{mmse} \left( \frac{1}{\sigma_X^2} \right)} \right) - \frac{1}{2} \int_0^{\frac{1}{\gamma}} \text{mmse}(\gamma) d\gamma$$

where we have used (7), (11), and (8). The difference on the RHS of (32) depends on $\text{mmse}(\gamma)$, which is bounded by $0 \leq \text{mmse}(\gamma) \leq \frac{\sigma_X^2}{1 + \sigma_X^2 \gamma}$, where the RHS is the MMSE in case $X$ is Gaussian. The upper bound on $\text{mmse}(\gamma)$ holds as Gaussian random variables are hardest to estimate [12, Prop. 15].

$UB_{new}$ is in general tighter than $UB_{Sha}$ as, by using $\text{mmse}(\gamma) \leq \frac{\sigma_X^2}{1 + \sigma_X^2 \gamma}$ and the relation between the MMSE and the mutual information given in [4], (32) is lower-bounded by

$$UB_{Sha} - UB_{new} \leq \frac{1}{2} \log \left( 1 + \frac{\sigma_X^2}{\sigma_X^2} \right) = \frac{1}{2} \log (2) - 1 = -1$$

Thus, $UB_{Sha} - UB_{new}$ converges to zero for $D \to \sigma_X^2$.

**D. Evaluation of the Upper Bound in Theorem 1**

In Fig. 1 we have evaluated\(^3\) the new upper bound on the rate distortion function given by Theorem 1 for a uniform source distribution (Fig. 1(a)), a source with Laplace distribution and a bipolar input distribution (both Fig. 1(b)) all with zero-mean and variance $\sigma_X^2 = 1$, i.e.,

$$p_X(x) = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } |x| \leq \sqrt{3}, \\ 0 & \text{otherwise}, \end{cases}$$

$$p_X(x) = \frac{1}{\sqrt{2}} \exp \left( -\sqrt{2} |x| \right),$$

$$p_X(x) = \begin{cases} \frac{1}{2} & \text{for } x = \pm 1, \\ 0 & \text{otherwise}. \end{cases}$$

For comparison the Shannon upper bound in (7) corresponding to the rate distortion function for a Gaussian source, the Shannon lower bound in (30), and the actual rate distortion function calculated numerically based on the Blahut-Arimoto algorithm [13], [14] are shown. The Shannon lower bound (30) is not shown for the bipolar source as $h(X)$ does not exist in this case.

Fig. 1 shows that the new upper bound and the Shannon lower bound on the rate distortion function become tight for $D \to 0$. On the other hand, for $D \to \sigma_X^2$ the new upper bound converges to the Shannon upper bound. The comparison of the new upper bound with the numerical evaluation of $R(D)$ based on the Blahut-Arimoto shows that the new upper bound is quite tight for the displayed source distributions.

\(^3\)The MATLAB source code is available on our websites.

---

**REFERENCES**


On the Information Loss in Memoryless Systems: The Multivariate Case

Bernhard C. Geiger*, Gernot Kubin*

*Signal Processing and Speech Communication Laboratory, Graz University of Technology, Austria
{geiger,gernot.kubin}@tugraz.at

Abstract—In this work we give a concise definition of information loss from a system-theoretic point of view. Based on this definition, we analyze the information loss in memoryless input-output systems subject to a continuous-valued input. For a certain class of multiple-input, multiple-output systems the information loss is quantified. An interpretation of this loss is accompanied by upper bounds which are simple to evaluate.

Finally, a class of systems is identified for which the information loss is necessarily infinite. Quantizers and limiters are shown to belong to this class.

I. INTRODUCTION

In the XXXI. Shannon lecture Han argued that information theory links information-theoretic quantities, such as entropy and mutual information, to operational quantities such as source, channel, capacity, and error probability [1]. In this work, we try to make a new link to an operational quantity not mentioned by Han: information loss. Information can be lost, on the one hand, in erasures or due to superposition of noise as it is known from communication theory. Dating back to Shannon [2] this loss is linked to the conditional entropy of the input given the output, at least in discrete-amplitude, memoryless settings. On the other hand, as stated by the data processing inequality (DPI, [3]), information can be lost in deterministic, noiseless systems. It is this kind of loss that we will treat in this work, and we will show that it makes sense to link it to the same information-theoretic quantity.

The information loss in input-output systems is very sparsely covered in the literature. Aside from the DPI for discrete random variables (RV) and static systems, some results are available for jointly stationary stochastic processes [4]. Yet, all these results just state that information is lost, without quantifying this loss. Only in [5] the information lost by collapsing states of a discrete-valued stochastic process is quantified as the difference between the entropy rates at the input and the output of the memoryless system.

Conversely, energy loss in input-output systems has been deeply analyzed, leading to meaningful definitions of transfer functions and notions of passivity, stability, and losslessness. Essentially, it is our aim to develop a system theory not from an energetic, but from an information-theoretic point of view. So far we analyzed the information loss of discrete-valued stationary stochastic processes in finite-dimensional dynamical input-output systems [6], where we proposed an upper bound on the information loss and identified a class of information-preserving systems (the information-theoretic counterpart to lossless systems). In [7] the information loss of continuous RVs in memoryless systems was quantified and bounded in a preliminary way. In this work, extending [7], we analyze the information loss for static multiple-input, multiple-output systems which are subject to a continuous input RV. Unlike in our previous work, we permit functions which lose an infinite amount of information and present the according conditions. Aside from that we provide a link between information loss and differential entropy, a quantity which is not invariant under changes of variables. The next steps towards an information-centered system theory are the analysis of discrete-time dynamical systems with continuous-valued stationary input processes and a treatment of information loss in multirate systems.

In the remainder of this paper we give a mathematically concise definition of information loss (Section II). After restricting the class of systems in Section III, in Section IV we provide exact results for information loss together with simple bounds, and establish a link to differential entropies. Finally, in Section V we show under which conditions the information loss becomes infinite. An extended version of this paper, accompanied by several examples illustrating the theoretical results, is available in [8].

II. A DEFINITION OF INFORMATION LOSS

When talking about the information loss induced by processing of signals, it is of prime importance to accompany this discussion by a well-based definition of information loss going beyond, but without lacking, intuition. We try to meet this objective with the following

Definition 1. Let $X$ be an RV\(^1\) on the sample space $\mathcal{X}$, and let $Y$ be obtained by transforming $X$. We define the information loss induced by this transform as

$$L(X \rightarrow Y) = \sup_{\mathcal{P}} \left( I(\hat{X};X) - I(\hat{X};Y) \right)$$

where the supremum is over all partitions $\mathcal{P}$ of $\mathcal{X}$, and where $\hat{X}$ is obtained by quantizing $X$ according to the partition $\mathcal{P}$ (see Fig. 1).

This Definition is motivated by the data processing inequality (cf. [3]), which states that the expression under the supremum is always non-negative: Information loss is the worst-case reduction of information about $X$ induced by

\(^1\)Note that $X$ and all other involved RVs need not be scalar-valued.
Fig. 1. Model for computing the information loss of a memoryless input-output system $g$, $Q$ is a quantizer with partition $\mathcal{P}$.

transforming $X$. We now try to shed a little more light on Definition 1 in the following

**Theorem 1.** The information loss of Definition 1 is given by the conditional entropy of the input given the output, i.e.,

$$I(X \rightarrow Y) = H(X|Y).$$

(2)

**Proof:** We start by noticing that

$$I(\hat{X};X) - I(\hat{X};Y) = H(\hat{X}|Y) \leq H(X|Y)$$

(3)

by the definition of mutual information and since both $\hat{X}$ and $Y$ are functions of $X$. The inequality in (3) is due to data processing ($\hat{X}$ is a function of $X$). We now show that in the supremum over all partitions equality can be achieved.

To this end, observe that among all partitions of the sample space of $X$ there is a sequence $\{\mathcal{P}_n\}$ of increasingly fine partitions such that

$$\lim_{n \rightarrow \infty} \hat{X}_n = X$$

(4)

where $\hat{X}_n$ is the quantization of $X$ induced by partition $\mathcal{P}_n$. By the axioms of entropy (e.g., [9, Ch. 14]), $H(\hat{X}_n|Y)$ is an increasing sequence in $n$ with limit $H(X|Y)$. Thus, equality in (3) is achieved, which completes the proof.

This Theorem shows that the supremum in Definition 1 is achieved for $\hat{X} \equiv X$, i.e., when we compute the difference between the self-information of the input and the information the output of the system contains about its input. In addition to that, the Theorem suggests a natural way to measure the information loss via measuring mutual informations, as it is depicted in Fig. 1. As we will see later (cf. Theorem 3), the considered partition does not have to be absolutely fine.

**III. Problem Statement**

Let $X = [X_1, X_2, \ldots, X_N]$ be an $N$-dimensional RV with a probability measure $P_X$ and the Lebesgue measure $\mu$ ($P_X \ll \mu$). We require $P_X$ to be concentrated on $\chi \subseteq \mathbb{R}^N$. This RV, which possesses a unique probability density function (PDF) $f_X$, is the input to the following multivariate, vector-valued function:

**Definition 2.** Let $g : \chi \rightarrow \chi'$, $\chi', \chi \subseteq \mathbb{R}^N$, be a surjective, Borel-measurable function defined in a piecewise manner:

$$g(x) = \begin{cases} g_1(x), & \text{if } x \in \chi_1 \\ g_2(x), & \text{if } x \in \chi_2 \\ \vdots \end{cases}$$

(5)

\[\text{where } x = [x_1, x_2, \ldots, x_N] \text{ and } g_i : \chi_i \rightarrow \chi_i \text{ bijectively. Furthermore, let the Jacobian matrix } J(g_i) \text{ exist on the closures of } \chi_i. \text{ In addition to that, we require the Jacobian determinant, } |\det J(g_i)|, \text{ to be non-zero } f_X - \text{almost everywhere.} \]

In accordance with previous work [7] the $\chi_i$ are disjoint sets of positive $P_X$-measure which unite to $\chi$, i.e., $\bigcup_i \chi_i = \chi$ and $\chi_i \cap \chi_j = \emptyset$ if $i \neq j$. Clearly, also the $\chi_i$ unite to $\chi$, but need not be disjoint. This definition ensures that the preimage $g^{-1}[y]$ of each element $y \in \chi$ is a countable set. Using the method of transformation [9, pp. 244] one obtains the PDF of the $N$-dimensional output RV $Y = [Y_1, Y_2, \ldots, Y_N]$ as

$$f_Y(y) = \sum_{x_i \in g^{-1}[y]} \frac{f_X(x_i)}{|\det J(g_i)|}$$

(6)

where the sum is over all elements of the preimage. Note that since $Y$ possesses a density, the corresponding probability measure $P_Y$ is also absolutely continuous w.r.t. the Lebesgue measure.

**IV. Main Results**

We now state our main results:

**Theorem 2.** The information loss induced by a function $g$ satisfying Definition 2 is given as

$$H(X|Y) = \int f_X(x) \log \left( \sum_{y \in g^{-1}[x]} \frac{f_X(x_i)}{|\det J(g_i)|} \right) dx.$$  

(7)

The proof of this Theorem can be found in the Appendix of [8] and, in a modified version for univariate functions, in [7]. Note that for univariate functions the Jacobian determinant is replaced by the derivative of the function.

**Corollary 1.** The information loss induced by a function $g$ satisfying Definition 2 is given as

$$H(X|Y) = h(X) - h(Y) + E \{ \log |\det J(g_i)| \}$$

(8)

**Proof:** The proof is obtained by recognizing the PDF of $Y$ inside the logarithm in (7) and by splitting the logarithm.

This result is particularly interesting because it provides a link between information loss and differential entropies already anticipated in [9, pp. 660]. There, it was claimed that

$$h(Y) \leq h(X) + E \{ \log |\det J(g_i)| \}$$

(9)

where equality holds iff $g$ is bijective. While (9) is actually another version of the DPI, Corollary 1 quantifies how much information is lost by processing. In addition to that, a very similar expression denoted as folding entropy has been presented in [10], although in a completely different setting analyzing the entropy production of autonomous dynamical systems.

We now introduce a discrete RV $W$ which depends on the set $\chi_i$ from which $X$ was taken. In other words, we have $W = \chi_i$. Let $g_i$ be a surjective function defined in a piecewise manner:

$$g_i(x) = \begin{cases} g_{i1}(x), & \text{if } x \in \chi_{i1} \\ g_{i2}(x), & \text{if } x \in \chi_{i2} \\ \vdots \end{cases}$$

\[\text{where } x = [x_1, x_2, \ldots, x_N] \text{ and } g_{i1}, g_{i2}, \ldots \text{ bijectively. Furthermore, let the Jacobian matrix } J(g_{i1}) \text{ exist on the closures of } \chi_{i1}. \text{ In addition to that, we require the Jacobian determinant, } |\det J(g_{i1})|, \text{ to be non-zero } f_X - \text{almost everywhere.} \]
Theorem 3. The information loss is identical to the uncertainty about the set \( X_i \) from which the input was taken, i.e.,

\[
H(X|Y) = H(W|Y).
\]

(10)

The proof closely follows the proof provided in [7] and thus is omitted. However, this equivalence suggests a way of measuring information loss by means of proper quantization: Since \( H(W|Y) = I(W;X) - I(W;Y) \) the loss can be determined by measuring mutual informations, which in this case are always finite (or, at least, bounded by \( H(W) \)).

The interpretation derived from Theorem 3 allows us now to provide upper bounds on the information loss:

Theorem 4. The information loss is upper bounded by

\[
H(X|Y) \leq \int_Y f_Y(y) \log \text{card}(g^{-1}[y])dy \tag{11}
\]

\[
\leq \log \left( \sum_i \int_{y_i} f_X(y)dy \right) \tag{12}
\]

\[
\leq \max_y \log \text{card}(g^{-1}[y]) \tag{13}
\]

where \( \text{card}(B) \) is the cardinality of the set \( B \).

Proof: We give here only a sketch of the proof: The first inequality results from bounding \( H(W|Y = y) \) by the entropy of a uniform distribution on the preimage of \( y \). Jensen's inequality yields the second line of the Theorem. The coarsest bound is obtained by replacing the cardinality of the preimage by its maximal value.

In this Theorem, we bounded the information loss given a certain output by the cardinality of the preimage. While the first bound considers the fact that the cardinality may actually depend on the output itself, the last bound incorporates the maximum cardinality only. In cases where the function from Definition 2 is defined not on a countable but on a finite number of subdomains, this finite number can act as an upper bound (cf. [7]). Another straightforward upper bound, which is independent from the bounds in Theorem 4 is obtained from Theorem 3 by removing conditioning:

\[
H(X|Y) \leq H(W) = -\sum_i p_i \log p_i \tag{14}
\]

where \( p_i = P_X(x_i) = \int_{x_i} f_X(x)dx \).

A further implication of introducing this discrete RV \( W \) is that it allows us to perform investigations about reconstructing the input from the output. Currently, a Fano-type inequality bounding the reconstruction error by the information loss is under development. In addition to that, new upper bounds on the information loss related to the reconstruction error of optimal (in the maximum a posteriori sense) and of simpler, sub-optimal estimators are analyzed.

V. Functions with Infinite Information Loss

We now drop the requirement in Definition 2 that the \( g \) possesses partial inverses to analyze a wider class of Borel-measurable functions \( g : X \rightarrow Y \). We keep the requirement that \( P_X \ll \mu \) and thus \( X \) possesses a density \( f_X \) (positive on \( X \) and zero elsewhere). We maintain

Theorem 5. Let \( g : X \rightarrow Y \) be a Borel-measurable function and let the continuous RV \( X \) be the input to this function. If there exists a set \( B \subseteq Y \) of positive \( P_Y \)-measure such that the preimage \( g^{-1}[y] \) is uncountable for every \( y \in B \), then the information loss is infinite.

Proof: With \( B \subseteq Y \) we notice that

\[
H(X|Y) \geq \int_B H(X|Y = y)dP_Y(y) \tag{15}
\]

where the integral is now written as Lebesgue integral, since \( P_X \) now not necessarily possesses a density.

Since on \( B \) the preimage of every element is uncountable, we obtain with [4] and the references therein \( H(X|Y = y) = \infty \) for all \( y \in B \), and, thus, \( H(X|Y) = \infty \).

Note that the requirement of \( B \) being a set of positive \( P_Y \)-measure cannot be dropped, as one of the examples provided in [8] illustrates. We immediately obtain the following

Corollary 2. Let \( g : X \rightarrow Y \) be a Borel-measurable function and let the continuous RV \( X \) be the input to this function. If the probability measure of the output, \( P_Y \), possesses a non-vanishing discrete component, the information loss is infinite.

Proof: According to the Lebesgue-Radon-Nikodym theorem [11, pp. 121] every measure can be decomposed in a component absolutely continuous w.r.t. \( \mu \) and a component singular to \( \mu \). The latter part can further be decomposed into a singular continuous and a discrete part, where the latter places positive \( P_Y \)-mass on points. Let \( y^* \) be such a point, i.e., \( P_Y(y^*) > 0 \). As an immediate consequence, \( P_X(g^{-1}[y^*]) > 0 \), which is only possible if \( g^{-1}[y^*] \) is uncountable (\( P_X \ll \mu \)).

This result is also in accordance with intuition, as the analysis of a simple quantizer shows: While the entropy of the input RV is infinite (\( I(X;X) \rightarrow \infty \) for \( X \rightarrow X \); cf. [9, pp. 654]), the quantized output can contain only a finite amount of information (\( I(X;X) \rightarrow H(Y) < \infty \)). In addition to that, the preimage of each possible output value \( y \) is a set of positive \( P_X \)-measure. The loss, as a consequence, is infinite.

While for the quantizer the preimage of each possible output value is a set of positive measure, there certainly are functions for which some outputs have a countable preimage and some whose preimage is a non-null set. An example of such a system is the limiter [9, Ex. 5-4]. For such systems it can be shown that both the information loss \( I(X \rightarrow Y) = H(X|Y) \) and the information transfer \( I(X;Y) \) are infinite.

Finally, there exist functions \( g \) for which the preimages of all output values \( y \) are null sets, but which still fulfill the conditions of Theorem 5. Functions which project \( X \) on a lower-dimensional subspace of \( \mathbb{R}^N \) fall into that category.
VI. EXAMPLE: A TWO-DIMENSIONAL TRANSFORM WITH FINITE INFORMATION LOSS

In this Section we illustrate our theoretical results with the help of an example. The logarithm is taken to base 2. Let $X$ be uniformly distributed on the square $\mathcal{X} = [-a,a] \times [-a,a]$, or equivalently, the two constituting RVs $X_1$ and $X_2$ are independent and uniformly distributed on $[-a,a]$. In other words, if $f_X(x) = 1/4a^2$ for all $x \in \mathcal{X}$, we have $f_X(x_i) = 1/2a$ for $x_i \in [-a,a]$ and $i = 1,2$.

We consider a function $g$ performing the mapping:

$$Y_1 = X_1 \tag{16}$$
$$Y_2 = |X_1 - X_2| \tag{17}$$

The corresponding Jacobian matrix is a triangular matrix

$$J_g(x) = \begin{bmatrix} 1 & 0 \\ \text{sgn}(x_1 - x_2) & \text{sgn}(x_2 - x_1) \end{bmatrix} \tag{18}$$

where $\text{sgn} \cdot \cdot \cdot \cdot \cdot$ is the sign-function. From this immediately follows that the magnitude of the determinant of the Jacobian matrix is unity for all possible values of $X$, i.e., $|\text{det}J_g(x)| = 1$ for all $x \in \mathcal{X}$. The subsets of $\mathcal{X}$ on which the partitioned functions $g_i$ are bijective are no intervals in this case; they are the triangular halves of the square induced by $x_1 = x_2$ (see Fig. 2).

$$\mathcal{X}_1' = \{x_1, x_2 \in \mathcal{X} : x_1 > x_2\} \tag{19}$$
$$\mathcal{X}_2' = \{x_1, x_2 \in \mathcal{X} : x_1 < x_2\} \tag{20}$$

The preimage of $g(x)$ is, in any case,

$$\{(x_1, x_2) \mid (x_1 + x_2) / 2 \notin \mathcal{X} \} \cap \mathcal{X} \tag{21}$$

The transform $g$ is bijective whenever $|x_1 + x_2| / 2 \notin \mathcal{X}$, i.e., if $|2x_1 - x_2| > a$.

With the PDF of $X$ and of its components we obtain for the information loss

$$H(X|Y) = \int_{-a}^{a} \int_{-a}^{a} \frac{1}{4a^2} \log \left( \frac{\frac{1}{2a} + f_X(2x_1 - x_2)}{2a} \right) \, dx_1 \, dx_2 \tag{22}$$

which is non-zero only if $-a \leq 2x_1 - x_2 \leq a$ (numerator and denominator cancel otherwise; no loss occurs in the bijective domain of the function). As a consequence,

$$H(X|Y) = \int_{-a}^{a} \int_{-a}^{a} \frac{\log 2}{4a^2} \, dx_1 \, dx_2 = \int_{-a}^{a} \frac{1}{4a} \, dx_2 = \frac{1}{2} \tag{23}$$

The information loss is identical to a half bit. This is intuitive by looking at Fig. 2, where it can be seen that any information loss occurs only on one half of the domain $\mathcal{X}$ (shaded in stronger colors). By destroying the sign information, in this area the information loss is equal to one bit.

VII. CONCLUSION

In this work, we proposed a mathematically concise definition of information loss for the purpose of establishing a system theory from an information-theoretic point of view. For a certain class of multivariate, vector-valued functions and continuous input variables this information loss was quantified, and the result is accompanied by convenient upper bounds. We further showed a connection between information loss and the differential entropies of the input and output variables.

Finally, a class of systems has been identified for which the information loss is infinite. Vector-quantizers and limiters belong to that class, but also functions which project the input space onto a space of lower dimensionality.

REFERENCES


One-Bit Quantizers for Fading Channels

Tobias Koch
University of Cambridge
Cambridge, CB2 1PZ, UK
Email: tobi.koch@eng.cam.ac.uk

Amos Lapidoth
ETH Zurich
CH-8092 Zurich, Switzerland
Email: lapidoth@ist.ee.ethz.ch

Abstract—We study channel capacity when a one-bit quantizer is employed at the output of the discrete-time average-power-limited Rayleigh-fading channel. We focus on the low signal-to-noise ratio regime, where communication at very low spectral efficiencies takes place, as in Spread Spectrum and Ultra-Wideband communications. We demonstrate that, in this regime, the best one-bit quantizer does not reduce the asymptotic capacity of the coherent channel, but it does reduce that of the noncoherent channel.

I. INTRODUCTION

We study the effect on channel capacity of quantizing the output of the discrete-time average-power-limited Rayleigh-fading channel using a one-bit quantizer. This problem arises in communication systems where the receiver uses digital signal processing techniques, which require the analog received signal to be quantized using an analog-to-digital converter (ADC). The effects of quantization are particularly pronounced when high-resolution ADCs are not practical and low-resolution ADCs must be used [1].

We focus on the low signal-to-noise ratio (SNR) regime, where communication at very low spectral efficiencies takes place (as in Spread-Spectrum and Ultra-Wideband communications). For the average-power-limited real-valued Gaussian channel, it is well-known that, in this regime, a symmetric one-bit quantizer (which produces 1 if the channel output is nonnegative and 0 otherwise) reduces the capacity by a factor of $2/\pi$, corresponding to a 2dB power loss [2]. It was recently shown that, by allowing for asymmetric one-bit quantizers with corresponding asymmetric signal constellations, these two decibels can be recovered in full [3]. A similar result was shown for the average-power-limited complex-valued Gaussian channel [4]: using binary on-off keying and a radial quantizer (which produces 1 if the magnitude of the channel output is above some threshold and 0 otherwise), one can achieve the low-SNR asymptotic capacity of the unquantized channel by judiciously choosing the threshold and the on-level as functions of the SNR. Here we extend [3], [4] to Rayleigh-fading channels. Specifically, we study the capacity per unit-energy [5] of such channels when the channel output is quantized using a one-bit quantizer.

For coherent fading channels, where the receiver has perfect channel knowledge, we show that quantizing the channel output with a one-bit quantizer causes no loss in the capacity per unit-energy. As in [4], the capacity per unit-energy can be achieved using binary on-off keying and a radial quantizer by choosing the threshold as a function of the SNR and the fading, with the threshold and the on-level both tending to infinity as the SNR tends to zero. This result might mislead one to think that quantizing the channel output with a one-bit quantizer causes no loss in the capacity per unit-energy also for noncoherent fading channels, where the receiver does not have perfect channel knowledge. Indeed, in the absence of a quantizer the capacity per unit-energy does not depend on whether the receiver has perfect channel knowledge or not [6], [7]. Since this capacity per unit-energy can be achieved using binary on-off keying with diverging on-level, it might therefore seem plausible that also in the presence of a quantizer the capacity per unit-energy would not depend on whether the receiver has perfect channel knowledge or not. But this is not the case: in contrast to the coherent case, quantizing the output of the noncoherent Rayleigh-fading channel with a one-bit quantizer reduces the capacity per unit-energy.

The rest of the paper is organized as follows. Section II describes the channel model and introduces the capacity per unit-energy. Section III presents the main results. Section IV discusses the capacity per unit-energy when the real and the imaginary part of the channel output are quantized separately with one-bit quantizers. And Section V presents the proofs of the main results.

II. CHANNEL MODEL AND CAPACITY PER UNIT-ENERGY

We consider a discrete-time Rayleigh-fading channel whose complex-valued output $\tilde{Y}_k$ at time $k \in \mathbb{Z}$ corresponding to the channel input $x_k \in \mathbb{C}$ (where $\mathbb{C}$ and $\mathbb{Z}$ denote the set of complex numbers and the set of integers) is given by

$$\tilde{Y}_k = H_k x_k + Z_k, \quad k \in \mathbb{Z}. \quad (1)$$

Here $\{Z_k, k \in \mathbb{Z}\}$ and $\{H_k, k \in \mathbb{Z}\}$ are independent sequences of independent and identically distributed (i.i.d.), zero-mean, circularly-symmetric, complex Gaussian random variables, the former with unit variance and the latter with variance $\sigma^2$. We say that the channel is coherent if the receiver is cognizant of the realization of $\{H_k, k \in \mathbb{Z}\}$ and that it is noncoherent if the receiver is cognizant only of the statistics of $\{H_k, k \in \mathbb{Z}\}$.

The receiver does not have access to the channel outputs $\{\tilde{Y}_k, k \in \mathbb{Z}\}$ but only to a quantized version thereof. Specifically, the complex channel output $\tilde{Y}_k$ is fed to a one-bit
quantizer which produces \( Y_k = 1 \) if \( \hat{Y}_k \) is in the quantization region \( D \) and \( Y_k = 0 \) otherwise, for some Borel set \( D \subset \mathbb{C} \). In the coherent case, \( \mathcal{D} \) may depend on the fading \( \{ H_k, k \in \mathbb{Z} \} \).

We assume that the average power of the channel inputs is limited by \( P \). The capacity of the above channel is [8], [9]

\[ C(P) = \sup I(X;Y|H), \quad \text{coherent case} \]  
(2)

\[ C(P) = \sup I(X;Y), \quad \text{noncoherent case} \]  
(3)

where the suprema on the right-hand side (RHS) of (2) and (3) are over all distributions on \( X \) satisfying \( E[|X|^2] \leq P \) and over all quantization regions \( D \). (Since the above channel is memoryless, we omit the time indices.)

The capacity per unit-energy is given by [5, Th. 2]

\[ \hat{C}(0) = \frac{C(P)}{P} \]  
(4)

It can be shown that

\[ \hat{C}(0) = \lim_{P \to 0} \frac{C(P)}{P} \]  
(5)

Thus, the capacity per unit-energy is equal to the slope at zero of the capacity-vs-power curve. It can be further shown that [5, Th. 3] (see also [6])

\[ \hat{C}(0) = \sup_{\xi \neq 0,D} \frac{D(P_{Y|H,X=\xi} \| P_{Y|H,X=0})}{|\xi|^2} \]  
(6)

in the coherent case and

\[ \hat{C}(0) = \sup_{\xi \neq 0,D} \frac{D(P_{Y|X=\xi} \| P_{Y|X=0})}{|\xi|^2} \]  
(7)

in the noncoherent case. Here \( D(\cdot \| \cdot) \) denotes relative entropy

\[ D(P||Q) = \begin{cases} \int \log \left( \frac{dP}{dQ} \right) dP, & \text{if } P \ll Q \\ \infty, & \text{otherwise} \end{cases} \]

where \( P \ll Q \) indicates that \( P \) is absolutely continuous with respect to \( Q \); \( D(\cdot || \cdot) \) denotes conditional relative entropy

\[ D(P_{Y|H,X=\xi} \| P_{Y|H,X=0}) = \int D(P_{Y|H=\xi,h,X=x} \| P_{Y|H=\xi,h,X=0}) dP_H(h); \]

\( P_H \) denotes the distribution of the fading \( H \); \( P_{Y|X=x} \) denotes the output distribution given that the input is \( x \); and \( P_{Y|H=h,X=x} \) denotes the output distribution conditioned on \((H,X) = (h,x)\).

By the Data Processing Inequality [10, Th. 2.8.1], the capacity per unit-energy of the quantized channel is upper-bounded by that of the unquantized channel [7], [6]

\[ \hat{C}(0) \leq \frac{1}{\sigma^2}. \]  
(8)

We show that in the coherent case this upper bound holds with equality, while in the noncoherent case it is strict.

### III. Main Result

We restrict ourselves to radial quantizers, for which

\[ D = \{ \hat{y} \in \mathbb{C} : |\hat{y}| \geq T \}, \quad \text{for some } T > 0. \]  
(9)

In the noncoherent case—as we show in Section V-B—such quantizers are optimal in the sense that they maximize the relative entropy on the RHS of (7) for every \( \xi \neq 0 \). In the coherent case such quantizers need not be optimal in the above sense. However, they suffice to achieve the capacity per unit-energy. And such quantizers have the practical advantage of not requiring knowledge of the phase of \( \hat{y} \).

**Theorem 1**: Consider the above channel model, and assume that the channel output is quantized using a one-bit quantizer. Then

1) In the coherent case,

\[ \hat{C}(0) = \frac{1}{\sigma^2} \]  
(10)

which can be achieved by some radial quantizer (9) with \( T \) depending on \( H \) and \( \xi \).

2) In the noncoherent case,

\[ \hat{C}(0) < \frac{1}{\sigma^2} \]  
(11)

with the inequality being strict.

**Proof**: See Section V.

### IV. Quantizing the Real and Imaginary Part

Instead of quantizing \( \hat{Y} \) using a one-bit quantizer, often the real and imaginary parts of \( \hat{Y} \) are quantized separately using a one-bit quantizer for each; see, e.g., [11]–[14], [6]. Thus, the first quantizer produces \( Y_{k,1} = 1 \) if \( \text{Re}(\hat{Y}_k) \in D_k \) and \( Y_{k,1} = 0 \) otherwise, and the second quantizer produces \( Y_{1,k} = 1 \) if \( \text{Im}(\hat{Y}_k) \in D_l \) and \( Y_{1,k} = 0 \) otherwise, for some Borel sets \( D_k, D_l \subset \mathbb{R} \). (Here \( \mathbb{R} \) denotes the set of real numbers. \( \text{Re}(\cdot) \) denotes the real part, and \( \text{Im}(\cdot) \) denotes the imaginary part.)

In the coherent case, \( D_k \) and \( D_l \) may depend on the fading \( \{ H_k, k \in \mathbb{Z} \} \).

The capacity per unit-energy of this channel is given by (6) or (7), but with \( Y \) replaced by \( (Y_{k,1}, Y_{1,k}) \), and with \( D \subset \mathbb{C} \) replaced by \( (D_k, D_l) \subset \mathbb{R} \times \mathbb{R} \).

For symmetric quantizers, i.e., for

\[ D_k = D_l = \{ u \in \mathbb{R} : u \geq 0 \} \]  
(12)

it follows from [11] and [15, Th. 2] that, in the coherent case,

\[ \hat{C}_{\text{sym}}(0) = \frac{2}{\pi \sigma^2}. \]  
(13)

In the noncoherent case, symmetric quantizers result in zero capacity and hence, by (4), in zero capacity per unit-energy. Indeed, for (12)

\[ \text{Pr}(Y_{k,1} = 1 | X = x) = \text{Pr}(Y_{1,k} = 1 | X = x) = \frac{1}{2}, \quad x \in \mathbb{C}. \]

Since, conditioned on \( X \), the random variables \( Y_{k,1} \) and \( Y_{1,k} \) are independent, this implies that the capacity is zero. Thus, quantizing the real and imaginary parts of the Rayleigh-fading

2Here and throughout this paper, \( \xi \) refers to the parameter in (6) or (7).
channel using symmetric one-bit quantizers reduces the capacity per unit-energy by a factor of $2/\pi$ in the coherent case, and it reduces it to zero in the noncoherent case. In the following, we show that if we allow for asymmetric quantizers, then we can fully recover the loss of $2/\pi$ incurred on the coherent Rayleigh-fading channel. For the noncoherent channel, we show that asymmetric quantizers achieve a positive capacity per unit-energy, albeit strictly smaller than $1/\sigma^2$.

**Theorem 2:** Consider the above channel model, and assume that the real and imaginary parts of $Y$ are quantized separately using a one-bit quantizer for each.

1) In the **coherent case**,\[ C(0) = \frac{1}{\sigma^2} \] (14)
which can be achieved by some quantization regions
\[ D_k = \{ u \in \mathbb{R} : u \geq 2k \} \]
(15)
\[ D_1 = \{ u \in \mathbb{R} : u \geq 1 \} \]
(16)
where $T_k$ and $T_1$ depend on $\text{Re}(H\xi)$ and $\text{Im}(H\xi)$, respectively.

2) In the **noncoherent case**,\[ \frac{2Q(1)}{\sigma^2} \leq C(0) < \frac{1}{\sigma^2} \] (17)
with the upper bound being strict. Here $Q(\cdot)$ denotes the Gaussian $Q$-function [16, Eq. (1.3)]. The lower bound can be achieved by the quantization regions (15) and (16) with $T_k = T_1 = (\|\xi\|^2 + \sigma^2)/2$.

**Proof:** Omitted. \[ \Box \]

**V. PROOF OF THEOREM 1**

**A. Part 1**

We show that a radial quantizer (9) achieves the rate per unit-energy $1/\sigma^2$. Together with (8), this proves Theorem 1.

To this end, we first note that, conditioned on $(H, X) = (h, x)$, the squared magnitude of $\frac{h}{\xi} Y$ is a noncentral chi-square distribution with degree 2 and noncentrality parameter $\frac{2}{\sigma^2} |h|^2 |x|^2$ [16, p. 8]. Consequently, a radial quantizer yields [16, Sec. 2-E]
\[ \Pr(Y = 1 \mid H = h, X = x) = Q_1(\sqrt{\frac{2}{\sigma^2} |h||x|}, \sqrt{\frac{2}{\sigma^2} 1}) \]
where $Q_1(\cdot, \cdot)$ denotes the first-order Marcum $Q$-function [16, Eq. (2.20)]. Furthermore, for $x = 0$ this becomes
\[ \Pr(Y = 1 \mid H = h, X = 0) = e^{-\frac{\pi^2}{2h}}. \]

We thus obtain
\[ D(P_{Y|H,X=\xi} \parallel P_{Y|H,X=0} \parallel P_H) = \mathbb{E}[Q_1(\sqrt{\frac{2}{\sigma^2} |H||\xi|}, \sqrt{\frac{2}{\sigma^2} 1}) - \log \frac{1}{1 - e^{\frac{-\pi^2}{2h}}} + \mathbb{E}[\{1 - Q_1(\sqrt{\frac{2}{\sigma^2} |H||\xi|}, \sqrt{\frac{2}{\sigma^2} 1})\} - \log \frac{1}{1 - e^{\frac{-\pi^2}{2h}}} - \mathbb{E}[H_b(\sqrt{\frac{2}{\sigma^2} |H||\xi|}, \sqrt{\frac{2}{\sigma^2} 1})] + \mathbb{E}[Q_1(\sqrt{\frac{2}{\sigma^2} |H||\xi|}, \sqrt{\frac{2}{\sigma^2} 1})] - \log 2] \]
(18)
where $H_b(\cdot)$ denotes the binary entropy function, i.e.,
\[ H_b(p) = \left\{ \begin{array}{ll} p \log_2 \frac{1}{p} + (1 - p) \log_2 \frac{1}{1 - p}, & 0 < p < 1 \\ 0, & p = 0 \text{ or } p = 1. \end{array} \right. \]

Here the inequality follows because the second term in the first step is nonnegative, and because the binary entropy function is upper-bounded by $\log 2$.

We choose $T = \mu |h||\xi|$ for some fixed $\mu \in (0, 1)$ and lower-bound the RHS of (18) using the general lower bound on the first-order Marcum $Q$-function [16, Sec. C-2, Eq. (C.24)]
\[ Q_1(\alpha, \beta) \geq 1 - \frac{1}{2} e^{-\frac{\alpha^2 + \beta^2}{4}} e^{-\frac{\beta^2}{4}}, \quad \alpha > \beta > 0. \]

We thus obtain for the first term on the RHS of (18)
\[ \mathbb{E}[Q_1(\sqrt{\frac{2}{\sigma^2} |H||\xi|}, \sqrt{\frac{2}{\sigma^2} 1})] \leq \frac{\mu^2 \mathbb{E}[|H|^2]}{\sigma^2} \frac{|\xi|^2}{\sigma^2} \]
\[ - \frac{1}{2} \mathbb{E}[\exp\left(-\frac{|H|^2}{\sigma^2}(1 - \mu)^2\right) \mu^2 |H|^2 |\xi|^2 \sigma^2] + \frac{1}{2} \mathbb{E}[\exp\left(-\frac{|H|^2}{\sigma^2}(1 + \mu)^2\right) \mu^2 |H|^2 |\xi|^2 \sigma^2] \]
\[ \geq \frac{\mu^2 \mathbb{E}[|H|^2]}{\sigma^2} \frac{|\xi|^2}{\sigma^2} - \frac{\mu^2}{2} \frac{1}{e(1 - \mu)^2} \]
(19)
where the last step follows because $0 \leq xe^{-ax} \leq 1/(ea)$ for every $x \geq 0$ and $a > 0$.

Combining (19) with (18), and computing its ratio to $|\xi|^2$ in the limit as $|\xi|^2$ tends to infinity, yields
\[ C(0) \geq \frac{\mu^2 \mathbb{E}[|H|^2]}{\sigma^2} = \frac{\mu^2}{\sigma^2}. \]
(20)
Theorem 1 follows then by letting $\mu$ tend to one.

**B. Part 2**

We first note that, by the Data Processing Inequality for relative entropy [10, Sec. 2.9], the relative entropy on the RHS of (7) is upper-bounded by the relative entropy corresponding to the unquantized channel, i.e., [6, Eq. (64)]
\[ D(P_{Y|X=\xi} \parallel P_{Y|X=0}) \leq \frac{1}{\sigma^2} \log \frac{1 + |\xi|^2}{\sigma^2}. \]
(21)
Consequently, the capacity per unit-cost (7) is strictly smaller than $1/\sigma^2$ unless $|\xi|$ tends to infinity. It thus remains to show that
\[
\limsup_{|\xi| \to \infty} \sup_D D(P_{Y|X=\xi} \parallel P_{Y|X=0}) \leq \frac{1}{\sigma^2}.
\] (22)
To this end, we first note that, for every $\xi \neq 0$, the supremum in (22) over all quantization regions $D$ can be replaced with the supremum over all radial quantizers (9). Indeed, for every quantization region satisfying
\[
\Pr(Y = 1 \mid X = \xi) = \beta, \quad 0 < \beta < 1
\]
the relative entropy
\[
D(P_{Y|X=\xi} \parallel P_{Y|X=0}) = \beta \log \frac{1}{\Pr(Y = 1 \mid X = 0)} + (1 - \beta) \log \frac{1}{1 - \Pr(Y = 1 \mid X = 0)} - H_b(\beta)
\] (23)
is a convex function of $\Pr(Y = 1 \mid X = 0)$. Thus, for every $0 < \beta < 1$, the RHS of (23) is maximized for the quantization region that minimizes (or maximizes) $\Pr(Y = 1 \mid X = 0)$ while holding $\Pr(Y = 1 \mid X = \xi) = \beta$ fixed. By the Neyman-Pearson lemma [17], such a quantization region has the form
\[
D_\Lambda = \left\{ \hat{y} \in \mathbb{C} : \frac{f(\hat{y},0)}{f(\hat{y},\xi)} \leq \Lambda, \quad \Lambda > 0 \right\}
\] (24)
(or the complement thereof), where $f(\hat{y},x)$ denotes the conditional density of $Y$, conditioned on $X = x$, and $\Lambda$ is such that $\Pr(Y \in D_\Lambda \mid X = \xi) = \beta$. (Note that for every $0 < \beta < 1$ there exists such a $\Lambda$ since, for the above channel model (1), $\Pr(Y \in D_\Lambda \mid X = \xi)$ is a continuous, strictly increasing function of $\Lambda > 0$.) The likelihood ratio on the RHS of (24) is readily evaluated as
\[
\frac{f(\hat{y},0)}{f(\hat{y},\xi)} = \left( 1 + \frac{\xi^2}{\sigma^2} \right) \exp \left( -\frac{\xi^2 + \sigma^2}{2\sigma^2} \right), \quad \hat{y} \in \mathbb{C}.
\]
Consequently, $D_\Lambda$ is the same as (9) with
\[
T = \sqrt{\left( 1 + \frac{\sigma^2}{\xi^2} \right) \log \left( 1 + \frac{\xi^2}{\sigma^2} \right)}.
\]
We thus obtain that, for every $\xi \neq 0$, the relative entropy $D(P_{Y|X=\xi} \parallel P_{Y|X=0})$ is maximized by a radial quantizer (9).

For such a quantizer, we have
\[
\Pr(Y = 1 \mid X = x) = \exp \left( -\frac{T^2}{|x|^2 + \sigma^2} \right)
\]
which yields
\[
\begin{align*}
D(P_{Y|X=\xi} \parallel P_{Y|X=0}) &= e^{-\frac{T^2}{|\xi|^2 + \sigma^2}} \log \frac{1}{e^{-\frac{T^2}{|\xi|^2}}} \\
&\quad + \left[ 1 - e^{-\frac{T^2}{|\xi|^2 + \sigma^2}} \right] \log \frac{1}{1 - e^{-\frac{T^2}{|\xi|^2}}} - H_b \left( e^{-\frac{T^2}{|\xi|^2 + \sigma^2}} \right) \\
&\leq \frac{T^2}{\sigma^2} e^{-\frac{T^2}{|\xi|^2 + \sigma^2}} - \left[ 1 - e^{-\frac{T^2}{|\xi|^2 + \sigma^2}} \right] \log \left( 1 - e^{-\frac{T^2}{|\xi|^2 + \sigma^2}} \right) \\
&\leq \frac{T^2}{\sigma^2} e^{-\frac{T^2}{|\xi|^2 + \sigma^2}} + \frac{1}{e} \\
&\leq \frac{T^2}{\sigma^2} e^{-\frac{T^2}{|\xi|^2 + \sigma^2}} + \frac{2}{\sigma^2}
\end{align*}
\] (25)
where the second step follows because $H_b(x) \geq 0$ and $\exp(-T^2/(|\xi|^2 + \sigma^2)) \geq \exp(-T^2/\sigma^2)$, and the third step follows because $-x \log x \leq \frac{1}{2e}$, $0 < x < 1$.

The first term on the RHS of (25) is maximized for $T^2 = |\xi|^2 + \sigma^2$. The RHS of (25) is thus upper-bounded by
\[
\limsup_{|\xi| \to \infty} \sup_D D(P_{Y|X=\xi} \parallel P_{Y|X=0}) \leq \frac{|\xi|^2 + \sigma^2}{e \sigma^2} + \frac{2}{e}.
\] (26)
Dividing the RHS of (26) by $|\xi|^2$, and computing the limit as $|\xi|$ tends to infinity, yields
\[
\limsup_{|\xi| \to \infty} \sup_D D(P_{Y|X=\xi} \parallel P_{Y|X=0}) \leq \frac{1}{e \sigma^2} + \frac{1}{e^2}.
\] (27)
This proves Theorem 2.

ACKNOWLEDGMENT

Enlightening discussions with A. Martinez and comments from S. Verdú are gratefully acknowledged.

REFERENCES

Model Switching and other Unconventional Fun with Experts

Steven de Rooij
Centrum Wiskunde & Informatica (CWI)
Science Park 123, Amsterdam, Netherlands
Email: s.de.rooij@cwi.nl

I. UNIVERSAL CODING AND REGRET

Universal codes combine a number of codes into one that achieves code lengths close to those of the best of the bunch. We consider the strongest notion of universality, where we compete with the best code for any data sequence, not just in expectation. When this is the goal, a natural performance measure is regret: the difference between the universal code length and the length of the best original code. Given a fixed finite set of \( K \) codes, a simple universal combination can be obtained with a two-part scheme, where the first \( \lfloor \log_2 K \rfloor \) bits are used to identify the original codes to be used for the data, and the second part is the actual encoding of the data using that code. The regret of this code is bounded by the length of the first part of the code. (From here on, all logarithms are binary and we ignore the rounding of code lengths to integer numbers of bits.)

This idea can be taken a step further. Consider the case where a good code is available for any reasonably small segment of the input sequence, but either the data or the codes themselves are so changeable that none of the \( K \) original codes perform very well on the entire data sequence. In such cases, a stronger regret guarantee is desirable. For simplicity, fix the number of source symbols at \( n \). Then, starting from \( K \) codes, first construct a new set of all “tracking” codes that split the input sequence into \( m \) blocks, and use one of the \( K \) original codes for each block. The goal is now to minimise the regret with respect to this extended set of codes. Counting how many such codes there are is a simple combinatorial exercise; we can subsequently use a two-part code as before to obtain a regret \( R(x^n, m) \) satisfying

\[
R(x^n, m) \leq \log \left( \frac{n - 1}{m - 1} \right) + m \log K. \tag{1}
\]

This solution is not quite satisfactory. First, it requires finding the best tracking code; while this can be done using dynamic programming, we will see that it is possible to obtain similar regret guarantees with much faster algorithms. Second, the number of segments \( m \) is a parameter of the algorithm, but \( m \) is typically unknown so we would like to have a regret guarantee that is valid for all \( m \) simultaneously. Third, the code is not sequential, making it unusable in an online setting.

A number of very simple and elegant algorithms have been developed that avoid these issues. Some derive from the information theory literature [Wit07], [VW98], [SM99], but there are also important, and strongly related results in online learning that deserve more exposure [HW98], [Vov99], [BW02], [CBL06], [GLL08]. This text is based on our paper [KdR08], which outlines the common structure to these algorithms.

II. EXPERT SEQUENCE PRIORS

Universal codes of the kind described above are most conveniently defined and analysed in the Bayesian framework. As a first step, we switch to probability theory and replace the length functions \( L_1, \ldots, L_K \) of the \( K \) original codes by probabilistic sources \( P_1, \ldots, P_K \), such that \( -\log P_k(x^n) = L_k(x^n) \) for all \( k \) and all data sequences \( x^n = x_1, \ldots, x_n \in X^n \) (for simplicity assume the alphabet to be discrete). Such probabilistic sources can now be used in a sequential prediction setting; the accumulated logarithmic loss exactly corresponds to the code length:

\[
\sum_{t=1}^{n} -\log P_k(x_t|x_{t-1}) = -\log P_k(x^n = x^n) = L(x^n).
\]

In the online learning literature, the prediction strategies \( P_k \) are called “experts” because, unless we have additional knowledge about the data generating process, which we do not wish to assume, we have to use their predictions. We now extend the model class as follows. Let \( k^n \in \{1, \ldots, K\}^n \) denote a sequence of expert identifiers. Define

\[
P_{k^n}(X^n) = P_{k_1}(X_1) \cdot P_{k_2}(X_2|X_1) \cdots P_{k_n}(X_n|X_{n-1}).
\]

We do not know which sequence of experts will yield the best predictions, but we can represent our uncertainty in the usual Bayesian way in the form of a prior \( \pi \) on sequences of experts. We can use this prior to make predictions by conditioning the Bayesian marginal distribution:

\[
P(X_{n+1}|x^n) = \frac{P(x_{n+1})}{P(x^n)} = \frac{\sum_{k^n+1} P_{k^n+1}(x^n; X_{n+1})}{\sum_{k^n} P_{k^n}(x^n)}. \tag{2}
\]

It is now also very easy to obtain a reasonable regret bound, by dropping every term from the marginal likelihood except the one we are interested in. Choose any reference expert sequence \( \hat{k}_n \), and let \( m \) be the number of blocks it has. Then

\[
R(x^n, \hat{k}_m) = -\log P(x^n) - (-\log P_{\hat{k}_m}^n)
\]

\[
= -\log \sum_{k^n} P_{k^n}(x^n)\pi(k^n) + \log P_{\hat{k}_m}(x^n)
\]

\[
\leq -\log \left( P_{\hat{k}_m}^n(x^n)\pi(\hat{k}_m^n) \right) + \log P_{\hat{k}_m}(x^n)
\]

\[
= -\log \pi(\hat{k}_m^n).
\]
A. Example

In general, the best sequence of experts may have a very low prior probability, so it will be hard to compete with. However, we can construct the prior such that expert sequences with relatively few blocks receive a high prior probability. For example, we may set

\[
\pi(k^n) = K^{-m} \alpha^{n-1} \left( \frac{1 - \alpha}{\alpha} \right)^{m-1},
\]

where \( m \) is the number of blocks in \( k^n \), and \( \alpha \in [0, 1] \) is a hyperparameter that can be interpreted as the “switching rate”: it is the probability of a new block starting at any given time. It may be verified that this prior sums to one. An algorithm to efficiently compute the predictions of this distribution, called Fixed Share, was first published in COLT 1995; see [HW98] for the journal version. For Fixed Share, the regret bound (3) reduces to

\[
R(x^n, k^n) \leq m \log K - n \log \alpha - (m - 1) \log \left( \frac{1 - \alpha}{\alpha} \right).
\]

The bound is optimised by setting the hyperparameter to \( \alpha^* = (n - m + 1)/(n - 1) \), which yields

\[
R(x^n, k^n) \leq m \log K - (n - 1)H(\alpha^*),
\]

where \( H \) denotes the binary entropy function. Note that this bound is very close to (1); the slack is necessary because this new regret bound is valid for all \( m \), whereas (1) is tight for a single fixed \( m \).

B. Second Example: Hyperpriors

In case the expert sequence prior is burdened by an unknown parameter, we can “integrate it out” using a hyperprior. Let \( \pi(\alpha) \) denote an expert sequence prior \( \pi \) with hyperparameter \( \alpha \), and let \( P(\alpha) \) denote the tracking distribution with expert sequence prior \( \pi(\alpha) \). Now impose a hyperprior \( \rho \) on \( \alpha \). We then obtain the following parameterless distribution:

\[
P(X^n) = \int P(\alpha) P(X^n | \alpha) \rho(\alpha) d\alpha
= \sum_k k^n P(k^n) \int \pi(\alpha)(k^n) \rho(\alpha) d\alpha.
\]

Note that the final integral itself defines an expert sequence prior. We will call it \( \sigma \). Thus, to obtain a parameterless tracking distribution we do not have to look at the data; it suffices to mix the expert sequence prior using \( \rho \).

In case of the example of the previous section, with \( \pi(\alpha) \) given as in (4) and hyperprior \( \rho(\alpha) = \text{Beta}(1/2, 1/2) \), we find

\[
\sigma(k^n) = K^{-m} \frac{\Gamma(n - m + 1/2) \Gamma(n + 1/2)}{\pi \Gamma(n + 1)}.
\]

Around the time that Fixed Share was published, but completely independently, Paul Volf and Frans Willems developed an algorithm dubbed the “Switching Method”, that efficiently predicts according to this distribution [Vil07], [VW98]. It can be shown that its regret exceeds (5) by at most \( 1/2 \log n + 1 \) bits, a modest price to pay for a completely parameterless strategy.

In the next section we proceed to show how Fixed Share and the Switching Method can be implemented efficiently.

III. Computation using HMMs

In general, the predictions of an expert distribution based on an expert sequence prior (2) may require summing an exponential number of terms, but the complexity can be reduced dramatically if the expert sequence prior admits a simple sufficient statistic. This is the case for the two example priors of the previous section: in Fixed Share, the marginal prior distribution on the next expert \( \pi(K_{n+1} | k^n) \) can be expressed as a function of only the current expert \( k_n \). Namely,

\[
\pi(K_{n+1} = k | k^n) = \frac{\alpha}{R} + 1 \{ k = k_n \} (1 - \alpha).
\]

Such expert distributions can be represented by a Hidden Markov Model (HMM), where the hidden state is such a sufficient statistic. The value of the hidden variable determines the distribution on experts; the experts’ predictions are weighed according to this distribution. It is convenient to represent the prior in the form of a state transition diagram for the hidden states. For the Fixed Share algorithm with four experts, the diagram looks as follows:

![State transition diagram](image)

All straight transitions have probability \( 1 - \alpha \); they represent that no new block is started at that sample size. The transitions into the centre nodes have probability \( \alpha \); they represent a switch to a new block. The centre nodes are “silent states” where no prediction has to be produced; such states redistribute some of the probability mass over the experts. The arcs out of the silent states all have probability \( 1/K \).

Using the Forward algorithm with these state transitions, the distribution on the hidden state can be updated in \( O(K) \), so that the entire data set can be processed in \( O(n \cdot K) \).

The switching method is only sightly more complicated. In this case, the currently used expert is not sufficient to determined the probability of starting a new block: we also need to know how many blocks there were previously. The following state transition diagram visualises the state transition diagram required to calculate \( \sigma(K_{n+1} | k^n) \). For simplicity it is for two experts, and we also assume that the first expert always predicts first. Then the current expert is determined by the number of switches modulo 2.
The power of expert tracking algorithms is twofold. First, they are black box: any online prediction strategy can be used as an expert. As such they can be applied very generally. One example is in [vEGdR11], where the estimators associated with each a series of model classes are used as “experts”. The paper subsequently uses an algorithm similar to Fixed Share to improve the performance of Bayesian Model Averaging under many circumstances. Second, very attractive computational complexity can sometimes be achieved. For example, the Fixed Share algorithm runs exactly as fast as Bayesian Model Averaging itself.

Based on our paper [KdR08], we have seen two expert tracking algorithms as examples of how you can do more with a bunch of prediction strategies than what everyone always seems to be doing, which is to just take the Bayesian mixture. The preprint of the full paper discusses several other expert sequence priors for tracking, as well as more sophisticated methods to analyse the resulting performance.

Much of current online learning research is related to prediction with expert advice, but it may not be instantly recognisable as such, as over time the problem has been generalised substantially. Rather than that the learner uses a set of experts who make predictions in the form of probability distributions, is subsequently penalized with logarithmic loss, before finally observing the actual outcome, in the more general setting both experts and outcomes are abstracted away altogether: the learner chooses an action from some space of allowed actions, and receives an arbitrary loss. This more general setup allows the theory to encompass online linear optimisation and prediction problems with loss functions other than logarithmic loss. It also makes the problem harder, and perhaps less directly applicable to information theory. As a starting point, see [CBL06].

**REFERENCES**


Elegant relationships between classical quantities in information and estimation arise in the presence of additive Gaussian noise. In this talk, we review and present these relationships as expectations of random quantities, and subsequently present some of our recent findings pertaining to their ‘pointwise’ analogues.

Duncan, in [1], showed that for the continuous-time additive white Gaussian noise channel, the minimum mean squared filtering (causal estimation) error is twice the input-output mutual information for any underlying signal distribution. Another discovery was made by Guo et al. in [2], where the derivative of the mutual information was found to equal half the minimum mean squared error in non-causal estimation. By combining these two intriguing results, the authors of [2] also establish the remarkable equality of the causal mean squared error (at some ‘signal to noise’ level snr) and the non-causal error averaged over ‘signal to noise’ ratio uniformly distributed between 0 and snr. There have been extensions of these results to the presence of mismatch. In this case, the relative entropy and the difference of the mismatched and matched mean squared errors are bridged together: Mismatched estimation in the scalar Gaussian channel was considered by Verdú in [3]. In [4], a generalization of Duncan’s result to incorporate mismatch for the full generality of continuous time processes is provided. In [5], Kadota et al. generalize Duncan’s theorem to the presence of feedback.

In [6], the pointwise analogues of these relationships are obtained, giving considerable insight into the above results. As an illustration, consider Duncan’s 1970 result, which can equivalently be expressed saying that the difference between the input-output information density and half the causal estimation error is a zero mean random variable (regardless of the distribution of the channel input). We characterize this random variable explicitly, rather than merely its expectation. Classical estimation and information theoretic quantities emerge with new and surprising roles. For example, the variance of this random variable turns out to be given by the causal MMSE (which, in turn, is equal to twice the mutual information by Duncan’s result).

REFERENCES


* Stanford University. Email: kvenkat@stanford.edu, tsachy@stanford.edu
Polar martingales of maximal spread

Emmanuel Abbe
Faculté Informatique et Communications
Ecole Polytechnique Fédérale de Lausanne
Email: emmanuel.abbe@epfl.ch

Abstract—This paper investigates polar codes for $q$-ary input channels using other kernels than $\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$ to construct the generator matrix. The kernel is optimized by maximizing the spread of the polar martingale and it is shown that in certain cases, the resulting construction has a significantly improved error probability (even at small block length like $n = 1024$).

I. INTRODUCTION

The generator matrix of a polar code $[1]$ of block length $n$ is obtained by deleting some rows of the matrix $G_n = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \otimes \log_2 n$. Which rows to delete depends on the channel and on the targeted error probability (or rate). For a symmetric discrete memoryless channel $W$, the rows to be deleted are indexed by

$$B_{\varepsilon,n} := \{i \in [n] : I(U_i; Y^n U^{i-1}) \leq 1 - \varepsilon\},$$

where $\varepsilon$ is a parameter governing the error probability. $U^n$ has i.i.d. components which are uniform on the input alphabet, $X^n = U^n G_n$ and $Y^n$ is the output of $n$ independent uses of $W$ when $X^n$ is the input. The key observation at the root of this construction can be explained by considering $n = 2$. Let $u_1, u_2 \in F_2 = \{0, 1\}$,

$$x_1 = u_1 + u_2$$
$$x_2 = u_2$$

and let $Y_1, Y_2$ be the output of two independent uses of a binary input channel $W$ when $x_1, x_2$ are the inputs. Note that the binary input channels

$$W^- : u_1 \rightarrow Y_1, Y_2$$
$$W^+ : u_2 \rightarrow Y_1, Y_2, u_1$$

satisfy the relations

$$I(W^-) \leq I(W) \leq I(W^+)$$
$$I(W^+) + I(W^-) = 2I(W)$$

where $I(W)$ denotes the mutual information of the channel $W$ when the input distribution is uniform over $\mathbb{F}_2$. Therefore, we have transformed two independent uses of the channel $W$ into two uses of the channels $W^-$ and $W^+$, with overall the same amount of mutual information. If one had to transmit only one bit uniformly drawn, using $W^+$ rather than $W$ would lead to a lower error probability. One can then iterate this argument, after $\log_2(n)$ iterations, one obtains the channels $u_i \rightarrow Y^n u^{i-1}$. For a given number of information bits to be transmitted (rate), selecting the channels having largest mutual informations leads to the least error probability. As explained in the next section, polarization happens in the sense that as $n$ tends to infinity, these synthesized channels have mutual information tending to either 0 or 1. Hence, sending information bits on the high mutual information channels (equivalently, deleting rows of $G_n$ corresponding to low mutual information channels) allows to achieve communication rates as large as the mutual information of the original binary input channel. The construction extends to $q$-ary input alphabets when $q$ is prime, using the same matrix $G_n = \begin{bmatrix} 1 & 0 \end{bmatrix} \otimes \log_2 n$, while carrying the operations over $\mathbb{F}_q$.

It is tempting to investigate what happens if one keeps the Kronecker structure for the generator matrix but modifies the kernel $\begin{bmatrix} 1 & 1 \end{bmatrix}$. For binary input alphabets, there might not be another interesting choice. One can consider kernel matrices which are not $2 \times 2$ but $k \times k$. In this case, $[3]$ shows that a significant improvement in the error probability rate can be achieved, however, to the cost of suffering from a complexity increase and a much larger block length requirement. Not as much has been studied concerning modifying the kernel matrix over $\mathbb{F}_q$. In $[4]$, the error probability of non-binary polar codes constructed on the basis of Reed-Solomon matrices is calculated using numerical simulations on $q$-ary erasure channels. It is confirmed that 4-ary polar codes can have significantly better performance than binary polar codes. In this paper, our goal is to investigate potential improvements at finite block length using modified kernels over $\mathbb{F}_q$. We propose to pick kernels not by optimizing the polar code exponent as in [4] but by maximizing the polar martingale spread. The resulting improvement is illustrated with numerical simulations. Preliminary theoretical results are provided and it is shown how the proposed criteria connects to new entropy maximization problems.

A. Polar martingale

In order to show that polarization happens, namely that

$$|\{i \in [n] : I(U_i; Y^n U^{i-1}) \in (\varepsilon, 1 - \varepsilon)\}| \rightarrow 0,$$

it is helpful to rely on a random process having a uniform measure on the possible realizations of $I(U_i; Y^n U^{i-1})$, so that counting the number of such mutual informations in $(\varepsilon, 1 - \varepsilon)$ is obtained by evaluating the probability that the process lies in this interval. The process is defined by taking $\{B_n\}_{n \geq 1}$ to be

---

1If the channel is symmetric the generator matrix is indeed obtained by deleting rows, otherwise in addition to deleting rows one may also have to translate the codewords (affine code)
The larger the spread binary input channels (the tick on the horizontal axis is at 1 and the tick on vertical axis is at 1/4).

The polarization result is then expressed as

$$I(W) = \arg \max_{K \in \mathcal{M}_2(F_q)} I(W^+(W, K))$$

where $W^+(W, K)$ is the channel $u_2 \mapsto Y_1 Y_2 u_1$, $(Y_1, Y_2)$ are the output of two independent uses of $W$ when $(x_1, x_2) = (u_1, u_2)K$ are the inputs. We call $K^*$ the 2-optimal kernel for $W$.

We consider in particular channels which are ‘additive noise’, in which case one can equivalently study the ‘source’ version of this problem as follows:

$$\lambda^*(\mu) = \arg \max_{\lambda \in \mathbb{R}} H(X + \lambda X')$$

where $X, X'$ are i.i.d. under $\mu$. As discussed next, the connection with previous problem is obtained by choosing

$$K^*(W) = \left[ \begin{array}{cc} 1 & 0 \\ \lambda^*(\mu) & 1 \end{array} \right],$$

where $\mu$ is the distribution of the noise of the channel $W$.

III. EQUIVALENT PROBLEMS AND DEFINITIONS

A general kernel is a matrix over $\mathbb{F}_q$ which is invertible. For $2 \times 2$ matrices, let $K = [a b; c d]$ be such a matrix and consider $(X_1, X_2)$ to be i.i.d. under $\mu$ over $\mathbb{F}_q$ and $(U_1, U_2) = (X_1, X_2)K$. Then, since $K$ is invertible, we have

$$H(\mu) = H(X_1, X_2) = H(U_1, U_2) = H(U_1) + H(U_2|U_1)$$

and

$$H(U_1) - H(\mu) = H(U_2|U_1)$$

is the entropy spread gained by using the transformation $K$. To maximize the spread, one may maximize $H(U_1) = H(aX_1 + cX_2)$ over the choice of $a$ and $c$, or simply $H(X_1 + cX_2)$ over the choice of $c$ (this leads to the definition of $\lambda^*$). Hence, the maximization problem depends on the variable $c$, $a$ can be set to 1, and $b$ and $d$ need to ensure that $K$ is invertible, which leads w.l.o.g. to a kernel of the form $K = [1 b; c d]$. Note that to maximize the spread, one may alternatively minimize $H(U_2|U_1) = H(X_2|X_1 + cX_2)$.

IV. RESULTS FOR $F_3$

**Lemma 1.** Let $X, X'$ be i.i.d. random variables valued in $\mathbb{F}_3$, then

$$H(X + X') \leq H(X - X').$$

(The arithmetic is over $F_3$.)

**Proof outline:** Let $p$ denote the distribution of $X + X'$ and $\tilde{p}$ denote the distribution of $X - X'$. Note that

$$\|F(p)\|_2 = \|F(\tilde{p})\|_2$$

where $F$ is the Fourier transform and $\|\cdot\|_2$ the $l_2$-norm. Hence,

$$\|p\|_2 = \|\tilde{p}\|_2.$$  

This implies that $p$ and $\tilde{p}$ lie on an Euclidean ball (and on the simplex). Moreover, note that $\tilde{p}$ is such that $\tilde{p}(1) = \tilde{p}(2)$ and
\[\hat{p}(0) \geq \hat{p}(1).\] Finally, because of the triangular shape of the entropy ball, one can show that the entropy ball passing by \(\hat{p}\) is contained in the Euclidean ball, implying the desired result.

**Corollary 1.** For a probability distribution \(\mu\) over \(\mathbb{F}_3\),
\[
\lambda^*(\mu) = 2 \\
\text{if } \mu(1) \neq \mu(2), \text{ and } \lambda^*(\mu) = \{1, 2\} \text{ if } \mu(1) = \mu(2).
\]

The gap between \(H(X + X')\) and \(H(X - X')\) is illustrated in Figure 2. Figure 3 illustrates the improvements of the error probability of a polar code using the kernel \([\frac{1}{2} \ 0\]\) instead of \([\frac{1}{2} \ 1]\) for a block length \(n = 1024\) when the channel if an additive noise channel over \(\mathbb{F}_3\) with noise distribution \(\{0.7, 0.3, 0\}\).

**V. RESULTS FOR \(\mathbb{F}_q\)**

When \(\mu\) is over \(\mathbb{F}_q\) with \(q \geq 5\), \(\lambda^*(\mu)\) varies with \(\mu\). One can check numerically that for the distribution \(\{0, 0.8, 0.1, 0.1, 0, 0\}\) we have \(\gamma^* = 4\) whereas for the distribution \(\{0.7, 0.2, 0.1, 0, 0\}\) we have \(\gamma^* = \{2, 3\}\).

**Lemma 2.** Let \(\mu\) be a probability distribution over \(\mathbb{F}_q\) with support \(S_\mu\). If there exists \(\gamma \in \mathbb{F}_q\) such that
\[
|S_\mu + \gamma S_\mu| = |S_\mu|^2 \quad (16)
\]
then
\[
H(X_2|X_1 + \gamma X_2) = 0 \quad (17)
\]
where \(X_1, X_2\) are i.i.d. under \(\mu\).

Note: the condition on the support could me simplified but as such it makes the conclusion of the Lemma immediate. Also note that \(\lambda\) for which \(H(X_2|X_1 + \gamma X_2) = 0\) is clearly optimal to maximize the spread.

**Proof of Lemma 2:** The condition \(|S_\mu + \gamma S_\mu| = |S_\mu|^2\) ensures that knowing \(x_1 + \gamma x_2\) with \(x_1, x_2 \in S_\mu\) allows to exactly recover both \(x_1\) and \(x_2\).

**Examples of distributions satisfying (16):** Let \(\mu\) over \(\mathbb{F}_5\) be such that \(S_\mu = \{0, 1\}\). Picking \(\gamma = 2\), one obtains \(2S_\mu = \{0, 2\}\) and \(S_\mu + 2S_\mu = \{0, 1, 2, 3\}\), and (16) is verified. In this case, using \(\gamma = 1\) can only provide a strictly smaller spread since it will not set \(H(X_2|X_1 + \gamma X_2) = 0\). It is hence better to use the 2-optimal kernel \([\frac{1}{2} 0]\) rather than the original kernel \([\frac{1}{2} 1]\), and as illustrated in Figure 4, this leads to significant improvements in the error probability at finite block length. Also note that a channel with noise \(\mu\) satisfying (16) has positive zero-error capacity, which is captured by the 2-optimal kernel as shown with the rapid drop of the error probability (it is 0 at low enough rates since half of the synthesized channels have noise entropy exactly zero). If \(\mu\) is close to a distribution satisfying (16), the error probability is also significantly improved with respect to the original kernel \([\frac{1}{2} 1]\).

Over \(\mathbb{F}_7\), let \(\mu\) be such that \(S_\mu = \{0, 1, 2\}\). Picking \(\gamma = 2\), one obtains \(2S_\mu = \{0, 2, 4\}\) and (16) does not hold. However, picking \(\gamma = 3\) leads to \(3S_\mu = \{0, 3, 6\}\) and (16) holds. Therefore, \(\gamma^*\) varies with \(q\).

Over \(\mathbb{F}_q\), defining \(k = \lfloor \sqrt{q} - 1 \rfloor\), if \(S_\mu = \{0, 1, \ldots, k - 1\}\) and \(\gamma = k\), (16) holds.

**VI. DISCUSSION**

We have shown that over \(\mathbb{F}_q\), the martingale spread can be significantly enlarged by using 2-optimal kernels rather than the original kernel \([\frac{1}{2} 1]\). In some cases, this leads to significant improvements on the error probability of polar codes, even at low block length \((n = 1024)\). We have considered additive noise channels, and shown that the improvement is particularly significant when the noise distribution...
is concentrated on ‘small’ supports. For distributions which have full support, the improvement may not be significant. It would be interesting to investigate the improvement for the AWGN channel, and check whether this channel falls in the cases where the improvement is significant. It would also be interesting to consider polar codes that have a kernel which changes adaptively in the recursive construction. This will require a more elaborate decoding algorithm but may not hurt the complexity. Finally, it would be interesting to show analytical results on the error probability rate or exponent of 2-optimal kernels, to consider extensions to $k \times k$ rather than $2 \times 2$ kernels, or to consider other criteria for the optimization of the kernel.

ACKNOWLEDGEMENT

I would like to thank Emre Telatar for helpful comments and stimulating discussions.

REFERENCES

Sumset Differential Entropy Bounds

Ioannis Kontoyiannis
Department of Informatics
Athens University of Economics & Business
Patission 76, Athens 10434, Greece
Email: yiannis@aueb.gr
Web: http://pages.cs.aueb.gr/ yiannisk/

Abstract—Terence Tao recently developed a series of new inequalities for the (discrete) Shannon entropy, which parallel sumset and inverse sumset bounds in additive combinatorics. We examine some of their natural analogs for differential entropy. Since the main ingredient in Tao’s proofs (the submodularity of the entropy function) fails in the continuous case, interesting new phenomena arise. We review some known results and present a series of new differential entropy inequalities. These results are based, in part, on joint work with Mokshay Madiman.

I. ADDITIVE COMBINATORICS AND ENTROPY

The field of additive combinatorics [see, e.g., [7] for an introduction] is the theory of additive structures in sets equipped with a group structure. The prototypical example is the study of arithmetic progressions in sets of integers, as opposed to the multiplicative structure that underlies prime factorization and much of classical combinatorics and number theory. There have been several major developments and a lot of high-profile mathematical activity in additive combinatorics in recent years, with perhaps the most famous example being the celebrated Green-Tao theorem on the existence of arbitrarily long arithmetic progressions within the primes.

An important collection of tools in the study of additive combinatorics is a variety of sumset inequalities. Here, the sumset \( A + B \) of two discrete sets \( A \) and \( B \) is defined as, \( A + B = \{a + b : a \in A, b \in B\} \), and a sumset inequality is an inequality connecting the cardinality \( |A + B| \) of \( A + B \) with the cardinalities of \( A \) and \( B \). Therefore, roughly and somewhat incorrectly speaking, it might be said that additive combinatorics gives bounds on the sizes of discrete additive sets.

In this setting, we recall the natural connection between entropy and set cardinality established by the AEP: The entropy \( H(X) \) can be thought of as the logarithm of the effective cardinality of the alphabet of a discrete random variable \( X \). This suggests a correspondence between bounds for the cardinalities of sumsets like, e.g., \(|A + B|\), and corresponding bounds for the entropy of sums of discrete random variables, e.g., \(H(X + Y)\). This connection appears to have first been identified by Imre Ruzsa, and in the last few years it has also been explored in different directions by, among others, Tao and Vu [6], Lapidoth and Pete [1], Ruzsa [4], Madiman, Marcus and Tetali [3], Madiman and Kontoyiannis [2], and Tao [5].

II. SUMSET BOUNDS AND DIFFERENTIAL ENTROPY

This analogy can be carried further: The continuous AEP states that the differential entropy \( h(X) \) of a continuous random vector \( X \) can be thought of as the logarithm of the (Euclidean or Lebesgue) “volume of the effective support” of \( X \). It is then natural to consider whether the recent discrete sumset entropy bounds can be extended to the continuous case.

Taking as our starting point the results in Tao’s recent work [5], we provide natural differential entropy analogs for the following inequalities:
- Rusza triangle inequality
- Sum-difference inequality
- Pl"{a}nmecke-Rusza inequality
- Iterated sum bound
- Balog-Szemer"{e}di-Gowers lemma
- Rusza’s covering lemma
- Green-Rusza-Freiman inverse-sumset theorem

The difficulty as well as the excitement of this work stem from the fact that, in the continuous case, the natural generalizations of the proofs of the discrete versions of many of these results fail at a deep level. One reason is that the main ingredient in earlier proofs is the submodularity of the discrete entropy functional, which does not hold for differential entropy. And, further, it is the overall structure of the proofs that does not carry on to the continuous case: Not the method, but the actual intermediate steps fail to hold for differential entropy. Therefore, it is necessary to employ different tools and fundamentally new proof strategies.

REFERENCES

Coding Theory and Compressed Sensing through Convex Relaxations

Arash Saber Tehrani, Alexandros G. Dimakis
Dept. of Electrical Eng. University of Southern California
Los Angeles, CA, USA
emails: {saberteh,dimakis}@usc.edu

I. INTRODUCTION

Sparse approximation theory and compressed sensing have recently received significant attention in signal processing and statistics. We focus on two developments that connect this area with coding theory and allow the translation of results and techniques. The first involves mathematical links [1]–[3] between two linear programming relaxations: basis pursuit for CS reconstruction and the fundamental polytope relaxation for the decoding of binary linear codes [4], [5]. In coding theory, the linear-program (LP) relaxation was introduced by Feldman, Wainwright and Karger [4] and is closely related to the work by Koetter and Vontobel [5]. This work defined the so-called fundamental polytope and structures called pseudocodewords that cause problems to channel decoders (see also the work of Wiberg [6] and Forney et al. [7].

Our recent results [1], [8] connect this theory to the basis pursuit LP relaxation used in compressed sensing. The high-level message is that ‘good’ channel codes can be used over the reals as provably good CS measurement matrices under basis pursuit. This connection allowed us to solve a well-known open problem in compressed sensing, the deterministic construction of measurement matrices with an order-optimal number of rows [1], [8].

The second connection involves the nature of iterative message-passing algorithms used both for channel coding and compressed sensing. A number of iterative algorithms (e.g. [9]–[13]) have been proposed for compressed sensing recovery, with the benefit of faster decoding complexity and comparable performance to basis pursuit. Very similar iterative message-passing algorithms have been developed and used for the decoding of error-correcting codes [14]. More interestingly, the techniques for their analysis have recently found applicability for compressed sensing. Channel-coding-inspired message-passing decoders for compressed sensing problems have been also discussed in [15]–[17].

The fact that both iterative message-passing algorithms and LP relaxations are mathematically connected for these two problems is not an accident: For channel coding, the LP relaxation is currently understood as an approximation to iterative message-passing [18] and the theory of pseudocodewords allows a rigorous non-asymptotic theory for decoding and design of error-correcting codes under polynomial time decoding algorithms.

II. RECENT RESULTS

Assume we observe $m$ linear measurements of an unknown vector $e \in \mathbb{R}^n$:

$$
H \cdot e = s,
$$

where $H$ is a real-valued matrix of size $m \times n$, called the measurement matrix. When $m < n$, this is an underdetermined system of linear equations and one fundamental compressed sensing problem involves recovering $e$ assuming that it is also $k$-sparse, i.e. it has $k$ or less non-zero entries.

The sparse approximation problem goes beyond exactly sparse vectors and requires the recovery of a $k$-sparse vector $\hat{e}$ that is close to $e$, even if $e$ is not exactly $k$-sparse itself. Recent breakthrough results [19], [20] showed that it is possible to construct measurement matrices with $m = O(k \log(n/k))$ rows that recover $k$-sparse signals exactly in polynomial time. These results rely on randomized matrix constructions and establish that the optimal number of measurements will be sufficient with high probability over the choice of the matrix and/or the signal. Unfortunately, the required properties of Restricted Isometry Property (RIP) [20], Nullspace [21] and high expansion (expansion quality $\epsilon < 1/6$) have no known ways to be deterministically constructed or efficiently checked.

There are several explicit constructions of measurement matrices (e.g. [22], [23]) which, however, require a slightly sub-optimal number of measurements ($m$ growing super-linearly as a function of $n$, for $k = p \cdot n$).

We focus on the linear sparsity regime where $k$ is a fraction of $n$ and optimal number of measurements will also be a fraction of $n$. The explicit construction of measurement matrices with an optimal number of rows is a well-known open problem in compressed sensing theory (see e.g. [24] and references therein). A closely related issue is that of checking or certifying in polynomial time that a given candidate matrix has good recovery guarantees or satisfies the RIP.

Consider a sparse matrix $H$ in $\{0,1\}^{m \times n}$ that has $d_c$ ones per row and $d_v$ ones per column. If the bipartite graph
corresponding to $H$ has $\Omega(\log n)$ girth, then for $k = p \cdot n$ and an optimal number of measurements $m = c_2 \cdot n$, we show that $H$ offers $\ell_1/\ell_2$ sparse approximation under basis pursuit decoding. Our technical requirement of girth, unlike expansion or RIP, is easy to check and several deterministic constructions of matrices with $m = c_1 \cdot n$ and $\Omega(\log n)$ exist, starting with the early construction in Gallager’s thesis [25], and progressive edge-growth Tanner graphs.

Our result is a weak bound, also known as a ‘for-every signal’ guarantee [24]. This means that we have a fixed deterministic matrix and show the $\ell_1/\ell_2$ sparse approximation guarantee with high probability over the support of the signal. To the best of our knowledge, this is the first deterministic construction of matrices with an optimal number of measurements.

Our proof relies on the discussed connections of channel decoding LP and compressed sensing. We rely on a primal-based density evolution technique initiated by Koetter and Vontobel [26] and analytically strengthened in the break-through paper of Arora et al. [27] that established the best known finite-length threshold results for LDPC codes under LP decoding. In our presentation we will survey this area, present recent results and discuss open problems.

REFERENCES

Polar Codes for the Deterministic Broadcast Channel

Naveen Goela∗†, Emmanuel Abbe∗, and Michael Gastpar∗†

∗ Ecole Polytechnique Fédérale de Lausanne, Faculté Informatique et Communications, CH-1015 Lausanne, Switzerland
Email: {naveen.goela, emmanuel.abbe, michael.gastpar}@epfl.ch
† University of California, Berkeley, Department of E.E.C.S., Berkeley, CA 94720-1770 USA
Email: {ngoela, gastpar}@eecs.berkeley.edu

Abstract—We construct and analyze polar codes for the deterministic broadcast channel (DBC) which is comprised of one source transmitting independent messages to m receivers. For any input distribution, polar codes with rate-splitting achieve all optimal rates within the corresponding multi-user rate region, a polyhedron in $\mathbb{R}^m_+$. For fixed m, the encoding and decoding complexities remain $O(n \log n)$ with $O(2^{-n})$ average block error probability where n is the block length and $0 < \beta < \frac{1}{2}$. In addition, a related joint polarization coding scheme without rate-splitting achieves points on the dominant face of the polyhedron.

I. INTRODUCTION

Traditionally, Arıkan’s channel polarization [1] was shown to achieve the capacity of binary-input symmetric channels with low encoding and decoding complexity. Subsequently, source polarization was introduced for lossless compression of binary sources [2], lossless Slepian-Wolf communication based on a reduction to the single-source setting [2], and lossy source compression [3]. In this note, we analyze the performance of polar codes for the noiseless multi-receiver deterministic broadcast channel (DBC). Several researchers have characterized (functional and random coding) dualities between the DBC, Slepian-Wolf distributed source coding, and deterministic multiple-access channel (MAC), see e.g. [4].

Polar codes were analyzed for the latter two scenarios using both single-user and joint polarization techniques in [5] and [6]; however, polar codes for the DBC shed new light on explicit, low complexity, capacity-achieving codes in contrast to random binning and joint typicality encoding strategies.

A. The Multi-Receiver DBC

Definition 1: (Deterministic BC) The DBC with m receivers consists of an input alphabet $\mathcal{X}$, output alphabets $\{\mathcal{Y}_i\}_{i=1}^m$, and m deterministic functions $y_i = f_i(x)$ for $i \in [m]$, $x \in \mathcal{X}$, $y_i \in \mathcal{Y}_i$.

A $((2^{nR_1}, 2^{nR_2}, \ldots, 2^{nR_m}), n)$ code for the DBC with independent messages consists of an encoder

$$x^n : 2^{nR_1} \times 2^{nR_2} \times \cdots \times 2^{nR_m} \rightarrow \mathcal{X}^n,$$

and m decoders specified by $\hat{W}_i : \mathcal{Y}_i^n \rightarrow 2^{nR_i}$. The probability of error $P_e^{(n)}$ is defined to be the probability that the decoded message is not equal to the transmitted message, i.e.,

$$P_e^{(n)} = P \left( \bigvee_{i=1}^m \hat{W}_i(Y^n) \neq W_i \right),$$

where the message $(W_1, W_2, \ldots, W_m)$ is assumed to be uniformly distributed over $2^{nR_1} \times 2^{nR_2} \times \cdots \times 2^{nR_m}$.

Theorem 1: (Channel Capacity Region [7] [8]) Let $R = (R_1, R_2, \ldots, R_m)$. The capacity region $C_{DBC}$ given deterministic functions $f_j(\cdot)$ is specified as

$$C_{DBC} := \text{clo} \left( \left\{ \frac{1}{n} \sum_{i \in S} R_i < H(Y(S)), \forall S \subseteq [m] \right\} \right),$$

where

$$R(X) = \left\{ R \in \mathbb{R}_+^m \mid \sum_{i \in S} R_i < H(Y(S)), \forall S \subseteq [m] \right\}. \quad (2)$$

The notation $Y(S) = \{Y_i : i \in S\}$, clo(S) the convex hull operation over set S. The union in Eqn. (1) is over random variable $X \sim P_X(x)$ which induces a joint distribution over $X, Y_1, Y_2, \ldots, Y_m$ via $Y_i = f_i(X)$.

B. Random Coding vs. Low Complexity Codes

Shannon-theoretic coding strategies for broadcast channels (see e.g. [9] or [10, Lec. 9] for a survey) rely on random coding. Specifically for the DBC, random binning and joint typicality encoding yield the optimal rates. At the encoder, assuming for a moment $m = 2$ receivers, a fixed $P_X(x)$ induces a joint distribution $P_{Y_1,Y_2}(y_1,y_2)$. All $y_1^n$ and $y_2^n$ sequences are randomly assigned bin indices $i \in [2^{nR_1}]$ and $j \in [2^{nR_2}]$ respectively. The rates $R_1$ and $R_2$ are selected to ensure that at least one jointly typical pair $(y_1^n, y_2^n)$ exists in each $(i,j)$ product bin. To send information i to the first receiver, and information j to the second receiver, the encoder selects a jointly typical pair $(y_1^n, y_2^n)$ in the product bin $(i,j)$ and computes the $x^n$ sequence which results in $y_1^n$ and $y_2^n$. A valid $x^n$ codeword exists due to the deterministic property of $f_1(\cdot)$ and $f_2(\cdot)$. Each receiver decodes its message by looking up the bin index of its received sequence. Random binning and joint typicality encoding require a search over an exponential number of sequences.

Designing low-complexity codes for the DBC requires “binning” in a sense, and constraint satisfaction to ensure an $x^n$ codeword exists which generates the proper $y_1^n$ and $y_2^n$ output sequences. If $(y_1^n, y_2^n)$ are jointly typical with respect to
where the first region is achieved with input distribution \( P_{X_1,Y_2}(y_1,y_2) \) induced by \( P_X(x) \), a matching codeword \( x^m \) is guaranteed because \( P_{X_1,Y_2}(y_1,y_2) \) has zero mass over inconsistent pairs \((y_1,y_2)\). In [11], the authors propose sparse linear coset codes to emulate binning (i.e., the syndrome is the message), combined with survey propagation to enforce the DBC channel constraints. In [12], the authors relate interference management in wireless scenarios with low-complexity coding over the DBC, and in [13], the authors describe a reinforced belief propagation algorithm with non-linear coding for the DBC. However, going beyond ad-hoc strategies in the present paper, we design polar codes that are low-complexity and provably optimal for achieving points in the DBC capacity region.

C. Simplifying Assumptions

We make the following simplifications to Definition 1: (a) The number of receivers \( m = 2 \); (b) Polar code design is for achieving vertices of the polyhedron given in Eqn. (2) for fixed \( P_X(x) \); (c) The output alphabets \( Y_i \) are all \( \mathbb{F}_2 \). The first two simplifications are without loss of generality. In particular (a) is for compactness of notation and our coding scheme applies for arbitrary pairs \((m,n)\). and for (b) it was shown that an arbitrary point in the achievable rate region of a DBC with \( m \) receivers maps to a corresponding vertex point in the achievable region of a DBC with \((2m-1)\) receivers if rate-splitting is used [14]. Item (c) is a mild restriction due to the algebraic construction of polar codes. It is possible to generalize the polar code construction to include output alphabets of prime, prime power, and different cardinalities.

Example 1: (Blackwell Channel) In Fig. 1, the Blackwell channel is depicted with \( X = \{0,1,2\} \) and \( Y_1 = \mathbb{F}_2 \). For any fixed distribution \( P_X(x) \), it is seen that \( P_{Y_1,Y_2}(y_1,y_2) \) has zero mass for the pair \((1,0)\). Let \( \alpha \in [\frac{1}{3}, \frac{2}{3}] \), and \( h_0(\cdot) \) denote the binary entropy function. Due to the symmetry of this channel, the capacity region is the union of two regions,

\[
\{(R_1, R_2): R_1 \leq h_0(\alpha), R_2 \leq h_0(\frac{\alpha}{2}), R_1 + R_2 \leq h_0(\alpha + \alpha), R_1 + R_2 \leq h_0(\alpha + \alpha)\},
\]

where the first region is achieved with input distribution \( P_X(0) = P_X(1) = \frac{\alpha}{2} \), and the second region is achieved with \( P_X(1) = P_X(2) = \frac{\alpha}{2} \) [10, Lec. 9]. The sum rate is maximized for a uniform input distribution which yields a pentagonal achievable rate region: \( R_1 \leq h_0(\frac{1}{3}), R_2 \leq h_0(\frac{1}{3}), R_1 + R_2 \leq \log_2 3 \). We construct polar codes to achieve the vertices of this pentagon, and more generally the vertices of a polyhedron in \( \mathbb{R}^m \).

II. POLAR CODES

Using the simplifications of Sec. I-C for compactness, we analyze coding over \( n \) channel uses of the DBC. Let \( m = 2 \) receivers, \( i \in [m] \) and \( j \in [n] \). The channel input is given by \( X = (X_1,X_2,\ldots,X_n) \) where \( X_j \sim P_X \) are i.i.d. random variables. The channel output \( Y_{ij} = f_i(X_j) \) and \( Y_{ij} \sim P_{Y_1,Y_2} \) where \( P_{Y_1,Y_2} \) is the joint distribution on \( \mathbb{F}_2 \) induced by \( P_X(x) \).

A. Polarization

Define the following row vectors of a matrix,

\[
\bar{Y}_1 = \begin{bmatrix} Y_{1,1} & Y_{1,2} & Y_{1,3} & \cdots & Y_{1,n} \end{bmatrix},
\]

the joint distribution of all output variables is

\[
P_{Y_{1,Y_2}}(y_1,y_2) = \prod_{j=1}^n P_{Y_{1,Y_2}}(y_{1,j},y_{2,j}).
\]

The polar algebraic transformation is defined in terms of an invertible matrix \( G_n \in \mathbb{F}_2^{n \times n} \) where \( n = 2^\ell \) for \( \ell \geq 1 \) [2]. Denote the polar transformation of the output variables by

\[
\bar{U}_1 = \begin{bmatrix} U_{1,1} & U_{1,2} & U_{1,3} & \cdots & U_{1,n} \end{bmatrix},
\]

where the transformed variables are defined

\[
\bar{U}_1 = \begin{bmatrix} U_{1,1} & U_{1,2} & U_{1,3} & \cdots & U_{1,n} \end{bmatrix}.
\]

Theorem 2 (Polarization [2]): Let the random variables \( \bar{Y}_1, \bar{Y}_2, \bar{U}_1, \bar{U}_2 \) be related according to Eqs. (3)-(6). As \( n \to \infty \), for any \( \epsilon \in (0,1) \),

\[
\left\{ j \in [n]: H(U_{1,j}|U_{1,j-1}^{j-1}) \geq 1 - \epsilon \right\} \to nH(Y_1),
\]

\[
\left\{ j \in [n]: H(U_{2,j}|U_{2,j-1}^{j-1}, \bar{Y}_1) \geq 1 - \epsilon \right\} \to nH(Y_2|Y_1).
\]

Theorem 2 shows how the randomness of \( \bar{Y}_1 \) and \( \bar{Y}_2 \) is extracted into \( nH(Y_1) \) and \( nH(Y_2|Y_1) \) nearly uniform components.

B. Message Indices

To design a polar code for the DBC, we identify the indices of variables in \( \bar{U}_1 \) and \( \bar{U}_2 \) that have (conditional) entropy close to one. The Bhattacharyya parameter is closely related to the conditional entropy.

Definition 2: (Bhattacharyya Parameter): Let \( (S,V) \sim P_{S,V} \) where \( S \in \{0,1\} \) and \( V \in \mathcal{V} \) where \( \mathcal{V} \) is an arbitrary countable alphabet. The Bhattacharyya parameter \( Z(S|V) \in [0,1] \) is defined

\[
Z(S|V) = 2 \sum_{v \in \mathcal{V}} P_S(v) \sqrt{P_{S|V}(0|v)P_{S|V}(1|v)}.
\]

As shown by Arikan in [2], \( H(S|V) \) is close to 0 or 1 if and only if \( Z(S|V) \) is close to 0 or 1. We define the message indices for \( m = 2 \) receivers based on the Bhattacharyya parameter of variables comprising the random vectors \( \bar{U}_1, \bar{U}_2, \bar{Y}_1, \bar{Y}_2 \).

\[
I_1^{(n)} = \{ j \in [n]: Z(U_{1,j}|U_{11}^{j-1}) \geq 1 - 2\delta_n \},
\]

\[
I_2^{(n)} = \{ j \in [n]: Z(U_{2,j}|U_{21}^{j-1}, \bar{Y}_1) \geq 1 - 2\delta_n \},
\]
where $\delta_n > 0$. According to the rate of polarization [15], we will select $\delta_n \downarrow 0$. As a result of Theorem 2 and the properties of the Bhattacharyya source parameter, for any $\tau > 0$ there exists an $n$ large enough so that both $|I_1^{(n)}| \geq n(H(Y_1) - \tau)$ and $|I_2^{(n)}| \geq n(H(Y_2|Y_1) - \tau)$.

III. Bounding the Total Variation Distance between Probability Measures

To code over $n$ channel uses of the DBC, we insert uniformly random message bits into the position indexed by message sets $I_1^{(n)}$ and $I_2^{(n)}$. We now begin to establish that for any $\tau > 0$, with $n \to \infty$, the message rates $R_1 \geq H(Y_1) - \tau$ and $R_2 \geq H(Y_2|Y_1) - \tau$ are achievable with $P_n(\tau)$ to $0$. For a DBC with $m = 2$ receivers, the rate pair $(R_1, R_2)$ approaches one vertex point of the pentagonal rate region (defined more generally as a polyhedron in Eqn. (2)). Other optimal vertex points of the polyhedral rate region are achieved similarly by changing the ordering of receivers.

A. Two Probability Measures on a Set of Binary Sequences

We define two probability measures on the set of all $n$-length binary sequences (here we assume $m = 2$) which will help in bounding the average block error probability $P_n(\tau)$ of our encoding/decoding scheme. The i.i.d. joint distribution $P_{U_1, U_2}(u_1, u_2)$ defined in Eqn. (4) specifies a joint distribution $P_{U_1, U_2}(u_1, u_2)$ via the one-to-one linear transform $G_n$ of Eqn. (5). We write this distribution as

$$P_{U_1, U_2}(u_1, u_2) = P_{U_1}(u_1)P_{U_2}(u_2|u_1).$$

We further factorize the above as follows,

$$P_{U_1}(u_1) = \prod_{j=1}^{n} P_{U_{1j}}(u_{1j}^{(j-1)}),$$

$$P_{U_2}(u_2|u_1) = \prod_{j=1}^{n} P_{U_{2j}|U_{1j}^{(j-1)}}(u_{2j})|u_{1j}^{(j-1)}, u_1).$$

We now define closely related conditional probability distributions $\forall j \in [n]$,

$$Q_{U_{1j}}(u_{1j}^{(j-1)}|u_{1j}^{(j-1)}) \triangleq \begin{cases} \frac{1}{2}, & \text{if } j \in I_1^{(n)}, \\ P_{U_{1j}}(u_{1j}^{(j-1)}), & \text{otherwise} \end{cases}$$

$$Q_{U_{2j}|U_{1j}^{(j-1)}}(u_{2j}|u_{2j}^{(j-1)}, u_1) \triangleq \begin{cases} \frac{1}{2}, & \text{if } j \in I_2^{(n)}, \\ P_{U_{2j}|U_{1j}^{(j-1)}}(u_{2j}^{(j-1)}, u_1), & \text{otherwise} \end{cases}$$

Then analogous to Eqn. (11) and Eqn. (12),

$$Q_{U_1}(u_1) \triangleq \prod_{j=1}^{n} Q_{U_{1j}}(u_{1j}^{(j-1)}|u_{1j}^{(j-1)}),$$

$$Q_{U_2}(u_2|u_1) \triangleq \prod_{j=1}^{n} Q_{U_{2j}|U_{1j}^{(j-1)}}(u_{2j})|u_{2j}^{(j-1)}, u_1).$$

Essentially $Q_{U_1}(u_1) = \left(\frac{1}{2}\right)^{2|I_1^{(n)}|} g_1(u_1)$ and $Q_{U_2}(u_2|u_1) = \left(\frac{1}{2}\right)^{2|I_2^{(n)}|} g_2(u_2|u_1)$ where $g_1(u_1)$ and $g_2(u_2|u_1)$ are nearly indicator functions; i.e., the distributions are approximately composed of i.i.d. Bernoulli(1/2) random variables.

B. Total Variation Bound

Theorem 3: (Total Variation Bound for Joint Distributions)

Let probability measure $P_{U_1, U_2}(u_1, u_2)$ be defined as in Eqn. (10). Let $Q_{U_1, U_2}(u_1, u_2) = Q_{U_1}(u_1)Q_{U_2}(u_2|u_1)$ be defined via Eqn. (13) and Eqn. (14). Then the total variation distance between $P$ and $Q$ is bounded as

$$\sum_{u_1, u_2} |Q(u_1, u_2) - P(u_1, u_2)| \leq 2|I_1^{(n)}|\delta_n + 2|I_2^{(n)}|\delta_n,$$

where we set $\delta_n = \frac{2^{-n/2}}{m}$ for $0 < \beta < \frac{1}{2}$. Thus the total variation distance is bounded by $O(2^{-n})$.

Proof: Let probability distribution $Q_{U_1, U_2}(u_1, u_2) \triangleq P_{U_1}(u_1)Q_{U_2}(u_2|u_1)$. Observe by the triangle inequality for total variation distance,

$$\sum_{u_1, u_2} |Q(u_1, u_2) - P(u_1, u_2)| \leq \sum_{u_1, u_2} |Q(u_1, u_2) - L(u_1, u_2)|$$

$$+ \sum_{u_1, u_2} |L(u_1, u_2) - P(u_1, u_2)|.$$
message bits for the first receiver. Likewise, the bits $u_{2j}$ with indices $j \in I(n)$ are set to the uniformly selected message bits for the second receiver. Note that the message indices $I(n)$ and $I(2n)$ defined in Eqs. (8) and Eqn. (9) are computed once during polar code construction, and do not depend on the realization of variables during the encoding process. The non-message indices in $\bar{u}_1$ and $\bar{u}_2$ are obtained via a bit-by-bit successive randomization calculation as follows.

$$\forall j \notin I(2n) : u_{1j} = \begin{cases} 0, & \text{w.p. } P_{U_1 | U_1^{j-1}}(0 | u_{1j-1}), \\ 1, & \text{w.p. } P_{U_1 | U_1^{j-1}}(1 | u_{1j-1}), \end{cases}$$

$$\forall j \notin I(2n) : u_{2j} = \begin{cases} 0, & \text{w.p. } P_{U_2 | U_2^{j-1}}(0 | u_{2j-1}, \bar{u}_1), \\ 1, & \text{w.p. } P_{U_2 | U_2^{j-1}}(1 | u_{2j-1}, \bar{u}_1). \end{cases}$$

The probabilities $P_{U_1 | U_1^{j-1}}(0 | u_{1j-1})$ and $P_{U_2 | U_2^{j-1}}(0 | u_{2j-1}, \bar{u}_1)$ above are efficiently computable in $O(n \log n)$ time due to Arıkan’s recursive polarization decomposition (see [2] for the recursive likelihood formulas).

Having formed the binary sequences $u_1$ and $u_2$, the encoder applies the inverse polar transform, $[y_1; y_2] = [u_1; u_2]G_n^{-1}$ where $G_n^{-1} = G_n$ in the binary case. Each channel symbol $x_j$ of the codeword $x^n$ is formed by taking any $x_j \in \gamma_{m-1} \cap \gamma_{m-2}$ if this intersection set is non-empty, otherwise the encoder declares a block error. If the encoder succeeds in transmitting a codeword, each receiver $i \in [m]$ obtains the sequence $y_i$ exactly and applies the $G_n$ transform to recover $u_i$, exactly. Knowing the location of the message indices in $u_i$, each receiver decodes its own message correctly.

B. Error Performance

An error event for the DBC occurs exclusively at the decoder if it is unable to construct a codeword $x^n$. In the following theorem, we assume $m = 2$. $P_{Y_1, Y_2}(y_1, y_2)$ is the induced distribution for arbitrary input distribution $P_X$, and the target rate pair is $(R_1, R_2) = (H(Y_1), H(Y_2 | Y_1))$.

Theorem 4: (Average Block Error Probability) Fix $R_1 < H(Y_1)$, $R_2 < H(Y_2 | Y_1)$, and $0 < \beta < \frac{1}{2}$. There exists a sequence of polar codes of length $n$ for which the achieved rates $(R_1, R_2)$, so that under successive randomized encoding, the expected average block error probability is $P_e^n = O(2^{-n^\beta})$.

Proof: Theorem 2 established that the message sets have cardinalities $|I(n)^1| \rightarrow nH(Y_1)$ and $|I(n)^2| \rightarrow nH(Y_2 | Y_1)$. Let $\mathcal{E}$ denote the set of all $2n$-length binary sequences written as $(y_{11}, y_{21})$ which are inconsistent, i.e. $\exists j \in [n]$ for which $(y_{1j}, y_{2j})$ is not a valid output of the DBC for any input symbol $x_j$. Let $\mathcal{E}$ denote the set of $2n$-length binary sequences $(y_{11}, y_{21})$ that correspond to the inconsistent sequences via the one-to-one polar transform $G_n$. The distribution $P$ is such that $\sum_{(u_1, u_2) \in \mathcal{E}} P_{U_1, U_2} \bar{u}_1, \bar{u}_2 = 0$. Due to Theorem 3,

$$\sum_{(u_1, u_2) \in \mathcal{E}} Q_{U_1, U_2} \bar{u}_1, \bar{u}_2 = O(2^{-n^\beta}).$$

The successive randomized encoding procedure of Sec. IV-A selects a codeword by effectively drawing from the distribution $Q$ for a uniformly selected message. The expected average block error probability of drawing an inconsistent pair is therefore $P_e^n = O(2^{-n^\beta})$.

V. JOINT/SIMULTANEOUS POLARIZATION METHOD

Let matrix $\bar{U} = [\bar{U}_1; \bar{U}_2]$ where $\bar{U} \in \mathbb{F}_2^{2n \times n}$ was defined in Eqn. (6). We make a brief remark about joint polarization methods for the DBC. It was shown in [5, Lemma 1] that,

$$\frac{1}{n} | \{ \bar{y}_j \in [n] : H(U_j | S)(U_j^{-1}) |_{r=e} = \sum_{S \subseteq [m]} \left| \left\{ r - r_1, r_2, \ldots, r_e \right\} \right| \rightarrow 1, \ (19)$$

where here the new notation $U_j$ now denotes the $j$-th column of matrix $\bar{U}$, $U_j(i)$ the $(i,j)$-entry of matrix $\bar{U}$, $U_j[S] = [U_j(i) : i \in S]$, and $U^j = [U_1, \ldots, U_j]$. When comparing Arıkan’s polarization of Theorem 2 with that of Eqn. (19), one sees that the latter involves joint polarization column by column, whereas the former involves successive polarization bit by bit. It can be shown that joint polarization methods allow for DBC coding schemes that achieve (sum rate) points on the dominant face of the polyhedron in $\mathbb{R}_n^m$ without rate-splitting.

VI. CONCLUSION

The polar encoding strategy is a non-linear mapping from $m$ messages to one codeword that requires $O(n \log n)$ complexity to compute. We believe polar codes will be applied in other network communication scenarios to address the high complexity of random binning and joint typicality coding.

ACKNOWLEDGMENT

The authors thank S. B. Korada for the helpful discussions regarding the journal paper [3].

REFERENCES


Polar Codes for Coordination in Cascade Networks

Ricardo Blasco-Serrano, Ragnar Thobaben, and Mikael Skoglund
KTH Royal Institute of Technology and ACCESS Linnaeus Centre
SE-100 44, Stockholm, Sweden
E-mail: {ricardo.blasco, ragnar.thobaben, mikael.skoglund}@ee.kth.se

Abstract—We consider coordination in cascade networks and construct sequences of polar codes that achieve any point in a special region of the empirical coordination capacity region. Our design combines elements of source coding to generate actions with the desired type with elements of channel coding to minimize the communication rate. Moreover, we bound the probability of malfunction of a polar code for empirical coordination. Possible generalizations and open problems are discussed.

I. INTRODUCTION

The limits of coordination in networks have been recently studied from a mathematical point of view (e.g. [1], [2]). Simple questions like how to measure coordination or how much communication is necessary to achieve a desired level of coordination have been posed and to some extent answered. The authors of [1] developed elements of a fundamental theory with two notions of coordination. In the first one, empirical coordination, the sequences of actions generated in the network must have a type that is close to a desired probability distribution. Empirical coordination is closely related to rate-distortion theory [3]. In fact any good code for rate-distortion is useful for empirical coordination and vice versa [1]. The second notion is that of strong coordination. Here the sequences of actions generated by the nodes must be statistically indistinguishable from those obtained by sampling a certain distribution.

In this work we construct sequences of polar codes (PCs) that achieve a special region of the empirical coordination capacity region for two-node and three-node cascade networks. PCs were introduced by Arikan as a method to achieve the capacity of any symmetric binary-input discrete memoryless channel (BI-DMC) [4]. Since then they have emerged as a powerful technique to develop achievability results in information theory with structured codes (as opposed to random coding). Korada and Urbanke established in [5] the optimality of PCs for lossy source coding of (symmetric) discrete memoryless sources (DMS) with binary reproduction alphabets. These results were later extended to non-binary channels and reproduction alphabets in [6] and [7], respectively.

Our constructions combine elements of PCs for source compression with PCs for channel coding. In addition, we show that, in line with [1], common randomness is not necessary for implementing PCs for empirical coordination, although it is useful in the proofs. We use the properties of PCs to extend the application of results on the rate of polarization [8] to PCs designed for empirical coordination. Finally, we discuss possible generalizations of our constructions as well as some open problems for future research.

This paper is organized as follows. In Section II we summarize basic results on empirical coordination and polar codes along with the notation. We analyze a simple two-node network in Section III. This serves as a building block for the cascade network, which is addressed in Section IV. We conclude our work in Section V with a discussion on the results of the paper as well as some open problems.

II. PRELIMINARIES

A. Notation

Scalars are written using normal face \( x \) and vectors using bold face \( \mathbf{x} \). The \( i^{th} \) element of a vector \( \mathbf{x} \) is denoted by \( x_i \). For a given set of natural numbers \( F \) with size \(|F| = N\), \( x_F \) is shorthand for the subvector with elements whose positions belong to \( F \). We use upper case letters for random variables (RVs) \( Y \) and lower case letters for their realizations \( y \). The joint probability distribution on \((X, Y)\) is denoted by \( P_{X,Y}(x, y) \). For convenience we shall alternatively drop the subindices or arguments whenever they are clear from the context. We follow the standard information-theoretic notation from [3].

B. Empirical Coordination

Consider the three-node network in Fig. 1. Node X observes a sequence of \( N \) external actions \( \mathbf{X} \) chosen independently and identically distributed (i.i.d.) according to \( P_X \). Communication from Node X to Node Y and from Node Y to Node Z is possible at rates \( R_1 \) and \( R_2 \) (bits per action), respectively. We use these resources to have Node Y and Node Z generate sequences of actions \( \mathbf{Y} \) and \( \mathbf{Z} \), respectively, with length \( N \) and joint type close to a desired probability function \( P_{Y,Z|X}(P_X) \).

\[ X \sim P_X \]

Fig. 1. Cascade network.

A \((2^NR_1, 2^NR_2, N)\) coordination code for the network in Fig. 1 consists of an encoding, a recoding, and two decoding functions (see [1]). All of them may have access to a source of common randomness (CR) independent of the external actions. Each coordination code induces a joint distribution on the actions \( Q_{X,Y,Z} \).

In this paper we are interested in the joint type of a tuple of action sequences \((x, y, z)\) which is defined as

\[
T_{x,y,z}(x, y, z) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1} \{ (x_i, y_i, z_i) = (x, y, z) \}
\]  (1)
for all \((x, y, z) \in X \times Y \times Z\), where \(\mathbb{1}\{\cdot\}\) is the indicator function. In order to measure the distance between two probability distributions \(P_{X,Y,Z}\) and \(Q_{X,Y,Z}\) we consider their total variation, which is defined as

\[
\|P_{X,Y,Z} - Q_{X,Y,Z}\| \triangleq \frac{1}{2} \sum_{x,y,z} |P(x, y, z) - Q(x, y, z)|.
\]

We say that a triple \((R_1, R_2, P_{X,Y,Z})\) is achievable for empirical coordination if for any \(\epsilon > 0\) there exists a sequence of \((2^{NR_1}, 2^{NR_2}, N)\) coordination codes and a choice of CR such that

\[
\Pr(\|P_{X,Y,Z} - T_{X,Y,Z}\| > \epsilon) < \epsilon
\]

for sufficiently large \(N\) under the distribution induced by the codes. The empirical coordination capacity region, denoted by \(C_{P_X}\), is the closure of the set of achievable triples \((R_1, R_2, P_{X,Y,Z})\).

Due to the nature of PCs we restrict our attention to binary actions \(X\) and \(Z\) (although no restriction is placed on \(Y\)) and to choices of \(P_{Y,Z|X}\) that induce uniform distributions on \(X\) and \(Z\) for the given \(P_X\). All the results in this paper are for the given \(P_X\), \(P_Y\) and \(P_{Z|Y}\) uniformly distributed) of these synthetic channels is noise-free while the rest is virtually useless.

1) Channel Coding: Channel polarization leads naturally to a code construction that achieves the capacity of any symmetric BI-DMC. These codes are known as PCs and consist of two elements: an encoding matrix \(G_N\) and a Successive Cancellation (SC) decoding algorithm. \(G_N\) synthesizes the channels with extremal properties. Fixed (i.e. frozen) bits are put into the “bad” channels, which are those in the frozen set \(F\) defined as

\[
F = \{i : Z(P^{(i)}_0) \geq \delta_N\}
\]

for some \(\delta_N > 0\), where \(Z(P^{(i)}_0)\) denotes the Bhattacharyya parameter (an upper bound on the error probability for uncoded transmission [4]) of the BI-DMC \(P^{(i)}_0\). Information is transmitted at full rate through the rest of the channels, i.e. those in the complement of the frozen set \(F\). The SC decoding algorithm generates sequentially estimates \(\hat{u}_i\) for the information bits using the channel distribution \(P^{(i)}_0(y, \hat{u}^{(i)}_i | u_i)\) and the previous estimates.

The following observation about PCs designed for degraded channels (as defined in [3]) will turn out to be important in the sequel.

Lemma 1 (Lemma 21 in [5]). Let \(P_{Y|X}\) and \(\hat{P}_{Y|X}\) be two BI-DMCs with \(\hat{P}_{Y|X}\) degraded with respect to \(P_{Y|X}\) (denoted by \(\hat{P}_{Y|X} \preceq P_{Y|X}\)). Then for all \(i \in \{1, \ldots, N\}\) the synthesized channels satisfy \(\hat{P}_{Y|X}^{(i)} \preceq P_Y^{(i)}\) and \(Z(\hat{P}_{Y|X}^{(i)}) \geq Z(P_Y^{(i)})\).

2) Source Coding: Korada and Urbanke showed in [5] that PCs are also optimal for lossy source compression of a DMS \(X \sim P_X\) into \(Y\). In this context the SC algorithm is used for compression while reconstruction is performed with the matrix \(G_N\). PCs for source compression are designed upon the test channel \(P_{Y|X}\) that solves the minimization defining the (symmetric) rate-distortion function. Their fundamental property is that the distribution they induce \(Q_{X,Y}\) is arbitrarily close to the \(N\)-product design distribution \(P_{X,Y} = \prod P_{Y|X} P_X\), as summarized in the following Lemma:

Lemma 2 (Lemmas 2, 5, and Theorem 16 in [5]). For any compression rate \(R > I(Y;X)\) (\(Y\) uniformly distributed) we can choose \(\delta_N > 0\) such that if the frozen set \(F\) is defined as

\[
F = \{i : Z(P^{(i)}_0) \geq 1 - \delta_N\}
\]

then for any \(0 < \beta < \frac{1}{2}\) we have

\[
\frac{1}{2} \sum_{x,y} |P(x,y) - Q(x,y)| \leq O(2^{-N^\beta}).
\]

The empirical coordination capacity region for the network in Fig. 2 is given by [1]

\[
C_{P_X} = \{(R, P_{Y|X}) : R \geq I(Y;X)\}.
\]

Theorem 1. For any pair \((R, P_{Y|X}) \in C_{P_X}\) (i.e. considering the aforementioned additional restrictions with respect to \(C_{P_X}\)) there exists a sequence of PCs that achieves empirical coordination in the two-node network in Fig. 2.

Proof: Let \(P_{Y|X}\) be the desired distribution and let

\[
P_{X|Y} = \frac{P_{X,Y}}{P_Y} = \frac{P_{Y|X} P_X}{P_Y}
\]

where \(P_Y\) is the uniform distribution. Construct a PC for source compression upon \(P_{X|Y}\) with the frozen set defined

\[
\hat{P}_{Y|X} = \prod P_{Y|X} P_X,
\]
as in (3). For any compression rate $R > I(Y;X)$ we know from the results on PCs for source coding in [5] that $R > \frac{I(Y)}{N}$ if $N$ is sufficiently large. Using the CR both nodes generate $|F|$ i.i.d. random bits $U_F$ according to a uniform distribution. Node $X$ compresses its observation $X$ using the SC encoding algorithm with frozen bits set to $U_F$. The output of the algorithm is the vector $U_F$ which contains $|F|$ information bits. Therefore a rate of $R$ bits per action suffices to communicate them to Node $Y$. This allows Node $Y$ to generate the sequence $Y = UG_N$. From Lemma 2 we know that the induced distribution is close to the desired $N$-product distribution $P_{X,Y} = \prod_{Y} P_{Y|X} P_{X}$:
\[
\|P_{X,Y} - Q_{X,Y}\| = \frac{1}{2} \sum_{x,y} |P(x,y) - Q(x,y)| \leq O(2^{-N\beta})
\]
for any $0 < \beta < \frac{1}{2}$.

We now verify that this implies empirical coordination. Let $C_{P,Q}(x_P,y_P,x_Q,y_Q)$ denote the optimal coupling between $P_{X,Y}(x_P,y_P)$ and $Q_{X,Y}(x_Q,y_Q)$. Consider the events
\[
E = \{|P_{X,Y} - T_{X,Y}(x_Q,y_Q)| > \epsilon\},
\]
and their complements $E^c$ and $E_X^c$. Since $C_{P,Q}$ has marginal $Q_{X,Y}(x_Q,y_Q)$, $E$ evaluated over $C_{P,Q}$ is our event of interest. We upper bound its probability using the properties of the optimal coupling (see (4)). For any $0 < \beta < \frac{1}{2}$ we have that
\[
\Pr(E) = \Pr(E|E_X) \Pr(E_X) + \Pr(E|E_X^c) \Pr(E_X^c) \leq \Pr(E_X) + \Pr(E|E_X^c) \leq O(2^{-N\beta}).
\]
The second term in (5) is upper bounded by the probability that a tuple $(X,Y) \sim P_{X,Y}$ has a type with total variation with respect to $P_{X,Y}$ larger than $\epsilon$. From the properties of the Asymptotic Equipartition Property (AEP) (see [1], [3]) we know that this probability goes to zero exponentially fast with $N$. Therefore the bound for the first term in (5), which is due to the properties of the optimal coupling, dominates. Hence we conclude that the sequence of PCs achieves empirical coordination.

The above proof requires both nodes to generate the frozen bits using CR. The following corollary shows that there exists a fixed choice of the values of these bits such that the sequence of PCs achieves empirical coordination.

**Corollary 1.** For any pair $(R, P_{Y|X}) \in C_{P,X}$ there exists a sequence of PCs and a choice of frozen bits $\tilde{u}_F$ that achieves empirical coordination in the two-node network in Fig. 2.

**Proof:** The proof is given in the Appendix.

IV. CASCADE NETWORK

Consider again the network given in Fig. 1. Its empirical coordination capacity region is given by [1]
\[
C_{P,X} \triangleq \begin{cases} 
(R_1, R_2, P_{Y|X}) : 
R_1 \geq I(Y,Z;X) \quad \text{and} \quad R_2 \geq I(Z;X)
\end{cases}
\]

**Theorem 2.** For any triple $(R_1, R_2, P_{Y|X}) \in C_{P,X}$ (i.e., considering the aforementioned additional restrictions with respect to $C_{P,X}$) there exists a sequence of PCs that achieves empirical coordination in the cascade network in Fig. 1.

**Proof:** Let $P_{Y|X}$ be the desired distribution. Design a PC for source compression based on $P_{X|Z}$ (obtained from $P_{Y|Z,X} P_X$ by conditioning and marginalizing). For any compression rate $R > I(Z;X)$ (for $Z$ uniformly distributed) and any $0 < \beta < \frac{1}{2}$ we know that
\[
\|P_{X,Z} - Q_{X,Z}\| \leq O(2^{-N\beta})
\]
where $P_{X,Z} = \prod_{Y} P_{Y|Z,X} P_{X}$ and $Q_{X,Z}$ is the distribution induced by the code. The values of the frozen bits are chosen i.i.d. according to a uniform distribution using CR so that they are also available at Nodes $Y$ and $Z$. The vector resulting from source compression is sent to Node $Y$ and then forwarded to Node $Z$. This allows both nodes to generate $Z$. Therefore
\[
R_1 \geq R_1^* + I(Z;X)
\]
and
\[
R_2 \geq I(Z;X)
\]
where $R_1^*$ is the fraction of $R_1$ which is not yet used. Clearly we need $R_1^*$ to be arbitrarily close to $I(Y;Z;X) - I(Z;X)$.

Now assume that $(X,Z) \sim P_{X,Z}$ (instead of $Q_{X,Z}$). Design a new PC for compression of $(X,Z)$ into $Y$ using the conditional probability $P_{X,Z|Y}$. Let $Q_{Y|X,Z}$ be the distribution induced by this new PC. We know from Section III that for any compression rate $R > I(Y;X;Z)$ (for $Y$ uniformly distributed) and any $0 < \beta < \frac{1}{2}$ we can choose the frozen set $F_Q$ as in (3) so that for $P_{Y|X,Z} \triangleq \prod_{Y} P_{Y|X,Z}$
\[
\|P_{Y|X;Z} F_{X,Z} - Q_{Y|X,Z} F_{X,Z}\| \leq O(2^{-N\beta}).
\]

Using the triangle inequality it is easy to show that (6) and (7) imply that for the true distribution on $(X,Z)$, i.e., $Q_{X,Z}$,
\[
\|P_{Y|X,Z} F_{X,Z} - Q_{Y|X,Z} F_{X,Z}\| \leq O(2^{-N\beta})
\]
for any $0 < \beta < \frac{1}{2}$. As in Section III we use this result to build the optimal coupling $C_{P,Q}$ between $P_{X,Y,Z}$ and $Q_{X,Y,Z}$ which satisfies, for any $0 < \beta < \frac{1}{2}$,
\[
\Pr(E_{XY}) \leq \Pr((X_P, Y_P, Z_P) \neq (X_Q, Y_Q, Z_Q)) \leq O(2^{-N\beta}).
\]

This allows us to evaluate the probability of the event
\[
\Pr(E_1) \triangleq \{\|P_{X,Y,Z} - T_{X,Y,Z}\| > \epsilon\}
\]
and bound it, for any $0 < \beta < \frac{1}{2}$, as
\[
\Pr(E_1) \leq \Pr(E_{XY}) \leq O(2^{-N\beta}).
\]

This means that our construction will yield sequences of actions with the desired type with high probability. Now we only need to make $Y$ available at Node $Y$. To accomplish this at the desired rate we take advantage of the fact that $Z$ is already available at Node $Y$ and that it is indeed correlated with $Y$. We model this correlation as transmitting $Y = UG_N$ (i.e. a codeword from a PC) through the DMC $P_{Z|Y}$ and use the SC decoding algorithm to obtain $Y$, an estimate of $Y$. 


Such a PC for transmission over $P_{Z|Y}$ would have a frozen set $F_Y$ as defined in (2). To obtain $Y$ the decoder needs to have access to the frozen bits $U_{F_Y}$. Moreover, since due to Lemma 1 we know that $F_Q \subseteq F_Y$ for $\delta_2^X < \delta_N$ and sufficiently large $N$, only the bits in the set $F_V \cap F_Q$ need to be conveyed (see Figure 3). This can be done at any rate
\[
R' \geq \frac{|F_V| - |F_Q|}{N} \geq I(Y; X, Z) - I(Y; Z),
\]
or equivalently $R' \geq I(Y; Z; X) - I(Z; X)$, as desired.

It is easy to show that this design will also work fine when $(\hat{Y}, Z)\sim \tilde{Q}_{Y, Z}$ by considering the optimal coupling between $P_{Y, Z}$ and $\tilde{Q}_{Y, Z}$ and the fact that, for any $0 < \beta < \frac{1}{2}$,
\[
\sum_{y, z} P_{Y, Z}(y, z) - \tilde{Q}_{Y, Z}(y, z) \leq O(2^{-N^\beta})
\]
which is derived from (8) with the triangle inequality. That is,
\[
Pr(\mathcal{E}_Y) \triangleq Pr(Y \neq \hat{Y}) \leq O(2^{-N^\beta})
\]
for any $0 < \beta < \frac{1}{2}$.

Finally, we see that the probability of the event
\[
\mathcal{E} = \{||P_{X, Y, Z} - T_{X, Y, Z}| > \epsilon\},
\]
when evaluated over the distribution induced by the code
\[
Q_{X, Y, Z} = P_{X}Q_{Z|X} \tilde{Q}_{Y|X}ZQ_{Y|\hat{Y}},
\]
where $Q_{Y|\hat{Y}}$ accounts for the possible errors in reproducing $\hat{Y}$ at Node Y, is arbitrarily low. That is, for any $0 < \beta < \frac{1}{2}$
\[
Pr(\mathcal{E}) \leq Pr(\mathcal{E}_Y) + Pr(\mathcal{E}'|\mathcal{E}_Y) = Pr(\mathcal{E}_Y) + Pr(\mathcal{E}_1) \leq O(2^{-N^\beta}).
\]

Hence the sequence of PC achieves empirical coordination.

As for the two-node network, there exists a choice of the values of the frozen bits such that the sequence of PCs achieves empirical coordination.

**Corollary 2.** For any triple $(R_1, R_2, P_{Y, Z|X}) \in C_{PK}$, there exists a sequence of PCs and a choice of frozen bits that achieves empirical coordination in the network in Figure 1.

**Proof:** The proof is similar to that of Corollary 1.

**V. DISCUSSION**

The previous sections show that PCs are suitable for empirical coordination. However, a few questions remain open. First of all, our discussion has been restricted to the symmetrical empirical coordination capacity region, a portion of the region $C_{PK}$. Preliminary results show that the usual techniques for extending the applicability of PCs (see [4]–[7]) may also be useful here to make more general statements as one would expect from [1, Section VII]. Moreover, we have only considered empirical coordination. No practical codes for strong coordination have been designed so far. The problem gets more involved due to the need for exploiting common randomness.

**ACKNOWLEDGMENT**

The authors would like to thank one anonymous reviewer for helpful comments.

**APPENDIX**

**Proof of Corollary 1:** Since the total variation is a bounded measure, Theorem 1 implies that there is a sequence of coordination codes such that
\[
\lim_{N \to \infty} E_{X, Y} \{||T_{X, Y} - P_{X, Y}||\} = 0.
\]

The expectation, which is taken with respect to the induced distribution, can be written as
\[
E_{X, Y} \{||T_{X, Y} - P_{X, Y}||\} = E_{X, U} \{||T_{X, U} - P_{X, Y}||\}
\]
\[
= \sum_{u, x} Q(x) ||T_{X, U} - P_{X, Y}||
\]
\[
= \sum_{u, x} Q(u) \sum_{\tilde{u}_F} Q(\tilde{x}, \tilde{u}_F||u) ||T_{X, U} - P_{X, Y}||
\]
\[
= E_{\tilde{u}_F} \{E_{X, U_{\tilde{F}}} \{||T_{X, U_{\tilde{F}}, U_{\tilde{F}}} - P_{X, Y}||\}\}.
\]

In (9) and (10) we have used the fact that $G_N$ is a one-to-one mapping to modify slightly the definitions of $Q_{X, Y}$ and of the type function to depend on $u$ by setting $y = uG_N$. The notation $T_{X, U_{\tilde{F}}, U_{\tilde{F}}} = T_{X, U}$ in (11) simply makes explicit the fact that $u = (u_{\tilde{F}}, u_F)$. For the outer expectation in (11) to go to zero with the block length there must be a choice of the values of the frozen bits $u_{\tilde{F}}$ such that
\[
E_{X, U_{\tilde{F}}} \{||T_{X, U_{\tilde{F}}, U_{\tilde{F}}} - P_{X, Y}||\} \leq E_{X, U} \{||T_{X, U} - P_{X, Y}||\}.
\]

Moreover, since convergence in the first mean implies convergence in probability, we conclude that there exists a sequence of PCs for coordination and a choice of the frozen bits that achieves empirical coordination.

**REFERENCES**


Polar Codes for Degraded Relay Channels

Mohammad Karzand
Ecole Polytechnique Fédérale de Lausanne
CH-1015 Lausanne, Switzerland
mohammad.karzand@epfl.ch

Abstract—Polar coding is a recent channel coding technique invented by Arıkan to achieve the ‘symmetric capacity’ of binary-input memoryless channels. Subsequently it was observed that such codes are also good for lossy channel coding, general memoryless channels without feedback and several multiterminal settings. In this paper, we apply polar codes to achieve the capacity of binary input symmetric degraded relay channels. The main novelty is to show how polar codes can implement block Markov coding in this degraded setting, and consequently yielding a low complexity method to achieve “decode-and-forward” rates.

I. INTRODUCTION

The relay channel was introduced by van der Meulen, and has been studied for forty years in the community. Some of the well-known capacity theorems for different classes of relay channels is proved by El Gamal and Cover in [1]. In this note, we are going to construct a polar coding scheme to achieve the rate given by [1] for binary stochastically-degraded relay channels. The well-known capacity theorems for different classes of relay channels is proved by El Gamal and Cover in [1]. In this note, we are going to construct a polar coding scheme to achieve the rate given by [1] for binary stochastically-degraded relay channels. The important challenge is to show how polar codes can implement block Markov coding in this degraded setting, and consequently yielding a low complexity method to achieve “decode-and-forward” rates.

II. REVIEW

A discrete memoryless relay channel consists of four finite sets $X, X', Y, Y'$ and a collection of conditional probability mass functions $Pr(y, y'|x, x')$ on $Y \times Y'$. A $(M, n)$ code consists of an encoder that assigns a codeword $x^n(w)$ to each message $w \in \{1, \ldots, M\}$, a relay that assigns at time $i \in \{1, \ldots, n\}$ a symbol $x_i(y_i^{-1})$ to each received sequence $y_i^{-1}$, and a decoder that assigns a message $\hat{w}$ to each received sequence $y^n$. The channel is memoryless in the sense that the current received symbols and the past symbols are conditionally independent given the current transmitted symbols.

The average probability of error is $P_e = Pr\{W \neq \hat{W}\}$ with message $W$ chosen uniformly at random. The rate $R = \frac{\log M}{n}$ is achievable if there exists a sequence of $\binom{2nR}{n}$ codes with vanishing average probability of error. The capacity $C$ is the supremum of all achievable rates.

The relay channel is physically degraded if $Pr(y, y'|x, x')$ can be written in the form

$$Pr(y, y'|x, x') = Pr(\tilde{y}|\tilde{x}, x) Pr(y|\tilde{y}, \tilde{x}).$$

Furthermore, it is stochastically degraded if there exists a set of conditional distributions $q(x|x', \tilde{x})$ such that

$$Pr(y|x, x') = \sum_{\tilde{y}} Pr(y|x', \tilde{x}) q(y|\tilde{y}, \tilde{x}),$$

Theorem 1 ([1]). The capacity of a physically degraded relay channel is given by

$$C = \sup_{Pr(x, \tilde{x})} \min \{I(X; \tilde{X} ; Y), I(X; Y|X)\}$$

where the supremum is over all joint distributions $Pr(x, \tilde{x})$ on $X \times \tilde{X}$. Furthermore, this rate gives a lower bound on the capacity of a stochastically degraded relay channel.

Proof: We present an overview of the achievability here, so we will identify the requirements on decoder that can perform well on this channel.

Consider $B$ blocks of transmission, each of $n$ symbols. A sequence of $B-1$ indices $u^{(i)} \in \{1, \ldots, 2^n R\}$ will be transmitted over the channels in $nB$ transmissions. We partition the set $\{1, \ldots, 2^n R\}$ through a appropriate hash function $s^{(i)} = hash(u^{(i-1)})$, $s^{(i)} \in \{1, \ldots, 2^n R\}$.

A doubly indexed set of codewords is generated using random coding as follows. We fix $Pr(\tilde{x}) Pr(x|x')$, and generate $2^{nR}$ i.i.d. $n$-sequences in $X^n$, each drawn according to $Pr(\tilde{x})$, and for each of them generate $2^nR$ i.i.d. $n$-sequences, each drawn independently according to $Pr(x|x')$. 

59
Let $w^{(i)}$ be the new index to be sent in block $i$ and $s^{(i)}$ be defined as the partition corresponding to $w^{(i-1)}$. The transmitter sends $x(w^{(i)}, s^{(i)})$. The relay has an estimate of $w^{(i-1)}$ and it sends $\tilde{x}(s^{(i)})$ in block $i$.

Decoding is accomplished in three steps:
1. Given $s^{(i)}$ and upon receiving $\tilde{s}^{(i)}$, the relay estimates the transmitted message $w^{(i)}$. This will succeed with low probability of error if $R < I(X; Y|X)$ and $n$ is sufficiently large.
2. The receiver estimates $s^{(i)}$ using a jointly typical decoder for the pair $(\tilde{x}^{(i)}(s), y^{(i)})$. This can be done with arbitrarily small probability of error if $R_0 < I(X; Y)$.
3. Assuming $s^{(i)}$ is decoded correctly at the receiver, the receiver looks for a unique $w^{(i-1)}$ among the $q^n(R-R_0)$ indices that satisfies $\text{hash}(w^{(i-1)}) = s^{(i-1)}$ and is jointly typical with $y^{(i-1)}$. This can be done if $n$ is sufficiently large and if $R < I(X; Y|X) + R_0$.

In this paper we aim to introduce the code so that all three decoding steps can be done using the polar decoder and thus with low complexity. Before we proceed to present our result, let us give a brief overview of polar codes.

We briefly recall how polarization is done for the $q$-ary input channels when $q$ is a prime number.

The basic idea behind the polarization technique is to take two independent uses of the same channel, and by preprocessing their inputs construct two unequal channels with different mutual informations such that the sum of mutual informations remains unchanged. The polarization construction is obtained by repeating this operation $l$ times to obtain from $N = 2^l$ independent uses of the channel $W$ the $N$ synthesized channels. The synthesized channels are denoted by $I(W^s), s \in \{+, -\}^l$. Polarization means that as $l$ gets large, almost all the channels at level $l$ are either almost perfect or almost pure noise. As discussed earlier, the transformation preserves mutual information; putting these two facts together, one concludes that the fraction of almost perfect channels is equal to the symmetric channel capacity. In order to construct a polar code $C_n(F, u_F)$ of blocklength $n = 2^l$ based on this, one finds $F^c = \{s \in \{+, -\}^l | I(W^s) > 1 - \delta\}$ for some sufficiently small $\delta > 0$ and sends information through these ‘good’ channels. For the other channels indexed by $F = \{+, -\}^l \setminus F^c$, the decoder and encoder, prior to the communication, agree on a sequence $u_F$ that will be sent over these channels; these values are said to be ‘frozen’.

The relation between $U$ and $X$ is as follows:

$$X = UH_1,$$

where $H_1$ is the $l$-fold Kronecker product of the matrix $[1, 0]$. The encoder represents the message as $u_F$, and concatenates it with a randomly selected $u_F$ to form the vector $u$. $u_F$ is revealed to the decoder.

**Theorem 2** ([2],[5]). Given a $q$-ary discrete channel, with $q$ prime, and a fixed rate $R < I(W)$, where $I(W)$ is the symmetric channel capacity of the channel, there exists a sequence of polar codes $C_n(F, u_F)$ of rate $R_N > R$ with low encoding and decoding complexity, and with vanishing error probability.

In the next section, this theorem provides the essential tools to perform the first two decoding steps successfully in the achievability proof of the degraded relay channel. However, we still need another result from the polar coding literature to guarantee the success of the third step.

We say $W_b(y|x)$ is a stochastically degraded version of $W_g(y|x)$ if there exists a distribution $q(y|x)$ such that

$$W_b(y|x) = \sum W_g(y|x)q(y|x).$$

Consider further, two polar constructions $C_N(F_b, u_{F_b})$ and $C_N(F_g, u_{F_g})$ for $W_b(y|x)$ and $W_g(y|x)$. The following lemma is proven in [6]:

**Lemma 1.** [6] If $W_b(y|x)$ is a degraded version of $W_g(y|x)$, then for a given $\delta$ the frozen set for the good channel $F_g$ is a subset of the frozen set for the bad channel $F_b$.

Thus, all good channels for $W_b$ are also good channels for $W_g$. Comparing the two polar constructions and considering Lemma 1, we label the channels in three groups. $I(W_g)$ fraction of all channels consists of channels which are noiseless for both $W_b$ and $W_g$. $I(W_b) - I(W_g)$ fraction of channels consist of those channels which are noiseless for $W_g$ but completely noisy for $W_b$. The remaining $1 - I(W_g)$ fraction of channels are completely noisy for both $W_b$ and $W_g$. Suppose one uses polar codes to multicast common information with the rate $I(W_g)$ over both these channels. Knowing the frozen bits, the decoder for the channel $W_g$ is able to decode everything, but the decoder for the channel $W_b$ cannot decode anything because the transmission rate is above its capacity. At this point a list decoder for the channel $W_b$ is able to pin down the transmitted codeword to a list of codewords of size $q^n(I(W_b) - I(W_g))$. Suppose a genie reveals the value of the information bits in the second group of channels to the decoder at $W_b$ in each block, which results in decreasing the rate of the code to $I(W_b)$. Now, the decoder is able to decode the remaining bits using the polar decoder for $W_g$.

We will see in the following that the concept of coding for the degraded relay channel regarding the transmission from relay to receiver consists in sending only the values corresponding to the intersection of the two frozen sets in order to support the decoding at the receiver side of the codeword that is sent by the transmitter over the degraded channel.

**III. Polar Codes for the Degraded Relay Channel**

In what follows, we consider $|X| = |\tilde{X}| = 2$. We consider the marginal probability distribution of $x$ and $\tilde{x}$ to be uniform and $Pr(x \neq \tilde{x}) = 0$, $0 < \alpha < 1$. We define the symmetric rate of the stochastically degraded relay channel to be $R_N = \min\{I(X, \tilde{X}|Y), I(X|\tilde{Y}|X)\}$.

**Theorem 3** (Polar codes achieve $\sup_\alpha R_\alpha$ of degraded relay channels). Given a binary-input stochastically degraded relay channel $Pr(y|x_1, x_1) = 0 \leq \alpha \leq 1$ and $R < R_\alpha$, there exists
a sequence of codes based on the polar encoder and decoder with low encoding and decoding complexity of rate $R_N > R$ with vanishing error probability.

**Proof:** Since $\alpha \to R_{\alpha}$ is continuous, it suffices to consider $\alpha$ of the form $\alpha = \frac{k}{N}$ where $k \in \mathbb{N}$ and $q$ is prime.

As in [1], we consider $B$ blocks of transmission, each consisting of $n$ symbols. A sequence of $B - 1$ indices will be transmitted over the channel in $nB$ transmissions. We fix $R = \frac{\log M}{n}$ and $R_0$ as design parameter.

Let $w^{(i)} \in \{1, \ldots, M\}$ be the new index to be sent in block $i$, and let $s^{(i)}$ be an integer in the set $\{1, \ldots, 2^n R_0\}$ and be a hash of $w^{(i-1)}$. We will define this hash function further into the proof. Note that $W^{(i)}$ and $S^{(i)}$ are independent. We assume that at the end of block $i - 1$ the receiver knows $(w^{(1)}, w^{(2)}, \ldots, w^{(i-2)})$ and $(s^{(1)}, s^{(2)}, \ldots, s^{(i-1)})$ and the relay knows $(w^{(1)}, w^{(2)}, \ldots, w^{(i-1)})$ and $(s^{(1)}, s^{(2)}, \ldots, s^{(i)})$.

As the proof of Theorem 1 suggests, the following three tasks should be completed at the end of block $i$.

- Both the relay and the transmitter have access to $s^{(i)}$, so they should both cooperatively transmit $s^{(i)}$ successfully to the receiver.
- The transmitter should transmit $w^{(i)}$ successfully to the relay.
- The receiver should decode $w^{(i-1)}$ knowing $y^{(i-1)}$, $s^{(i-1)}$ (that has been obtained in the previous block) and $s^{(i)}$.

An overview of the system can be found in the next column and as we go through our proof we explain each part of it.

We suggest using the following scheme. Suppose $\tilde{x} = u H_n$ and $x = j + \tilde{x}$. For now, to satisfy the condition on the probability distribution, we only assume that $U$ is an $n$-tuple vector which is derived from $S^{(i)}$ and has independent uniformly distributed components, $j$ is an $n$-tuple vector which is a function of $w^{(i)}$ and has independent and identical components distributed as Bernoulli with parameter $\alpha$, and $H_n$ is the $n$-fold Kronecker product of the matrix $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Observe that the channel between $\tilde{X}$ and $Y$ is governed by the following conditional probability distribution:

$$
\Pr(y|\tilde{x}) = \sum_j \Pr(y|x, j) \Pr(j) \\
= \prod_{i=1}^n \sum_{j_i} \Pr(y_i|\tilde{x}_i, j_i + \tilde{x}_i) \Pr(j_i) \\
= \prod_{i=1}^n \sum_{j_i} \left( \sum_{\tilde{y}_i} \Pr(y_i, \tilde{y}_i|\tilde{x}_i, j_i + \tilde{x}_i) \right) \Pr(j_i),
$$

so that the channel is memoryless. Therefore, we consider $\tilde{X}$ to be a codeword of a binary polar code $C_n(F, U_F)$ designed for the channel

$$
\Pr(y|\tilde{x}) = \sum_j \left( \sum_{\tilde{y}} \Pr(y, \tilde{y}|\tilde{x}, j + \tilde{x}) \right) \Pr(j).
$$

Choosing $U_F$ uniformly at random makes $\tilde{X}$, the codeword of a binary polar code, a uniform distribution, and hence $\tilde{X}$ satisfies the constraint on its distribution. By Theorem 2, we know that a sequence of codes with rate $R_{0N} > R_0$ exists for

$$
R_0 < I(\tilde{X}; Y) \tag{1}
$$

with vanishing error probability as the length increases. With this, our scheme succeeds in performing the first task in the achievability scheme.

Similarly, observe that there exists an equivalent channel between $J$ and $\tilde{Y}$ which is also memoryless. If we follow the steps of the previous part, we will not be able to construct a codeword which satisfies the condition on the probability...
distribution of J. In this case, we need to proceed as follows.

Recall \( \alpha = \frac{1}{2^{q}} \). Consider the operator \( \mathcal{L}(x_q) : \{0, \cdots, q - 1\} \to \{0, 1\} \) as follows

\[
\mathcal{L}(x_q) = \begin{cases}
1 & \text{if } x_q < k, \\
0 & \text{if } x_q \geq k.
\end{cases}
\]

Observe that if \( x_q \) is uniformly distributed on the set \( \{0, \cdots, q - 1\} \), then \( \mathcal{L}(x) \) has a Bernoulli distribution with parameter \( \alpha \). This trick is mentioned in [7]. This enables us to provide arbitrary correlation between the output of the relay and the transmitter.

Now, we consider \( J = \mathcal{L}(X_q) \), where \( X_q = VH_n \) is a codeword of a \( q \)-ary polar code \( C_n(\hat{G}, V \hat{G}) \) designed to encode \( w(i) \) for the auxiliary channel governed by the following transition probability:

\[
Pr(\hat{Y}, \hat{X}|X_q) = Pr(\hat{Y}|\mathcal{L}(X_q), \hat{X})Pr(\hat{X}) = \prod_{i=1}^{n} Pr(\hat{y}_i|\hat{x}_i, \mathcal{L}(x_{qi}) + \hat{x}_i).
\]

Observe that \( J \) has Bernoulli distribution with parameter \( \alpha \). We can transmit the message to the relay by using a polar encoder to encode \( w(i) \) to \( J \). As the relay has access to \( Y \) and \( \hat{X} \) such a power encoder exits for rates up to \( I(J; Y|\hat{X}) = I(\hat{J}; Y, \hat{X}) \). By Theorem 2, we know a sequence of codes with rate \( R_N > R \) exists for

\[
R < I(X_q; \hat{Y}, \hat{X}) = I(X; \hat{Y}|\hat{X}) \tag{2}
\]

with vanishing error probability as the length increases. With this, our scheme succeeds in performing the second task in the achievable scheme. Note that it is easy to extend the ordinary polar decoder with the same complexity to this case by using \( (Y, \hat{X}) \) as a super alphabet as the output.

Now the only remaining task is to decode \( w(i-1) \) at the receiver by using \( y^{(i-1)}, v^{(i-1)} \) and \( s^{(i)} \). Up to now, we have not chosen the hash function relating \( s^{(i)} \) and \( w^{(i-1)} \). Now, we should choose the relation between \( s^{(i)} \) and \( w^{(i-1)} \) so that we help the receiver to decode \( w^{(i-1)} \). Consider the two channels

\[
X_q^{(i-1)} \rightarrow (Y^{(i-1)}, \hat{X}^{(i-1)})
\]

and

\[
X_q^{(i-1)} \rightarrow (Y^{(i-1)}, \hat{X}^{(i-1)}).
\]

Note that the second of these is a degraded version of the first one. Further, consider the polar code \( C_n(\hat{G}, V \hat{G}) \) used to transmit \( w^{(i-1)} \) to the relay. According to Lemma 1 and the discussion that follows, we need to reveal at least \( n[1(X_q^{(i)}; Y, \hat{X}) - I(X_q^{(i)}; Y|\hat{X}) + I(X_q^{(i)}; Y|\hat{X})] \) bits to the receiver to ensure a successful decoding of \( w^{(i-1)} \) at the receiver. We will use \( s^{(i)} \) to reveal these values to the receiver. Observe that \( s^{(i)} \) should be the value of those positions that are not in the frozen set \( \hat{G} \) but should be considered frozen and revealed to the decoder for the channel \( X_q^{(i-1)} \rightarrow (Y^{(i-1)}, \hat{X}^{(i-1)}) \) because of a lower symmetric channel capacity. We denote the frozen set of the polar code designed for the channel \( X_q^{(i-1)} \rightarrow (Y^{(i-1)}, \hat{X}^{(i-1)}) \) to be \( G \).

The figure below shows the relationship between \( W(i), V(i), U^{(i+1)} \) and \( S^{(i+1)} \).

![Construction of hash function relating W and S](image)

By Lemma 1, we know that \( \hat{G} \subset G \). We need to transmit \( |G| - |\hat{G}| \) \( q \)-ary digits to the receiver through \( nR_0 \) bits (i.e., \( s^{(i)} \)) to the receiver, so we should have

\[
R_0 > I(X_q^{(i)}; \hat{Y}, \hat{X}) - I(X_q^{(i)}; Y|\hat{X}). \tag{3}
\]

Putting (1), (2) and (3) together, we have the following conditions on the successful transmission of \( w^{(i-1)} \) to the receiver at the end of time \( i \):

\[
R < I(X_q^{(i)}; \hat{Y}|\hat{X}) \tag{4}
\]

\[
R < I(X_q^{(i)}; Y). \tag{5}
\]

By theorem (2), this implies that there exist specific values of the frozen bits so that all error probabilities are small for all decoders.

IV. CONCLUSION

The main novelty is to adapt polar codes to implement the block Markov scheme of [1] and to show that in the degraded case the polar decoder will succeed with low complexity.

ACKNOWLEDGMENT

The author wishes to thank Prof. Emre Telatar for fruitful discussions and helpful comments on the paper.

REFERENCES


62
Unambiguous Decoding of Generalized Reed–Solomon Codes Beyond Half the Minimum Distance

Alexander Zeh1,2, Antonia Wachter1,3, and Martin Bossert1
1 Institute of Telecommunications and Applied Information Theory, Ulm University, Ulm, Germany
2 École Polytechnique, Paris, France
3 Inst. de recherche math. de Rennes (IRMAR), Université de Rennes 1, Rennes, France
{alexander.zeh, antonia.wachter, martin.bossert}@uni-ulm.de

Abstract—The Schmidt–Sidorenko–Bossert scheme extends a low-rate Reed–Solomon code to an Interleaved Reed–Solomon code and achieves the decoding radius of Sudan’s original list decoding algorithm while the decoding result remains unambiguous. We adapt this result to the case of Generalized Reed–Solomon codes and calculate the parameters of the corresponding Interleaved Generalized Reed–Solomon code. Furthermore, the failure probability is derived.

Index Terms—Failure Probability, (Interleaved) Generalized Reed–Solomon Codes, Multi–Sequence Shift–Register Synthesis

I. INTRODUCTION

Two approaches to decode an Reed–Solomon (RS) code of length n and dimension k beyond half the minimum distance \((n-k)/2\) with polynomial time and space complexity exist. The first one — formulated by Guruswami and Sudan (GS, [1], [2]) — considers the decoding problem of an RS code as interpolation problem and returns a list of all possible candidates within the increased decoding radius. The second one — invented by Schmidt, Sidorenko and Bossert (SSB, [3]) — extends an RS code virtually to an Interleaved RS code [4]–[8] and achieves the same decoding radius as Sudan’s original approach, where a multiplicity one [1] for the interpolation is used. The decoding result is unambiguous and therefore it is interesting to investigate its failure probability. For further parallels between these two schemes see [9], [10].

While the GS approach was formulated for Generalized RS (GRS) codes, the SSB adaption for GRS codes is still missing. We fill the lack in this contribution by giving an explicit expression for the virtually generated Interleaved Generalized RS (IGRS) code. Furthermore, we bound the failure probability for this unique decoding approach.

Our contribution is organized as follows. In Section II, Interleaved Generalized Reed–Solomon (IGRS) codes are defined and basic properties are discussed. The joint decoding of IGRS codes is explained in Section III. The parameters and the technique of a virtual extension of an low-rate GRS to an IGRS codes is explained in Section IV. The failure probability for this particular IGRS case is derived in Section V. Section VI concludes our contribution.

II. INTERLEAVED GENERALIZED REED–SOLOMON CODES

Let \(\mathcal{L} = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}\) denote a set containing \(n\) non-zero distinct elements (code locators) of the finite field \(\mathbb{F} = \text{GF}(q)\) of cardinality \(n < q\). Let \(\mathcal{V}' = \{\nu'_1, \nu'_2, \ldots, \nu'_n\}\) be a set of non-zero elements of \(\mathbb{F}\). Denote

\[
\mathcal{G}(\mathcal{V}', \mathcal{L}) = (\nu'_1 f(\alpha_1), \nu'_2 f(\alpha_2), \ldots, \nu'_n f(\alpha_n))
\]

for a given polynomial \(f(x)\) over \(\mathbb{F}\).

An GRS code \(\mathcal{G}\mathcal{R}\mathcal{S}(\mathcal{V}';n,k)\) of length \(n\) and dimension \(k\) over \(\mathbb{F}\) with \(n < q\) is given by

\[
\mathcal{G}\mathcal{R}\mathcal{S}(\mathcal{V}';n,k) = \{c = \mathcal{G}(\mathcal{V}', \mathcal{L}) : f(x) \in \mathbb{F}_k[x]\},
\]

where \(\mathbb{F}_k[x]\) denotes the set of all univariate polynomials with degree less than \(k\) and indeterminate \(x\). The generator matrix \(\mathbf{G}\) of the GRS code is:

\[
\mathbf{G} = \begin{pmatrix}
1 & 1 & \cdots & 1 \\
\alpha_1 & \alpha_2 & \cdots & \alpha_n \\
\alpha_1^2 & \alpha_2^2 & \cdots & \alpha_n^2 \\
\vdots & \vdots & \ddots & \vdots \\
\alpha_1^{n-1} & \alpha_2^{n-1} & \cdots & \alpha_n^{n-1}
\end{pmatrix}
\begin{pmatrix}
\nu'_1 \\
0 \\
\nu'_2 \\
0 \\
\vdots \\
\nu'_n \\
0
\end{pmatrix},
\]

and the parity check matrix \(\mathbf{H}\) is given by:

\[
\mathbf{H} = \begin{pmatrix}
1 & 1 & \cdots & 1 \\
\alpha_1 & \alpha_2 & \cdots & \alpha_n \\
\vdots & \vdots & \ddots & \vdots \\
\alpha_1^{n-k-1} & \alpha_2^{n-k-1} & \cdots & \alpha_n^{n-k-1}
\end{pmatrix}
\begin{pmatrix}
\nu_1 \\
0 \\
\vdots \\
0
\end{pmatrix}.
\]

Since \(\mathbf{GH}^T = 0\), we obtain an explicit relation for the column multipliers [11]:

\[
\sum_{j=1}^n \nu_j \nu'_j \alpha_j^\ell = 0, \quad 0 \leq \ell \leq n - 2.
\]

GRS codes are maximum distance separable (MDS), i.e., their minimum Hamming distance is \(d = n - k + 1\). Analog to interleaved RS (IRS) codes, we introduce interleaved generalized RS codes as follows.
Definition 1 (Interleaved GRS (IGRS) Code) Let the set
\[ K = \{k_1, k_2, \ldots, k_s\}, \]
consist of \(s\) integers, where all \(k_i < n\). Let \(M = (V'_1, V'_2, \ldots, V'_s)\),
be a tuple of \(s\) sets of \(n\) non-zero elements in \(\mathbb{F}\).

An Interleaved Generalized Reed–Solomon code \(\text{IGRS}(n, K, M, s)\) of interleaving order \(s\) is given by
\[
\text{IGRS}(n, K, M, s) = \left\{ \left( c^{(1)}, c^{(2)}, \ldots, c^{(s)} \right) : f^{(i)}(x) \equiv f^{(i)}(V'_j, \mathcal{L}) \mod x^{n-k_i}, \text{ for } i = 1, \ldots, s \right\}.
\]

If \(k_i = k \forall i = 1, \ldots, s\), then the IGRS code is called homogeneous, otherwise heterogeneous.

Let the code polynomials \(c^{(1)}, c^{(2)}, \ldots, c^{(s)}\) of an \(\text{IGRS}(n, K, M, s)\) code be corrupted by \(s\) error words \(e^{(1)}, e^{(2)}, \ldots, e^{(s)}\) of weight \(wt(e^{(i)}) = t_i\). We denote each received word by \(r^{(i)} = e^{(i)} + f^{(i)}(x) = (r^{(i)}_1, r^{(i)}_2, \ldots, r^{(i)}_n)\), respectively each received polynomial by \(r^{(i)}(x) = \sum_{j=1}^{n} r^{(i)}_j x^{j-1}\), for all \(i = 1, \ldots, s\).

We assume (as usual for interleaved codes) that the channel adds a so-called burst error of weight \(t\), i.e., each of the \(s\) codewords is affected by an error at the same \(t\) positions. Let \(E^{(i)}\) denote the set of error positions for the \(i\)th word. Since a burst error occurred, the union of the \(s\) sets of error positions \(E = \{E^{(1)} \cup E^{(2)} \cup \ldots \cup E^{(s)}\} \subseteq \{1, \ldots, n\}\) has cardinality \(|E| = t\).

III. Joint Decoding of IGRS Codes

Joint or collaborative decoding of IGRS codes makes use of the special structure of the burst error. In the first step, \(s\) syndrome polynomials \(S^{(1)}(x), S^{(2)}(x), \ldots, S^{(s)}(x)\) of degree smaller than \(n - k_i\) are calculated. The coefficients of \(S^{(i)}(x) = \sum_{j=1}^{n-k_i} c^{(i)}_j x^{j-1}\) for a GRS code as defined in (1) are calculated by:
\[
S^{(i)}_j = \sum_{k=1}^{n} \tau^{(i)}_k c^{(i)}_j \alpha^j_k, \quad (4)
\]
for all \(j = 1, \ldots, n - k_i\) and \(i = 1, \ldots, s\). These syndromes provide \(s\) equations with one common error-locator polynomial \(\Lambda(x)\):
\[
S^{(i)}(x) \cdot \Lambda(x) \equiv \Omega^{(i)}(x) \mod x^{n-k_i}, \quad \text{for } i = 1, \ldots, s.
\]

These equations can be combined to one system of equations \[ S \cdot \Lambda = T \] with the coefficients of \(\Lambda(x) = \Lambda_1 + \Lambda_2 x + \cdots + \Lambda_t x^{t-1} + x^t\) as unknowns:
\[
\begin{bmatrix}
S^{(1)}_1 \\
S^{(1)}_2 \\
\vdots \\
S^{(s)}_1 \\
S^{(s)}_2 \\
\vdots \\
S^{(s)}_1 \\
\vdots \\
S^{(s)}_s \\
\end{bmatrix}
\begin{bmatrix}
\Lambda_1 \\
\Lambda_2 \\
\vdots \\
\Lambda_1 \\
\vdots \\
\Lambda_2 \\
\vdots \\
\Lambda_1 \\
\vdots \\
\Lambda_1 \\
\end{bmatrix}
= \begin{bmatrix}
T^{(1)}_1 \\
T^{(1)}_2 \\
\vdots \\
T^{(s)}_1 \\
T^{(s)}_2 \\
\vdots \\
T^{(s)}_s \\
\end{bmatrix},
\]
where each submatrix \(S^{(i)}_j\) is a \((n-k_i-t) \times t\) matrix and each \(T^{(i)}_j\) is a column vector of length \(n-k_i-t\):
\[
S^{(i)} = \begin{bmatrix}
S^{(i)}_{11} & S^{(i)}_{12} & \cdots & S^{(i)}_{1s} \\
S^{(i)}_{21} & S^{(i)}_{22} & \cdots & S^{(i)}_{2s} \\
\vdots & \vdots & \ddots & \vdots \\
S^{(i)}_{n-k_i} & S^{(i)}_{n-k_i+1} & \cdots & S^{(i)}_{n-k_i+t+1} \\
\end{bmatrix},
\]
where each submatrix \(S^{(i)}_j\) is a \((n-k_i-t) \times t\) matrix and each \(T^{(i)}_j\) is a column vector of length \(n-k_i-t\):
\[
S^{(i)} = \begin{bmatrix}
S^{(i)}_{11} & S^{(i)}_{12} & \cdots & S^{(i)}_{1s} \\
S^{(i)}_{21} & S^{(i)}_{22} & \cdots & S^{(i)}_{2s} \\
\vdots & \vdots & \ddots & \vdots \\
S^{(i)}_{n-k_i} & S^{(i)}_{n-k_i+1} & \cdots & S^{(i)}_{n-k_i+t+1} \\
\end{bmatrix},
\]
\[
\tau_{IRS} = \left[ \frac{s}{s+1} \left( n - \frac{s}{s+1} \sum_{i=1}^{s} k_i \right) \right].
\]

To obtain a unique solution of (5), the number of equations has to be greater than or equal to the number of unknowns. By counting both, we obtain the joint error–correcting capability of a heterogeneous \(\text{IGRS}(n, K, M, s)\) code [7]:
\[
\tau_{IRS} = \left[ \frac{s}{s+1} \left( n - \frac{s}{s+1} \sum_{i=1}^{s} k_i \right) \right].
\]

IV. Virtual Extension to an IGRS Code

We extend the Schmidt–Siderenko–Bostrom scheme [3] to the case of Interleaved Generalized Reed–Solomon codes (see Definition 1) and give the corresponding parameters.

A low-rate GRS code can be virtually extended to an IGRS code. This IGRS code is denoted by \(\text{VIRS}(\mathcal{V}' ; n, k, s)\), where \(n, k\) and \(\mathcal{V}'\) are the original parameters of the \(\text{GRS}(\mathcal{V} ; n, k)\) code. The parameter \(s\) denotes the order of (virtual) interleaving. Let the vector \(c = (c_1, c_2, \ldots, c_n)\) and let \(c^{(i)}\) denote the vector, where each coefficient is raised to the power \(i\), i.e., \(c^{(i)} = (c_1^i, c_2^i, \ldots, c_n^i)\). The virtual IGRS code is defined as follows.

Definition 2 (Virtual Extension to an IGRS code) Let \(\text{GRS}(\mathcal{V} ; n, k)\) be an GRS code with the evaluation polynomials \(f(x)\) as defined in (2). The virtually extended Interleaved Generalized Reed–Solomon code \(\text{VIRS}(\mathcal{V}' ; n, k, s)\) of extension order \(s\) is given by
\[
\text{VIRS}(\mathcal{V}' ; n, k, s) = \left\{ \left( c^{(1)}, c^{(2)}, \ldots, c^{(s)} \right) : f^{(i)}(x) \equiv f^{(i)}(\mathcal{L}, \mathcal{V}'_j) \mod x^{n-k_i}, \text{ for } i = 1, \ldots, s \right\}.
\]

Through element-wise powering of the original received vector \(r^{(1)}\), we obtain a virtually created error in the received word \(r^{(1)}\) with non-zero values at the same positions as the original one. Therefore, we create a virtual burst error. This fact can be used to increase the error-correcting capability of low-rate
GRS codes (see [3]). We obtain a heterogeneous Interleaved Generalized Reed–Solomon code of order s, where s is chosen s.t. \( s(k-1) < n \) and the decoding radius is maximized. The parameters of the IGRS code are:

\[
k_i = i(k-1) + 1,
\]

\[
\nu_j^{(i)} = (\nu_j^i)_i,
\]

for \( i = 1, \ldots, s \) and \( j = 1, \ldots, n \). The following theorem shows the relation between the column multipliers \( \nu_j^{(i)} \) and \( \nu_j^i \) of the virtually created GRS code.

**Theorem 1 (Column Multipliers for VIRS Code)** Let an \( \text{GRS}(V'; n, k) \) code with evaluation polynomial \( f(x) \) as defined in (2) be given and let \( \nu_j, j = 1, \ldots, n \), be given, such that (3) holds.

Furthermore, let the virtually extended Interleaved Generalized Reed–Solomon code \( \text{VIRS}(V'; n, k, s) \) for a given parameter \( s \) with \( k_i \) and \( \nu_j^{(i)} \) be given as in (9). Then, the corresponding column multipliers of the \( i \)th subcode \( \{ f^{(i)}(L, V_i') \} \) are given by

\[
\nu_j^{(i)} = \nu_j^i (\nu_j^i)_i^{-1}.
\]

**Proof:** With (3) we have for the \( i \)th subcode \( \{ f^{(i)}(L, V_i') \} \):

\[
\sum_{j=1}^n \nu_j^{(i)} (\nu_j^i)^{\alpha_j^i} = \sum_{j=1}^n \nu_j^i = \nu_j^i (\nu_j^i)^{\alpha_j^i} = 0, \quad 0 \leq \ell \leq n - 2.
\]

The decoding radius of a \( \text{VIRS}(V'; n, k, s) \) code is the same as in [3]:

\[
\tau_{\text{VIRS}} = \left[ \frac{s}{s+1} \left( n - \left\lfloor \frac{s+1}{2} (k-1)+1 \right\rfloor \right) \right],
\]

where we choose \( s \) such that \( \tau_{\text{VIRS}} \) is maximized. Note that this decoding radius results from (8) using an IGRS code with mean dimension \( \bar{k} = \frac{1}{s} \sum k_i = \frac{1}{s} \sum (i(k-1)+1) = \frac{(k-1)(s+1)}{2} + 1 \).

Decoding can be done as for IGRS codes by solving (5) with multi-sequence shift-register synthesis.

**V. Failure Probability**

Since the virtual extension searches a unique solution for the error locator polynomial, a decoding failure occurs if the system of equations (5) has more than one solution. In the following, we derive an upper bound on the failure probability, which is the same as for the virtual extension of the RS codes used in [3] and therefore independent of the column multipliers.

We bound the probability that the \( s(n - \bar{k} - t) \times t \) matrix \( S \) does not have full rank \( t \) and denote the failure probability, if \( t \) errors occurred, by:

\[
P_f(t) \leq P(\text{rank}(S) < t \mid |E| = t).
\]

The following theorem states the failure probability.

**Theorem 2 (Upper Bound on the Failure Probability)**

Let a codeword of an \( \text{GRS}(V'; n, k) \) code be corrupted by an error of weight \( t \leq \tau_{\text{VIRS}} \) (11). Let the GRS code be virtually extended to an IGRS code \( \text{VIRS}(V'; n, k, s) \) of extension order \( s \) with the dimensions \( k_i \) and the set of column multipliers \( \nu_j^{(i)} \), for \( i = 1, \ldots, s \) as in (9).

Assume, for decoding we solve the system of equations from (5). The probability for a decoding failure is upper bounded by:

\[
P_f(t) \leq \left( \frac{q}{q-1} + \frac{1}{q} \right)^t \frac{q^{-3(\tau_{\text{VIRS}}-t)}}{q-1}.
\]

**Proof:** As in the proof of [3, Theorem 3], this is equivalent to the case that there exists a vector \( u \neq 0 \) of length \( t \), such that

\[
\exists u \neq 0 : S^{(i)} \cdot u^T = 0, \quad \forall i = 1, \ldots, s.
\]

Each syndrome matrix can be decomposed into five matrices (in [3], the decomposition consists only of four matrices):

\[
S^{(i)} = H^{(i)} \cdot \nu_j^{(i)} \cdot F^{(i)} \cdot D \cdot V,
\]

where \( D \) and \( V \) are the same full-rank \( t \times t \) matrices as in [3, Proof of Theorem 3] and

\[
H^{(i)} = (e^{(\mu-1)i})_{\mu=1, \ldots, n-k_i-t, \nu=1, \ldots, t}
\]

\[
\nu_j^{(i)} = \text{diag}\left( \nu_j^{(i)}(j_1), \nu_j^{(i)}(j_2), \ldots, \nu_j^{(i)}(j_t) \right),
\]

\[
F^{(i)} = \text{diag}\left( e_j^{(i)}(j_1), e_j^{(i)}(j_2), \ldots, e_j^{(i)}(j_t) \right),
\]

where \( E = \{ j_1, j_2, \ldots, j_t \} \) denotes the union of the error positions and \( e_j^{(i)} \) denotes the error value at the \( j_i \)th position of the \( i \)th received word. Since \( \nu_j^{(i)} \) and \( F^{(i)} \) are both diagonal matrices, we can switch their order. Since \( \nu_j^{(i)} \), \( D \), \( V \) are nonsingular, there is a one-to-one mapping from \( u \) to \( v \), where \( v^T = \nu_j^{(i)} \cdot D \cdot V \cdot u^T = (v_1, v_2, \ldots, v_t)^T \). Hence, (13) is equivalent to

\[
\exists v \neq 0 : H^{(i)} \cdot F^{(i)} \cdot v^T = 0, \quad \forall i = 1, \ldots, s.
\]

This is the same as [3, Proof of Theorem 3, Equation (22)] and using Lemma 1 in the appendix for arbitrary \( q \), the rest of the proof is analog.

Thus, the upper bound on the failure probability is independent of the column multipliers and in particular independent of using GRS codes or the RS codes from [3].

Note that this upper bound generalizes [3, Theorem 3] from RS to GRS codes and also to arbitrary \( q \) (in [3, Theorem 3] it was only shown for \( q = 2 \)). Moreover, the bound does not only hold for \( s = 2 \) (as indicated in [3]), but also upper bounds the
cases with \( s > 2 \), although the bound is not very tight then. This is due to the fact that we can upper bound the probability that all sub-matrices are singular by the probability that some sub-matrices are singular.

VI. CONCLUSION

We considered unambiguous decoding of low-rate GRS codes beyond half the minimum distance. For this purpose, the received word of a usual GRS codeword was virtually extended to an interleaved GRS code, similar to the Schmidt–Sidorenko–Bossert scheme. We derived the parameters of the extended code and proved that the failure probability is the same as for the RS codes from [3]. Moreover, we generalized the known bound on the failure probability to arbitrary fields.

VII. ACKNOWLEDGMENT

We would like to thank Markus Ullmschneider for implementing the decoding algorithm.

APPENDIX

In the following, we extend [3, Lemma 4] to arbitrary fields.

Lemma 1 Let \((v, e, c)\) be three non-zero elements from a field \(\mathbb{F} = \mathbb{GF}(q)\). Then, the set

\[
\mathcal{V} = \left\{ \left. \begin{pmatrix} v \cdot e \\ v \cdot e^{(2)} \right) : v, e \in \mathbb{F} \setminus \{0\} \right\}
\]

with \(e^{(2)} = 2c \cdot e + e^2\) forms the set of all full weight vectors of length 2, i.e.

\[
\mathcal{V} = \left\{ v \in \mathbb{F}^2 : \text{wt}(v) = 2 \right\} = (\mathbb{F} \setminus \{0\})^2.
\]

Proof: It is sufficient to show that all \((q - 1)^2\) possible vectors \(v \in \mathcal{V}\) are pairwise different. For any fixed \(c\), consider two vectors \(v, \tilde{v} \in \mathcal{V}\), and assume that \(v = \tilde{v}\), then

\[
\begin{align*}
v \cdot e &= \tilde{v} \cdot \tilde{e}, \\
v \cdot e^{(2)} &= \tilde{v} \cdot \tilde{e}^{(2)}. & \quad (16)
\end{align*}
\]

Dividing (17) by (16) yields

\[
\frac{2c \cdot e + e^2}{e} = \frac{e^{(2)}}{e} = \frac{\tilde{e}^{(2)}}{\tilde{e}} = 2c \cdot \tilde{e} + \tilde{e}^2 = \frac{2c \cdot \tilde{e} + \tilde{e}^2}{\tilde{e}}.
\]

Therefore, \(2c + e = 2c + \tilde{e}\) and hence, \(e = \tilde{e}\). Inserting this into (16), we obtain \(v = \tilde{v}\). Thus, for any \(c\), two different pairs \((v, e) \neq (\tilde{v}, \tilde{e})\) always result in two different vectors \(v, \tilde{v}\). ■

References


66
Iteration-constrained design of IRA codes

Alex Grant, Ingmar Land and Guangsong Wang
Institute for Telecommunications Research
University of South Australia
{alex.grant,ingmar.land}@unisa.edu.au, wangy012@mymail.unisa.edu.au

Abstract—This paper addresses extrinsic information chart based design of non-systematic irregular repeat-accumulate codes for a given number of decoding iterations. This criterion is of practical importance in many applications where complexity or latency is limited. Our main contribution is a novel formulation of the optimization problem in a particular way that makes its actual evaluation possible. This is achieved by introducing additional optimization variables and constraints which have the effect of “unrolling” the iteration. Our approach enforces the finite iteration condition, while avoiding the need to know arbitrary repeated functional compositions of the component EXIT functions. We restrict attention to the binary erasure channel, for which the EXIT chart approach is exact. Extension to other sparse-graph codes and other communication channels is straightforward.

I. INTRODUCTION

Sparse-graph codes with iterative message-passing decoding represent the state-of-the-art of modern channel coding. The two major classes are low-density parity-check (LDPC) codes, and irregular repeat-accumulate (IRA) codes. For both classes, encoding and decoding complexity are linear in the code length, and there are efficient code design methods [1–3].

The typical design goal in the literature is capacity-approaching codes. These codes that have maximum rate while the probability of error diminishes with increasing code length and increasing number of iterations. The predominant approaches are density evolution and extrinsic information transfer (EXIT) charts [1, 4, 5]. While density evolution provides exact results for the decoding threshold, the EXIT chart method employs a Gaussian approximation. The advantage of the latter, however, is that code design can often be formulated in terms of a convex optimization problem, allowing convenient numerical evaluation. Tradeoffs between the two methods for the design of IRA codes were discussed in [6]. For erasure channels, the two methods are equivalent.

Two important parameters characterizing implementation complexity are the parity-check matrix density and the number of decoding iterations. The relationship between parity-check density and performance in terms of the gap to capacity have been analyzed in [7, 8], the required number of iterations for successful decoding and corresponding bounds have been addressed in [9]. As opposed to LDPC codes, non-systematic IRA codes can achieve capacity on the binary erasure channel with bounded complexity per information bit [7].

Whereas the above approaches assume infinite iteration, the number of iterations required to achieve a target performance is of practical interest. In [10] a method for the design of irregular LDPC codes was proposed, which minimizes the number of iterations for a given target error probability. The key concept is to approximate the number of iterations by a continuous function, and use this as the objective function in the optimization. Starting from a capacity-achieving code, the variable-node degree distribution is iteratively adapted to improve this objective function.

We take a different approach, seeking a code that is optimized for a fixed number of iterations. We restrict ourselves to non-systematic IRA codes over the binary erasure channel (BEC). Extension to other sparses-graph codes and to other communication channels (given the usual assumptions and approximations with EXIT charts) is possible.

II. IRA CODES AND EXIT CHARTS

We assume familiarity with low-density parity-check codes, their graphical representation, iterative decoding, and EXIT chart based design [1–3, 5]. We focus on non-systematic irregular repeat accumulate codes, where information bits are repeated, interleaved, passed through parity-checks, and finally scrambled by an accumulator [6, 11]. Only the code bits produced by the accumulator are transmitted over the channel. We restrict our analysis to the binary erasure channel, for which the EXIT chart approach is exact [12].

The encoder maps \( k \) information bits onto \( n \) code bits, the code rate is \( R = k/n \). The code is defined by the edge perspective degree polynomials, \( \lambda(z) = \sum_i \lambda_i z^i \) for the variable-nodes and \( \rho(z) = \sum_i \rho_i z^i \) for the check-nodes, where \( \lambda_i, \rho_i \) denote proportion of edges connected to variable nodes of degree \( i \), and \( \lambda_i, \rho_i \) denote proportion of edges connected to check nodes of degree \( i \). According to common practice in the literature, \( \lambda_i \) does not count the connection to the accumulator. The design rate is

\[
R_d = \frac{\sum_i \lambda_i/i}{\sum_i \rho_i/i} = \int_0^1 \frac{\lambda(z)dz}{\int_0^1 \rho(z)dz} \leq R,
\]

The code bits are transmitted over a binary erasure channel with erasure probability \( \delta \) and capacity \( C = 1 - \delta \).

The iterative decoder consists of three components: the variable-node decoder, the check-node decoder, and the accumulator decoder. Only the accumulator obtains observations from the channel. In every iteration the component decoders are activated in the order variable-check-accumulator-check.

In the EXIT chart method, the mutual information transfer between the component decoders is modeled and used to analyze convergence behavior [5]. For the following analysis,
we combine the check-node decoder and the accumulator decoder into a single unit, henceforth referred to as the check-accumulator decoder. Since we have assumed a binary erasure channel, all virtual channels are also erasure channels and EXIT chart analysis is exact [12].

As shown in Figure 1, we orient the extrinsic information transfer chart with the $x$-axis representing the variable-node extrinsic information $I_{Ev}$ and the check-accumulator a-priori information $I_{Av}$. The $y$-axis carries the variable-node a-priori information $I_{Av}$ and check-accumulator extrinsic information, $I_{Eca}$. In order to avoid proliferation of subscripts, where it will not cause confusion, we simply use $x$ and $y$ to represent these quantities. The variable-node EXIT function $f(y)$ maps from $y = I_{Av}$ onto $x = I_{Ev}$. Conversely, the check-accumulator function $g(x)$ maps from $x = I_{Aca}$ onto $y = I_{Eca}$.

![EXIT chart](image)

**Fig. 1.** Example EXIT chart of a non-systematic IRA code: variable-node EXIT function $f(y)$ and check-accumulator EXIT function $g(x)$.

The variable node EXIT function is given by [12]

$$f(y) = \sum_i \lambda_i f_i(y)$$  

$$f_i(y) = 1 - (1 - y)^{i-1}, \quad i = 1, 2, \ldots$$

where $f_i$ is the transfer function for a degree-$i$ variable node. The check-accumulator EXIT function is [12]

$$g(x) = \sum_i \rho_i g_i(x),$$

$$g_i(x) = x^{i-1} \frac{C}{1 - (1 - C)x^i}^2, \quad i = 1, 2, \ldots$$

where $g_i$ is the transfer function for a degree-$i$ check node combined with the accumulator. We can show that $f(y)$ and $g(x)$ are linear in $\lambda$ and $\rho$ and convex in $y$ and $x$, respectively.

Denote $x^{(\ell)} = I_{Ev}^{(\ell)}$ the extrinsic information at the output of the variable node decoder after iteration $\ell$, and let $y^{(\ell)} = I_{Av}^{(\ell)}$ be the corresponding a-priori information. With reference to Figure 1, the corresponding values of the trajectory for $L$ iterations are given recursively for $\ell = 1, 2, \ldots$ by

$$x^{(\ell)} = f(y^{(\ell-1)}), \quad y^{(\ell)} = g(x^{(\ell)}).$$  \hspace{1cm} (6)

The initial values are $x^{(0)} = 0$ and $y^{(0)} = g(0)$, resulting from the fact that the variable nodes are not directly connected to the communication channel.

We can eliminate the $y^{(\ell)}$ from (6) to obtain $x^{(\ell)} = f(g(x^{(\ell-1)}))$. Going a step further, we may also substitute these equalities into each other to express the variable node extrinsic information after $L$ iterations $I_{Ev}^{(L)}$ as

$$x^{(L)} = h_L(x)$$  \hspace{1cm} (7)

where $h(x) = f(g(x))$ and $h_L$ denotes $L$-fold functional composition.

The areas under the variable node curve (with the orientation indicated in Figure 1) and under the check-accumulator curve are given by

$$1 - \int_0^1 f(y) \, dy = \sum_i \lambda_i = \int_0^1 \lambda(z) \, dz,$$  \hspace{1cm} (8)

$$\int_0^1 g(x) \, dx = C \sum_j \rho_j = C \int_0^1 \rho(z) \, dz,$$  \hspace{1cm} (9)

respectively. Substituting from (1) we have the following theorem, first proved for low-density parity-check codes in [12].

**Theorem 1 (Area Theorem):**

$$C - R_d = \frac{A}{\int_0^1 \rho(z) \, dz},$$

$$A = \int_0^1 g(x) \, dx + 1 - \int_0^1 f(y) \, dy.$$  \hspace{1cm} (11)

where $A$ is the area between the check-accumulator curve $g$ and the variable node curve $f$.

Thus, finding the code with the smallest rate gap from capacity corresponds to minimizing the area between $f$ and $g$. Note a minor difference from the original LDPC result, where $\int \rho$ sets a “scaling factor”. We do not require the inner check-accumulator code to have rate 1.

For code analysis and design, it is useful to express the (information) bit error rate of the decoder as a function of the mutual information exchanged during the iterative process.

**Theorem 2 (Bit Error Rate):** The bit error rate $P_b$ of a non-systematic IRA code transmitted over a BEC, with variable node degree distribution $\lambda$ and a-priori information $I_{Av}$ (to the variable nodes) is given by

$$P_b = 1 - \frac{I_v}{2},$$

$$I_v = \left( \sum_j \frac{\lambda_j}{j} \right)^{-1} \sum_i \frac{\lambda_i}{i} \left( 1 - (1 - I_{Av})^i \right).$$  \hspace{1cm} (13)

The value $I_v$ denotes the overall variable node information.
Note that $P_b$ is monotonically increasing in $I_{Av}$. For fixed design rate $R_d$, 
\[ I_v = \left( R_d \sum_j \frac{\rho_j}{j} \right)^{-1} - \sum_i \lambda_i \left( 1 - (1 - I_{Av})^i \right) \] (14)

which is linear in the $\lambda_i$, convex in the $\rho_j$ and concave in $I_{Av}$.

With sufficiently many iterations, the decoder will converge to the fixed point $x = f(g(x) = h(x)$, which for a well-designed code will occur at a value of $x$ very close to 1, corresponding to a low bit error rate. To prevent an undesirable error floor, we may wish to enforce a condition that the iteration does not get stuck at a fixed point prior to some target value $x = 1 - \xi$ for some small $\xi$. To this end, we require $x < h(x)$ for all $x \in (1 - \xi, 1)$ . For this range of $x$, we use a Taylor series expansion around $x = 1$, and obtain the condition $x < h(1) + h'(1) \cdot (x - 1)$, which is equivalent to
\[ h'(1) = 1 + \frac{x}{\lambda} \cdot \left( \frac{2 - C}{C} \sum_{j \geq 1} j \rho_j - 1 \right) < 1, \] (15)
referred to as the stability condition. This condition is always met if $\lambda_2 = 0$ (this choice is usually bad for capacity-achieving codes). Otherwise it is equivalent to
\[ \sum_{j \geq 1} j \rho_j < 1 + \frac{\lambda_2}{\lambda} \cdot \frac{C}{2 - C}. \] (16)

III. CODE DESIGN

The prevailing approach to code design finds the degree distribution which results in the minimal rate loss from capacity. According to Theorem 1, this corresponds to a curve fitting problem. Equivalently, for fixed $\rho$, the maximum design rate is achieved by maximization of $\sum_i \lambda_i/i$. This results in the following linear program.

\textbf{Problem 1 (Capacity Approaching Code):} Given the channel capacity $C$ and check-node degree distribution $\rho$, find the variable-node degree distribution $\lambda^* \cdot$ as the solution to the following linear program:
\[ \max_{\lambda} \sum_i \frac{\lambda_i}{i} \] s.t. \[ \sum_i \lambda_i \cdot f_i(y) > g^{-1}(y), \quad y \in [0, 1] \] (18)
\[ (1 + \lambda_2) \cdot \frac{C}{2 - C} - \lambda_2 \sum_j j \rho_j > 0 \] (19)
\[ \lambda_i > 0, \quad i = 1, 2, \ldots \] (20)
\[ \sum_i \lambda_i = 1. \] (21)

The constraints (18) ensure that there are no sub-optimal fixed points and (19) is the stability condition.

In many situations of practical interest, the implementation complexity of the decoder is limited, and it is therefore of interest to consider the design of codes that reduce decoder complexity, even if this is at the expense of reduced code rate. Complexity constraints may be introduced into the optimization problem (17) to a certain extent via (a) fixing the maximal variable node degree (which needs to be done anyhow in order to have a finite number of variables in Problem 1), or (b) introducing other linear constraints on the $\lambda_i$, e.g. corresponding to the cost of high degree nodes.

For a code optimized according to (17), operating at rates close to capacity, it may take many iterations to achieve the target bit error rate. It is therefore of great practical interest to consider the design of codes that achieve the best possible performance after a fixed number of iterations, denoted by $L$. One obvious approach would be to replace the objective function in (17) with the bit error rate at iteration $L$. From (12) and (13), we see that this is equivalent to maximizing the variable node information at iteration $L$,
\[ I_v^{(L)} = \left( \sum_i \frac{\lambda_i}{j} \right)^{-1} \sum_i \lambda_i \left( 1 - (1 - g(h^{L-1}(0)))^i \right) \] (22)

Note that $I_v^{(L)}$ is a multinomial of degree $L$ times the square of the maximal variable node degree. This could potentially present numerical difficulties for optimization. Furthermore, this formulation requires us to compute $h^L$ for each value of $L$ of interest. In the case of IRA codes and the BEC, we have analytical expressions for $f_i(y)$ and $g_i(x)$. In more general settings, we may only have numerical samples of these functions.

Motivated by (6), the key novel step in this paper is to introduce dummy variables $x_i, y_i, \ell = 0, 1, \ldots, L$. With reference to Figure 1, these are the values of the variable node extrinsic and a-priori information at iteration $\ell$. For given target rate $R$, we formulate our design problem as follows (where we have used (14)).

\textbf{Problem 2 (Minimum bit error rate after $L$ iterations):} Given $R < C$ determine the variable node degree distribution $\lambda^L$ and check node degree distribution $\rho^L$ that minimize the bit error rate after $L$ iterations as the solution to:
\[ \max_{\lambda, \rho, x, y} \left( R_d \sum_i \frac{\rho_i}{i} \right)^{-1} \sum_i \lambda_i \left( 1 - (1 - y_\ell)^i \right) \] (22)

s.t. (1), (19), (20), and (21) and
\[ \rho_i \geq 0, \quad i = 1, 2, \ldots \] (23)
\[ \sum_i \rho_i = 1. \] (24)
\[ x_0 = 0, y_0 = g(x_0) \] (25)
\[ x_\ell = \sum_j \lambda_j f_j(y_{\ell-1}), \quad \ell = 1, 2, \ldots, L \] (26)
\[ y_\ell = \sum_j \rho_j g_j(x_\ell), \quad \ell = 1, 2, \ldots, L \] (27)

In this setup, the target rate $R$ would be chosen to be less than capacity, sacrificing some code rate for potential performance improvements after $L$ iterations.
The new constraints (25),(26),(27) ensure consistency of the decoding trajectory according to (6). The trajectory constraints are convex, however as discussed earlier, the objective function is neither convex nor concave.

We have empirically observed that there is very little bit error rate penalty incurred by replacing the bit error rate objective function (22) with \( y_L \) or \( x_L \) (i.e. maximize the information passed between the decoders). In fact, this may be of interest if the code is an inner component of some other outer iteration, e.g. a multi-user decoder. This results in the following optimization problem:

**Problem 3 (Maximal information after \( L \) iterations):**

Given \( R < C \) determine the variable node degree distribution \( \lambda^{xL} \) and check node degree distribution \( \rho^{yL} \) that maximizes the variable node extrinsic information after \( L \) iterations as the solution to the following convex optimization problem:

\[
\max_{\lambda,\rho, x, y} x_L \quad \text{s.t.} \quad (1), (19), (20), (21), (23), (24), (25), (26), (27)
\]

Note that by “unrolling” the iteration, we obtain constraints that are independent of the maximum number of iterations \( L \). Furthermore, we do not need to know the \( L \)-fold functional composition \( f(g(...)) \). Instead, we only need to know \( f(y) \) and \( g(x) \) individually. This could be an additional advantage if these characteristics are only available numerically (e.g. for other kinds of codes).

**IV. NUMERICAL EXAMPLE**

We seek for a code of rate \( R = 0.5 \) for transmission over a BEC with erasure rate \( \delta = 0.3 \), that achieves a low bit error probability with \( L = 10 \) decoding iterations. We consider variable node degrees from the set \( \{2, 3, 4, 5, 6, 13, 20, 30\} \), and (for simplicity) the fixed check-node distribution \( \rho_1 = 0.2, \rho_0 = 0.8 \). (Note that degree-1 check nodes are required for non-systematic IRA codes.) Solving Problem 3 for maximal information, we obtain the variable node distribution \( \lambda^{(1)} \), see Table I, and the EXIT function \( f^{(1)} \), depicted in Fig. 2.

As an alternative approach, we designed a capacity-approaching code, according to Problem 1, for a BEC with a higher erasure rate, namely \( \delta = 0.47 \), to obtain a code rate of \( R = 0.5 \). We then use this code with \( L = 10 \) decoding iterations for the given BEC with \( \delta = 0.3 \). The resulting variable node distribution \( \lambda^{(2)} \) is given in Table I, and the EXIT function is depicted in Fig. 2.

For the maximal-information code, the tunnel between the EXIT functions is clearly more open towards the end, and for a given number of iterations a better performance can be achieved. Resulting from that, the bit error rate after \( L = 10 \) iterations is \( 10^{-11} \) for the maximal-information code and only \( 10^{-3} \) for the ‘capacity-achieving’ code.

**TABLE I**

**VARIABLE NODE DEGREE DISTRIBUTIONS (TWO-DIGIT PRECISION).**

<table>
<thead>
<tr>
<th>( i )</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>13</th>
<th>20</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \lambda^{(1)} )</td>
<td>0</td>
<td>0.11</td>
<td>0.66</td>
<td>0.08</td>
<td>0.06</td>
<td>0.06</td>
<td>0.02</td>
<td>0.01</td>
</tr>
<tr>
<td>( \lambda^{(2)} )</td>
<td>0.16</td>
<td>0.15</td>
<td>0</td>
<td>0</td>
<td>0.54</td>
<td>0.14</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

![Fig. 2. EXIT chart and decoding trajectories: check-accumulator EXIT function \( g \); variable-node EXIT functions for the maximal-information code (Problem 3), \( f^{(1)} \), and the ‘capacity-achieving’ code (Problem 1), \( f^{(2)} \).](image)

**ACKNOWLEDGMENT**

This work has been supported in parts by the Australian Research Council under the ARC Discovery Grant DP0986089.

**REFERENCES**

Abstract—This paper presents a fundamental information-theoretic scaling law for a heterogeneous cascade of links corrupted by Gaussian noise. For any set of fixed-rate encoding/decoding strategies applied at the links, it is claimed that a scaling of $\Theta(\log n)$ in block length is both sufficient and necessary for reliable communication. A proof of this claim is provided using tools from the convergence theory for inhomogeneous Markov chains.

I. INTRODUCTION

Consider a line network in which a message originating at a certain node is conveyed to the intended destination through a series of hops. The reception at every hop is impaired by zero-mean additive white Gaussian noise (AWGN) of a certain variance. The originating node as well as the intermediate nodes are subject to an average transmit-power constraint $P_0$ for the duration of transmission of any message. The messages, drawn from a certain set of size $M$, are encoded and transmitted as codewords in $\mathbb{R}^N$ that satisfy the transmit power constraint. Every intermediate hop decodes (possibly erroneously) the original message from its noisy observation and re-encodes it before transmitting to the next hop. One can form the (row-stochastic) probability transition matrix $P_i$ for the $i$th hop in which each row gives the conditional probability distribution of the message decoded at hop $i$, given a certain message sent by the previous hop:

$$P_i = \begin{pmatrix}
p_i(1|1) & p_i(2|1) & \cdots & p_i(M|1) \\
p_i(1|2) & p_i(2|2) & \cdots & p_i(M|2) \\
\vdots & \vdots & \ddots & \vdots \\
p_i(1|M) & p_i(2|M) & \cdots & p_i(M|M)
\end{pmatrix}.$$ 

Assume that a network of the type outlined above consists of $n$ hops. If the encoder-decoder pairs are identical at the hops, and if the noise variance at every hop is of $N$ to be identical for large $n$, the originating node as well as the intermediate nodes are subject to an average transmit-power constraint $P_0$ for the duration of transmission of any message. The messages, drawn from a certain set of size $M$, are encoded and transmitted as codewords in $\mathbb{R}^N$ that satisfy the transmit power constraint. Every intermediate hop decodes (possibly erroneously) the original message from its noisy observation and re-encodes it before transmitting to the next hop. One can form the (row-stochastic) probability transition matrix $P_i$ for the $i$th hop in which each row gives the conditional probability distribution of the message decoded at hop $i$, given a certain message sent by the previous hop:

$$P_i = \begin{pmatrix}
p_i(1|1) & p_i(2|1) & \cdots & p_i(M|1) \\
p_i(1|2) & p_i(2|2) & \cdots & p_i(M|2) \\
\vdots & \vdots & \ddots & \vdots \\
p_i(1|M) & p_i(2|M) & \cdots & p_i(M|M)
\end{pmatrix}.$$ 

The probability distribution of $\hat{W}_n$ will be almost independent of the original message $W$. This implies that the mutual information $I(W; \hat{W}_n)$ between the original message and the decoded message tends to zero with $n$. Hence, communication in the network is highly unreliable if $N$ and $M$ do not vary with the number of hops $n$.

On the other hand, if $N$ is large and if one chooses $M = 2^N R$ for any $R < \frac{1}{2} \log(1 + \frac{P}{\sigma^2})$, random-coding theorem states that there exists a channel code of block length $N$ such that the probability of error during decoding at any link is bounded above by $e^{-N E(R,P/\sigma^2)}$, where $E$ is the random coding exponent for the corresponding AWGN channel [2], [3]. By employing such a code for each link in the line network model, choosing any $f(n) \in o(1)^\dagger$, and letting $N$ vary with $n$ as $\frac{\log n}{f(n)}$, one obtains the following union bound on the probability of communication failure in the network:

$$P \left[ W_n \neq \hat{W}_n \right] \leq n P_e \leq n e^{-N E(R,P/\sigma^2)} = e^{-f(n)} \to 0.$$ 

Hence, it is possible to transmit messages with very high reliability in the network for a fixed code rate $R$, if the code length $N$ satisfies $N(n) \in \Theta(\log n)^\dagger$. On the other hand, the previous discussion about diminishing $I(W; \hat{W}_n)$ implies that any fixed-rate fixed-length coding scheme is not scalable for the line network. This leads to the natural question whether the scaling $\Theta(\log n)$ for $N$ is indeed optimal. In this paper, we show that for any scaling of $N$ satisfying $N(n) \in o(\log n)^\dagger$, the mutual information $I(W; \hat{W}_n)$ diminishes to zero with $n$ for any set of fixed-rate codes. This implies that the scaling order $\Theta(\log n)$ for $N$ is necessary for reliable communication in the network. Hence, the sufficiency criterion for $N$ given by the union bound in Eqn. (1) is indeed optimal up to a constant factor.

While this scaling problem has been solved partially in the past for cascades of homogeneous DMCs by Niesen et al. in [4], it has also been pointed out in the same work that the tools developed there apply neither to non-homogeneous cascades nor to cascades of continuous-input channels (as is the case in the current paper). In contrast, the results developed here are applicable to continuous-input AWGN links where the links can be non-identical.

1 For any $f(n) > 0$ and $g(n) > 0$, $f(n) \in o(g(n)) \iff \lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$, $f(n) \in \Theta(g(n)) \iff \exists c > 0 \text{ s.t. } \lim_{n \to \infty} \frac{f(n)}{g(n)} = c$. 

This work has been supported by the Australian Research Council under the ARC Discovery Grant D90986089.
II. NOTATIONS AND DEFINITIONS

The set of all real numbers is denoted by \( \mathbb{R} \) and the set of all natural numbers by \( \mathbb{N} \). Natural logarithms are assumed in the notations, unless the base is specified. The notation \( \| \cdot \| \) represents \( L^2 \) norm throughout.

We employ the following terminologies in this paper. Let \( N \in \mathbb{N} \), called the code length or block length of the transmission scheme. A code rate \( R > 0 \) is a real number such that \( 2^{NR} \) is an integer. Let \( \mathcal{M} \triangleq \{1, 2, 3, \cdots, 2^{NR}\} \), called the message alphabet. Let \( P_0 > 0 \) be the input power constraint for transmission.

Definition 1: A rate-\( R \) length-\( N \) code \( \mathcal{C} \) with a power constraint of \( P_0 \) is an ordering of \( M = 2^{NR} \) elements from \( \mathbb{R}^N \), called code words that satisfies
\[
\mathcal{C} = (x_1, x_2, x_3, \cdots, x_M) \quad \text{s.t.} \quad \frac{1}{N} \|x_w\|^2 \leq P_0. \tag{2}
\]

Definition 2: A rate-\( R \) length-\( N \) decision rule \( \mathcal{A} \) is an ordering of \( M = 2^{NR} \) sub-sets of \( \mathbb{R}^N \), called decision regions, spanning \( \mathbb{R}^N \) and pairwise disjoint:
\[
\mathcal{A} = (\mathcal{R}_1, \mathcal{R}_2, \cdots, \mathcal{R}_M) \quad \text{s.t.} \quad \bigcup_{w \in \mathcal{A}} \mathcal{R}_w = \mathbb{R}^N, \quad \text{and} \quad \mathcal{R}_{w_1} \cap \mathcal{R}_{w_2} = \emptyset. \tag{3}
\]

Definition 3: The encoding function \( ENC_{\mathcal{C}} : \mathcal{M} \to \mathbb{R}^N \) for a code \( \mathcal{C} \) is defined by:
\[
ENC_{\mathcal{C}}(w) = x_w, \tag{4}
\]
where \( x_w \) is the \( w \)-th code word in \( \mathcal{C} \).

Definition 4: The decoding function \( DEC_{\mathcal{A}} : \mathbb{R}^N \to \mathcal{M} \) for a decision rule \( \mathcal{A} \) is defined by:
\[
DEC_{\mathcal{A}}(y) = \sum_{i \in \mathcal{A}} w I_{\mathcal{R}_i}(y), \tag{5}
\]
where \( I \) is the indicator function and \( \mathcal{R}_w \) is the \( w \)-th subset in \( \mathcal{A} \).

III. NETWORK MODEL

The line network model to be considered is described in Fig. 1. There are \( n+1 \) nodes in the network identified by the indices \( \{0, 1, 2, \cdots, n\} \). The \( n \) hops in the network are each associated with noise variances \( \sigma_i^2 \), \( 1 \leq i \leq n \). For encoding, nodes \( 0, 1, \cdots, n-1 \) choose \( C_0, C_1, \cdots, C_{n-1} \) respectively to transmit, each code being rate-\( R \) length-\( N \) with a power constraint of \( P_0 \). For decoding, nodes \( 1, 2, \cdots, n \) choose rate-\( R \) length-\( N \) decision rules \( \mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_n \) for reception. From here on, we write \( ENC_{\mathcal{C}_i} \) and \( DEC_{\mathcal{A}_i} \) in place of \( ENC_{\mathcal{C}_0} \) and \( DEC_{\mathcal{A}_0} \) respectively, for simplicity. Node 0 generates a random message \( W \in \mathcal{M} \) with probability distribution \( p_W(w) \) and intends to convey the same to node \( n \) through the cascade of noisy links in a multihop fashion. Each of the \( n-1 \) intermediate nodes estimates the message (as sent by the node in the previous hop) from its own noisy observation, re-encodes the decoded message, and transmits the resulting codeword to the next hop. The codeword transmitted by node \( i \), for any \( 0 \leq i \leq n-1 \) is given by
\[
X_i = ENC_{\mathcal{C}_i}(\hat{W}_i), \tag{6}
\]
where \( \hat{W}_i \) is the estimate of the message at node \( i \) after decoding (Note that \( \hat{W}_0 = W \) in this notation). The observation received by node \( i \), for any \( 1 \leq i \leq n \), is given by \( Y_i \), which follows a conditional density function that depends on the codeword \( X_{i-1} \) sent by the previous node:
\[
p_{Y_i|X_{i-1}}(y|x) = \frac{1}{(2\pi\sigma_i^2)^{n/2}} e^{-\frac{|y-x|^2}{2\sigma_i^2}}. \tag{7}
\]
The above density function follows from the assumptions of AWGN noise and memoryless-ness in the channel. The message \( \hat{W}_i \) decoded by node \( i \) is given by
\[
\hat{W}_i = DEC_{\mathcal{A}_i}(Y_i). \tag{8}
\]
Note that the random variable \( \hat{W}_n \) represents the message decoded by the final destination, as per the above notation.

IV. AN UPPER BOUND ON THE RELIABILITY OF THE NETWORK

The key result presented here is summarized by the following theorem and its corollary:

Theorem 1: In a line network consisting of a cascade of \( n \) AWGN links, let for \( m \leq n \) of the links the noise variances satisfy \( \sigma_m^2 \geq \sigma_d^2 > 0 \). Then, for any choice of codes \( \mathcal{C}_0, \mathcal{C}_1, \cdots, \mathcal{C}_{n-1} \) and decision rules \( \mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_n \) with rate \( R \) length \( N \), and for any \( \epsilon > 0 \), the following holds for sufficiently large \( m \):
\[
I(W; \hat{W}_n) \leq 2^{NR} \left( 1 - \frac{\epsilon^2}{\Gamma(\frac{n}{2} + 1)} \right)^{m(1-\epsilon)} E(S), \tag{9}
\]
where
\[
E(S) = \exp \left\{ \cosh^{-1} \left( \frac{S}{2} \right) \right\} + \cosh^{-1} \left( \frac{S}{2} + 1 \right). \tag{10}
\]

The corollary to the above theorem leads to the upper bound on the reliability vs. block-length scaling as claimed in the introductory section:

Corollary 1: Let \( R > 0 \), \( \sigma_0 > 0 \), and \( 1 \geq r > 0 \) be independent of \( n \). In a line network of \( n \) links, let for \( r \) of the hops the noise variance be at least \( \sigma_0^2 > 0 \). Then, for any \( N(n) \in (0, n) \),
\[
\lim_{n \to \infty} I(W; \hat{W}_n) = 0. \tag{11}
\]

Proof of Theorem 1: At any hop \( i \), the process of channel encoding, noisy reception, followed by decoding induces the conditional probabilities \( p_{\hat{W}_i|W_i}(k|j), \forall j, k \in \mathcal{M} \). For every hop \( i \), let \( P_i \) be the \( M \times M \) row-stochastic matrix whose entry in row \( j \) and column \( k \) is \( p_{\hat{W}_i|W_i}(k|j) \). Note that the \( j \)-th row in \( P_i \) gives the conditional probability mass function on the estimate of message \( W \) at hop \( i \) (i.e., \( \hat{W}_i \)) given that the message sent by hop \( i - 1 \) was \( j \). Let
\[
P \triangleq \prod_{i=1}^{n} P_i. \tag{12}
\]
Then, \( P \) clearly represents the row-stochastic probability transition matrix between the original message \( W \) and the message decoded at the final destination, \( \hat{W}_n \). The transition matrix \( P \)
along with $p_W$, the probability mass function of the original message $W$ together induce a joint distribution between $W$ and $\hat{W}_n$. Our goal is to find an upper bound on $I(W;\hat{W}_n)$, with the constraints as given in the statement of Theorem 1.

For any $M \times M$ row-stochastic matrix $Q = \{Q_{jk}\}$, let us consider the following measures of deviation:

$$
\delta(Q) \triangleq \max_{j_1,j_2} \max_k |Q_{j_1,k} - Q_{j_2,k}|, \quad (10)
$$

$$
\lambda(Q) \triangleq \lambda_{\min} \left( \sum_{k=1}^{M} \min_{j_1,j_2} (Q_{j_1,k}, Q_{j_2,k}) \right). \quad (11)
$$

Both the above deviations measure the degree to which the rows of $Q$ differ. The measure $\lambda(Q)$ is known as Dobrushin’s coefficient of ergodicity and plays a role in the convergence of non-homogeneous stochastic processes [5]. Moreover, for any stochastic matrix $Q$,

$$
0 \leq \lambda(Q) \leq 1. \quad (12)
$$

Now consider the total variational distance between the joint distribution of $W$ and $\hat{W}_n$ and the product of the respective marginal distributions:

$$
\tau(W;\hat{W}_n) \triangleq \sum_{w_1,\hat{w} \in R^n} |p_W(w_1)p_{\hat{W}_n}(\hat{w}) - p_W(\hat{w})p_{\hat{W}_n}(w_1, \hat{w})|. \quad (13)
$$

In the above expression, substituting $p_W(w_1)p_{\hat{W}_n}(\hat{w}|w_1)$ for $p_W(\hat{w})p_{\hat{W}_n}(w_1, \hat{w})$, $\sum_{w_1,\hat{w} \in R^n} p_W(w_1)p_{\hat{W}_n}(\hat{w}|w_2)p_W(w_2)$ for $p_W(\hat{w}_2)p_{\hat{W}_n}(w_1, \hat{w})$ and by using the triangle inequality, one obtains the following bound:

$$
\tau(W;\hat{W}_n) \leq \sum_{w_1,\hat{w} \in R^n} p_W(w_1)p_{\hat{W}_n}(\hat{w}|w_2)p_W(w_2)\delta(P) \leq M\delta(P) = 2^{N_R}\delta(P). \quad (13)
$$

Recalling the definition of $P$ in our case, and from Theorem 2 in [6] (equivalently, Lemma 2 in [7]), we have:

$$
\delta(P) = \delta \left( \prod_{i=1}^{n} P_i \right) \leq \prod_{i=1}^{n} \lambda(P_i) \leq \prod_{i \in [n]} \lambda(P_i). \quad (14)
$$

The second inequality in the above follows by applying Eqn. (12) for those hops for which $\sigma_i < \sigma_0$. Note that as per the statement of the theorem, we have at least $m$ terms in the product in the last expression above. Now consider $\lambda(P_i)$ for any hop $i$ such that $\sigma_i \geq \sigma_0$. Suppose $P_i$ has $L$ distinct columns with indices $k_1, k_2, \ldots, k_L$ such that

$$
\forall j, P_{j,k_1} = p_{W_i|\hat{W}_{i-1}}(k_1|j) \geq p_1
$$

$$
\vdots
$$

$$
\forall j, P_{j,k_L} = p_{W_i|\hat{W}_{i-1}}(k_L|j) \geq p_L. \quad (15)
$$

Then from the definition in Eqn. (11), it follows that

$$
1 - \lambda(P_i) \geq \sum_{\ell=1}^{L} p_{\ell}. \quad (17)
$$

We now show that the conditions given by Eqn. (15)-(16) hold true in our case, so that we can apply Eqn. (17) to our bound. Let $\alpha > 0$. Consider one particular hop $i$ satisfying $\sigma_i \geq \sigma_0$. Define the spherical regions

$$
S_1 = \left\{ y \in R^{N} : \|y\| \leq \sqrt{N_P} \right\}
$$

$$
S_{\alpha} = \left\{ y \in R^{N} : \|y\| \leq \alpha \sqrt{N_P} \right\}. \quad (18)
$$

Clearly, any code word in $S_{\alpha} \cap \hat{S}_{\alpha}$ (transmitted by the previous hop) has to be in $S_1$. Next, observe that $S_{\alpha}$ has to contain decision regions, say $L$ in number, spanning $S_{\alpha}$ entirely. Let us designate them as $R_{k_1}, R_{k_2}, \ldots, R_{k_L}$. Also define for each $1 \leq \ell \leq L$, $V_\ell \triangleq R_{k_\ell} \cap S_\alpha$. Hence, we have:

$$
\bigcup_{\ell=1}^{L} V_\ell = S_\alpha, \quad \text{and} \quad \forall \ell_1 \neq \ell_2 V_{\ell_1} \cap V_{\ell_2} = \emptyset. \quad (18)
$$

Let $\hat{W}_{i-1} = j$ be the message as decoded by the previous hop. Hence, the code word $x_i$ from $\hat{W}_{i-1}$ was transmitted by the node $i - 1$. The probability that this was decoded as message $k_\ell$ at hop $i$ is given by:

$$
p_{W_i|\hat{W}_{i-1}}(k_\ell|j) = \int_{y \in R^{N}} p_{Y_i|X_i}\left(y|x_i\right) dy 
\geq \int_{y \in V_\ell} p_{Y_i|X_i}\left(y|x_i\right) dy 
\geq \frac{1}{2\pi \sigma_i^2} \int_{y \in V_\ell} \frac{e^{-\frac{\|y-x_i\|^2}{2\sigma_i^2}}}{\sqrt{2\pi}} dy 
\geq \frac{\left(2\pi \alpha^2\sigma_i^2\right)^{-\frac{N}{2}}}{\sqrt{2\pi}} \int_{y \in V_\ell} e^{-\frac{\|y-x_i\|^2}{2\alpha^2\sigma_i^2}} dy 
\geq \frac{\left(2\pi \alpha^2\sigma_i^2\right)^{-\frac{N}{2}}}{\sqrt{2\pi}} \int_{y \in V_\ell} e^{-\frac{\|y-x_i\|^2}{2\alpha^2\sigma_i^2}} dy 
\geq e^{-\frac{1}{2\alpha^2\sigma_i^2} \|x_i\|^2} |V_\ell|. \quad (19)
where $|V_i|$ denotes the $N$-dimensional volume of $V_i$. In the above series of inequalities, step (a) follows from Eqn. (7), (b) from the triangle inequality, and (c) from the fact that any $y \in V_i$ must satisfy $|y| \leq \alpha \sqrt{NF_0}$, since $V_i \subseteq S_0$. Since at least $m$ hops satisfy the minimum noise variance criterion $\sigma_i \geq \sigma_0$, we can now combine Eqn. (24) with Eqn. (14) to obtain

$$
\delta(P) \leq \left( 1 - \frac{\left( \frac{N}{2} \right)^2}{\Gamma \left( \frac{N}{2} + 1 \right)} e^{-NE'(\rho_b/\sigma_0^2)} \right)^m.
$$

Applying the above to Eqn. (13), we will have

$$
\tau(W;W_n) \leq 2^{NR} \left( 1 - \frac{\left( \frac{N}{2} \right)^2}{\Gamma \left( \frac{N}{2} + 1 \right)} e^{-NE'(\rho_b/\sigma_0^2)} \right)^m.
$$

The bound given by the statement of the theorem then follows from combining the above with Eqn. (26) and by observing that for sufficiently large $m$, $\tau(W;W_n) \leq e^{-1}$, that $-x \log x$ is non-decreasing for $x \leq e^{-1}$, and finally from the fact that for any $\epsilon > 0$, $x^\epsilon \geq \log x$ for sufficiently large $x$.

**Proof of Corollary 1:** For the corollary, we have $m = nr$. Moreover, $\Gamma \left( \frac{N}{2} + 1 \right) \leq \left( \frac{N}{2} \right)^{\frac{N}{2}}$. Hence, the bound given by Theorem 1 reduces to:

$$
\tau(W;W_n) \leq 2^{NR} \left( 1 - e^{-NE'(\rho_b/\sigma_0^2)} \right)^{nr(1-\epsilon)}
$$

$$
\leq 2^{NR} e^{-nr(1-\epsilon)e^{-NE'(\rho_b/\sigma_0^2)}}
$$

$$
= 2^{NR} \log 2 e^{-nr(1-\epsilon)\log(1-e^{-NE'(\rho_b/\sigma_0^2)})}
$$

The last expression diminishes to 0 with $n \to \infty$ if $N = N(n) \in o(\log n)$. Hence, we will have $\tau(W;W_n) \to 0$ for any $N \in o(\log n)$.

---

**REFERENCES**


Secure Multiplex Coding Over Interference Channel with Confidential Messages

Xiaolin Li
Dept. of Electrical and Computer Engineering
Hong Kong University of Science and Technology
darwinlx1@ust.hk

Ryutaroh Matsumoto
Dept. of Communications and Integrated Systems
Tokyo Institute of Technology, 152–8550 Japan, and Dept. of Mathematical Sciences, Aalborg University, Denmark.
ryutaroh@it.ss.titech.ac.jp

Abstract—In this paper, an inner bound on the capacity region of two-user interference channels with two confidential messages has been proposed. With secure multiplex coding, we show that the Han-Kobayashi region can be achieved with strong security requirement. The equivocation rate for a collection of secret messages has also been evaluated, when the length of the message is finite or the information rate is high, our result provides a good approximation for bounding the worst case equivocation rate.

Index Terms—Information theoretic security, capacity region, interference channel, secure multiplex coding, strong security.

I. INTRODUCTION

In [12] the authors proposed the celebrated Han-Kobayashi region which provided the best inner bound on the channel capacity of the interference channel known to now. In the coding scheme proposed in that paper, the information from each transmitter was divided into two parts, the first part was for only one receiver (we say this part of the information is sent over the “private channel”), and the other part could be decoded by both receivers (we say this part of the information is sent over the “common channel”). A natural question raises: what is the secrecy transmission rate if only confidential messages are sent? In [2], the authors proposed an inner bound of secure transmission over interference channel by modifying the coding scheme in [11], and information is only sent over the “private channel”, which is a natural solution. But in this paper, we show somewhat surprisingly that even if we transmit over the “common channel”, confidentiality can also be guaranteed, thus we propose a larger achievable security rate region.

Also in [2], outer and inner bounds have been provided, but are under the weak secrecy requirement [3], which requires that the mutual information divided by the length of the codeword goes to zero as the codeword length goes to infinity. But this requirement is not strong enough for some applications [8] [9], because even if this rate goes to zero asymptotically, vital information bits can still be easily leaked to an illegitimate receiver. Moreover, secrecy is achieved by adding dummy random bits into the transmitted signal, which inevitably decreases the information rate.

The authors in [2] did not evaluate the equivocation rate when the information rates of the secret messages are large or the length of the message is finite. This means that their results are only valid for the cases where secrecy can be asymptotically achieved, but if the secrecy requirement is not achieved, they are not able to evaluate how much information may be leaked out.

In [4], the authors proposed the secure multiplex coding scheme for wiretap channels, the goal of which is to remove the rate loss incurred by the random dummy message. In [5] and [7], the authors applied the secure multiplex coding in different scenarios: broadcast channels with a common message and secure network coding. They showed that secure multiplex coding can not only remove the information rate loss, but can also achieve strong security within the capacity region.

In this paper, the model of interference channel with confidential messages as in [2] is considered. By applying the technique of secure multiplex coding, we propose an inner bound that can achieve the Han-Kobayashi region with the strong security requirement. Moreover, we give the dominating term approximation for a lower bound on the equivocation rate with finite message length.

This paper is organized as the following: in Section II, the system model and preliminaries shall be introduced. In Section III, we present an inner bound on the secure capacity of the interference channel. Section IV concludes the paper.

II. SYSTEM MODEL AND PRELIMINARIES

A. System Model

We adopt the same channel model as in [2]. Consider a discrete memoryless interference channel with finite input alphabets $\mathcal{X}_1$, $\mathcal{X}_2$, finite output alphabets $\mathcal{Y}_1$, $\mathcal{Y}_2$, and the channel transition probability distribution $P_{Y_1,Y_2|X_1,X_2}$. Two transmitters wish to send independent, confidential messages to their respective receivers.

The definition of the capacity region for the interference channel with secure multiplex coding is given as follow:

Definition 1: The rate tuple $(R_{1,1}, \ldots, R_{1,T_1}, R_{2,1}, \ldots, R_{2,T_2})$ is achievable for the secure multiplex coding with $T_1$ secret messages for sender 1 and $T_2$ messages for sender 2, if there exists a sequence of stochastic encoders for sender 1 denoted as $\zeta^1_n: \mathcal{C}^n_{1,1} \times \cdots \times \mathcal{C}^n_{1,T_1} \rightarrow \mathcal{X}_1^n$, and for sender 2 denoted as $\zeta^2_n: \mathcal{C}^n_{2,1} \times \cdots \times \mathcal{C}^n_{2,T_2} \rightarrow \mathcal{X}_2^n$, and for receiver 1 deterministic
decoder $\phi_n^1 : Y^n \rightarrow C^n_1 \times \cdots \times C^n_{1,T_1}$, and for receiver 2 deterministic decoder $\phi_n^2 : Y^n \rightarrow C^n_2 \times \cdots \times C^n_{2,T_2}$ such that

$$\lim_{n \to \infty} \Pr((C^n_{1,1}, \ldots, C^n_{1,T_1}) \neq \phi_n^1(Y^n)) = 0,$$

or

$$\lim_{n \to \infty} I(C^n_{1,1}; Y^n) = 0 \quad \text{if} \quad R_1 x_1, e = \sum_{i \in I_1},$$

$$\lim_{n \to \infty} I(C^n_{2,1}; Y^n) = 0 \quad \text{if} \quad R_2 x_2, e = \sum_{i \in I_2},$$

$$\liminf_{n \to \infty} \frac{\log|C^n_{1,1}|}{n} \geq R_1,$$

$$\liminf_{n \to \infty} \frac{\log|C^n_{2,1}|}{n} \geq R_2.$$

for $i = 1, \ldots, T_1$ and $j = 1, \ldots, T_2$, where $C^n_{1,i}$ and $C^n_{2,j}$ represent the $i$-th secret message from sender 1 and the $j$-th secret message from sender 2 respectively. All of $C^n_{1,i}$ and $C^n_{2,j}$ have uniform distribution on $C^n_{1,i}$ and $C^n_{2,j}$ and are statistically independent. Both of $C^n_{2,j}$ and $C^n_{2,j}$ are collections of random variables: $C^n_{2,1} = \{C^n_{2,j} | j \in I_2\}$ and $C^n_{2,2} = \{C^n_{2,j} | j \in I_2\}$. The received signals by the two receivers are denoted as $Y^n_1$ and $Y^n_2$, with the transmitted signals $\xi^n(C^n_{1,1}, \ldots, C^n_{1,T_1})$, $\xi^n(C^n_{2,1}, \ldots, C^n_{2,T_2})$, and the channel transition probability $P_{Y_1, Y_2|I_1, x_1, x_2}$. The capacity region of the secure multiplex coding is the closure of the achievable rate tuples.

**Remark 2:** Note that (2) and (3) reflect the requirement of strong secrecy [3].

The main idea behind the multiplex coding is that instead of making the mutual information between $C_1 = \{C^n_{1,1}, \ldots, C^n_{1,T_1}\}$ and $Y^n$ to be zero, we now only need to ensure $I(C^n_{1,1}, Y^n)$ vanishes, which means each multiplex channel is secure. Since other messages $(C^n_{1,1}, \ldots, C^n_{1,T_1})$, $(C^n_{1,1}, \ldots, C^n_{1,T_1}, C^n_{1,T_1+1}, \ldots, C^n_{1,T_1+T_2})$ are independent with $C^n_{1,i}$, they acted as noise and provide protection for $C^n_{1,i}$.

**B. Preliminaries**

Throughout the whole paper we use natural log.

The definition of interference channel can be found in [1], in the following we introduce the Han-Kobayashi region:

**Theorem 3:** [11] Let $P^*_1(\cdot)$ be the set of probability distribution $P^*_1(\cdot)$ that factor as

$$P^*(u, w_1, w_2, v_1, v_2) = p(u)p(v_1, w_1|u)p(v_2, w_2|u) \quad (8)$$

Let $R_{HK}(P^*_1(\cdot))$ be the set of nonnegative rate-tuples $(R_1, R_2)$ that satisfy

$$R_1 \leq I(V_1; Y_1 | W_1 U),$$

$$R_1 \leq I(V_1; Y_1 | W_1 U) + I(V_2 W_1; Y_2 W_2 U) + I(V_2 W_1; Y_1 W_2 U). \quad (11)$$

and

$$R_2 \leq I(V_2; Y_2 | W_1 U),$$

$$R_2 \leq I(V_2; Y_2 | W_1 U) + I(V_2 W_1; Y_2 W_2 U),$$

$$R_1 + R_2 \leq I(V_1 W_1; Y_1 W_2 U) + I(V_2 W_1; Y_2 W_2 U),$$

$$R_1 + R_2 \leq I(V_1 W_1; Y_1 W_2 U) + I(V_2 W_1; Y_2 W_2 U) + I(V_2 W_1; Y_2 W_2 U). \quad (17)$$

Then we have

$$R_{HK} = \cup_{P^*_1(\cdot)} R_{HK}(P^*_1(\cdot))$$

is an achievable rate region for the discrete memoryless interference channel.

**III. Inner Bound on the Secure Multiplex Coding with Strong Secrecy Requirement**

Denote the total rate of the sender $t$ by $R_t = \sum_{i=1}^{t-1} R_i \leq I(V_t; Y_t | U)$ (here $t = 1$ or 2, and we adopt this notation throughout the paper). An inner bound is proposed as the following:

**Theorem 4:** Let $P^*_2(\cdot)$ that factor as

$$P(u, w_1, w_2, v_1, x_1, x_2, y_1, y_2) = p(u)p(w_1, w_2, v_1, x_1, x_2|u)p(y_1, y_2|x_1, x_2) \quad (20)$$

$$P(u, v_1, v_2|w_1, w_2)P(x_1|v_1)p(x_2|v_2) \quad (21)$$

$$P(y_1, y_2|x_1, x_2) \quad (22)$$

In the above, $x_1$, $x_2$ and $y_1$, $y_2$ are the inputs and outputs of the interference channel respectively.

And $R_{HK}(P^*_2(\cdot))$ be the set of nonnegative rate-tuples and $(R_1, R_2, R_{1,x_1, e}, R_{2,x_2, e})$ satisfy

$$R_{1,x_1, e} = R_1 + I(V_1; Y_1 | U, V_2) \quad (23)$$

$$R_{2,x_2, e} = R_2 + I(V_2; Y_1 | U, V_1) \quad (24)$$

$$(R_1, R_2) \in R_{HK}(P^*_2(\cdot)) \quad (25)$$

$$0 \leq R_{1,x_1, e} \leq \sum_{i \in I_1} R_{1,i} \quad (26)$$

Note in the above we abuse the notation a little by writing $R_{HK}(P^*_2(\cdot))$, we can write this because if $P(u, w_1, v_1, w_2, v_2, x_1, x_2, y_1, y_2) \in P^*_2(\cdot)$, then the marginal distribution $P(u, w_1, v_1, w_2, v_2, x_1, x_2, y_1, y_2) \in P^*_2(\cdot)$.

An inner bound for the interference channels with secure multiplex coding is
\[ R_{in} = \cup_{P_2} C P_{in}(P_2) \]

**Proof:**

**Part 1: Random Coding Scheme**

Before presenting the random coding scheme, some notations are introduced: let \( (c_{i1}, \ldots, c_{iT_i}) \in (C_{i1}^1, \ldots, C_{iT_i}^1) \) be the secret messages for transmitter \( t \), and denote \( C_t = \prod_{i=1}^{T_t} C_{iT_i}^t \), where \( C_{iT_i}^t \) is the alphabet of randomness used by the stochastic encoder, and \( n \) denotes the code length. Let \( F_t^i \) be the set of all linear bijective maps from \( C_t^i \) to itself.

We modify the random coding scheme proposed in [11], and apply the secure multiplex coding techniques. The new scheme is described in detail as follows:

Fix the distribution of \( P(u), P(w_t, u_t) \) and \( P(x_t | v_t) \), also since the channel distribution \( P(y_1, y_2 | x_1, x_2) \) is given, all the distributions in \( (20) \) are now fixed.

1. **Codebook Generation:** Sender \( t \) and receiver \( t \) fix and agree on the choice of a bijective function \( f_t^i \in F_t^i \). Given \( T_t \) secret messages \( (c_{i1}, \ldots, c_{iT_t}) \in (C_{i1}^t, \ldots, C_{iT_t}^t) \), uniformly choose \( c_{iT_t+1} \in C_{iT_t+1}^t \), let \( \alpha = (f_t^i)^{-1}(c_{i1}, \ldots, c_{iT_t+1}) \). Here the message \( c_{iT_t+1} \) is used by the stochastic encoder to increase the randomness in the secret message. In order to use multiplex coding, we write

\[ C_t^n = (E_1^n, B_t^n) = (f_t^n)^{-1}(C_{1T_t}^1, \ldots, C_{iT_t}^t) \]

Here \( f_t^n \) belongs to the family of linear bijective maps \( F_t^n \), and this is achieved by matrix multiplication. Applying \( (f_t^n)^{-1} \) on the secret messages can be achieved by matrix multiplication:

\[ C_t^n = (L_1^n)^{-1} [C_{1T_t}^1, \ldots, C_{iT_t}^t]^T \]

Note that if the length of \( C_{iT_t}^t \) is \( k_{iT_t} \), then \( L_t = k_{iT_t} \) is a nonsingular matrix of size \( k_t \times k_t \). Randomly generate a sequence \( u \) with probability \( P(u) = \prod_{i=0}^{n-1} P(u_i) \), and assume that both transmitters and receivers know the time-sharing sequence \( u \).

For transmitter \( t \), generate \( 2^n S_t \) independent sequences \( S_t \) is the information rate over the “common channel” \( w_t \), each with probability \( P(w_t | u) = \prod_{i=0}^{n-1} P(w_{i+1} | u_i) \). Then generate \( 2^n R_t \) \((R_t = S_t + T_t, \text{ and } T_t \) is the rate of information over “private channel” \) independent sequences \( v_t \) each with probability \( P(v_t | u) = \prod_{i=0}^{n-1} P(v_{i+1} | u_i) \).

2. **Encoding:** Encode \( c_t \) with encoder \( t1 \) and \( t2 \) (here \( t = 1 \) or \( 2 \), refer to Fig. 1), according to the codebook generated in the previous step and obtain the codeword \( u^n \).

The transmitters generate the channel input sequences based on respective mappings \( PX_i | V_i \) and \( PX_2 | V_2 \).

The structure of the encoder is illustrated in Fig. 1. The codebook for encoder \( i \) is denoted as \( A_{ij} \in A_{ij} (i, j = 1, 2) \), for the simplicity of notations, let \( \Lambda = (\Lambda_{11}, \Lambda_{12}, \Lambda_{21}, \Lambda_{22}) \), and \( \Lambda = (\Lambda_{11}, \Lambda_{12}, \Lambda_{21}, \Lambda_{22}) \).

3. **Decoding:** Without loss of generality, we consider for receiver \( 1 \). Let \( A^{(n)}(W_1, V_1, W_2, Y_1, U) \) denote the set of jointly typical sequences defined in [13]. Since receiver 1 is supposed to know the time sharing sequence \( U^n \), after receiving \( Y^n_2 \), the receiver will try to find \( W_1^n, W_2^n, Y_1^n, U^n \) such that \( (W_1^n, V_1^n, W_2^n, Y_1^n, U^n) \in A^{(n)}(W_1, V_1, W_2, Y_1, U) \).

Fig. 1. Code construction for the interference channel with confidential messages.
Decoding error probability \( \leq 2^{T+2} \delta_n \).

In the above expressions, \( \epsilon(\rho) \) is a constant that only depends on \( \rho \), and \( \lim_{\rho \to 0} \epsilon(\rho) = 0. \)

By Eq. (29) we can see
\[
I(C_1; Y_1^n | F_1^n = f_1^n, \Lambda = \lambda, U^n = u)) \\
\leq 1 + \log\left(\frac{2^{T+2}}{n\rho}\right) + R_T - R_P + I(V_1; Y_2 | U V_2) + \epsilon(\rho).
\]

for \( R_T - R_P + I(V; Z | U) + \epsilon(\rho) \geq 0. \) By Eq. (30) we can see that the equivocation rate \( H(C_1, Y_2^n | F_1^n = f_1^n, \Lambda = \lambda) / n \) becomes larger than the required value \( R_{1,\epsilon} \) for sufficiently large \( n \). This completes the analysis of the equivocation rates and the mutual information for all \( \emptyset \neq I \subseteq \{1, \ldots, T_1\} \).

Remark 5: This inner bound of secret capacity over interference channel given above shows that the whole Han-Kobayashi region can be achieved. Moreover from (27) and (28), it can be observed that strong security can be achieved. Note that (28) – (30) also provides an upper bound for the leaked information, which is not analyzed in [2].

IV. GAUSSIAN INTERFERENCE CHANNEL AND NUMERICAL RESULTS

In this section we use the same model and notations of Gaussian interference channel as in Section III part D in [1]. Then we compare the inner bound we proposed with that in [1]. For the sake of “fair” comparison, in our inner bound we only use a “trivial” multiplex coding, namely we let \( T_1 = T_2 = 1 \), this is equivalent to use dummy message to achieve secrecy. We can see in Fig. 2 our inner bound (denoted as “Secure-Han-Kobayashi region”) is strictly larger than the one proposed in [2].

V. CONCLUSIONS

In this paper an inner bound that achieves Han-Kobayashi region for secure multiplex coding over interference channel has been proposed. We have also presented a random coding scheme that achieves the inner bound. We have substituted the weak secrecy requirement by the strong one, and removed the information rate loss caused by the dummy message. Moreover we evaluated the equivocation rate for a collection of secret messages.

VI. ACKNOWLEDGEMENT

This work was partly supported by the AoE Grant E-02/08 from the University Grants Committee of the Hong Kong Special Administration Region, China, and also by the MEXT Grant-in-Aid for Young Scientists (B) No. 22760267, Grant-in-Aid for Scientific Research (A) No. 23246071, and the VELUX Foundation through their VELUX Visiting Professor Programme 2011–2012.

\[\begin{array}{c}
\text{Fig. 2. Achievable regions with } \alpha_1 = 0.1, \alpha_2 = 0.3 \text{ and } P_1 = 80, P_2 = 10. \\
\end{array}\]

REFERENCES

Polymer Expansions for Cycle LDPC Codes

Nicolas Macris and Marc Vuffray
LTHC, IC, EPFL,
CH-1015 Lausanne, Switzerland
nicolas.macris@epfl.ch, marc.vuffray@epfl.ch

Abstract—We prove that the Bethe expression for the conditional input-output entropy of cycle LDPC codes on binary symmetric channels above the MAP threshold is exact in the large block length limit. The analysis relies on methods from statistical physics. The finite size corrections to the Bethe expression are expressed through a polymer expansion which is controlled thanks to expander and counting arguments.

I. INTRODUCTION

A few years ago Cherktov and Chernyak [1] devised a loop series which represents the partition function of a general vertex model as the product of the Bethe mean field expression and a residual partition function over a system of loops. In this representation all quantities are entirely expressible in terms of Belief Propagation (BP) marginals or messages. However it has not been clear so far if this representation leads to a controlled series expansions for the log-partition, in other words the free energy. If this is the case it should hopefully allow to control the difference between the true free energy and the Bethe free energy.

The loop expansion has a potential interest in coding theory since Low-Density-Parity-Check (LDPC) and Low-Density-Generator-Matrix (LDGM) codes on general binary-input memoryless symmetric (BMS) channels fit in the framework of (generalized) vertex models. In this context free energy is just another name for conditional input-output Shannon entropy. For these models it is believed that the Bethe formula for the conditional entropy/free energy is exact. However there is no general proof, except for the cases of the binary erasure channel [2], LDGM codes for high noise, and in special situations for LDPC codes at low noise [3].

We consider cycle LDPC codes for high noise (above the MAP threshold) on the binary symmetric channel (BSC) according to the framework of (generalized) vertex models. In this context free energy is just another name for conditional input-output Shannon entropy. For these models it is believed that the Bethe formula for the conditional entropy/free energy is exact. However there is no general proof, except for the cases of the binary erasure channel [2], LDGM codes for high noise, and in special situations for LDPC codes at low noise [3].

We consider cycle LDPC codes for high noise (above the MAP threshold) on the binary symmetric channel (BSC). We show that, under the assumption that there exists a fixed point for the BP equations, the average conditional entropy/free energy is given by the Bethe expression. The novelty of the approach is to turn the loop expansion into a rigorous tool allowing to derive provably convergent polymer expansions [4]. Controlling the loop expansion is a non-trivial task because in most situations of interest the number of loops proliferates. For example, this is the case (for the system of fundamental cycles) in capacity approaching codes even under MAP decoding [5].

II. LOOP AND POLYMER REPRESENTATIONS

Let \( \Gamma = (V, E) \) be a graph with vertices \( a \in V \) of regular degree \( d \) and edges \( ab \in E \). The symbol \( \partial a \) denotes the set of \( d \) neighbors of \( a \). In vertex models the degrees of freedom are spins \( \sigma_{ab} \in \{-1, +1\} \) attached to each edge. At each function node \( a \in V \) we attach a non-negative function \( f_a(\sigma_{ba}) \) depending only on neighboring variables \( \sigma_{ba} \equiv (\sigma_{ab})_{b \in \partial a} \). We study probability distributions which can be factorized as

\[
\eta_a(\sigma_{ba}) = f_a(\sigma_{ba}) \quad \text{and} \quad Z_\eta = \sum_{\sigma_{ba}} f_a(\sigma_{ba}).
\]

and their associated free energy \( f_a \). For each edge \( ab \in E \) we introduce two directed “messages” \( \eta_{a \to b} \) and \( \eta_{b \to a} \). For the moment these variables are arbitrary and are collectively denoted by \( \vec{\eta} \). One has the identity [1]

\[
f_n = \frac{1}{n} \ln Z_{\text{Bethe}}(\vec{\eta}) + \frac{1}{n} \ln Z_{\text{corr}}(\vec{\eta}).
\]

The first term is the Bethe free energy functional,

\[
\ln Z_{\text{Bethe}}(\vec{\eta}) = \sum_{a \in V} \ln \left( \sum_{\sigma_{ba}} f_a(\sigma_{ba}) \prod_{b \in \partial a} e^{\eta_{b \to a} - \sigma_{ab}} \right)
\]

(3)

The “partition function” in the second term can be expressed as a sum over all subgraphs (not necessarily connected) \( g \subset \Gamma \)

\[
Z_{\text{corr}}(\vec{\eta}) = \sum_{g \subset \Gamma} K(g)
\]

(4)

and \( K(g) = \prod_{a \in g} K_a \) with

\[
K_a = \sum_{\sigma_{ba}} p_a(\sigma_{ba}) \prod_{b \in \partial a \setminus g} \sigma_{ab} e^{-\sigma_{ab}(\eta_{a \to b} + \eta_{b \to a})}
\]

\[
p_a(\sigma_{ba}) = \frac{f_a(\sigma_{ba}) \prod_{b \in \partial a \setminus g} e^{\eta_{b \to a} - \sigma_{ab}}}{\sum_{\sigma_{ba}} f_a(\sigma_{ba}) \prod_{b \in \partial a \setminus g} e^{\eta_{b \to a} - \sigma_{ab}}}.
\]

It is well known that the stationary points of (3) satisfy the BP fixed point equations,

\[
\eta_{a \to b} = \frac{\sum_{\sigma_{ba}} \sigma_{ab} f_a(\sigma_{ba}) \prod_{c \in \partial a \setminus b} e^{\eta_{c \to b} - \sigma_{ac}}}{\sum_{\sigma_{ba}} f_a(\sigma_{ba}) \prod_{c \in \partial a \setminus b} e^{\eta_{c \to b} - \sigma_{ab}}}.
\]

(5)

Remarkably, for any solution of (5), \( K(g) = 0 \) if \( g \) contains a degree one node. Thus if \( \vec{\eta} \) is a fixed point of the BP equations then \( Z_{\text{corr}}(\vec{\eta}) \) is given by the sum in (4) over \( g \subset \Gamma \) with no degree one nodes. Such graphs are called loops (see figure 1).

One can recognize that \( Z_{\text{corr}} \) can be interpreted as the partition function of a system of polymers. A loop \( g \subset \Gamma \) can be decomposed into its connected parts in a unique way as
illustrated on figure 1. Connected loops are called polymers and are generically denoted by the letter \( \gamma \). The important point is that by definition the polymers do not intersect. For each polymer \( \gamma \) we define a weight (also called activity), \( K(\gamma) = \prod_{a \in \gamma} K_a \). Let \( g = \bigcup_{i=1}^M \gamma_i \). Since the \( \gamma_i \) are disjoint, \( K_g = \prod_{i=1}^M K(\gamma_i) \). Thus equation (4) can be cast in the form

\[
Z_{\text{corr}}(\tilde{\eta}) = \frac{1}{n!} \sum_{M=0}^n \sum_{\gamma_1, \ldots, \gamma_M} \prod_{i<j} \prod_{\gamma_i \cap \gamma_j = \emptyset} K(\gamma_i) \prod_{\gamma_i \in \gamma} (\gamma_i \cap \gamma_j = \emptyset).
\]

In this sum each \( \gamma_i \) runs over all connected subgraphs with no dangling edges of the underlying graph \( \Gamma \). The sum over the number of polymers \( M \) has a finite number of terms because the polymers cannot intersect.

In the next paragraphs \( \Gamma \) is a random \( d \)-regular graph. We denote by \( P \) and \( E \) the relevant probability and expectation over this ensemble.

### III. POLYMER EXPANSION

We wish to compute the correction to the Bethe free energy in (2), namely \( f_{\text{corr}}(\tilde{\eta}) = \frac{1}{n} \ln Z_{\text{corr}}(\tilde{\eta}) \) when \( \tilde{\eta} \) is a BP fixed point. Using (6) the logarithm can be expanded as a power series in \( K(\gamma_i) \)'s. This yields the polymer (or Mayer) expansion [4]

\[
f_{\text{corr}}(\tilde{\eta}) = \frac{1}{n} \sum_{M=1}^\infty \frac{1}{M!} \sum_{\gamma_1, \ldots, \gamma_M} \prod_{i=1}^M K(\gamma_i) \times \sum_{G \subseteq \gamma} \prod_{(i,j) \in G} \{ -1 \} (\gamma_i \cap \gamma_j = \emptyset). \tag{7}
\]

The third sum is over the set \( G_M \) of all connected graphs with \( M \) vertices labeled by \( \gamma_1, \ldots, \gamma_M \), and at most one edge between each pair of vertices. The product of indicator functions is over edges \( (i,j) \in G \). It constrains the set of polymers \( \gamma_1, \ldots, \gamma_M \) to intersect according to the structure of \( G \). In this expansion one sums over an infinite number of terms so it is important to address the question of convergence. A criterion which ensures the convergence of the expansion uniformly in system size \( n \) (and thus ensures convergence in the infinite size limit) is

\[
\sum_{t=0}^{\infty} \frac{1}{t!} \sup_{\gamma \in V} |\gamma| |K(\gamma)| < 1 \tag{8}
\]

To illustrate the use of the polymer expansion in a simple case, consider a vertex model at high temperature defined by

\[
f_a(\sigma_{ab}) = \frac{1}{2} (1 + \tanh J_a \prod_{b \in \partial a} \sigma_{ab} - h_{ab}) \sigma_{ab}
\]

where \( J_a \) and \( h_{ab} \) are in \( \mathbb{R} \) with \( \sup_{a \in V} J_a \equiv J < < 1 \) and \( \sup_{a,b \in E} h_{ab} < h < +\infty \). For \( J \) small enough the BP (5) equations have a unique fixed point solution [6]. We call \( \tilde{\eta}_a^n \) this fixed point. The subscript \( n \) indicates (with some abuse of notation) that this fixed point depends on the finite instance, that is, the graph \( \Gamma \), and \( J_a, h_{ab} \). For the activities of the polymers, computed at the fixed point, we have the bounds \( |K(\gamma)| \leq (2J)^{|\gamma|} \). Moreover the number of polymers \( |\gamma| \geq 3 \) is (for each \( a \)) at most \( \exp (d|\gamma|) \) with \( c_d \geq 0 \) a numerical constant depending only on \( d \). Using also that the smallest polymer must have \( |\gamma| \geq 3 \), it is then easily shown that the left hand side of (8) is \( O(J^3) << 1 \). By standard methods [4] one can then estimate the sum over \( M \) in (7) term by term, which yields

\[
|f_{\text{corr}}(\tilde{\eta}_a^n)| \leq (1 + O(J^3)) \frac{1}{n} \sum_{a \in V} \sum_{\gamma \in a} (2J)^{|\gamma|} e^{|\gamma|} \] \tag{9}

### IV. CYCLE LDPC CODES OVER THE BSC

Random \( d \)-regular graphs are equivalent to the LDPC(2, \( d \)) ensemble of cycle codes. Code bits \( x_{ab} = 0 \) are attached to the edges \( ab \in E \). In the spin language bits are \( \sigma_{ab} = \pm 1 \) and the parity check constraints are \( \prod_{b \in \partial a} \sigma_{ab} = 1 \). For definiteness we assume transmission over the BSC(\( p \)), \( p \in [0, \frac{1}{2}] \). Without loss of generality one can assume that the transmitted word is \( (1, \ldots, 1) \) so that MAP decoding is based on the posterior distribution (1) with

\[
f_a(\sigma_{ab}) = \frac{1}{2} (1 + \prod_{b \in \partial a} \sigma_{ab}) \prod_{b \in \partial a} e^{\frac{1}{2} h_{ab} \sigma_{ab}} \tag{10}
\]

Here \( h_{ab} \) is the half-log-likelihood for the bit \( \sigma_{ab} = \pm 1 \), based on the channel output. The Shannon conditional input-output entropy and free energy are essentially equivalent, and related by the simple formula,

\[
\frac{1}{n} H(\tilde{X}|\tilde{Y}) = E_{\tilde{h}}[f_a(\tilde{h})] - \frac{1 - 2p}{2} \ln \frac{1 - p}{p} \tag{11}
\]
where $E_h$ is the average over channel outputs (or the log-likelihood vector).

We interested in the high noise regime where $p$ is close to $1/2$. Therefore we seek solutions of the BP equations such that $\sup_{\omega \in E} |h[\omega]| \leq h$ where $h > 0$ is a fixed small number. We assume that for $h$ small enough there exists a fixed point of the BP equations for each finite instance. We denote it $\vec{\eta}_n$ as before. Note that assuming its unicity is not needed.

**Proposition 4.1:** Assuming the existence of a fixed point $\vec{\eta}_n$ of the BP equations for $h$ small enough, we have

$$\lim_{n \to +\infty} \mathbb{E} \left[ \frac{1}{n} \ln Z_{\text{corr}}(\vec{\eta}_n) \right] = 0.$$  

In view of (2), (11) the proposition implies that the average conditional entropy is given by the average of the Bethe expression computed at the fixed point (for $|p - \frac{1}{2}| < \epsilon$).

In order to prove the proposition we will use the identity

$$\ln Z_{\text{corr}}(\vec{\eta}) = \ln Z_p(\vec{\eta}) + \left\{ 1 + \sum_{\gamma \subset \Gamma} |\gamma| \frac{Z_p(\vec{\eta})}{Z_p(\vec{\eta}_{\gamma})} \right\}$$  

where

$$Z_p(\vec{\eta}_{\gamma} | \gamma) = \frac{1}{M!} \sum_{\gamma_1, ..., \gamma_M} \prod_{i=1}^{M} K(\gamma_i) \prod_{i<j} (\gamma_i \cap \gamma_j)$$  

and $Z_p(\vec{\eta}) \equiv Z_p(\vec{\eta} | \emptyset)$. This identity is derived by splitting the sum over $\gamma_1, ..., \gamma_M$ in (6), into a sum where all polymers are small ($\forall i, |\gamma_i| < n/2$), and a sum where there exists at least one large polymer ($\exists i, |\gamma_i| \geq n/2$); and by noting that when there exists a large polymer it has to be unique.

We will need three lemmas.

**Lemma 4.2:** For $h$ small enough we have

$$\lim_{n \to +\infty} \mathbb{E} \left[ \frac{1}{n} \ln Z_p(\vec{\eta}_n) \right] = 0.$$  

**Sketch of Proof:** It is possible to estimate the activities computed at the fixed point,

$$|K(\gamma)| \leq (1 - \alpha d B^2)^{n_1(\gamma)} \prod_{i=2}^{d-1} (\alpha_i h^{d-i})^{n_i(\gamma)}.$$  

Here $0 < \alpha_i < 1$, and $\alpha_i > 1$, $i = 2, ..., d - 1$ are fixed numerical constants (that we can take close to 1). The $n_i(\gamma)$ denotes the number of nodes of degree $i$ in the polymer $\gamma$.

Estimate (14) is essentially optimal for $h$ small, as can be checked by Taylor expanding $K(\gamma)$ in powers of $h_{\text{inh}}$. Hard constraints manifest themselves in the factor $(1 - \alpha d B^2)^{n_1(\gamma)}$ which is not small enough to compensate the entropic term $e^{\epsilon d |\gamma|}$ in the convergence criterion. However for polymers of size $|\gamma| \leq \frac{n}{2}$ we can use expander arguments to circumvent this problem. Let $e(\gamma)$ the set of edges in $E$ connecting $g$ to $\Gamma \setminus g$. We say that $\Gamma$ is an expander if for all $g \subset \Gamma$ with $|g| \leq \frac{n}{2}$ we have $|e(\gamma)| \geq \kappa |g|$. For all $d \geq 3$ [8]

$$\mathbb{P}[\Gamma \text{ is an expander with } \kappa = 0.18 d] = 1 - o_n(1).$$  

Now note that for polymers $e(\gamma) \leq \sum_{i=2}^{d-1} (d-i)n_i(\gamma) \leq d \sum_{i=2}^{d-1} n_i(\gamma)$. Therefore we deduce thanks to (15) that with high probability $\sum_{i=2}^{d-1} n_i(\gamma) \geq 0.18 |\gamma|$ and $K(\gamma) \leq (2\mathbb{E}[e(d)])^{0.18 |\gamma|}$ for $|\gamma| \leq \frac{n}{2}$. This is sufficient to control the convergence criterion, and achieve the proof of this lemma by methods similarly to the high temperature case. $
$

**Lemma 4.3:** Fix $\epsilon > 0$. Then

$$\mathbb{P}[|\gamma| \subset \Gamma : e^{-2n \epsilon} \leq Z_p(\vec{\eta} | \gamma) \leq e^{2n \epsilon}] \geq 1 - \frac{1}{\epsilon} o_n(1).$$  

The proof uses rather trivial bounds on the partition functions. We omit the details.

**Lemma 4.4:** Fix $\delta > 0$. There exists a numerical constant $C > 0$ and such that for $h$ small enough

$$\mathbb{P} \left[ \sum_{g \subset \Gamma} |K(\gamma)| \geq \delta \right] \leq C \epsilon^{-\alpha d B^2 h^2}.$$  

This inequality is a fortiori valid for $g$ replaced by $\gamma$’s in the sum.

**Sketch of Proof:** We denote by $K_n$ the complete graph with $n$ vertices. By Markov’s inequality,

$$\mathbb{P} \left[ \sum_{g \subset \Gamma} |K(\gamma)| \geq \delta \right] \leq \frac{1}{\delta} \sum_{g \subset \Gamma} \mathbb{E}[|K(\gamma)| | g \subset \Gamma] \leq \frac{1}{\delta} \sum_{g \subset \Gamma} \left(1 - \alpha d h^{2}(\gamma)^n(g) \prod_{i=2}^{d-1} (\alpha_i h^{d-i})^{n_i(g)} \right) \mathbb{P}[g \subset \Gamma].$$  

Consider graphs $g$ with $n_i(g), i = 2, ..., d$ fixed. Mackay [9] provides a bound for the probability $\mathbb{P}[g \subset \Gamma]$ of finding a particular subgraph into a regular graph $\Gamma$. Namely for $\frac{1}{2} \sum_{i=2}^{d} n_i(g) + 2d^2 \leq \frac{n d}{2}$,

$$\mathbb{P}[g \subset \Gamma] \leq \frac{1! \sum_{i=2}^{d} n_i(g)}{\frac{n}{2}^d \sum_{i=2}^{d} n_i(g)} \prod_{i=2}^{d} (\alpha_i h^{d-i})^{n_i(g)}.$$  

The number of subgraphs of $K_n$ with given $n_i(g)$ is

$$\frac{n^d}{n!} \prod_{i=2}^{d} (\alpha_i h^{d-i})^{n_i(g)}.$$  

Replacing (18) in (17), using (19), setting $x_i = \frac{n_i(g)}{n}$, and performing an asymptotic calculation for $n$ large, we show (here $\vec{x} \equiv (x_2, ..., x_d)$ and $\Delta \equiv \{\vec{x} | \frac{1}{2} \sum_{i=2}^{d} n_i(g) + 2d^2 \leq \frac{n d}{2}\}$)

$$\mathbb{P} \left[ \sum_{g \subset \Gamma} |K(\gamma)| \geq \delta \right] \leq \frac{1}{\delta} \int_{\Delta} |d \vec{x} g_n(\vec{x}) \exp[n f_0(\vec{x}) + x_1 \ln(1 - \alpha d h^2) + \sum_{i=2}^{d-1} x_i \ln(\alpha_i h^{d-i})]|.$$  

$^1$This assumption can be relaxed by softening the hard constraint in (10) and using existence results [7]. Indeed all our estimates are uniform in the softening parameter. We omit this discussion here due to lack of space.
The large \( n \) behavior of the integral asymptotic is controlled by \( f_n(\vec{x}) \), and \( g_n(\vec{x}) \) gives sub-dominant contributions that do not concern us here. We have:

\[
f_n(\vec{x}) = \frac{1}{2} \sum_{i=2}^{d} ix_i \ln \left( \frac{1}{2} \sum_{i=2}^{d} ix_i - \sum_{i=2}^{d} x_i \ln x_i \right)
\]

\[
(1 - \frac{r}{n}) \ln(1 - \frac{r}{n}) - \ln(\frac{r}{n}) - \sum_{i=2}^{d} \frac{x_i}{2} - \frac{2r^2}{n}
\]

For \( h \) small enough, in the domain \( \Delta \), the exponent in (20) is strictly negative and attains its maximum at the corner point \( x_2 = \cdots = x_{d-1} = 0, x_d = 1 \). At this point it is equal to \( \ln(1 - \alpha_d \frac{d}{n} h^2) \) which allows to conclude (16).

We are now in a position to prove proposition 4.1.

**Proof of proposition 4.1:** In view of (12), we must show that for \( h \) small enough,

\[
\frac{1}{n} \ln \left( 1 + \sum_{\gamma \subset \Gamma} \frac{Z_p(\gamma)}{Z_p}\right) = o_n(1). \tag{22}
\]

Call \( I_\epsilon \) the event

\[
\{ |\gamma| \geq n/2, \sum_{\gamma \subset \Gamma} |K(\gamma)| \frac{Z_p(\gamma)}{Z_p} < \epsilon \}
\]

where \( \epsilon \) is a positive constant that will be adjusted later on. We split the expectation in two terms \( A + B \) by conditioning over \( I_\epsilon \) and its complement \( I_\epsilon' \), and estimate each contribution.

For the first contribution, using \( |\ln(1 + x)| \leq |\ln(1 - |x|)| \) for \( |x| < 1 \), \( A \leq \frac{1}{n} \ln(1 - \epsilon) \). For the second contribution we have to estimate \( \mathbb{P}[I_\epsilon'] \). The events \( \{ |\gamma| \subset \Gamma : e^{-2\epsilon n} \leq \frac{Z_p(\gamma)}{Z_p(\Gamma)} \leq e^{2\epsilon n} \} \) imply \( I_{\delta_2n} \). Therefore \( I_{\delta_2n} \) implies the union of the complementary events, so that for the union bound together with lemmas 4.3 and 4.4,

\[
\mathbb{P}[I_{\delta_2n}] \leq C \frac{e^{-\alpha_n \delta^2 h^2}}{\delta} + \frac{1}{n} \epsilon o_n(1).
\]

Now suppose for a moment that there exist a positive constant independent of \( n \) such that

\[
\frac{1}{n} \ln \left( 1 + \sum_{\gamma \subset \Gamma} K(\gamma) \frac{Z_p(\gamma)}{Z_p}\right) \leq C'
\]

Then \( B \leq C'' \mathbb{P}[I_\epsilon'] \). Setting \( \epsilon = \delta e^{2\alpha_n} \), the above arguments imply

\[
E \left[ \frac{1}{n} \ln \left( 1 + \sum_{\gamma \subset \Gamma} K(\gamma) \frac{Z_p(\gamma)}{Z_p}\right) \right] \equiv A + B
\]

\[
\leq \frac{1}{n} \ln(1 - \delta e^{2\alpha_n} + C \frac{e^{-\alpha_n \delta^2 h^2}}{\delta} + \frac{1}{n} \epsilon o_n(1).
\]

We are free to choose \( \delta = e^{-\alpha_n \frac{d}{n} h^2} \) and \( \epsilon = \alpha_n \frac{d}{n} h^2 \) (lemmas 4.3, 4.4 hold) and this choice \( A + B = o_n(1) \), which proves (22).

It remains to justify (23). From the convergence of the polymer expansion we deduce that \( \frac{1}{n} \ln Z_p(\gamma_n) \) is bounded uniformly in \( n \). From (1), (10) we easily show that \( \frac{1}{n} \ln Z_p \leq \ln 2 + \frac{3}{2} h \). In the high noise regime the BP messages are bounded so that from (3) we deduce that \( \frac{1}{n} \ln Z_p(\gamma_n) \) is bounded by a constant independent of \( n \). Finally the triangle inequality implies that \( \frac{1}{n} \ln Z_p - \ln \mathbb{P}[I_\epsilon] \ln Z_p(\gamma_n) \) is bounded uniformly in \( n \). This is precisely the statement (23).

\[\text{V. Conclusion}\]

The approach is quite general and can hopefully be generalized to standard irregular LDPC codes with bounded degrees and binary-input memoryless output-symmetric channels with bounded log-likelihood variables. This will be the subject of future work.

\[\text{Acknowledgment}\]

M.V acknowledges supported by the Swiss National Foundation for Scientific Research, Grant no 2000-121903. N.M benefited from discussions with M. Chertkov and R. Urbanke.

\[\text{References}\]

Factoring linear trellises

David Conti
Claude Shannon Institute and
School of Mathematical Sciences
University College Dublin, Ireland

Nigel Boston
Departments of Mathematics and ECE
University of Wisconsin, USA

Abstract—Koetter/Vardy proved in [9] the remarkable result that any linear trellis can be factored into elementary factors. We prove that the spans (and their multiplicities) of the elementary factors of a linear trellis are uniquely determined, extending a known result for minimal trellises ([4], [8], [11]). In so doing, we give a graphical method to compute all the spans and their multiplicities. We show also how to determine all the possible edge-labelings of the factors of a linear trellis, and give exact conditions for their uniqueness. These results can help to compute and classify all trellises for a fixed code, once the labels play no role. We then show how to find all the possible factorizations of a linear trellis, once the code represented by the trellis is linear and that edge-labelings then unique factorization holds, which means that any linear trellis can be factored into elementary trellises. We will always mean a linear trellis.

I. INTRODUCTION

Linear trellises are amongst the most important graph representations of linear codes, both for theoretical and practical reasons. Towards the end of last millennium, due to the observation of striking low complexity achieved by nonconventional trellises ([11]), the interest expanded to the whole class of tail-biting trellises (of which conventional trellises can be seen as a particular case). As a consequence, in the recent years, linear tail-biting trellises has been the subject of major research (as a particular case). As a result, in the recent years, linear tail-biting trellises has been the subject of major research. In this paper we investigate further linear trellis factorizations and give graphical/combinatorial methods to compute such factorizations. We first prove that if we forget about edge-labels then unique factorization holds, which means that all the spans of factors and their multiplicities are uniquely determined. Our proof yields at the same time a method for computing all the spans and multiplicities. This result was proven before for minimal trellises ([11]), and the striking fact is that this holds true for any linear trellis and that edge-labels play no role. We then show how to find all the possible labelings of the elementary factors of a linear trellis, once the spans’ multiplicities are known. As a corollary, we get precise conditions under which such labelings are uniquely determined. Finally, we mention how these results help us to compute and classify all trellises representing a fixed code, which is important for the search of trellis representations that perform well under iterative or LP decoding.

A. Basics on Trellises

All trellises considered will be over a fixed finite field \( \mathbb{F} \equiv \mathbb{F}_q \) and have a fixed length \( n \). Also, we want to do modular arithmetic on the coordinate indices of \( \mathbb{F}^n \), so \( \mathbb{Z}_n \equiv \mathbb{Z}/n\mathbb{Z} \) will be the index set of coordinates. A trellis (over \( \mathbb{F} \), and of length \( n \)) is a directed graph \( T = (V,E) \) together with a partition \( V = \bigcup_{i\in\mathbb{Z}_n} V_i \) indexed over \( \mathbb{Z}_n \), such that the edges are \( \mathbb{F} \)-labeled and can only go from \( V_i \) to \( V_{i+1} \), so that \( E \) has also a partition \( E = \bigcup_i V_i \times \mathbb{F} \times V_{i+1} \). For formal reasons we also allow multiple edges in a trellis, i.e. each edge \( e \in E \) also has a multiplicity \( m(e) \geq 1 \). A trellis is conventional if \( |V_0| = 1 \).

Two trellises \( T = (V,E), T' = (V',E') \) are isomorphic if there exists a bijection \( f : V \to V' \) such that \( e = (v,a,w) \in E \) if and only if \( f(e) := (f(v),a,f(w)) \in E' \), \( m(e) = m(f(e)) \) for all \( e \in E \), and \( f(V_0) = V'_0 \). If by forgetting their labels \( T \) and \( T' \) are isomorphic as directed multigraphs, then we say that \( T \) and \( T' \) are structurally isomorphic (again, we require \( V_0 \) to be sent to \( V'_0 \)) and write \( T \cong T' \).

By a cycle of a trellis \( T \) we mean a directed closed path in \( T \) of length \( n \) starting and ending at \( V_0 \). The subset \( C(T) := \{\text{edge-label sequences of cycles of } T \} \subseteq \mathbb{F}^n \) is the code represented by \( T \). A trellis is reduced if every edge and every vertex belongs to a cycle. A trellis is linear if it is reduced, the multiplicity function \( m : E \to \mathbb{N} \) on each \( E_i \) is a constant of the form \( q^i \) for some \( q \geq 1 \), and all \( V_i \)'s have an \( \mathbb{F} \)-vector space structure such that \( E_i \) is an \( \mathbb{F} \)-subspace of \( V_i \times \mathbb{F} \times V_{i+1} \) for all \( i \). If \( T \) and \( T' \) are isomorphic then \( T \) is linear if and only if \( T' \) is linear. Also, if \( T \) is linear then so is \( C(T) \). In this paper we will consider only linear trellises, so, to simplify the terminology, from now on by a trellis we will always mean a linear trellis.

We will denote by \( \mathbb{T} \) the set of trellises up to isomorphism, and by \( \mathbb{T} \) the set of unlabeled trellises, i.e. trellises up to structural isomorphism. Clearly \( \mathbb{T} \) can be identified with the subset of all trellises having all edge-labels equal to 0, so, we see \( \mathbb{T} \subseteq \mathbb{T} \). We have a projection map \( T \to \mathbb{T} \), which to each trellis \( T \) associates its underlying unlabeled trellis \( \mathbb{T} \) (equivalently, putting all edge-labels of \( T \) equal to 0).

For \( 0 \leq h \leq n \) and \( i \in \mathbb{Z}_n \), \( \{i,i+h\} := \{i+1,i+2,\ldots,i+h\} \subseteq \mathbb{Z}_n \) is a span of length \( h \) of \( v \in \mathbb{F}^n \) if \( \text{supp}(v) \subseteq [i,i+h] \). If \( h < n \) we say that the span \( \{i,i+h\} \) is proper, and that it starts at \( i \) and ends at \( j = i+h \).
The degenerate span \( Z_n \) is a span of any \( v \in F^n \). Now, let \( s \) be a span of \( v \in F^n \). Let \( V_t := F \) for all \( t \in s \) and else \( V_t := 0 \). Choose \( w_t \in V_t \setminus \{0\} \) for all \( t \in s \), while for all \( t \notin s \) put \( w_t := 0 \). Let then \( E_t := F(w_t, v_t, w_{t+1}) \), i.e. \( E_t \) is the subspace of \( V_t \times V_{t+1} \) generated by the vector \((w_t, v_t, w_{t+1})\). We also put \( m(e) := 1 \) for all \( e \in E \), except in the case \( v[s] = 0 \), where we put \( m(e) = q \) for the (only) edge \( e = (0,0,0) \in E_t \) (i.e. \( e \) is repeated \( q \) times). This gives the elementary trellis \( e[i] \) (up to trellis isomorphism, it doesn’t depend on the choice of the \( v_t \)’s). It is clearly a (linear) trellis, and it satisfies \( v[s] = \lambda v[s] \) for \( \lambda \neq 0 \), \( C(v[s]) = Fv \). Notice also that \( v[s] = v'[s'] \) if and only if \( s = s' \), and that since \( F \) is finite, the set \( T_{EI} \) of elementary trellises is finite too.

### B. Trellis products

\( T \) benefits of a natural monoid structure \((T, \otimes)\) given as follows. Let \( T, T' \in T \) be two trellises. For \( e = (v_i, a, v_{i+1}) \in E_i, e' = (v'_i, a', v'_{i+1}) \in E'_i \) define \( e \otimes e' := ((v_i, v'_i), a + a', (v_{i+1}, v'_{i+1})) \). For each \( i \) let \( V^{(i)} := V_i \times V'_i \) and let \( E^{(i)} := \{e \otimes e' | e \in E_i, e' \in E'_i\} \). Then the trellis \( T \otimes T' := (\cup V^{(i)}, \cup E^{(i)}) \) is called the trellis product of \( T \) and \( T' \). This is the usual definition of trellis product as given for example in [8], [9]. Since we also allow multiple edges in a trellis, we naturally define the multiplicity of \( e'' \in E'' \) as \( m(e'') := \sum_{e \in E''} m(e)m(e') \). It’s easy to see that if \( T \) is a product of elementary trellises then \( T \) has no multiple edges if and only if no factor of span length 0 is repeated and no factor is of the form \( 0[i, i] \).

The operation \( \otimes \) is associative, commutative, and has an identity, the zero trellis \( \emptyset \to \emptyset \to \emptyset \to \emptyset \), i.e. \((T, \otimes)\) is a commutative monoid. Note that the only invertible element of \((T, \otimes)\) is the zero trellis, that the subset \( T \) is a submonoid of \((T, \otimes)\), and \( T \otimes T' = T \otimes T' \).

Also, a fundamental property of the trellis product is the identity \( C(T \otimes T') = C(T) + C(T') \), from which follows that any linear code can be represented by a product of elementary trellises. That any trellis is a product of elementary trellises is the following famous factorization theorem (see [9]):

**Theorem 1 (KV).** Any (linear) trellis is a product of elementary trellises.

### II. Unique factorization in \( T \)

Knowing that any trellis can be factored into elementary trellises, a natural following question is whether the factorization must be unique or not. The answer is that in general it does not. For example, \( 111(0,2) \otimes 010(1,1) = 101|(0,2) \otimes 010(1,1) \), but the two factorizations are different, as \( 111(0,2) \neq 101(0,2) \). Nevertheless, one sees here that the two factorizations have precisely the same set of spans, i.e. up to labels the two factorizations are equal. This is no coincidence, as it is a well known result that different factorizations of the minimal conventional trellis for a linear code must have the same set of spans (for details on minimal and conventional trellises see [11], [8]), a result that is proved only as a byproduct of the minimality assumption. More recently the same result was also proven for minimal nonconventional trellises (see [8], [4]), again by exploiting the minimality assumption. We show that this holds actually for any trellis, and that the edge-labels really play no role in determining the spans of factors. The structure alone determines the spans. Here is our main result of this section:

**Theorem 2.** In \((\overline{T}, \otimes)\) unique factorization holds, that is, if \( T_1 \otimes T_2 \otimes \ldots \otimes T_r = T'_1 \otimes T'_2 \otimes \ldots \otimes T'_r \), with \( T_i, T'_i \in T_{EI} \), then \( r = r' \) and, up to order, \( T_i = T'_i \) for all \( i \). In other words, the spans (along with their multiplicities) appearing in a factorization of a trellis \( T \) are uniquely determined.

It is easy to see that the multiplicity of the degenerate span \( Z_n \) is uniquely determined and is equal to the number of connected components of \( T \), so, the real problem is understanding what happens with proper spans. In what follows we will tackle this problem by a graph-theoretical approach, which yields also a method to compute the structural factorization. Before going on to prove Theorem 2, we also inform the reader that we were recently told by Heide Gluesing-Luerssen that her PhD student Elizabeth Weaver has independently obtained uniqueness of spans of factors in the particular case of one-to-one trellises.

### A. Graph-theoretical approach

The graph-theoretical approach involves studying the earliest intersections of paths starting along different edges from a fixed vertex. Not only does this enable to prove unique factorization in \((\overline{T}, \otimes)\), but from the same intersection data one is also able to determine precisely all the spans of the factors.

Now, let us give some notation. Given a multiset \( S \), we write \( m(x, S) \) for the multiplicity of \( x \in S \) (in particular \( m(x, S) > 0 \) if and only if \( x \in S \)). If \( T \) is a trellis and \( e = (v, a, v') \) is an edge of \( T \), then \( h(e) := v' \) is the head of \( e \), and \( t(e) := v \) is its tail. A path \( p \equiv p_0|p_1| \ldots |p_r \) in \( T \) is a sequence of edges \( p_0, p_1, \ldots, p_r \) in \( T \) such that \( h(p_i) = t(p_{i+1}) \), \( i = 0, \ldots, r-1 \).

Now, let \( e \neq e' \) be two different edges of \( T \) such that \( t(e) = t(e') \). Define \( l(e, e') \) to be the smallest \( r \geq 0 \) such there exist \( r \) directed paths \( p = p_0|p_1| \ldots |p_r \) in \( T \) satisfying \( p_0 = e, p_0' = e' \), and \( h(p_r) = h(p_r') \). If there exist no directed paths satisfying those conditions then we put \( l(e,e') = \infty \). Define then the multiset \( I(e) := \{(l(e,e'|e' \neq e, t(e') = t(e)) \} \).

**Example 1.** Consider the unlabeled trellis

![Trellis](image)

Then \( I(e) = \{1\}, I(e') = \{2\}, \) and \( I(e'') = \{1, 1\} \).

Now, let \( T \) be a trellis and let \( e, e' \) be two edges with same tail. A priori, to compute \( I(e, e') \) one needs to compute the set of all possible coupled of paths \( p, p' \) starting along

---

84
Observation 1. Let $T$ be a trellis and let $e, e'$ be two edges such that $l(e) = l(e')$. Fix a path $p$ such that $p_0 = e$. Then $l(e, e') = \min \{ r \mid \exists p' = p_0 \cdots p_r \text{ such that } p'_0 = e', h(p'_r) = h(p_0) \}$. 

The highly symmetrical structure of (linear) trellises is further reflected in the following fundamental lemma:

**Lemma 1.** Let $T$ be a trellis of length $n$. Then:
- $l(e, e') \leq n - 1$ for any edges $e, e'$
- $I(I(e)) = I(e')$ if $l(e), l(e') \in V_i$ for some $i$

In sight of this result, for a trellis $T$ and $i \in \mathbb{Z}_n$ it is legitimate to define $I_i(T) := I(e)$, where $e$ is any edge such that $l(e) = V_i$. The next lemma tells us that $I_i(T)$ is determined only by those factors of $T$ whose spans start at $i$.

**Lemma 2.** Let $T$ be a trellis, and let $T' \in \mathbb{T}_{E(T)}$ be an elementary trellis with span not starting at $i$ or degenerate span. Then $I_i(T \otimes T') = I_i(T)$.

The above lemma is crucial for the following theorem, which is the key result:

**Theorem 3.** Let $T = T_1 \times \ldots \times T_n$, with $T_1, \ldots, T_n \in \mathbb{T}_{E(T)}$. Fix $i \in \mathbb{Z}_n$, and let $L_i(T)$ be the multiset of span lengths of those $T_j$'s with (proper) span starting at $i$. Then for $l = 0, \ldots, n - 1$

$$
\log_p \left( 1 + \sum_{j=0}^{l} m(j, L_i(T)) \right) = \sum_{j=0}^{l} m(0, L_i(T))
$$

The above equations can be recursively solved for $m(j, L_i(T))$, $j = 0, \ldots, n - 1$, and the solution is unique. Therefore, since the left-hand side depends only on $T$, we conclude that Theorem 2 holds. Moreover, by computing all $I_i(T)$'s we are able to determine all the (proper) spans and their multiplicities in any factorization of $T$.

**Example 2.** Let $T$ be the unlabeled connected trellis

\[\text{This trellis is } \mathbb{F}_2\text{-linear, its linear structure given by the shown } \mathbb{F}_2\text{-labels of vertices. Then one easily computes:}
- $I_0(T) = I(e_0) = \emptyset$
- $I_1(T) = I(e_1) = \{2\}$
- $I_2(T) = I(e_2) = \emptyset$
- $I_3(T) = I(e_3) = \{3, 4, 5\}$
- $I_4(T) = I(e_4) = 0$

So, by Theorem 3, we get that $L_0(T) = L_2(T) = L_4(T) = \emptyset$, $L_1(T) = \{\{3\}\}$, $L_3(T) = \{\{4, 5\}\}$. Therefore $T = \{1, 3\} \otimes \{3, 1\} \otimes \{3, 2\}$, as one can check.

**III. Factorization in $\mathbb{T}$**

We have already pointed out that in $(\mathbb{T}, \otimes)$ unique factorization doesn’t hold, since the labels are not necessarily uniquely determined. Nevertheless, we give a method to find a legitimate labeling of a structural factorization of a trellis. We will also deduce precise conditions for a trellis $T$ under which unique factorization in $(\mathbb{T}, \otimes)$ holds.

Let $T \in \mathbb{T}$, and let $S(T)$ be the span multiset of $T$, which by Theorem 2 is uniquely determined. If $(i, j) \in S(T)$, i.e. $m((i, j), S(T)) \geq 1$, we say that $(i, j)$ is a span of $T$. Also, we put $s := [i, j]$ for a proper span $s = (i, j)$, and $Z_n = \mathbb{Z}_n$.

By linearity there exists a zero vertex $z_i \in V_i$ for all $i$. Let $(i, j)$ be a proper span of $T$, and let $C_{(i,j)}(T)$ denote the code of all label-sequences of paths starting at $z_i$ and ending at $z_j$. This is a code of length equal to the span length of $(i, j)$, but it is naturally a code in $\mathbb{F}_n$ by setting the coordinates outside the span $(i, j)$ equal to 0. It is not hard to see that it is linear, since $T$ is so. Also, for the degenerate span $Z_n$ we put $C_{Z_n}(T) := C(T)$. Then we have the following:

**Theorem 4.** Let $T \in \mathbb{T}$, and let $s$ be a span of $T$ of multiplicity $r = m(s, S(T))$. Then $v_1 | s \otimes \ldots \otimes v_r | s$ divides $T$ if and only if

$$
\mathbb{F}_{v_1} + \ldots + \mathbb{F}_{v_r} + \sum_{\nu \in S(T) | \nu \subset s} C_{\nu}(T) = C_s(T)
$$

The following corollary extends the well known result that if there are no containments between different atomic spans of a code then the associated minimal conventional trellis factorizes uniquely, since all atomic generators are uniquely determined (up to scalar multiplication).

**Corollary 1.** Let $T = v_1 | (i_1, j_1) \otimes \ldots \otimes v_r | (i_r, j_r)$. Then $v_1, \ldots, v_r$ are uniquely determined up to scalar multiplication and ordering (i.e. the factorization of $T$ is unique) if and only if there are no containments between different spans of $T$ and all their multiplicities are equal to 1.

**Example 3.** Label the edges of $T$ from example 2 as follows (by using dashed lines for 0’s and full lines for 1’s):

Thus labeled, $T$ is again $\mathbb{F}_2$-linear. Then one gets:
- $C_{(1,3)}(T) = \{00000, 01010\}$
\[ C(3,1)(T) = \{00000, 01011\} \]
\[ C(3,2)(T) = \{00000, 01011\} \]

Therefore by Theorem 4 and its corollary, \( T \) has two (and only two) distinct factorizations,
\[
T = 01010[(1, 3) \otimes 01011](3, 1) \otimes 01011)(3, 2) \\
T = 01010[(1, 3) \otimes 01011](3, 1) \otimes 00000)(3, 2)
\]

Note also that \( T' = 01010[(1, 3) \otimes 00000](3, 1) \otimes 01111)(3, 2) \) (depicted below) is a different trellis from \( T \), according to Theorem 4. In fact, in \( T' \) it is possible to go out from \( 0 \in V_3 \) along two paths with all edge-labels equal to 0, and that meet at \( V_2 \), while this is not possible in \( T \).

IV. APPLICATIONS

The complexity of trellis-based decoding is directly proportional to the size of the chosen representing trellis, so, to achieve low complexity it is important to search for the smallest trellis representations of codes. If trellis size is all what one is interested in, then obviously the way labels are arranged on a trellis doesn’t matter (as long as the trellis represents the wanted code), which implies that it is sufficient to classify all the trellis representations up to structural isomorphism. In [8] Koetter and Vardy show that up to structural isomorphism all the minimal trellises for a given code are encoded into a single matrix (characteristic matrix), and show how to efficiently compute that matrix (see also [4]).

On the other hand, the performance of iterative or LP decoding algorithms on a trellis is affected by so-called pseudocodewords, which makes it important to sort trellises also with respect to their pseudocodewords, as one wants to find trellises that yield few bad pseudocodewords ([3], [6]). In [2] we have given an example which shows that by changing labels in a trellis it is possible to change the set of yielded pseudocodewords and yet represent the same code. This means that to sort all the trellis representations of a given code with respect to pseudocodewords it is not sufficient to do a classification of trellises up to structural isomorphism. We need a complete classification of trellises. Theorem 2 and Theorem 4 help in that direction. Indeed they yield an exact method to check whether two trellises are isomorphic or not.

To see this recall that (by Theorem 1) to define a trellis we need to specify a span multiset and a list of vectors that is compatible with that span multiset. Theorem 2 tells us first of all that two different span multisets always yield two different trellises (since they are not even structurally isomorphic!). Once a span multiset (and so by Theorem 2 the unlabeled graph structure of the trellis) is chosen, Theorem 4 tells us the precise relations that hold among all the possible lists of vectors that give the same trellis. More precisely, if the chosen spans are \( s_1, \ldots, s_r \) with respective multiplicities \( m_1, \ldots, m_r \), and \( \{v_{ij}\}_{1 \leq i \leq s_1, 1 \leq j \leq m_i} \) are lists of vectors associated to the spans, then it is easy to see from Theorem 4 that the corresponding trellises are isomorphic if and only if for all \( i = 1, \ldots, r \) the equation
\[
\sum_{h \in [s_i]} \sum_{1 \leq j \leq m_h} \mathbf{F}_{hj} = \sum_{h \in [s_i]} \sum_{1 \leq j \leq m_h} \mathbf{F}_{hj}
\]
holds.

**Remark 1.** It is easy to show that if a trellis \( T \) for a given code is optimal with respect to iterative or LP decoding (in the sense that it yields few bad pseudocodewords) then it must be one-to-one, i.e. the list of generating vectors of \( T \) must be a basis of the code. So, for iterative or LP decoding purposes, it is actually sufficient to do a complete classification of one-to-one trellises.

**ACKNOWLEDGMENTS**

This work is part of the Ph.D. research of D. Conti supported by the Science Foundation Ireland grant 06/MI/006, and supervised by N. Boston.

**REFERENCES**

Information Theoretic Authentication and Secrecy Codes in the Splitting Model

Michael Huber
University of Tuebingen
Wilhelm Schickard Institute for Computer Science
Sand 13, D-72076 Tuebingen, Germany
Email: michael.huber@uni-tuebingen.de

Abstract—In the splitting model, information theoretic authentication codes allow non-deterministic encoding, that is, several messages can be used to communicate a particular plaintext. Certain applications require that the aspect of secrecy should hold simultaneously. Ogata–Kurosawa–Stinson–Saido (2004) have constructed optimal splitting authentication codes achieving perfect secrecy for the special case when the number of keys equals the number of messages. In this paper, we establish a construction method for optimal splitting authentication codes with perfect secrecy in the more general case when the number of keys may differ from the number of messages. To the best knowledge, this is the first result of this type.

I. INTRODUCTION

The development of quantum computer resistant cryptographic schemes and security technologies is of crucial importance for maintaining cryptographic long-term security and/or confidentiality of digital data, as classical cryptographic primitives such as RSA, DSA, or ECC would be easily broken by future quantum computing based attacks (e.g., [1], [2]). Application areas where cryptographic long-term security and/or confidentiality is strongly required include archiving official documents, notarial contracts, court records, medical data, state secrets, copyright protection as well as further areas concerning e-government, e-health, e-publication, et cetera.

To this end, one promising approach is the design of information theoretic authentication and secrecy systems (e.g., [3], [4]). The information theoretic, or unconditional, security model does not depend on any complexity assumptions and hence cannot be broken given unlimited computational resources. This guarantees not only resistance against future quantum computing based attacks but also perfect security in the classical world.

This paper considers authentication and secrecy codes in the splitting model. Splitting is of importance, for instance, in the context of authentication with arbitration [5] (i.e., protection against insider attacks in addition to outsider attacks). Ogata–Kurosawa–Stinson–Saido [6] have constructed optimal splitting authentication codes with perfect secrecy for the special case when the number of keys equals the number of messages. In this work, we develop a construction method for optimal splitting authentication codes with perfect secrecy in the more general case when the number of keys may differ from the number of messages. To the best knowledge, this is the first result of this type. Our simple yet powerful approach is based on the notion of cyclic splitting designs and establishes an efficient method to construct optimal splitting authentication codes with perfect secrecy.

II. THE SPLITTING MODEL

We rely on the information theoretical, or unconditional secure, authentication model developed by Simmons (e.g., [7], [8]). Our notation follows [6], [9], [10]. In this model, three participants are involved: a transmitter, a receiver, and an opponent. The transmitter wants to communicate information to the receiver via a public communications channel. The receiver in return would like to be confident that any received information actually came from the transmitter and not from some opponent (integrity of information). The transmitter and the receiver are assumed to trust each other. An authentication code is sometimes called, for short, an A-code.

Let \( S \) denote a finite set of source states (or plaintexts), \( M \) a finite set of messages (or ciphertexts), and \( E \) a finite set of encoding rules (or keys). Using an encoding rule \( e \in E \), the transmitter encrypts a source state \( s \in S \) to obtain the message \( m = e(s) \) to be sent over the channel. The encoding rule is communicated to the receiver via a secure channel prior to any messages being sent. When it is possible that more than one message can be used to communicate a particular source state \( s \in S \) under the same encoding rule \( e \in E \), then the authentication code is said to have splitting. In this case, a message \( m \in M \) is computed as \( m = e(s, r) \), where \( r \) denotes a random number chosen from some specified finite set \( R \). If we define

\[
e(s) := \{ m \in M : m = e(s, r) \text{ for some } r \in R \}
\]

for each encoding rule \( e \in E \) and each source state \( s \in S \), then splitting means that \(|e(s)| > 1\) for some \( e \in E \) and some \( s \in S \). In order to ensure that the receiver can decrypt the message being sent, it is required for any \( e \in E \) that \( e(s) \cap e(s') = \emptyset \) if \( s \neq s' \). For a given encoding rule \( e \in E \), let

\[
M(e) := \bigcup_{s \in S} e(s)
\]

denote the set of valid messages. A received message \( m \) will be accepted by the receiver as being authentic if and only if
A splitting authentication code is called \( c \)-splitting if
\[
|e(s)| = c
\]
for every encoding rule \( e \in E \) and every source state \( s \in S \). We note that an authentication code can be represented algebraically by a \(|E| \times |S|\) encoding matrix with the rows indexed by the encoding rules \( e \in E \), the columns indexed by the source states \( s \in S \), and the entries defined by \( a_{es} := e(s) \).

We address the scenario of a spoofing attack of order \( i \) (cf. [11]): Suppose that an opponent observes \( i \geq 0 \) distinct messages, which are sent through the public channel using the same encoding rule. The opponent then inserts a new message \( m' \) (being distinct from the \( i \) messages already sent), hoping to have it accepted by the receiver as authentic. The cases \( i = 0 \) and \( i = 1 \) are called impersonation game and substitution game, respectively.

For any \( i \), we assume that there is some probability distribution on the set of \( i \)-subsets of source states, so that any set of \( i \) source states has a non-zero probability of occurring. For simplification, we ignore the order in which the \( i \) source states occur, and assume that no source state occurs more than once. Given this probability distribution on the set \( S \) of source states, the receiver and transmitter also choose a probability distribution on the set \( E \) of encoding rules, called an encoding strategy. It is assumed that the opponent knows the encoding strategy being used. If splitting occurs, then the receiver/receiver will also choose a splitting strategy to determine \( m \in M(e) \), given \( s \in S \) and \( e \in E \) (this corresponds to non-deterministic encoding). The transmitter/receiver will determine these strategies to minimize the chance of being deceived by the opponent. The deception probability \( P_{di} \) denotes the probability that the opponent can deceive the transmitter/receiver with a spoofing attack of order \( i \).

### III. COMBINATORIAL SPLITTING DESIGNS

The notion of splitting balanced incomplete block designs and, more generally, that of splitting \( t \)-designs have been introduced in [6] and [10], respectively.

**Definition 1**: For positive integers \( t, v, b, c, u, \lambda \) with \( t \leq u \) and \( cu \leq v \), a \( t-(v, b, l = cu, \lambda) \) splitting design \( D \) is a pair \((X, B)\), satisfying the following properties:

(i) \( X \) is a set of \( v \) elements, called points,

(ii) \( B \) is a family of \( l \)-subsets of \( X \), called blocks, such that every block \( B_i \in B \) (\( 1 \leq i \leq |B| := b \)) is expressed as a disjoint union
\[
B_i = B_{i,1} \cup \cdots \cup B_{i,u}
\]
with \(|B_{i,1}| = \cdots = |B_{i,u}| = c \) and \(|B_i| = l = cu\),

(iii) every \( t \)-subset \( \{x_m\}_{m=1}^{t} \) of \( X \) is contained in exactly \( \lambda \) blocks \( B_i = B_{i,1} \cup \cdots \cup B_{i,u} \) such that
\[
x_m \in B_{j,m} \quad (j_m \text{ between } 1 \text{ and } u)
\]
for each \( 1 \leq m \leq t \), and \( j_1, \ldots, j_t \) are mutually distinct.

We summarize some basic conditions concerning the existence of splitting designs (cf. [6], [10]).

**Proposition 1**: Let \( D = (X, B) \) be a \( t-(v, b, l = cu, \lambda) \) splitting design, and for a positive integer \( s \leq t \), let \( S \subseteq X \) with \(|S| = s \). Then the number of blocks containing each element of \( S \) as per Definition 1 is given by
\[
\lambda_s = \lambda \frac{(v-s)_{(t-s)\lambda}}{c!(t-s)}.
\]
In particular, for \( t \geq 2 \), a \( t-(v, b, l = cu, \lambda) \) splitting design is also an \( s-(v, b, l = cu, \lambda_s) \) splitting design.

**Proposition 2**: Let \( D = (X, B) \) be a \( t-(v, b, l = cu, \lambda) \) splitting design. Let \( r := \lambda_t \) denote the number of blocks containing a given point. Then the following holds:

(a) \( b! = vr \).

(b) \( \binom{v}{t} \lambda = b \binom{u}{t} \).

(c) \( r(e-1) = \lambda_2(v-1) \) for \( t \geq 2 \).

**Proposition 3**: Let \( D = (X, B) \) be a \( t-(v, b, l = cu, \lambda) \) splitting design. Then
\[
\lambda(v-s)(t-s) \equiv 0 \pmod{c^{(v-s)(t-s)}}
\]
for each positive integer \( s \leq t \).

**Proposition 4**: If \( D = (X, B) \) is a \( t-(v, b, l = cu, \lambda) \) splitting design with \( t \geq 2 \), then
\[
b \geq \frac{v}{u}
\]

### IV. OPTIMAL SPLITTING AUTHENTICATION CODES

We state lower bounds on cheating probabilities for splitting authentication codes (cf. [12], [13]).

**Theorem 1**: In a splitting authentication code, for every \( 0 \leq i \leq t \), the deception probabilities are bounded below by
\[
P_{di} \geq \min_{e \in E} \frac{|M(e)| - i \cdot \max_{e \in S} |e(s)|}{|M| - i}
\]
A splitting authentication code is called \( t \)-fold secure against spoofing if
\[
P_{di} = \min_{e \in E} \frac{|M(e)| - i \cdot \max_{e \in S} |e(s)|}{|M| - i}
\]
for all \( 0 \leq i \leq t \).

We indicate a lower bound on the size of encoding rules for splitting authentication codes (see [10], and [14], [15] for the case \( t = 2 \)).

**Theorem 2**: If a splitting authentication code is \((t - 1)\)-fold secure against spoofing, then the number of encoding rules is bounded below by
\[
|E| \geq \prod_{i=0}^{t-1} \frac{|M| - i \cdot \max_{e \in S} |e(s)|}{|M| - i}
\]
A splitting authentication code is called \textit{optimal} if the number of encoding rules meets the lower bound with equality.
Corollary 1: In a $c$-splitting authentication code, $$P_{di} = \frac{e((|S| - i)|M| - i)}{e^i(S)}$$ for every $0 \leq i \leq t$.

Corollary 2: If a $c$-splitting authentication code is $(t - 1)$-fold secure against spoofing, then $$|E| \geq \frac{(|M|)}{c^i(S)}.$$ Optimal splitting authentication codes can be characterized in terms of splitting designs (see [10], and [6] for the case $t = 2$) as follows.

Theorem 3: Suppose there is a $(v, b, l = cu, 1)$ splitting design with $t \geq 2$. Then there is an optimal $c$-splitting authentication code for $u$ equiprobable source states, having $v$ messages and $\binom{v}{i}/c^i(\binom{v}{i})$ encoding rules, that is $(t - 1)$-fold secure against spoofing. Conversely, if there is an optimal $c$-splitting authentication code for $u$ source states, having $v$ messages and $\binom{v}{i}/c^i(\binom{v}{i})$ encoding rules, that is $(t - 1)$-fold secure against spoofing, then there is a $(v, b, l = cu, 1)$ splitting design.

V. Optimal Splitting Authentication Codes with Perfect Secrecy

In what follows, we are interested in optimal splitting authentication codes that simultaneously achieve perfect secrecy. According to Shannon [16], an authentication code is said to have perfect secrecy if $$p_{\theta}(s|m) = p_{\theta}(s)$$ for every source state $s \in S$ and every message $m \in M$, that is, the a posteriori probability that the source state is $s$, given that the message $m$ is observed, is identical to the a priori probability that the source state is $s$.

By introducing the notion of an external difference family (EDF) (which yields a certain type of a splitting design), Ogata–Kurosawa–Stinson–Saido [6, Thm. 3.4] have given a construction scheme for optimal splitting authentication codes with perfect secrecy in the special case when the number of keys equals the number of messages.

Theorem 4: Suppose there exists a $(v, c, l)$-EDF over an Abelian group of order $v$, then there is an optimal $c$-splitting authentication code for $u$ equiprobable source states, having $v$ messages and $v$ encoding rules, that is one-fold secure against spoofing and simultaneously achieves perfect secrecy.

Example 1: An optimal 2-splitting authentication code for $u = 2$ equiprobable source states, having $v = 9$ messages and $b = 9$ encoding rules, that is one-fold secure against spoofing and achieves perfect secrecy can be constructed from a $2 \times 9$ matrix. Each encoding rule is used with probability $1/9$. An encoding matrix is given in Table I.

In the following, we develop a construction method for obtaining optimal splitting authentication codes with perfect secrecy in the more general case when the number of keys may differ from the number of messages:

1. We first introduce the notion of a cyclic splitting design. Let $D = (X, B)$ be a $2-(v, b, l = cu, 1)$ splitting design, and let $\sigma$ be a permutation on $X$. For a block $B_i \in \mathcal{B}$ as in (ii) of Definition 1, define $B_i^\sigma := \{B_i^\sigma, \ldots, B_i^\sigma\}$, satisfying

$$B_i^\sigma = B_i^1 \cup \cdots \cup B_i^l$$

with $|B_i^\sigma| = |B_i^1| = \cdots = |B_i^l| = \sigma$ and $|B_i^\sigma| = l = cu$. If $B_i^\sigma := \{B_i^\sigma, B_i^\sigma, \ldots, B_i^\sigma\}$, then $\sigma$ is called an automorphism of $D$. If there exists an automorphism $\sigma$ of order $v$, then $D$ is called cyclic. In this case, the point-set $X$ can be identified with $\mathbb{Z}_v$, the set of integers modulo $v$, and $\sigma$ can be represented by $\sigma : j \mapsto j + 1$ (mod $v$). A block $B_i = \{B_i^1, \ldots, B_i^l\}$, the set

$$B_i + j := \{B_i^1 + j \text{ (mod } v), \ldots, B_i^l + j \text{ (mod } v)\}$$

for $j \in \mathbb{Z}_v$ is called a translate of $B_i$, and the set of all distinct translates of $B_i$ is called the orbit containing $B_i$. If the length of an orbit is $v$, then the orbit is said to be full, otherwise short. A block chosen arbitrarily from an orbit is called a base block (or starter block).

2. Let us assume that there exists a cyclic $2-(v, b, l = cu, 1)$ splitting design without short orbit. Then, by Theorem 3, there is an optimal $c$-splitting authentication code for $u$ equiprobable source states, having $v$ messages and $\binom{v}{i}/c^i(\binom{v}{i})$ encoding rules, that is one-fold secure against spoofing. Furthermore, when considering the corresponding $b \times u$ encoding matrix, it follows by constructional reasons from the underlying cyclic splitting design without short orbit that the code simultaneously achieves perfect secrecy under the assumption that the encoding rules are used with equal probability.

Hence, we have proved the following theorem.

<table>
<thead>
<tr>
<th>$s_1$</th>
<th>$s_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_1$</td>
<td>${1, 2}$</td>
</tr>
<tr>
<td>$c_2$</td>
<td>${3, 5}$</td>
</tr>
<tr>
<td>$c_3$</td>
<td>${2, 3}$</td>
</tr>
<tr>
<td>$c_4$</td>
<td>${3, 4}$</td>
</tr>
<tr>
<td>$c_5$</td>
<td>${4, 5}$</td>
</tr>
<tr>
<td>$c_6$</td>
<td>${5, 6}$</td>
</tr>
<tr>
<td>$c_7$</td>
<td>${6, 7}$</td>
</tr>
<tr>
<td>$c_8$</td>
<td>${7, 8}$</td>
</tr>
<tr>
<td>$c_9$</td>
<td>${8, 9}$</td>
</tr>
<tr>
<td>$c_{10}$</td>
<td>${9, 1}$</td>
</tr>
<tr>
<td>$c_{11}$</td>
<td>${1, 2}$</td>
</tr>
</tbody>
</table>

TABLE I

Splitting authentication code with perfect secrecy from a $2-(9, 9, 4 = 2 \times 2, 1)$ splitting design.
Theorem 5: Suppose there is a cyclic 2-(v, b, l = cu, 1) splitting design without short orbit (that is, it holds that \( v \equiv 1 \pmod{u(u - 1)/c^2} \)). Then there is an optimal \( c \)-splitting authentication code for \( u \)-equiprobable source states, having \( v \) messages and \( \binom{v}{c}/\binom{n}{c} \) encoding rules, that is one-fold secure against spoofing and simultaneously achieves perfect secrecy.

Relying on some recent constructions of splitting designs (cf. [17, Sect. 3.2]), we give exemplarily a series of optimal splitting authentication codes with perfect secrecy.

Example 2: (i) An optimal 2-splitting authentication code for \( u = 2 \) equiprobable source states, having \( v = 17 \) messages and \( b = 34 \) encoding rules, that is one-fold secure against spoofing and achieves perfect secrecy can be constructed from a cyclic 2-(17, 34, 4 = 2 \times 2, 1) splitting design with base blocks \( \{1, 2\}, \{3, 5\} \) and \( \{1, 2\}, \{11, 13\} \). Each encoding rule is used with probability \( 1/34 \). An encoding matrix is given in Table II.

(ii) As generalization of (i), an optimal \( c \)-splitting authentication code for \( u = 2 \) equiprobable source states, having \( v = 2c^2n + 1 \) messages and \( b = (2c^2n + 1)n \) encoding rules, that is one-fold secure against spoofing and achieves perfect secrecy can be constructed from a cyclic 2-(2c^2n + 1, 2c^2n + 1) = c \times 2, 1) splitting design with base blocks \( \{1, 2, \ldots, c\}, \{2c^2n - 2c - c + 1, 2c^2h - (2c^2 - c) + c + 1, \ldots, 2c^2h - (2c^2 - c) + c(c - 1) + 1\} \) for all \( 1 \leq h \leq n \).

(iii) Further examples of splitting authentication codes with perfect secrecy, also for \( u > 2 \), can be obtained in the same way from various further constructions of splitting designs in [17, Sect. 3.2].

Acknowledgment

The author gratefully acknowledges support by the Deutsche Forschungsgemeinschaft (DFG) via a Heisenberg grant (Hu954/4) and a Heinz Maier-Leibnitz Prize grant (Hu954/5).

References

Anti-Structure Problems

Ram Zamir
Department of Electrical Engineering-Systems
Tel Aviv University, Israel
zamir@eng.tau.ac.il

Abstract—The recent success of structured solutions for a class of information-theoretic network problems, calls for exploring their limits. We show that sum-product channels resist a solution by structured (as well as random) codes. We conclude that the structured approach fails whenever the channel operations do not commute (or for general functional channels, when the channel function is non-decomposable).

Key words: sum-product channels, distributed coding, functional source coding, functional Gelfand-Pinsker problem, non-decomposable functions, commutativity, associativity.

I. INTRODUCTION

Structured codes can be effective, and significantly better than random codes, for various multiuser problems. Prominent examples include the well known Korner-Marton (KM) “modulo-two sum” problem [1], as well as more recent setups such as the “dirty” multiple-access channel (MAC) [2], the noisy linear network (along with the compute & forward relaying technique) [3], and more [4]. The effectiveness of structured codes, in particular linear or lattice codes, is due to a good match between their linear structure and the additive nature of the source or channel involved.

In the Korner-Marton problem, for example, the two components \((X,Y)\) of a doubly-symmetric binary source are encoded separately. The joint decoder is not interested in a full reconstruction of \(X\) and \(Y\), which are viewed as “helper sources”; rather the decoder is interested in their modulo-two sum \(X+Y\) (or Xor). Writing the statistical relation between \(X\) and \(Y\) as a modulo-additive noise channel

\[
Y = X + Z
\]

we can recast the problem as that of lossless reconstruction of the noise \(Z\) from separate encodings of \(X\) and \(Y\).

As shown by Korner and Marton, a linear structured coding scheme, which sends the syndromes of \(X^n\) and \(Y^n\) with respect to a suitable linear binary code, achieves the minimum possible rate of \(H(Z)\) - the entropy of \(Z\) - per each encoder. In contrast, a conventional random coding scheme cannot do better than to encode at a sum rate equal to the joint entropy of \(X\) and \(Y\). This corresponds to the Slepian-Wolf solution [5], i.e., to a lossless reconstruction of both \(X\) and \(Y\) at the decoder. The resulting sum rate can be therefore significantly higher than \(2H(Z)\) for highly correlated sources.

The binary KM problem can be generalized to a \(q\)-ary field, in which case a linear \(q\)-ary code replaces the linear binary code in the KM solution [6]. And it also has a quadratic-Gaussian version [7].

A dual example with a similar characteristics is that of the “doubly dirty” MAC. This channel extends Costa’s “writing on a dirty paper” problem [8], [9] to a MAC; i.e., an additive-noise channel with two inputs \(X_1\) and \(X_2\) and an output \(Y\) given by

\[
Y = X_1 + X_2 + S_1 + S_2 + \text{noise}
\]

where \(S_1\) and \(S_2\) are two interferences, each known as “side information” to one of the encoders. Addition in (2) is over some group in the discrete channel case, or the usual addition in the continuous case. The problem is made interesting by imposing input constraints upon \(X_1\) and \(X_2\), thus the encoders cannot simply subtract the interferences.

Similarly to the Korner-Marton problem, a linear/lattice pre-coding scheme (which subtracts the interference “modulo the code”) achieves the capacity region of this channel [10], [2].

And in contrast, the rates achieved by a more conventional random binning scheme vanish in the limit of strong interference signals.

This sharp discrepancy is due to the distributive nature of the side-information; if the knowledge of \(S_1\) and \(S_2\) were centralized - i.e., they were both known to one encoder or to the joint decoder, then random binning could be effective and (nearly) achieve capacity; see [2].

Sometimes structured codes are inferior to random codes. This situation occurs when the linear structure of the code causes ambiguity at the decoder; for example, the symmetric-rates point of the (clean) MAC capacity region, or of the Slepian-Wolf rate region.

In this short note we focus on another, perhaps obvious weakness of structured codes; they are sensitive to the structure of the channel. Specifically, we show that if the additive channel in (1) or in (2) is replaced by one involving both addition and multiplication, then structured codes - and in fact, any other coding scheme - are not effective.

II. SUM-PRODUCT KORNER-MARTON

Consider a generalization of the KM problem (1), where the statistical relation between the component sources is given by

\[1\] In the discrete noiseless case the linear coding scheme is exactly optimal, while in the continuous case it is asymptotically optimal in the continuous high SNR case.

\[2\] This problem can be resolved by using two different (linearly independent) linear codes [11].
functional source coding with side information at the decoder.

\[ \begin{align*}
X \to \text{enc} \quad & \xrightarrow{M} \quad \text{dec} \rightarrow \hat{Z} = F(X, Y) \\
\uparrow \quad & \\
Y
\end{align*} \]

Fig. 1. Functional source coding with side information at the decoder.

4They in fact consider a more general case where both \( X \) and \( Y \) are encoded.

Consider next a modification of the dirty MAC problem (2), in which the channel output is given by

\[ Y = A' + A'' + B \times C \] (7)

3Regarding the former case (\( B \) available at the decoder), note that a random parity-check matrix \( H \) is “good” with high probability for the classical KM problem (1) (i.e., \( Z^n \) can be reliably decoded from the syndromes \( H X^n \) and \( H Y^n \)); hence, \( H B^n \) is “good” with high probability for the generalized KM problem (3). In the latter case, the encoders simply divide their observations by \( B \), hence get back to the classical KM problem.

\[ 2^n R, \text{ with } n \text{ being the code block length and } R \text{ being the coding rate.} \]

The decoding function is \( g : M \times F_q^2 \rightarrow F_q^n \).

The probability of error \( P_e \) is the probability that \( g(M, Y^M) \) is not equal to the vector \( Z^n \), where \( Z_i = F(X_i, Y_i), \ i = 1 \ldots n \), and where \( M = f(X^n) \) is the encoded message. For a given memoryless double source \((X_1, Y_1), (X_2, Y_2), \ldots \) and a function \( F(\ldots) \), a rate \( R \) is said to be “achievable” if we can make \( P_e \) as small as desired for some functions \( f \) and \( g \), and large enough \( n \). Finally, \( R^* \) denotes the minimum achievable rate.

Clearly, the minimum achievable rate \( R^* \) satisfies

\[ H(X|Y) \geq R^* \geq H(F(X,Y)|Y) \] (5)

where the LHS corresponds to the case where the decoder fully reconstructs \( X \) before computing \( F(X,Y) \), while the RHS corresponds to the case where the encoder also has access to \( Y \), so it can first compute \( F(X,Y) \) and then compress it. In the sum-product case (4), if \( A \) is uniform over \( F_q \), then the bounds (5) become

\[ H(B) + H(C) \geq R^* \geq H(C). \] (6)

Note that in the classical KM problem \( B = 1 \), i.e., \( H(B) = 0 \); thus the bounds coincide, and the coding rate is merely the entropy of the desired variable \( C \).

Han and Kobayashi [6] give necessary and sufficient conditions for the LHS of (5) to be tight.4 These conditions are satisfied in the sum-product case.

Lemma 1. In the sum-product (functional source coding) problem (4), \( R^* = H(X|Y) \). Thus, if \( A \) is uniform over \( F_q \), then \( R^* = H(B) + H(C) \).

Proof: Follows since two different lines in \( F_q \) intersect in at most one point, implying the condition in [6, lem.1].

This result implies that the minimum coding rate \( R^* \) is in general larger than the entropy of the desired variable \( C \), which is the rate in the classical KM setting (1). In fact, the “extra” rate can be as large as \( \log(q-1) \), for which is uniform over \( F_q \setminus 0 \).

As a corollary from Lemma 1, it follows that for uniform channel states \( B' \) and \( B'' \) in the distributed coding setup of (3), the rate of each encoder is at least \( \log(q-1) + H(C) \). The interpretation is that the introduction of the multiplicative state variables breaks the structure of the classical KM problem; the states \( B' \) and \( B'' \) must be fully conveyed to the decoder before the linear structure of the channel can be utilized (by means of a linear “syndrome” coding) to encode the desired source \( C \).

III. THE SUM-PRODUCT DIRTY MAC

Consider next a modification of the dirty MAC problem (2), in which the channel output is given by

\[ Y = A' + A'' + B \times C \] (7)

4They in fact consider a more general case where both \( X \) and \( Y \) are encoded.
where as in (3) all variables belong to a finite field $F_q$ of size $q$, and the $+$ and $\times$ are the sum and product operations over $F_q$. The inputs of this MAC are $A'$ and $A''$ (corresponding to $X_1$ and $X_2$ in (2)), while $B$ and $C$ are the channel state variables (corresponding to $S_1$ and $S_2$ in (2)). There is no additional noise, nor input constraints.

As in the sum-product KM, the centralized state case is easy: if both state variables $B$ and $C$ are known to one encoder, or are known to the joint decoder, then the product $B \times C$ can be simply subtracted; hence the sum capacity is $\log(q)$, as if the channel was noiseless.

The interesting setup is, again, the de-centralized state case. That is, each encoder has access to only one of the channel states, while the decoder is completely ignorant of the states. The capacity in this case is bounded from above by that of the single-user channel shown in Fig. 2, with

$$Y = A + B \times C. \tag{8}$$

Here there is a single encoder that has access to one of the states ($S_1 = B$), while the decoder has access to the second state ($S_2 = C$), where $C$ is independent of both the input $X = A$ and $B$.

A precise definition of encoding and decoding over such a channel is as follows. The encoding function is $f : M \times \mathbb{F}_q^n \rightarrow F_q^n$ and the decoding function is $g : F_q^n \times F_q^n \rightarrow \mathcal{M}$, where the size of the message space $\mathcal{M}$ is $2^n R$, $R$ being the coding rate. The error probability $P_e$ is the probability that $g(Y^n, S_2^n) \neq M$, where $Y^n$ depends on $X^n$, $S_1^n$ and $S_2^n$, and $X^n = f(M, S_1^n)$ for $M \in \mathcal{M}$. A rate $R$ is said to be “achievable” if we can make $P_e$ as small as desired for some functions $f$ and $g$, and large enough $n$. Finally, the capacity $C$ is the highest achievable rate.

The sum-product channel (8) is, in fact, a deterministic channel, where the output $Y$ is a function of the input $X$, and the two states $S_1$ and $S_2$:

$$Y = F(X, S_1, S_2). \tag{9}$$

There is no additional noise in the channel, beyond the randomness of the two (known) states $S_1$ and $S_2$.

The setup of (9) is an instance of the Gelfand-Pinsker problem [9], i.e., a channel with non-causal side information at the encoder. Hence, it has a single letter solution of the form

$$C = \max I(U; Y, S_2) - I(U; S_1) \tag{10}$$

where the maximization is over a suitable set of auxiliary random variables $U$, and functions $X = X(U, S_1)$.

The structure of the function $F$ in (9) plays a key role in determining the capacity $C$. A favorable case is when $F$ has a composite form, where the dependence on the encoder variables $(X, S_1)$ is separate from the decoder state $S_2$.

**Lemma 2.** If the function $F : F_q \times F_q \times F_q \rightarrow F_q$ can be decomposed into $F(a, b, c) = \tilde{F}(G(a, b), c)$, where $\tilde{F}$ is invertible with respect to the first argument (i.e., the equation $y = \tilde{F}(t, c)$ has a solution $t$ for every $y$ and $c$), then (10) is optimized by $U = G(X, S_1)$. If also $G$ is invertible with respect to the first argument, then the capacity is

$$C = \log(q)$$

and it is achieved by an input $p(x|s_1)$ that makes $G(X, s_1)$ uniform over $F_q$ for all values of $s_1$.

**Proof:** The first part follows from [13, sec. III.F], and the invertibility of $\tilde{F}$. See also [14].

The sum-product channel (8) clearly does not satisfy the first condition of the lemma, as addition and multiplication do not commute. In fact, this channel is much worse. To assess its capacity, we shall first establish a relation to a “minimum entropy” problem.

**Lemma 3.** The capacity of a deterministic two-state channel of the form

$$Y = X + F(S_1, S_2) \tag{11}$$

where $+$ denotes addition in $F_q$, and where $S_1$ and $S_2$ are available at the encoder and the decoder, respectively, is given by

$$C = \log(q) - \inf \frac{1}{n} H\left(g(S_1^n) + F(S_1^n, S_2^n) | S_2^n\right) \tag{12}$$

where the second term is the average conditional entropy given $S_2^n$, and where the infimum is over all code block lengths $n$, and functions $g : F_q^n \rightarrow F_q^n$.

**Proof:** Easy and will be omitted.

Although (12) is not a single-letter expression, it is sometimes easier for analysis than the Gelfand-Pinsker solution (10). Specifically, for the sum-product channel (8) we have:

**Theorem 1. (Shany-Zamir [15])** For the case where $Y = X + S_1 \times S_2$, the minimum average conditional entropy in (12) is bounded by

$$\log\left(\frac{q}{2}\right) \leq H_{\min} \leq \log\left(\frac{q}{2-1/q}\right).$$

The upper bound is achieved (for all $n$) by a quadratic per-letter function $g(s) = s^2$.

As a corollary from this theorem, we conclude that the capacity of the sum-product channel (8) is at most one bit.

The admissible $U$’s are those for which $S_2$ is independent of $(U, X, S_1)$, and $U \leftrightarrow (X, S_1) \leftrightarrow Y$ form a Markov chain for each value of $S_2$. Since $S_2$ is independent of $U$, the first term in (10) can be written also as $I(U; Y | S_2)$.
for all $q$. This is quite disappointing when compared to the capacity of $\log(q)$, which is achievable in the centralized-state case. The same statement is true also for the sum-product dirty MAC (7); that is, the rate of each user is at most one bit.7

IV. DISCUSSION

The essence of the examples given in this paper is that the order of performing the sum and product operations matters; and in fact, they are very “different”. One aspect of this difference - which is related to the sum-product KM problem - is that the expression

$$a + b \times c$$  \hspace{1cm} (13)

cannot be decomposed into the form: function( function $(a, b), c$) (not even approximately). A second aspect - related to the sum-product dirty MAC - is that it is impossible to find a function of $a = a(b)$ such that (13) would be only a function of $c$ (not even approximately). In contrast, these two requirements are easily fulfilled if the expression in (13) is a pure sum $a + b + c$ (by the associativity of summation), or a pure product $a \times b \times c$ (by the associativity of multiplication).8

It would be interesting to explore further (and perhaps quantify) the information-theoretic aspects of function decomposition. Note that this question is almost “distribution free” (i.e., nearly independent of the probability distributions of sources and channels). A different aspect of “anti structure”, which is due to a “bad” noise distribution, can be found in [16].

ACKNOWLEDGMENT

I thank Yaron Shany for many interesting discussions about the sum-product channel. Thanks are also due to Aaron Wagner for an interesting discussion about the sum-product KM problems, and to Prakash Iswar and Sandeep Pradhan for pointing out the relation to the work of Han and Kobayashi.

REFERENCES


\[^7\]Perhaps even the stronger statement - that the sum rate is at most one bit - is true.

\[^8\]Note that pure multiplicative versions of the generalized KM and DMAC problems can be solved using linear codes over a “logarithmic” domain.
Real-Time Coding of Markov Sources over Erasure Channels: When is Binning Optimal?

Ashish Khisti and Farrokh Fetzadi
University of Toronto
10 King’s College Road
Toronto, ON, Canada
Email: {akhisti, fetzadi}@comm.utoronto.ca

Mitchell Trott
HP Labs
1501 Page Mill Road
Palo Alto, California
Email: mitchell.trott@hp.com

Abstract—We study sequential coding of a Markov source process under error propagation constraints. The channel can erase up to $B$ packets in a single burst, but reveals all other packets to the destination. The destination is required to reproduce all the (vector) source sequences sequentially, except those that occur in a window of length $B+W$ following the start of the erasure burst. Our earlier work establishes upper and lower bounds on the compression rate as a function of $B$ and $W$. In this work we show that for the class of symmetric sources, if we restrict to a memoryless encoding function, then a binning based scheme is optimal. Our converse involves a drawing connection between the sequential coding problem and a multiuser source coding problem called Zig-Zag source coding with side information.

I. INTRODUCTION

A tradeoff between compression efficiency and error resilience is fundamental to any video compression system. In live video encoding, an encoder observes a sequence of correlated video frames and produces a compressed bit-stream that is transmitted to the destination. If the underlying channel is an ideal bit-pipe, it is well known that predictive coding [1] achieves the optimum compression rate. Unfortunately in many emerging video distribution networks, such as peer-to-peer systems and mobile systems, packet losses are unavoidable. Predictive coding is highly sensitive to such packet losses and can lead to a significant amount of error propagation. Various techniques are used to practice prevent such losses. Commonly used video coding techniques use a group of picture (GOP) architecture, where intra-frames are periodically inserted to limit the effect of error propagation. Forward error correction codes can also be applied to compressed bit-streams to recover from missing packets [2], [3]. Modifications to predictive coding, such as leaky-DPCM [4], [5], have been proposed in the literature to deal with packet losses. The robustness of distributed video coding techniques in presence of packet losses has been studied in e.g., [6], [7].

Information theoretic analysis of video coding has received significant attention in recent times, see e.g., [8]–[10] and the references therein. These works focus primarily on the source coding aspects of video. The source process is a sequence of vectors, each of which is spatially i.i.d. and temporally correlated. Each source vector is sequentially compressed into a bit stream. The destination is required to recover the source vectors in a sequential manner as well. However all of these works assume an ideal channel with no packet losses. To our knowledge even the effect of a single isolated packet loss is not fully understood [11].

In an earlier work [12] we build upon [8], [9] and introduce an information theoretic framework to characterize the tradeoff between error propagation and compression rate. An encoder is revealed source vectors in a sequential manner and compresses them sequentially into channel packets that are then transmitted over a channel. An information theoretic notion of error propagation is defined and upper and lower bounds are obtained on the compression rate. The lower bound is based on a careful analysis of information flow during the decoding process whereas the upper bound is based on a binning technique. For a special class of sources, a new technique, prospicient coding is proposed, and shown to be optimal. However the optimal compression rate remains an open problem for a large class of sources including the binary symmetric Markov source.

In this paper we consider a class of symmetric Markov sources. For this class the minimum compression rate has not yet been characterized. Our main result in this paper is that if we restrict to the class of memoryless encoders, then the binning based scheme proposed in [12] is optimal. The converse is established by drawing a connection to a multiuser source coding problem called zig-zag source coding network [13]–[15].

II. PROBLEM STATEMENT

A. Source Model

We consider a semi-infinite stationary vector source process \( \{ s^n \}_{n \geq 0} \) whose symbols (defined over some finite alphabet \( \mathcal{S} \)) are drawn independently across the spatial dimension and from a first-order Markov chain across the temporal dimension, i.e., for each \( t \geq 1 \),

\[
\Pr( s^n = s^n_0 | s^{n-1}_0 = s^{n-1}_{n-1} ) = \prod_{j=1}^{n} p_{s_{j} | s_{j-1}}(s_{j} | s_{j-1}), \quad \forall t \geq 1.
\]  

(1)

We assume that the prior distribution \( p_{s_0}(\cdot) \) and \( p_{s_{j} | s_{j-1}}(\cdot | \cdot) \) are selected such that the underlying random variables \( \{ s_j \}_{j \geq 0} \) constitute a time-invariant and a first-order stationary Markov
chain. Of particular interest in this paper is the class of symmetric sources where the underlying Markov chain is also reversible i.e., the random variables satisfy $\{s_0, \ldots, s_t\} \equiv (s_0, \ldots, s_t)$, where the equality is in the sense of distribution [16]. Of particular interest to us is the following property satisfied for each $i \geq 1$.

$$p_{s_{i-1}, s_i}(s_{i+1}, s_{i+2}) = p_{s_{i-1}, s_i}(s_{i+2}, s_{i+1}), \quad \forall s_{i-1}, s_i \in S$$  \hspace{1cm} (2)

i.e., we can “exchange” the source pair $\{s_{i+1}, s_{i+2}\}$ with $\{s_{i+2}, s_{i+1}\}$ without affecting the joint distribution. An important class of sources that are symmetric are the binary sources: $s_i = s_{i+1} \oplus z_i$, where $\{z_i\}_{i=1}^\infty$ is an i.i.d. binary source process (in both temporal and spatial dimensions) with the marginal distribution $Pr(z_i = 0) = p$, the marginal distribution $Pr(s_i = 0) = 1/2$ and $\oplus$ denotes modulo-2 addition.

### B. Rate-Recovery Function

A rate-$R$ causal encoder maps the sequence $\{s^n_i\}_{i=0}^\infty$ to an index $f_i \in [1, 2^{\infty}]$ according to some function

$$f_i = f_i(s_0^i, s_1^i, \ldots, s_{i-1}^i)$$  \hspace{1cm} (3)

for each $i \geq 0$. A memoryless encoder satisfies $f_i(s_0^i, \ldots, s_{i-1}^i) = f_i(s_i^i)$ i.e., the encoder does not use the knowledge of the past sequences.

The channel introduces an erasure burst of size $B$, i.e. for some particular $j \geq 0$, it introduces an erasure burst such that $g_i = \star$ for $i \in \{j, j+1, \ldots, j+B-1\}$ and $g_i = f_i$ otherwise. Upon observing the sequence $\{g_j\}_{j \geq 0}$ the decoder is required to perfectly recover all the source sequences using decoding functions

$$s_i^j = g_i(s_0^i, g_1^i, \ldots, g_i^i), \quad i \notin \{j, \ldots, j+B+W-1\}.$$  \hspace{1cm} (4)

It is however not required to produce the source sequences in the window of length $B+W$ following the start of an erasure burst. We call this period the error propagation window. The setup is shown in Fig. 1.

A rate $R(B,W)$ is feasible if there exists a sequence of encoding and decoding functions and a sequence $e_n$ that approaches zero as $n \rightarrow \infty$ such that, $Pr(s_i^j \neq \hat{s}_i^j) \leq e_n$ for all $i \notin \{j, \ldots, j+B+W-1\}$. We seek the minimum feasible rate $R(B,W)$ which we define to be the rate-recovery function. The following upper and lower bounds have been established in [12].

**Theorem 1 ([12]):** For any stationary first-order Markov source process the rate-recovery function satisfies $R^-(B,W) \leq R(B,W) \leq R^+(B,W)$ where

$$R^-(B,W) = H(s_1|s_0) + \frac{1}{W+1} I(s_{B+1}; s_{B+W+1}|s_0).$$  \hspace{1cm} (5)

$$R^+(B,W) = H(s_1|s_0) + \frac{1}{W+1} I(s_B; s_{B+1}|s_0).$$  \hspace{1cm} (6)

Notice that the upper and lower bound coincide for $W = 0$ and $W \rightarrow \infty$, yielding the rate-recovery function in these cases. The upper bound is obtained via a memoryless binning based scheme. At each time the encoding function $f_i$ in (3) is obtained as the bin-index of an independent Slepian-Wolf codebook [17]. The rate expression for $R^+(B,W)$, which is equivalent to [12]

$$R^+(B,W) = \frac{1}{W+1} H(s_{B+1}, s_{B+2}, \ldots, s_{B+W+1}|s_0).$$  \hspace{1cm} (7)

guarantees that the decoder can recover $s_{j+B+W}^j$ following an erasure burst between $[j, j+B-1]$ using the $W+1$ bin indices $f_{j+B}, \ldots, f_{j+B+W}$ and the source sequence $s_{j-1}^j$ before the erasure.

In [12] some counter-examples are provided where the lower bound $R^-(B,W)$ (c.f. (5)) is tight and binning based upper bound $R^+(B,W)$ (c.f. (6)) is not optimal in general. Nevertheless such examples require a special structure and do not include many natural source models such as the binary symmetric sources. The optimal rate-recovery function for the class of symmetric sources remains open. In this paper we establish the optimality of binning based scheme if one restricts to the class of memoryless encoders.

**Theorem 2:** For the class of symmetric sources that satisfy (2) the rate-recovery function, restricted to the class of
memoryless encoders, is given by

\[ R(B,W) = \frac{1}{W+1} H(s_{B+1}, s_{B+2}, \ldots, s_{B+W+1} | s_0). \]  

(8)

Note that the achievability follows immediately from (7). Thus it only remains to show that the lower bound (5) needs to be improved. We have only been able to obtain this improvement for the class of memoryless encoders. For the general encoder structure (3) this remains an open problem. At first glance one may expect that the binning based scheme is always optimal for the class of memoryless encoders. This is however not true. Interestingly the *prospective encoders* in [12] that improve upon the binning based lower bound are also memoryless. Our proof involves an interesting connection a multi-terminal source coding problem called zig-zag source coding [13]–[15]. In particular we develop a simple approach to lower bound the sum-rate of a zig-zag source coding network with symmetric sources that may be of independent interest.

III. PROOF OF THEOREM 2

The special case when \( W = 0 \) follows directly from (5). We only need to consider the case when \( W \geq 1 \). For simplicity in exposition we consider the case when \( W = 1 \). Then we need to show that

\[ R(B,W = 2) \geq \frac{1}{2} H(s_{B+1}, s_{B+2} | s_0). \]  

(9)

The proof for general \( W \geq 1 \) follows along similar lines and will be sketched briefly.

Assume that an erasure-burst spans time indices \( j-B, \ldots, j-1 \). The decoder must recover

\[ s^n_{j+1} = \hat{g}_{j+1} \left( c^{j-B-1}, f_j, f_{j+1} \right). \]  

(10)

From Fano’s inequality, we have,

\[ H \left( s^n_{j+1} | \epsilon^1_{j-B-1}, f_j, f_{j+1} \right) \leq n \epsilon_n. \]  

(11)

Furthermore if there is no erasure until time \( j \) then

\[ s^n_j = \hat{g}_j \left( \epsilon^1_j \right) \]  

must hold. Hence from Fano’s Inequality,

\[ H \left( s^n_j | \epsilon^1_j \right) \leq n \epsilon_n. \]  

(13)

Our aim is to combine (11) and (13) to establish the following lower bound on the sum-rate

\[ R_j + R_{j+1} \geq H(s_{j+1} | s_j) + H(s_j | s_{j-B-1}). \]  

(14)

The lower bound then follows since

\[ R \geq \max(R_j, R_{j+1}) \]

\[ \geq \frac{1}{2} (R_j + R_{j+1}) \]

\[ \geq \frac{1}{2} \left( H(s_{j+1} | s_j) + H(s_j | s_{j-B-1}) \right) \]

\[ = \frac{1}{2} \left( H(s_{j+1} | s_j, s_{j-B-1}) + H(s_j | s_{j-B-1}) \right) \]

\[ = \frac{1}{2} H(s_{j+1}, s_j | s_{j-B-1}) = \frac{1}{2} H(s_{B+1}, s_{B+2} | s_0). \]  

(19)

thus establishing (9).

To establish (14) we make a connection to a multi-terminal source coding problem in Fig. 2.

A. Zig-Zag Source Coding

Consider the source problem from side information illustrated in Fig. 2(a). In this setup there are four source sequences drawn i.i.d. from a joint distribution \( p(s_j, s_{j-1}, s_{j-B-1}) \). The two encoders \( j \) and \( j+1 \) are revealed source sequences \( s^n_j \) and \( s^n_{j+1} \) and the two decoders \( j \) and \( j+1 \) are revealed sources \( s^n_{j+1} \) and \( s^n_{j-B-1} \). The encoders operate independently and compress the source sequences to \( f_j \) and \( f_{j+1} \) at rates \( R_j \) and \( R_{j+1} \) respectively. Decoder \( j \) has access to \( (f_j, s^n_{j+1}) \) while decoder \( j+1 \) has access to \( (f_{j+1}, s^n_{j-B-1}) \) and are interested in reproducing,

\[ s^n_j = \hat{g}_j(f_j, s^n_{j+1}) \]  

(20)

\[ s^n_{j+1} = \hat{g}_{j+1}(f_{j+1}, s^n_{j-B-1}) \]  

(21)

respectively such that \( \Pr(s^n_i \neq \hat{s}^n_i) \leq \epsilon_n \) for \( i = j, j+1 \).

When \( s^n_{j-B-1} \) is a constant sequence, the problem has been studied in [13], [15]. A complete single letter characterization involving an auxiliary random variable is obtained. Fortunately in the present case of symmetric sources a simple lower bound can be obtained using the following observation.

**Lemma 1**: The set of all achievable rate-pairs \( (R_j, R_{j+1}) \) for the problem in Fig. 2(a) is identical to the set of all achievable rate-pairs for the problem in Fig. 2(b) where the side information sequence \( s^n_{j+1} \) at decoder 1 is replaced by the side information sequence \( s^n_{j-B-1} \).

The proof of Lemma 1 follows by observing that the capacity region for the problem in Fig. 2(a) depends on the joint distribution \( p(s_j, s_{j+1}, s_{j-B-1}) \) only via the marginal distributions \( p(s_j, s_{j-1}) \) and \( p(s_{j+1}, s_{j-B-1}) \). When the source is symmetric the distributions \( p(s_j, s_{j-1}) \) and \( p(s_{j+1}, s_{j-B-1}) \) are identical. The formal proof will be omitted.

Thus it suffices to lower bound the achievable sum rate for the problem in Fig. 2(b). First upon applying the Slepian-Wolf lower bound to encoder \( j+1 \)

\[ nR_{j+1} \geq H(s^n_{j+1} | s^n_{j-B-1}, f_j) - n \epsilon_n \]  

(22)

and to bound \( R_j \)

\[ nR_j \geq H(f_j) = I(f_j; s^n_{j-B-1}) \]

\[ \geq H(s^n_{j-B-1} | s^n_{j-B-1}) - H(s^n_{j-B-1} | f_j) \]

\[ = nH(s^n_{j-B-1} | s^n_{j-B-1}) - nH(s^n_{j-B-1} | f_j) \]

\[ = nH(s^n_{j-B-1} | s^n_{j-B-1}) - nH(s^n_{j-B-1} | s^n_{j-B-1}, f_j) \]  

(23)

\[ = nH(s^n_{j-B-1} | s^n_{j-B-1}) - nH(s^n_{j-B-1} | s^n_{j-B-1}, f_j) \]  

(24)

where (23) follows by applying Fano’s inequality to since \( s^n_{j-B-1} \) can be recovered from \( (s^n_{j-B+1}, f_j) \) and hence

\[ H(s^n_{j-B-1} | s^n_{j-B-1}, f_j) \leq n \epsilon_n \]

holds and (24) follows from the Markov relation \( s^n_{j-B+1} \rightarrow s^n_{j-B} \rightarrow (f_j, s^n_{j-B-1}) \). Observe that (14) follows by summing (22) and (24).
B. Connection between Streaming and Zig-Zag Coding Problems

It remains to show that the lower bound on the Zig-Zag coding problem also constitutes a lower bound on the original problem.

Lemma 2: Suppose that the encoding function $f_j = \mathcal{E}_j (s^n_j)$ is memoryless, suppose that there exist decoding functions $\hat{s}_j^n = \mathcal{D}_j (\hat{s}_j^n)$ and $\hat{s}_{j+1}^n = \mathcal{D}_j (f_{j+1}, f_{j+2})$ such that $\Pr(\hat{s}_j^n \neq s_j^n)$ and $\Pr(\hat{s}_{j+1}^n \neq s_{j+1}^n)$ both vanish to zero as $n \to \infty$. Then

$$H(s_j^n | s_{j-1}^n, f_j) \leq nR_j \quad (25)$$

$$H(s_{j+1}^n | s_{j+B-1}^n, f_j, f_{j+1}) \leq nR_j \quad (26)$$

also hold.

We omit the proof due to space constraint. The conditions in (25) and (26) show that any rate that is achievable in the original problem is also achieved in the zig-zag source network. Hence a lower bound to the source network also constitutes a lower bound to the original problem.

C. Extension to Arbitrary $W > 1$

Finally we comment on the extension of the above approach to $W = 2$. We now consider three encoders $j \in \{ j, j+1, j+2 \}$. Encoder $j$ observes a source sequence $s_j^n$ and compresses it onto a channel $f_j$ into an index $f_j \in [1, 2^{nR_j}]$. The corresponding decoders are revealed $s_{j-B-1}^n$ for $t \in \{ j, j+1 \}$ and the decoder $j+2$ is revealed $s_{j+2}^n$. By an argument analogous to Lemma 1 the rate region is equivalent to the case when decoders $j$ and $j+1$ are instead revealed $s_{j+1}^n$ and $s_{j+2}^n$ respectively. For this new setup it is easy to show that decoder $j+2$ must reconstruct $(s_j^n, s_{j+1}^n, s_{j+2}^n)$ given $(s_{j-B-1}^n, f_j, f_{j+1}, f_{j+2})$. The sum rate must therefore satisfy $\mathcal{R}_j + \mathcal{R}_{j+1} + \mathcal{R}_{j+2} \geq 4H(s_j^n, s_{j+1}^n, s_{j+2}^n | s_{j-B-1}^n)$. Using an extension of Lemma 2 we can show that the proposed lower bound also continues to hold for the original streaming problem. This complements the proof. The extension to any arbitrary $W > 1$ is completely analogous.
SuperNyquist Rateless Coding for Intersymbol Interference Channels

Uri Erez
Tel Aviv University, Ramat Aviv, Israel
Email: uri@eng.tau.ac.il

Gregory W. Wornell
Dept. EECS, MIT, Cambridge, MA
Email: gww@mit.edu

II. SYSTEM AND CHANNEL MODEL

We consider a linear dispersive Gaussian channel for which the complex baseband channel output takes the form

\[ g(t) = h(t) * x(t) + z(t), \]

where \( z(t) \) is additive white Gaussian noise (AWGN) with one-sided power spectral density \( N_0 \), and where \( x(t) \) is the input, which is subject to a power constraint \( \mathbb{E}\{|z(t)|^2\} \leq P \) and bandwidth constraint \( W \). The associated white-input capacity of the channel is

\[ C_{[h]} = \int_{-W/2}^{W/2} \log \left( 1 + \frac{P|H(f)|^2}{N_0 W} \right) df. \]

We consider pulse-amplitude modulation, whereby

\[ x(t) = \sum_n s[n] g(t - nT) \text{,} \]

where \( T \) is the symbol duration and \( T_0 = 1/W \) is the Nyquist sampling time. The associated “over-signalizing” ratio is thus \( L = T_0/T \). After matched filtering and sampling at the symbol rate, the equivalent discrete-time channel is

\[ g[n] = k[n] * s[n] + z[n], \]

where \( k[n] = k(nT) \), where \( k(t) = h^*(t) * h(t) * g(-t)^* * g(t) \), and where

\[ S_{xx}(e^{j2\pi f}) = \frac{N_0}{2} K(e^{j2\pi f}) = \frac{N_0}{2T} \sum_j |H(f/T + j/T)|^2 |G(f/T + j/T)|^2. \]

The pulse shape is required to be limited to system bandwidth \( W \), i.e., \( g(t) \) satisfies \( G(f) = 0 \) for \( |f| > W/2 \). To simplify our development, we largely restrict our attention to the case

\[ g(t) = \sin(t/T_0), \text{ with } \sin(u) \triangleq \sin(\pi u)/\pi u. \]

Taking the symbols \( s[n] \) to be independent identically-distributed (i.i.d.) circularly symmetric complex Gaussian with power \( P/L \) results in a (proper) Gaussian random input signal...
$x(t)$ with power $P$. It follows that the capacity of the discrete-time channel (3) is
\[ C_{[b/\text{SNQ symbol}]} = \int_{-1/2}^{1/2} \log \left( 1 + \frac{P}{N_0} \sum_{i} |H(f+i/T)^2|G(f+i/T)^2 \right) df \]
\[ = T \int_{-1/2}^{1/2T} \log \left( 1 + \frac{P}{N_0} \sum_{i} |H(f+i/T)^2| \right) df \]
\[ = T \int_{-1/2}^{1/2T} \log \left( 1 + \frac{P}{N_0} \sum_{i} |H(f+i/T)^2| \right) df, \]
where the last equality follows from the fact that $g(t)$ is bandlimited.

Note that for sinc modulation (4), $x(t)$ has a flat power spectrum over the bandwidth $W$, and the modulation achieves the white-input capacity (1) for any $L$, i.e., (5) specializes to
\[ C_{[b/\text{SNQ symbol}]} = \frac{1}{LW} \int_{-W/2}^{W/2} \log \left( 1 + \frac{P}{N_0} \right) df \]
\[ = \frac{T}{W} \sum_{i} C_{[b/\text{SNQ symbol}]} \]

III. LINEAR SNQ RATELESS CODING

Consider now packetized transmission where the packet size is large but otherwise plays no role in the analysis. We consider a simplified model where the channel response experienced throughout transmission of the $m$th packet, $m = 1, \ldots, M$, is linear time-invariant (LTI) but the impulse response, which we denote by $h_m(t)$, may vary from packet to packet. The channel input-output relation for the transmission of the $m$th packet is therefore
\[ y_m[n] = s_m[n] * h_m[n] + z_m[n], \]
where $k_m[n] = k_m[nT]$ and $h_m(t) = h_m^*(-t) * g^*(-t) + g(t)$. Assuming discrete-time white-input transmission for all packets, it follows from (6), that the mutual information (in b/\text{SNQ symbol}) corresponding to each packet is
\[ C_m = \int_{-1/2}^{1/2} \log \left( 1 + \frac{P}{N_0} \frac{|H(f)|^2}{|G(f)|^2} \right) df \]
\[ = \frac{T}{W} \sum_{i} \log \left( 1 + \frac{P}{N_0} \right) df, \]
where the second equality holds for ideal sinc modulation.

Upon receiving a set $S \subseteq \{1, \ldots, M\}$ of packets, the aggregate mutual information is thus
\[ C(S) = \sum_{m \in S} C_m. \]

Our aim is to design a low complexity coding and modulation scheme that (simultaneously) approaches $C(S)$ for all sets $S \subseteq \{1, \ldots, M\}$ without requiring the transmitter to have knowledge of the capacities $C_m$. Rather, for any given chosen target rate $R$, and no knowledge of the channel, transmission should be successful whenever $C(S) > R$ holds for the received set of packets $S$.

We proceed to describe the proposed linear rateless SNQ construction. All the signals $s_m[n]$ are obtained from a single coded stream $s[n]$ according to
\[ s_m[n] = v_m[n] * s[n], \]
where $v_m[n]$ are sequences to be specified. The transmitted signal corresponding to packet $m$ is thus
\[ x_m(t) = \sum_{i} s[n] \cdot v_m[n] * g(t - iT). \]

Provided we choose the sequences $v_m[n]$ so that the transmitted signals $x_m(t)$ are statistically independent and the $s_m[n]$ are white circularly-symmetric complex Gaussian processes, the mutual information corresponding to each packet remains $C_m$ and furthermore the aggregate mutual information from the receipt of multiple packets is the sum of the individual ones.

A simple means to achieve this is by taking $v_m[n] = e^{-j2\pi mn/L}$. Accordingly, we define
\[ x_m[n] \triangleq x_m(nT) = \sum_{i} s[n] e^{-j2\pi mn/L} g(n - i), \]
where $g[n] = g(nT) = \text{sinc}(n/L)$, with $g(t)$ as in (4). Using (10), we see that the transmit signals “shifted-back in frequency” in this case are
\[ x_m[n] \triangleq e^{j2\pi mn/L} x_m[n] = \sum_{i} s[n] g(n - i), \]
where $g_m[n] = e^{j2\pi mn/L} g[n]$.

Clearly, requiring that the signals $\{x_m(t)\}$ be mutually independent is equivalent to requiring that the associated discrete-time signals $\{x_m[n]\}$ be. Furthermore, the latter holds if and only if $\{x_m[n]\}$ are mutually independent. Therefore, it suffices to verify the last condition. Since the signals $x_m[n]$ are jointly Gaussian and stationary, they are independent if their cross-spectra
\[ S_{x_m x_m}(e^{j2\pi f}) = S_{xx}(e^{j2\pi f}) \frac{G_m(f + m/L)}{G^*(f + m/L)} \frac{G^*(f + m/L)}{G^*(f + m/L) \frac{1}{T}} \]
vanish. Since $G(f/T)$ occupies no more than $1/L$ of the SNQ frequency band, it follows that there is no overlap between the frequency responses $G((f + m/L)/T \frac{1}{T})$ for different values of $m$, and hence $S_{x_m x_m}(e^{j2\pi f})$ indeed vanishes for $m_1 \neq m_2$.

A. RECEIVER ARCHITECTURE

A low-complexity receiver architecture suffices to approach the associated information-theoretic limits. In particular, specializing (7) to the case (9) with our choice $v_m[n] = e^{-j2\pi mn/L}$, the equivalent “shifted back” channel model is
\[ y_m[n] \triangleq y_m[n] e^{j2\pi mn/L} = \tilde{y}_m[n] * s[n] + \tilde{z}_m[n], \]
where $\tilde{k}_m[n] = k_m[n] e^{j2\pi mn/L}$. Since these channels do not overlap in frequency, they may be added without loss, resulting in the effective scalar ISI channel

$$\tilde{y}[n] = \sum_{m \in S} y_m[n] e^{j2\pi mn/L} = s[n] * \left( \sum_{m \in S} \tilde{k}_m[n] \right) + \sum_{m \in S} z_m[n].$$

For such ISI channels, the unbiased MMSE decision-feedback equalizer (DFE) is an information-lossless receiver structure [3], [4]. In particular, to approach capacity, one can use Guess-Vanassari interleaving [3] and a single (fixed-rate) base code designed for an AWGN channel. In essence, every symbol is replaced by a different codeword and thus the DFE decision device acts on codewords rather than symbols.

IV. MIMO-SNQ: EXTENDING SNQ TO MIMO SYSTEMS

A straightforward extension of the preceding architecture to a multi-input multi-output (MIMO) system, which we term MIMO-SNQ, is as follows. The particular channel model of interest is

$$y_m(t) = H_m(t) * x_m(t) + z_m(t),$$

where there are $N_t$ transmit and $N_r$ receive elements. In turn, with input of the form $x_m(t) = \sum_n x_m[n] g(t - nT)$, the associated discrete-time channel, after applying a matrix matched filter, is

$$y_m[n] = K_m[n] * x_m[n] + z_m[n],$$

where $K_m[n] = K_{m0}(nT)$ with $K_m(t) = H_m^\dagger(-t) * H_m(t) * g^\dagger(-t) * g(t)$.

We employ a single-stream transmission architecture based on the application of time-varying DFT beamforming to a scalar signal $s[n]$. Specifically, we assume that the transmitted signal corresponding to packet $m$ is formed as $x_m[n] = s[n] v_m[n]$, where $v_m[n] = v[n] e^{-j2\pi mn/L}$ with

$$v[n] = \left[ 1 \quad e^{-j2\pi n/N_1} \quad \ldots \quad e^{-j2\pi (N_1-1)n/N_1} \right]^T.$$

The effective received signal for packet $m$ is, after frequency shifting [cf. (11)],

$$\tilde{y}_m[n] = y_m[n] e^{j2\pi mn/L} = \sum_l K_m[l] v[n-l] s[r-l] + \tilde{z}_m[n],$$

where $\tilde{K}_m[n] = K_{m0}[n] e^{j2\pi mn/L}$. Note that the effective channel (12) is a periodically varying MIMO-ISI channel with period $N_1$. We refer to each of the $N_1$ induced substreams as “phases.” As in the SISO case, we may employ a DFE at the receiver, but due to the time-varying nature of the effective channel, for each phase a different set of $N_t$ feedforward filters is applied to the channel output vector sequence. Thus, the equalizer is also periodic with period $N_1$.

Note the covariance matrix of the transmitted vector for our SNQ modulation is white. We conclude that for this modulation, the transmitted signal is white over all degrees of freedom as long as the oversampling rate satisfies $L \geq N_1 M$.

However, this doesn’t guarantee capacity can be achieved. In particular, we associate with each of the $N_1$ “phases” a signal-to-interference-plus-noise ratio (SINR) value corresponding to the associated DFE slicer input. Equivalently, we may associate with each such phase a corresponding capacity. Hence, while the sum of the per-phase capacities equals the white-input capacity of the MIMO channel, the per-phase capacities are in general not equal. Moreover, since the variation is unknown to the transmitter, in a Guess-Vanassari transmission architecture, a fixed code rate is used, and thus the achievable rate is determined by the minimum of the per-phase capacities.

It is worth emphasizing that this SINR variation across phases is analogous to the SINR variation across streams in a V-BLAST system, in which independently coded streams are sent over the antennas [5]. For this reason, V-BLAST serves as a useful benchmark with which to compare the performance of SNQ modulation.

A. Parallel channels

In some cases, MIMO-SNQ is strictly capacity achieving. For example, consider the special case of $N$ parallel ISI channels. This model is essentially equivalent to using SNQ modulation for transmission over a block-varying ISI channel as considered in Section III, with the $N$ parallel channels replacing the $L$ consecutive blocks of the SISO ISI channel.

As we have shown that SNQ modulation is an optimal scheme in such a scenario, it follows that MIMO-SNQ is optimal for the case of parallel channels.

B. Channels Without Temporal ISI

As another class of channels of interest, consider the special case in which there is no temporal ISI and only inter-channel interference (ICI) is present, i.e., $H_m(t) = H_m$.

While for diagonal $H_m$ MIMO-SNQ is capacity achieving, there exist other $H_m$ for which MIMO-SNQ achieves zero rate. For example, if $H_m$ is the (rank-one) matrix of all 1’s, there will exist an SNQ Nyquist substream that experiences a zero-capacity channel since a vector of all 1’s is orthogonal to $v[n]$ for $n \neq 0, N_1, \ldots, l$. Hence, MIMO-SNQ achieves zero rate, while V-BLAST achieves a strictly positive rate. However, for all but such pathological $H_m$, MIMO-SNQ supports a rate that grows with SNR. By contrast, it is well known that for all rank-one channel matrices V-BLAST performance is interference-limited, i.e., is bounded with increasing SNR.

More generally, when the channel matrix $H_m$ is drawn from a random ensemble, its performance is on average never worse that V-BLAST, and has significant advantages, particularly when keeping in mind that in V-BLAST an ordering is forced in the detection process, while MIMO-SNQ requires no such ordering since the scheme is inherently more symmetric.

We consider the average throughput for an ensemble of $H_m$ with i.i.d. circularly symmetric complex Gaussian entries. The resulting average throughput of MIMO-SNQ is depicted in Fig. 1, along with that for both fixed- and optimized-decoding-order V-BLAST. As the plot reflects, the performance
of MIMO-SNQ modulation lies in between the two and approaches the latter at high SNR.

We can also relate MIMO-SNQ performance to that of D-BLAST [6]. In particular, as is well known, D-BLAST can achieve capacity, but to do so requires a base code designed for time-varying channels. In that sense, MIMO-SNQ can also achieve capacity provided practical such base codes exist. However, when we require that a communication architecture has the property that the base code sees an AWGN channel, Fig. 1 reflects that MIMO-SNQ can perform as well as V-BLAST with an optimized decoding order. In contrast, when the same constraint is imposed on D-BLAST, the result is V-BLAST with a fixed decoding order.

C. Spatio-Temporal ISI Channels

When there is also temporal ISI, MIMO-SNQ is even more attractive, as we next illustrate. We now consider a random MIMO channel model where the Nyquist-rate equivalent discrete-time matrix channel impulse response $K_m[n]$ is of finite length and each Nyquist-rate tap is drawn i.i.d. over spatial and time dimensions according to a circularly-symmetric complex Gaussian distribution. Fig. 2 depicts the expected (averaged over the ensemble) capacity of MIMO-SNQ for different channel lengths. We observe that the gap-to-capacity decreases as the channel length grows confirming that SNQ modulation is able to exploit the temporal diversity afforded by the channel.

Moreover, such behavior is not specific to such i.i.d. ensembles. Indeed, Table I describes the performance of MIMO-SNQ for a typical $2 \times 2$ underwater acoustic communication channel realization from the recent KAM-11 experiment. Performance was numerically evaluated for an oversignaling rate of $L = 2$. In this case, the equivalent discrete-time (Nyquist-rate) baseband channel impulse responses are 100 taps long. As the table reflects, MIMO-SNQ is effectively capacity achieving for this channel, with a gap to capacity of less than 0.5 dB.

REFERENCES

Successive Compute-and-Forward

Bobak Nazer
Boston University
ECE Department
Boston, MA 02215, USA
Email: bobak@bu.edu

Abstract—In prior work, we proposed the compute-and-forward framework for sending linear combinations of messages to relays. In this note, we extend the notion of successive interference cancellation to the compute-and-forward setting. We find that once a relay has decoded a linear combination, it can mix it with its channel output to create a new effective channel output. The resulting effective channel can be tuned so that it is more suitable for decoding a second linear combination than the original channel.

I. INTRODUCTION

The classical approach to communication over a wireless relay network treats interference between transmitters as a nuisance to be avoided. Typically, each relay observes a noisy linear combination of the transmitted codewords and attempts to decode one (or more) of them while treating the others as additional noise. Recent efforts have revealed that interference can in fact be exploited if we are willing to expand the set of decoding possibilities to include linear combinations of codewords. One natural approach, sometimes referred to as compute-and-forward, is to employ a lattice codebook so that integer combinations of codewords are themselves codewords [1]–[5]. Relays are then free to select integer coefficients that match the channel coefficients as closely as possible, thus reducing the effective noise and increasing the achievable rates.

Under the classical approach, a relay can employ successive interference cancellation to remove decoded codewords from its channel observation. This decreases the effective noise encountered in the next decoding step. In this paper, we devise an analogous technique for the compute-and-forward framework. After decoding a linear combination, a relay can combine it with its channel observation to obtain a new effective channel that is even better for decoding the next linear combination. For ease of exposition, we will focus on the case where each relay wants to recover just two linear combinations.

Owing to space limitations, we do not attempt a full survey of the literature. We refer interested readers to [2] for additional references pertaining to compute-and-forward and to [6], [7] for surveys of the closely related topic of physical-layer network coding.

II. PROBLEM STATEMENT

Our setting is nearly identical to that of [2] and we reproduce some of the key definitions below. For ease of exposition, we will limit ourselves to real-valued channels and symmetric rates. We will denote addition and summation over \( \mathbb{R} \) with \( + \) and \( \sum \), respectively, and use \( \oplus \) and \( \oplus \) to denote the same over \( \mathbb{F}_p \).

Each transmitter (indexed by \( \ell = 1, \ldots, M \)) has a length-\( k \) message that is drawn independently and uniformly over a prime-sized finite field, \( w_\ell \in \mathbb{F}_p \). An encoder, \( E_\ell : \mathbb{F}_p^k \rightarrow \mathbb{R}^n \), then maps the message into a length-\( n \) codeword, \( x_\ell = E(w_\ell) \), which must satisfy the usual power constraint \( \|x_\ell\|^2 \leq nP \). The message rate is \( R = (k/n) \log p \).

Each relay (indexed by \( m = 1, \ldots, M \)) observes a noisy linear combination of the codewords,

\[
y_m = \sum_{\ell=1}^L h_{ml} x_\ell + z_m, \tag{1}
\]

where the \( h_{ml} \in \mathbb{R} \) are the channel coefficients and \( z_m \sim \mathcal{N}(0, I) \) is i.i.d. Gaussian noise. Let \( h_m = [h_{m1} \ldots h_{mL}]^T \) denote the vector of channel coefficients. The goal is for each relay to recover two linear combinations of the messages of the form

\[
\mathbf{u}_m^{(1)} = \bigoplus_{\ell=1}^L q_{ml}^{(1)} \mathbf{w}_\ell \quad \mathbf{u}_m^{(2)} = \bigoplus_{\ell=1}^L q_{ml}^{(2)} \mathbf{w}_\ell \tag{2}
\]

where the \( q_{ml}^{(1)}, q_{ml}^{(2)} \in \mathbb{F}_p \) are finite field coefficients. To this end, each relay is equipped with a decoder, \( \mathcal{D} : \mathbb{R}^n \rightarrow \mathbb{F}_p^k \times \mathbb{F}_p^k \), that produces estimates \( \hat{\mathbf{u}}_m^{(1)} \) and \( \hat{\mathbf{u}}_m^{(2)} \) of its desired linear combinations. We will say that the average probability of error is at most \( \epsilon \) if

\[
P \left( \bigcup_m \{ \hat{\mathbf{u}}_m^{(1)} \neq \mathbf{u}_m^{(1)} \} \cup \{ \hat{\mathbf{u}}_m^{(2)} \neq \mathbf{u}_m^{(2)} \} \right) < \epsilon. \tag{3}
\]

To map between the realized-valued linear combination provided by the channel and the desired finite field linear combinations, we will need a bit of additional nomenclature. Specifically, we will refer to \( \hat{\mathbf{u}}_m^{(1)} \) as the linear combination with coefficient vector \( a_m = [a_{m1} \ldots a_{mL}]^T \in \mathbb{Z}^L \) if its finite field coefficients satisfy

\[
q_{ml}^{(1)} = [a_{ml}] \mod p. \tag{4}
\]

\footnote{This is a slight abuse of notation. More formally, we should explicitly define a mapping between \( \mathbb{F}_p \) and \( \{0, 1, \ldots, p-1\} \). See [2, Definition 6] for more details.}
Similarly, we will refer to $u^{(2)}_m$ as the linear combination with coefficient vector $b_m = [b_{m1} \cdots b_{mL}]^T \in \mathbb{Z}^L$ if
\[ q^{(2)}_m = [b_{m1} \mod p] \mod p. \] (5)

We will say that the computation rate region $R(h_m, a_m, b_m)$ is achievable if, for any $\epsilon > 0$ and $n$ large enough, there exist encoders and decoders, such that all relays can recover their desired linear combinations with average probability of error $\epsilon$ as long as
\[ R < \min_m R(h_m, a_m, b_m). \] (6)

Note that the relays are free to choose which linear combinations to decode so long as (6) is satisfied.

III. NESTED LATTICE CODES

One key requirement of our scheme is that all integer combinations of codewords must be afforded protection against noise. Nested lattice codes are a natural fit for this purpose. A lattice $\Lambda$ is a discrete subgroup of $\mathbb{R}^n$ with the property that if $t_1, t_2 \in \Lambda$ then $t_1 + t_2 \in \Lambda$. By construction, all of our lattices will contain the zero vector. A pair of lattices $\Lambda, \Lambda_{\text{FINE}}$ is nested if $\Lambda \subset \Lambda_{\text{FINE}}$.

A lattice quantizer is a function, $Q_{\Lambda} : \mathbb{R}^n \to \Lambda$, that maps vectors to the nearest lattice point in Euclidean distance,
\[ Q_{\Lambda}(x) = \arg \min_{t \in \Lambda} \|x - t\|_2. \] (7)
The fundamental Voronoi region is the subset of points in $\mathbb{R}^n$ that quantize to the zero vector, $\mathcal{V} = \{x : Q_{\Lambda}(x) = 0\}$. The modulo operation returns the quantization error with respect to the lattice,
\[ \lfloor x \rfloor \mod \Lambda = x - Q_{\Lambda}(x), \] (8)
and satisfies the distributive law,
\[ \lfloor ax + by \rfloor \mod \Lambda = \lfloor ax \rfloor + \lfloor by \rfloor \mod \Lambda, \] for any $a, b \in \mathbb{Z}$.

A nested lattice code $C$ is created by taking the set of fine lattice points that fall within the fundamental Voronoi region of the coarse lattice, $C = \Lambda_{\text{FINE}} \cap \mathcal{V}$. Erez and Zamir have shown that their exist nested lattice codes that can approach the capacity of a point-to-point Gaussian channel [8].

IV. COMPUTE-AND-FORWARD

In [2], we proposed the compute-and-forward framework as a way of communicating linear combinations of messages. Our focus was on the case where each relay decodes a single linear combination with coefficient vector $a_m$. Define
\[ R_{\text{CF}}(h, a) \triangleq \frac{1}{2} \log^+(\left(\|a_m\|^2 - \frac{(h_m^T a_m)^2 P}{1 + P\|h_m\|^2}\right)^{-1}) \] where $\log^+(x) \triangleq \max(\log(x), 0)$.

**Theorem 1** ([2, Theorem 2]): For any set of channel vectors $h_m \in \mathbb{R}^L$ and coefficient vectors $a_m \in \mathbb{Z}^L$, the following computation rate region is achievable:
\[ R(h_m, a_m) = R_{\text{CF}}(h_m, a_m). \] (9)

We now provide a brief overview of the basic compute-and-forward encoding and decoding functions which will be useful in the proof of our main result. Using Construction A [8], [9], we select a pair of nested lattices $\Lambda \subset \Lambda_{\text{FINE}}$ that can approach the capacity of a point-to-point Gaussian channel. It can be shown that there is a one-to-one map $\phi$ between $\mathbb{R}^L$ and the nested lattice code $C$ that preserves linearity (see [2, Lemma 5]). Using this mapping, the encoder chooses a lattice point $t_\ell = \phi(w_\ell)$. It then applies a dither $d_\ell$ that is drawn independent and uniformly over $\mathcal{V}$ and transmits the result,
\[ x_\ell = [t_\ell - d_\ell] \mod \Lambda. \] (10)

Relay $m$ observes $y_m$ and has access to every dither $d_\ell$. It scales its observation by the minimum mean-squared error (MMSE) coefficient
\[ \alpha_m = \frac{P h_m^T a_m}{1 + P\|h_m\|^2} \] (11)
and removes the dithers according to the desired coefficients $a_{m \ell}$. Afterwards, it quantizes the result onto the fine lattice and takes the modulus with respect to the coarse lattice,
\[ \hat{v}^{(1)}_m = \left[ Q_{\Lambda_{\text{FINE}}} (a_m y_m + \sum_{\ell=1}^L a_{m \ell} d_\ell) \right] \mod \Lambda. \] (12)

It can be shown that, with high probability, this is equal to
\[ v^{(1)}_m = \left[ \sum_{\ell=1}^L a_{m \ell} t_\ell \right] \mod \Lambda \] (13)
so long as
\[ R < \min_m \frac{1}{2} \log^+(\frac{P}{\alpha_m^2 + P\|a_m h_m - a_m\|^2}). \] (14)

Finally, the relay applies the inverse map to get its estimate $\hat{u}^{(1)}_m = \phi^{-1}(\hat{v}^{(1)}_m)$. Assuming that $\hat{v}^{(1)}_m = v^{(1)}_m$, it can be shown that $\hat{u}^{(1)}_M = u^{(1)}_M$ (see [2, Lemma 6]).

V. SUCCESSIVE COMPUTE-AND-FORWARD

Successive interference cancellation is a powerful technique for decoding several messages at a single receiver. Assume that, given the channel observation $y_m$, a relay has correctly decoded $x_i$. It can now completely remove the effect of $x_i$ from its observation,
\[ y_m - h_m x_i = \sum_{\ell \neq i} h_{m \ell} x_\ell + z_m, \] (15)
which reduces the interference and makes it easier to decode the next codeword.

As it turns out, we can employ a similar technique when decoding several linear combinations. Assuming the relay has decoded $\sum_{\ell} a_{m \ell} x_\ell$, it can create a new effective channel
\[ y_m + \gamma_m \sum_{\ell=1}^L a_{m \ell} x_\ell = \sum_{\ell=1}^L (h_{m \ell} + \gamma_m a_{m \ell}) x_\ell + z_m. \] (16)

2These dithers can be replaced with deterministic sequences.
By adjusting the effective channel coefficients, we can make it easier for the relay to decode its second linear combination, and thus increase the computation rate region. Note that unlike successive interference cancellation, it is not always optimal to subtract the recovered linear combination. Below, we develop a successive computation scheme that follows the concept outlined above. We begin by showing that we can always recover the real sum of codewords if we have access to the modulo sum and the dithers.

Remark 1: In [2, Theorem 12], we described a limited version of successive computation. The key drawback is that this scheme only allows for integer-valued $\gamma_m$, owing to the fact that it works directly with the modulo sum of codewords. As part of the compute-and-forward scheme, the relay recovers an estimate $\hat{v}_m^{(1)}$ of the modulo linear combination of codewords $v_m^{(1)}$ from (34). The lemma below shows that this modulo sum can be used to recover the real sum $\sum_{\ell=1}^L a_m x_\ell$ that is needed for successive computation.

**Lemma 1:** The relay can make an estimate $\hat{s}_m$ of the real sum of codewords

$$s_m = \sum_{\ell=1}^L a_m x_\ell$$

with vanishing probability of error, $\lim_{n \to \infty} P(\hat{s}_m \neq s_m) = 0$, so long as $R < R(h_m, a_m)$.

**Proof:** Since $R < R(h_m, a_m)$, we can use Theorem 1 to make an estimate $\hat{v}_m^{(1)}$ that is equal to $v_m^{(1)}$ with high probability. For the remainder of the proof, we will assume that this estimate is correct. First, the relay dithers this linear combination and takes the modulus with respect to $\Lambda$. This gives it access to a modulo combination of the dithered lattice points,

$$\left[ \hat{v}_m^{(1)} - \sum_{\ell=1}^L a_m d_\ell \right] \mod \Lambda$$

$$= \left[ \sum_{\ell=1}^L a_m x_\ell \right] \mod \Lambda - \sum_{\ell=1}^L a_m d_\ell \mod \Lambda$$

$$= \left[ \sum_{\ell=1}^L a_m x_\ell \right] \mod \Lambda \mod \Lambda.$$  

It then combines this quantity with $a_m y_m$,

$$r_m = a_m y_m - \left[ \sum_{\ell=1}^L a_m x_\ell \right] \mod \Lambda$$

$$= \sum_{\ell=1}^L a_m h_\ell x_\ell + a_m z_m - \sum_{\ell=1}^L a_m x_\ell + Q_\Lambda \left( \sum_{\ell=1}^L a_m x_\ell \right)$$

$$= Q_\Lambda \left( \sum_{\ell=1}^L a_m x_\ell \right) + a_m z_m + \sum_{\ell=1}^L (a_m h_\ell - a_m x_\ell) x_\ell,$$

to get a quantized version of the desired sum $s_m$ plus some effective noise with variance $a_m^2 + P||a_m h_m - a_m||^2$. To remove this noise, it applies the coarse lattice quantizer. This operation will be successful with high probability so long as the second moment of $\Lambda$ exceeds the effective noise variance, i.e., $P > a_m^2 + P||a_m h_m - a_m||^2$. Assuming that $R(h_m, a_m) > 0$, this condition holds and we have that

$$Q_\lambda(r_m) = Q_\Lambda \left( \sum_{\ell=1}^L a_m x_\ell \right) \mod \Lambda$$

with high probability. Finally, since the relay knows the quantized sum as well as its quantization error, it can infer the desired real sum. Assuming that (21) holds, we have that

$$s_m = Q_\lambda(r_m) + \sum_{\ell=1}^L a_m x_\ell \mod \Lambda$$

$$\sum_{\ell=1}^L a_m x_\ell = s_m.$$  

Using the real sum of the codewords, we can construct a successive compute-and-forward scheme. Define

$$\mathbb{R}_{SCF}(h, a, b) \triangleq \frac{1}{2} \log^+ \left( \frac{||b||^2 - ||a||^2}{||\epsilon||^2} - \frac{P \left( \frac{||b||^2 - ||a||^2}{||\epsilon||^2} \right)}{1 + P \left( \frac{||b||^2 - ||a||^2}{||\epsilon||^2} \right)} \right)$$

**Theorem 2:** For any set of channel vectors $h_m \in \mathbb{R}^L$ and coefficient vectors $a_m, b_m \in \mathbb{Z}^L$, the following computation rate region is achievable:

$$R(h_m, a_m, b_m) = \max \left( R_{AB}, R_{BA} \right)$$

$$R_{AB} = \min \left( \mathbb{R}_{SCF}(h_m, a_m), \mathbb{R}_{SCF}(h_m, b_m, a_m) \right)$$

$$R_{BA} = \min \left( \mathbb{R}_{SCF}(h_m, b_m, a_m), \mathbb{R}_{SCF}(h_m, b_m, a_m) \right).$$

**Proof:** Fix an $\epsilon > 0$. The expressions $R_{AB}$ and $R_{BA}$ correspond to the two possible decoding orders. We will prove that $R_{AB}$ is achievable by first decoding the linear combination with coefficient vector $a_m$ and then that with $b_m$. The proof of $R_{BA}$ follows identically by exchanging the roles of $a_m$ and $b_m$.

We employ the same encoding framework as in Theorem 1. Relay $m$ uses the same decoding framework to make an estimate $\hat{u}_m^{(1)}$ of $u_m^{(1)}$. For $n$ large enough, this estimate is incorrect with probability at most $\epsilon/3$ if

$$R < \min_m \mathbb{R}_{SCF}(h_m, a_m).$$

As a byproduct of successful decoding, the relay will obtain a correct estimate $\hat{v}_m^{(1)}$ of $v_m^{(1)}$. Using Lemma 1, it makes an estimate $\hat{s}_m$ of $s_m = \sum_{\ell=1}^L a_m x_\ell$ that has probability of error at most $\epsilon/3$ for $n$ large enough. Below, we assume $\hat{s}_m = s_m$.

The relay removes the projection of $\hat{s}_m$ onto $y_m$ from $y_m$ to get

$$y_m = y_m - \frac{a_m^T h_m}{||a_m||^2} s_m$$

$$= \sum_{\ell=1}^L \left( h_m x_\ell - \frac{a_m^T h_m}{||a_m||^2} a_m x_\ell \right) x_\ell + z_m.$$
Define $g_m = h_m a_m^T a_m$ and $g_m = [g_m_1, \ldots, g_m_L]^T$. It then forms a new effective channel observation

$$
\tilde{y}_m = \beta_m y_m + \mu_m z_m 
$$

and proceeds to decode the linear combination with coefficient vector $b_m$ as in Theorem 1. Specifically, it forms the estimate

$$
\hat{v}_m^{(2)} = [Q_{\text{max}}(\tilde{y}_m + \sum_{\ell=1}^L b_m \ell t_\ell)] \mod \Lambda.
$$

It can be shown that, with probability of error at most $\epsilon/3$, this is equal to

$$
\hat{v}_m^{(2)} = \left[ \sum_{\ell=1}^L b_m \ell t_\ell \right] \mod \Lambda
$$

so long as

$$
R < \min_m \frac{1}{2} \log^+ \left( \frac{P}{\beta_m^2 + P \|b_m\|^2} \right).
$$

Finally, the relay applies the inverse map to get its estimate $\hat{u}^{(2)}_m = \phi^{-1}(\hat{v}^{(2)}_m)$. It remains to solve for the $\beta_m$ and $\mu_m$ that minimize the effective noise variance

$$
f(\beta_m, \mu_m) = \beta_m^2 + P \|b_m\|^2 + \mu_m a_m - b_m \|^2.
$$

This is a convex function whose global minimum is attained at

$$
\beta_m^* = \frac{P \text{tr}(b_m)}{1 + P \|b_m\|^2}
$$

and

$$
\mu_m^* = \frac{a_m^T b_m}{\|a_m\|^2}.
$$

Plugging this back in, we find that

$$
f(\beta_m^*, \mu_m^*) = P \|b_m\|^2 - \frac{P (a_m^T b_m)^2}{\|a_m\|^2} - \frac{P^2 (g_m^T b_m)^2}{1 + P \|g_m\|^2}.
$$

Substituting this into (35), we get the desired condition

$$
R < \min_m R_{\text{SCF}}(h_m, a_m, b_m).
$$

By the union bound, the probability of error is at most $\epsilon$. \hfill \blacksquare

**Example 1:** Consider a single relay with channel vector $h_1 = [2 \ 1 \ 1]^T$ that wishes to decode the linear combinations with coefficient vectors $a_1 = [1 \ 1 \ 1]^T$ and $b_1 = [1 - 1 - 1]^T$ using Theorem 2. It is not possible to decode $b_1$ first as $R_{\text{SCF}}(h_1, b_1) = 0$. Decoding $a_1$ first requires

$$
R < \frac{1}{2} \log^+ \left( \frac{1 + 6P}{3 + 2P} \right).
$$

After recovering $a_1$, the relay can adjust the channel and decode $b_1$ so long as

$$
R < \frac{1}{2} \log^+ \left( \frac{9}{24} + \frac{P}{4} \right).
$$

The example above demonstrates that successive compute-and-forward can make it possible to recover linear combinations that are not available via a direct application of the original compute-and-forward framework. In other words, the relay can first target a linear combination that is “easy” to decode and then use it to create a better effective channel for decoding the second linear combination.

From another perspective, successive compute-and-forward can be used to enlarge the computation rate region for decoding a single linear combination with coefficient vector $b_m$. The relay should order all viable coefficient vectors (i.e., those satisfying $\|a_m\|^2 \leq 1 + P \|b_m\|^2$) by computation rate $R_{\text{SCF}}(h_m, a_m)$ and set aside those $a_m$ with rates larger than $R_{\text{SCF}}(h_m, b_m)$. It can then calculate which pair $(a_m, b_m)$ offers the highest rate using Theorem 2. Finally, it applies successive compute-and-forward for this pair and keeps only the second equation. Example 1 demonstrates that this procedure does indeed enlarge the rate region.

VI. GENERALIZATIONS AND EXTENSIONS

Following the framework in [2], successive compute-and-forward can be generalized to include complex-valued channel models as well as unequal message rates. One can also envision extending this technique to the case where each relay may want more than two linear combinations. In this case, the linear combinations obtained thus far should be mixed together with the original channel observation to create a new effective channel for the next targeted linear combination.

**REFERENCES**


Tunable Sparse Network Coding

Soheil Feizi1, Daniel E. Lucani2, Muriel Médard1

1Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA
2Instituto de Telecommunicações, DEEC, Faculdade de Engenharia da Universidade do Porto, Porto 4200-465, Portugal
Email:sfeizi@mit.edu; dlucani@fe.up.pt; medard@mit.edu

Abstract—A fundamental understanding of the relationship between delay performance and complexity in network coding is instrumental towards its application in practical systems. The main argument against delay-optimal random linear network coding (RLNC) is its decoding complexity, which is $O(n^3)$ for $n$ original packets. Fountain codes, such as LT and Raptor codes, reduce the decoding load on the receiver but at the cost of introducing additional, non-negligible delay. The source of this increased delay is the inherent sparsity of the code, which significantly reduces the impact of a new coded packet, i.e., the probability of a packet to be linearly independent from the knowledge at the receiver, as we approach the end of the transmission (when few degrees of freedom are missing). Thus, the additional overhead is mainly due to the transmission of the last data packets. Our key observation is that switching gears to denser codes as the transmission process progresses could considerably reduce the delay overhead while maintaining the complexity advantages of a sparse code. We propose tunable sparse network coding as a dynamic coding mechanism with a code structure with two regions: a sparse region, with various levels of sparsity, and a dense region, where packets are generated as per RLNC. We characterize the problem in multicast sessions on general networks and illustrate trade-offs in special cases of interest. We also present a novel mechanism to perform efficient Gaussian elimination for sparse matrices that guarantees linear decoding complexity with high probability.

I. INTRODUCTION

Network coding constitutes a disruptive concept for the operation of communication networks [1] that has evolved from an information theoretic result for achieving capacity in multicast to become an enabler of a wide variety of applications, including communications in ad-hoc networks, data gathering in sensor networks, and data storage. One of the key features of network coding is that it encourages the mixing (coding) of data packets at intermediate nodes, rather than limiting intermediate nodes to store and forward packets. Under this premise, it is no longer required for the system to keep track of which packets have been received: receivers need only accumulate enough coded packets in order to recover the information. Ref. [2] and [3] show that linear codes over a network are sufficient to achieve multicast capacity, while Ref. [4] proved that randomly generated linear codes achieve capacity with high probability.

One of the main arguments against the use of RLNC is that its decoding complexity is $O(n^3)$, where $n$ constitutes the number of original packets. Sparse end-to-end erasure correcting codes, such as LT [5] and Raptor codes [6], reduce the decoding load on the receiver but at the cost of introducing additional, non-negligible delay. More importantly, they lack RLNC’s capability to use re-encode at intermediate nodes. Sparse end-to-end erasure correcting codes tend to choose sparsity in a static manner. Although this sparsity helps to reduce the complexity, it also introduces additional delay overheads. In particular, this inherent (and static) sparsity reduces the impact of a coded packet as the transmission session progresses. In fact, the additional overhead of these codes is mainly due to the transmission of the latter packets.

We revisit the idea of using sparse coding to reduce the decoding complexity, but using a fundamentally different approach in which i) we tune the level of sparsity as the transmission process evolves, and ii) we transition to RLNC for the last transmissions. This tuning process can reduce the delay overhead by using denser codes towards the end of the transmission, while maintaining complexity advantages of a sparse code. We call this scheme Tunable Sparse Network Coding. Unlike RLNC, in tunable sparse network coding the coding coefficients are chosen sparsely, i.e., mostly zero. As the session progresses, these coefficients become denser, i.e., statistical properties of the coding coefficients vary in time.

There are three key ideas in tunable sparse network coding:

• Sparse coding is more beneficial at the beginning of a transmission session. Since destination nodes have only received a few degrees of freedom (independent linear combination of the original packets), any sparsely coded packet will constitute a new degree of freedom (or innovative packet) with high probability.

• Dense coding is required towards the end of the sessions for fast completion because dense coded packets are innovative with high probability, while sparse coded packets are innovative with low probability.

• Sparse coding translates in reduced complexity. A decoder can exploit an encoding matrix with a large number of zero coefficients for improved complexity.

This paper proposes a framework for studying our network coding mechanism over packet erasure networks to analyze the trade-off between the delay performance (i.e., mean number of transmissions) and the decoding complexity of multicast sessions. Finally, we introduce a simple, yet powerful algorithm to organize a sparse matrix in order to provide a linear decoding complexity with high probability.

II. MODEL AND PRELIMINARIES

Consider a general multicast network with one source and several receivers. We model the network as a directed graph, with independent edge erasure probabilities. This assumption may be relaxed in future studies, but it is a fair assumption in many networks. Suppose the source node has $n$ packets to transmit to a subset of the nodes in the network, i.e., a multicast session. Suppose $X$ is a $n \times 1$ vector of source packets whose $i^{th}$ component, $X_i$, represents the $i^{th}$ source packet in the finite field $\mathbb{F}_q$. Suppose the minimum min-cut between
each receiver and source is \( m \). Therefore, the network can deliver \( m \) packets at each transmission round. A transmission session is complete when all destination nodes receive \( n \) source packets. Throughout this paper, we assume \( m \ll n \). Hence, multiple transmission rounds of \( m \) coded packets are required in a session. Each transmission round is indexed by \( t \).

Let us leverage some of the ideas from the algebraic framework for linear network coding from Ref. [3] to model our problem. For the case of no erasures, the receiver \( i \) gets \( A_iX \), where \( A_i \) is a \( m \times n \) full-rank matrix (since the min-cut rate is \( m \)) after each round. \( A_i \) is called the network coding matrix associated with the receiver \( i \) and can be decomposed into two matrices \( B \) and \( G_i \) (i.e., \( A_i = GB \)). \( B \) is a \( m \times n \) pre-coding matrix which is applied at the source node. This matrix is the same for all receivers. \( G_i \) is a \( m \times n \) matrix representing the effect of performing network coding at intermediate nodes from the source node to the receiver \( i \).

Now, consider the case of having packet erasures over the network links. Since some packets are lost as they traverse the network, the min-cut rate of this network is smaller than the error-free min-cut rate of the network (i.e., \( m \)). For receiver \( i \), define a \( m \times m \) diagonal matrix \( P_i \), whose \( j \)-th component on the diagonal (i.e., \( P_{ij} \)) is zero with probability \( p_{ji} \), and one with probability \( 1 - p_{ji} \). Probabilities \( p_{ji} \) depend on edge erasure probabilities. Hence, at the end of each transmission round, the destination node \( i \), receives \( P_iA_iX \). In other words, some of coded packets are erased over the network. Considering multiple transmission rounds, \( Y(t) = P_tA_tX \) represents the received coded packets at receiver \( i \) in round \( t \).

### III. Tunable Sparse Network Coding

In this section, we introduce a framework for studying a time-varying coding scheme, with particular emphasis on the proposed case of tunable sparse network coding. A coded packet is uniformly coded when its coding coefficients are uniformly chosen from a finite field \( F_q \). On the other hand, a coded packet is \( k \)-sparse if it contains exactly \( k \) non-zero coding coefficients. Non-zero coefficients are chosen uniformly from \( F_q - \{0\} \). Finally, a coded packet is \( k \)-sparse with high probability (w.h.p.) when, with probability \( 1 - k/n \), each coding coefficient is zero, otherwise, it is chosen uniformly from \( F_q - \{0\} \). For simplicity, consider that \( q \) is sufficiently large, although our techniques extend naturally to small \( q \).

Define \( \rho(A_i(t)) \), the density of a matrix \( A_i(t) \), as the ratio of its non-zero components. A matrix is dense if this ratio is high, e.g., RLNC’s \( A_i(t) \) is dense w.h.p. If the ratio is low, the matrix is sparse. In a tunable sparse network coding scheme, the density of \( A_i(t) \) is tuned depending on the transmission round index \( t \). The key idea is that, at the beginning of the transmission session, receivers have only a few coded packets making a sparsely coded packet innovative to the receivers w.h.p. However, towards the end of the transmission session, sparsely coded packets are innovative with low probability because destination nodes have received most of the required degrees of freedom (doF). Hence, \( \rho(A_i(t)) \) should be increased, i.e., densely coded packets should be transmitted.

Depending on the function \( \rho(A_i(t)) \) and its control method, we can classify various tunable sparse network coding schemes. If the density control of \( A_i(t) \) is performed without having feedback from receivers, we say the coding scheme is open-loop. Otherwise, it is a closed-loop scheme. By having different time dependencies of elements of \( A_i(t) \), a variety of coding schemes, such as window-based coding, sliding window-based codings, and generation-based encoding, can be characterized and studied with our framework.

This work focuses on an open loop approach with prior knowledge of the network’s erasure probabilities. In particular, we characterize the delay and complexity performance from the perspective of a receiver and for the special case where the error-free min-cut rate is one. However, the analysis extends seamlessly to (i) multiple receivers with an error-free min-cut rate \( m \) greater than one, and (ii) close-loop schemes, because our analysis relies on of characterizing receivers’ performance given the number and type of received coded packets.

#### A. From a Receiver’s Perspective

A tunable sparse network coding scheme with \( k \)-sparsely coded packets for \( m = 1 \) can be described as follows: the source node first transmits \( n \) uncoded packets, where some of these packets are lost due to erasures. These uncoded packets are followed by sparse packets. We shall study two different approaches: i) transmission of \( n_k \) \( k \)-sparse coded packets, i.e., with exactly \( k \) non-zero coefficients chosen at random, where each non-zero coefficient is chosen uniformly from \( F_q - \{0\} \), which we call deterministic density, and ii) transmission of \( n_k \) \( k \)-sparse coded w.h.p., which we call randomized density. Finally, \( n_{\text{anti}} \) uniformly coded packets are transmitted.

We analyze the effect of \( n_k \) on the delay performance and the decoding complexity for the general case of several levels of sparsity. Without loss of generality, we restrict our delay performance achievability result in Theorem 3 to the case of one type of sparsely coded packets, i.e., only one \( k \neq 1 \) for which \( n_k > 0 \). This result extends seamlessly to the case of different levels of sparsely coded packets.

#### B. Delay Performance

This subsection characterizes the probability that a newly received, sparse coded packet is innovative under specific density distributions. This problem is key to modeling the general problem of a multicast session over a general network, where at each transmission time \( t \) up to \( m \) sparse packets will arrive to a receiver. In the general network case, the choice of precoding and the coding within the network as well as the link erasures will determine the sparsity of the received coded packets for that time \( t \). Considering a scheme with multiple levels of sparsity, a transmission of \( g(d_k|np) \) sparse packets for
a level $k$ yields $d_k$ dof, when $np$ dof were previously received. The following lemma characterizes $g_d(d_k|np)$ and $gr(d_k|np)$ for deterministic and randomized sparsity, respectively.

**Lemma 1.** For a receiver in a multicast session with $np < n$ previously received degrees of freedom,

\[
g_d(d_k|np) = \sum_{i=0}^{n_p + d_k} \frac{1}{i!} \left( \frac{1}{\alpha \cdot 1 - \frac{k}{n}} \right)^i \left( 1 - \frac{k}{n} \right)^{n - i}, \quad \text{if } k < (1 - \alpha)n, \tag{1}
\]

\[
gr(d_k|np) = \sum_{i=0}^{n_p + d_k} \frac{1}{i!} \left( \frac{1}{k/n(1 - i/n)} \right)^i. \tag{2}
\]

**Proof:** First we compute $g_d(d_k|np)$ for the case of deterministic density and then point out the necessary changes for $gr(d_k|np)$.

Suppose at time $t$, we have $i$ dof at the receiver. For the case of deterministic density, a $k$-sparse coded packet is not innovative with probability $\left( \frac{1}{\alpha \cdot 1 - \frac{k}{n}} \right)^i$, which represents the probability of all the $k$ non-zero coefficients being chosen amongst packets that do not increase the knowledge space of the receiver. We model the reception of a new packet as the Markov chain shown in Figure 1. The expected number of required sparsely coded packets to have $d_k$ new dof, conditioned on having $np$ innovative packets at time $t$ is Eq. (1).

For the case of randomized density, a $k$-sparsely coded packet is not innovative with probability $(1 - \frac{k}{n})(1 - i/n)^n$ and again model this as a Markov chain to show Eq. (2).

The following corollary provides approximate expressions for Eq. (1) and (2) for $n$ is large enough and $d_k << n - np$.

**Corollary 2.** For large $n$ and $d_k << n - np$. If $(np/n)^k << 1$ then, $g_d(d_k|np) \approx d_k(1 + (1 - np/n)^k)$.

If $(1 - (k/n)(1 - np/n))^n << 1$ then, $gr(d_k|np) \approx d_k(1 + (1 - (k/n)(1 - np/n))^n)$.

**Proof:** Under conditions of Corollary 2, we have that

\[
g_d(d_k|np) \approx \sum_{i=0}^{n_p + d_k/n} \frac{n}{1 - x^k} dx \approx \int_{np/n}^{n_p + d_k/n} 1 + x^k dx \approx d_k(1 + (1 - np/n)^k).
\]

We use similar approximations for $gr(d_k|np)$.

Define a $(nk, nuni)$ tunable sparse network coding scheme as a scheme that transmits $n$ systematic packets, $nuni$ uniformly coded packets, and $nk$ $k$-sparse coded packets. This scheme is achievable if the receiver is expected to recover all $n$ packets. The following result characterizes achievability for an erasure probability $\alpha$ as seen by the receiver. For this result, define $f(d_k)$ as the number of sparse coded packets to have $d_k$ innovative ones in average. $f(d_k) - d_k$ represents the performance loss due to using sparse network coding instead of RLNC with large $q$.

**Theorem 3.** For a network with a single receiver and $m = 1$, a $(nk, nuni)$ tunable sparse network coding scheme is achievable where $nk = \frac{f(d_k)}{d_k}$ and $d_k + (1 - \alpha)nuni = \alpha n$. For a deterministic density of $k < (1 - \alpha)n$, $f(d_k) = g_d(d_k|1 - \alpha)n$.

For a randomized density, $f(d_k) = gr(d_k|1 - \alpha)n$.

---

**Fig. 2.** Example of proposed decoding algorithm (Algorithm 1)

**Proof:** Since the expected value of $np$ is $(1 - \alpha)n$ we compute having $f(d_k) = E[g_d(d_k|D_1)]$ and $f(d_k) = E[gr(d_k|D_2)]$ using Lemma 1. We conclude the proof following a typical argument for large $n$ since the distribution of $np$ is given by a binomial distribution with success probability $1 - \alpha$.

**C. Complexity Analysis**

We present a simple algorithm that reorders the matrix to avoid operations in the forward pass w.h.p. for the processing of sparse packets. This has the crucial effect of avoiding additional density to be created in sparse packets, which keeps the number of operations in the backward pass for the sparse packets as $O(d_k)$. Define $P_1$ as the $i$-th original packet, $C_j(Z)$ as the set of coded packets of $j$-sparse packets drawn from $Z$ a subset of the original packets, and $S_1$ as the set of packets in a $1$-sparse combination. Define $Sum(F, Z)$ as the set of original packet of subset $Z$ used only once in the set of coded packets $F$. Finally, we define $\Omega$ as the set of all $n$ original packets.

Fig. 2 (a) illustrates the case of a transmission using tunable sparse network coding. For simplicity, we have assumed that received coded packets were either uncoded, i.e., from the systematic region, or sparse, although for this example $n$ is kept small for illustration purposes. Fig. 2 (a) shows that packets $P_1$ and $P_2$ were received uncoded and we use them to substract their contribution from coded packets with a non-zero coefficient for $P_1$ and $P_2$. i.e., all non-zero elements of the first two columns. The algorithm counts the number of times a packet corresponding to columns on the lower right-hand side, as defined by the dashed lines, is involved in a linear combination. Involvement of the packet in combinations below the dashed line are not counted. Rows and columns shall be reorganized so that packets with a single occurrence will be in the upper left corner, as defined by the dashed lines in Fig. 2 (b). At this point we repeat the process until no single-occurrence packets are identified or until all sparse coded packets have been processed. In Fig. 2 (b), it is clear that the latter is attained because packet $P_3$ is involved in only one linear combination below the dashed line. The algorithm is stated in the following.

**Algorithm 1**

**STEP 0:** $R(0) = \{C_k(\Omega) : k > 1\}$, $C(0) = \{P_j : P_j \notin S_1\}$

**STEP i:** If $Sum(R(i), C(i)) \neq \emptyset$

$R(i+1) = R(i) - \{C_k(Sum(R(i), C(i)) : k > 1\}$, $C(i+1) = C(i) - Sum(R(i), C(i))$

$i \leftarrow i + 1$, Go to next step

Else Stop. Proceed with Standard Gaussian Elimination.
Theorem 4 shows that the complexity of decoding $d_k$ $k$-sparsely coded packets with a randomized density, given that $n_P$ innovative packets have already been ordered as per Algorithm 1 is $O(d_k)$.

**Theorem 4.** For tunable sparse network coding with randomized density with $d_k$ $k$-sparsely random packets when $n_P$ degrees of freedom have been processed before with Algorithm 1, and , the average number of operations to process the $d_k$ sparse packets using Algorithm 1 with probability $\prod_{l=0}^{d_k-1} \left(1 - \left(1 - \frac{k}{n}d_k - 1 - l\right)^{n_P - l}\right)$ is $c(n_P, d_k) = k d_k$

**Proof:** First, the probability of a packet being used only once in $j$ independent linear $k$-sparsely coded packets is $P_j = \left(1 - \frac{k}{n}d_k + 1 - l\right)^{n_P - l}$. Second, we process on a row by row basis such that there is a decreasing number of linear combinations from the original $d_k$. For the $r$-th processed row, the probability of having at least one packet being used only once is given by $\mathcal{P}(r) = 1 - \left(1 - \frac{k}{n}d_k - 1 - r\right)^{n_P - r}$. If all $d_k$ packets can be processed exclusively with the reordering algorithm, only $O(d_k)$ operations for reordering during the forward pass and $O(d_k)$ operations for the backward elimination pass are required. The probability of this event is $\prod_{l=0}^{d_k-1} \mathcal{P}(l)$. \hfill \llap{■}

**IV. PERFORMANCE EVALUATION**

This section illustrates the trade-off between complexity and delay performance via numerical examples and show that the threshold to switch from sparse to dense codes is key in determining this trade-off. For simplicity, we assume that no packets are sent uncoded at the beginning of the transmission. We compare the following schemes:

- **3-sparse Network Coding:** This scheme considers the transmission of $3$-sparse packets only. For determining the delay performance, we use Eq. (1) and use Algorithm 1 to determine complexity.

- **RLNC:** This constitutes a classical RLNC scheme with cubic complexity. In particular, we consider that $GE(n) = n^3/3 + n^2/2 - 5n/6$, which is a usual estimate of operations necessary to perform Gaussian elimination. For RLNC we assume that every incoming packet is innovative, which is a good approximation when using large field sizes.

- **Tunable Sparse Network Coding Scheme:** This scheme relies on both a sparse and a dense region, with a tunable threshold to decide when to switch from one to the other. Packets generated in the sparse region are $k$-sparse, with $k$ being incremented every time $n/100$ coded packets sent. Fig. 3 shows the performance of the three schemes as more packets are coming into the system. The right and top-most point for each curve constitutes the final pair of values for delay and complexity. Fig. 3 shows our proposed scheme for different threshold levels, namely when $1$, $2$, $5$, $10$, $50$ and $75\%$ of the degrees of freedom are missing at the receiver. We observe that i) switching too fast to RLNC is counterproductive from a complexity perspective, and ii) gains in terms of complexity become important when the switch to RLNC happens for the very last packets (See case of $1\%$). Finally, the case of switching for the last $1\%$ helps us confirm our initial intuition, namely that a sparse code’s delay performance is primarily degraded when few degrees of freedom are missing.

**V. CONCLUSIONS**

We revisit the problem of trading off delay performance and decoding complexity but providing a novel framework that uses both sparse and dense coded packets and allows coding at intermediate nodes for multicast sessions in general networks. More generally, we propose that the sparsity level should evolve as the transmission session progresses. To enable this coding flexibility, we proposed an efficient and intuitive algorithm for reordering rows and columns of a matrix in order to perform efficient Gaussian elimination in sparse matrices. This algorithm guarantees linear decoding complexity of sparse packets w. h. p.

Our approach can benefit considerably from the use of feedback in order to change the density of the code and studying efficient feedback mechanisms will be the focus of future work. Future work will also consider efficient and practical pre-coding mechanisms at the source for the case of single and multi source multicast to deal with specific network topologies and practical coding policies at intermediate nodes.

**VI. ACKNOWLEDGEMENTS**

This work was supported in part by the Air Force Office of Scientific Research (AFOSR) under award No. FA9550-08-1-0159 and by FCT project PEst-OE/EEI/LA0008/2011.

**REFERENCES**


The Flatness Factor in Lattice Network Coding: Design Criterion and Decoding Algorithm

Jean-Claude Belfiore
Department of Communications and Electronics
Telecom ParisTech, LTCI UMR 5141
46, rue Barrault, 75013 Paris, France
Email: belfiore@enst.fr

Cong Ling
Department of Electrical and Electronic Engineering
Imperial College London
London SW7 2AZ, United Kingdom
Email: cling@ieee.org

Abstract—In a recent work, Nazer and Gastpar proposed the compute-and-forward strategy as a physical-layer network coding scheme. They described a code structure based on nested lattices whose algebraic structure makes the scheme reliable and efficient. In a more recent paper, Niesen and Whiting revealed a fundamental limitation of the decoder used by Nazer and Gastpar. In this work, we consider maximum-likelihood decoding of compute-and-forward, aiming to overcome its limitation. By examining the decoding metric from the viewpoint of Gaussian measures over lattices, we present a design criterion of the lattice code based on the flatness factor, and propose a new decoding algorithm based on inhomogeneous Diophantine approximation.

I. INTRODUCTION

In [1], Zhang et al. introduced the physical-layer network coding concept (PNC) in order to turn the broadcast property of the wireless channel into a capacity boosting advantage. Instead of considering the interference as a nuisance, each relay converts an interfering signal into a combination of simultaneously transmitted codewords. PNC concept has received a particular interest in the last years because it provides means of embracing interference and improving network capacity.

In a recent work [2], Nazer and Gastpar proposed a new physical-layer network coding scheme. The proposed strategy, called compute-and-forward (CF), exploits interference to obtain higher end-to-end transmission rates between users in a network. The relays are required to decode noiseless linear equations of the transmitted messages using the noisy linear combination provided by the channel. The destination, given enough linear combinations, can solve the linear system for its desired messages. This strategy is based on the use of structured codes, particularly nested lattice codes to ensure that integer combinations of codewords are themselves codewords. The authors demonstrated its asymptotic gain using information-theoretic tools. Standard minimum Euclidean distance decoding was assumed.

The authors in [3] followed and generalized the framework of Nazer and Gastpar by introducing the Lattice Network Coding (LNC). They related the Nazer-Gastpar’s approach to the theorem of finitely generated modules over a principle ideal domain (PID). They gave sufficient conditions for lattice measures over lattices, we present a design criterion of the lattice code based on the flatness factor, and propose a new decoding algorithm based on inhomogeneous Diophantine approximation.

II. COMPUTE AND FORWARD

In our model, we consider one relay receiving messages from $k$ sources $S_1, \ldots, S_k$ and transmitting a linear combination of these $k$ messages. The relay observes a noisy linear combination of the transmitted signals through the channel. Received signal at the relay is expressed as,

$$y = \sum_{j=1}^{k} h_j x_j + w$$

where $h_j$ is the channel coefficient between source $S_j$ and the relay and $x_j$ is the vector transmitted by source $S_j$.

The relay searches to transmit a noiseless linear combination of the received signals with integer coefficients given by vector $a = [a_1 \ a_2 \ \cdots \ a_k]^T$. To achieve that goal, it decodes a noiseless linear combination of the transmitted vectors,

$$x_R = \sum_{j=1}^{k} a_j x_j$$
and retransmits it to the destination or another relay. We consider a complex-valued channel model with complex inputs and outputs. The channel coefficients \( h_i \) are complex, circular, i.i.d. Gaussian, \( h_i \sim \mathcal{N}(0, 1) \). \( z \) is Gaussian, zero mean, with variance \( \sigma^2 \) \((z \sim \mathcal{N}(0, \sigma))\). Let \( \mathbf{h} = [h_1 \ h_2 \ \cdots \ h_k]^T \) denote the vector of channel coefficients. Source vectors \( x_j \) are carved from a lattice code. The sources have no channel side information (CSI). CSI is only available at the relay.

III. THE ML DECODER

A. Solving the System of Diophantine Equations

The relay aims to decode a linear system of equations of the transmitted messages and passes it towards the destination. After calculating the vector \( a \) chosen following some given criterion, the relay recovers a linear combination of the i.i.d. Gaussian, equations. This system admits the following solutions.

\[
\lambda \triangleq \frac{1}{\sigma^2}\left(\mathbf{h}^H\mathbf{h}\right)^{\frac{1}{2}}
\]

where matrices \( \mathbf{V}_j \) are square \( n \times n \) matrices. Then the solution to the system of Diophantine equations is,

\[
x_j = s_j + v_j
\]

where \( v_j = M_j V_j B^{-1} \lambda \) and \( s_j \) is any vector in the lattice \( L_j \subset \Lambda_j \) of generator matrix \( M_j U_j \).

B. ML Decoding Metric

The ML decoder maximizes \( p(y/\lambda) \) over all possible values of \( \lambda \). The conditional probability \( p(y/\lambda) \) can be expressed as,

\[
p(y/\lambda) = \sum_{\sum_{j=1}^k a_j x_j = \lambda} p(x_1, \ldots, x_k) p(x_1, \ldots, x_k)
\]

where

\[
\sum_{a_j x_j = \lambda} p(x_1, \ldots, x_k) \propto e^{-\frac{1}{2\sigma^2} \|y - \sum_{j=1}^k h_j x_j\|^2}
\]

and \( x_1, \ldots, x_k \) are \( (a \text{ priori}) \) supposed equiprobable and given by (6). The decoding rule is now to find,

\[
\hat{\lambda} = \arg \max_{\lambda \in \Lambda} \varrho(\lambda),
\]

with

\[
\varrho(\lambda) = \sum_{q \in \mathcal{L}} e^{-\frac{1}{2\sigma^2} \|u(\lambda) - q\|^2}
\]

where \( u(\lambda) = y - \left(\sum_{j=1}^k h_j M_j V_j \right) B^{-1} \lambda \) and \( \mathcal{L} \) the \( n \)-dimensional \( \mathbb{Z}^k \)-lattice with generator matrix \( C \triangleq \sum_{j=1}^k h_j M_j U_j \). We observe that, for some values of the parameters, the function \( \varrho(\lambda) \) can be almost constant or flat, which will yield a high error probability. Figure III-B illustrate the behavior of the likelihood function for difference noise variances. Clearly, we should avoid the second and third cases where it is almost flat. In the next section, we will use the flatness factor to assess the behavior of the likelihood function.

IV. THE FLATNESS FACTOR

A. Definition

For a given lattice \( \mathcal{L} \) and a vector \( y \), we consider the \( n \)-variable function

\[
f_{\sigma, \mathcal{L}}(y) = \frac{1}{(2\pi\sigma^2)^{n/2}} \sum_{q \in \mathcal{L}} e^{-\frac{1}{2\sigma^2} \|y - q\|^2}
\]

which is a Guassian measure over a lattice. Obviously, \( \varrho(\lambda) \propto f_{\sigma, \mathcal{L}}(u(\lambda)) \). This function has been studied in [8]. We want here, to avoid that this function be flat for the lattice \( \mathcal{L} \).

Definition 1. [Flatness factor] The flatness factor of the Gaussian measure is defined as the maximum ratio \( f_{\sigma, \mathcal{L}}(y) \) can deviate from the uniform distribution:

\[
\epsilon_{\mathcal{L}}(\sigma) \triangleq \max_{y \in \mathbb{R}^n} \left| \frac{f_{\sigma, \mathcal{L}}(y) - E_y f_{\sigma, \mathcal{L}}(y)}{E_y f_{\sigma, \mathcal{L}}(y)} \right|.
\]

Note that this definition is slightly different from that in [5], for the sake of consistency with [9] where the problem of physical-layer security is addressed (flatness is desired in there). The following proposition gives the expression of the
flatness factor in terms of the theta series (the proof is given in [9]).

**Proposition 2.** [Expression of $\epsilon_L(\sigma)$]

$$\epsilon_L(\sigma) = \mu^n \Theta_L \left( \frac{1}{2\pi \sigma^2} \right) - 1 \quad (11)$$

where $\mu = \frac{V(L)^{\frac{1}{2}}}{2\pi \sigma^2}$ is the generalized signal-to-noise ratio, and $\Theta_L(z) = \sum_{\lambda \in \mathcal{L}} e^{-\pi i \langle \lambda, z \rangle^2}$ is the theta series of $\mathcal{L}$.

The flatness factor is closely related to the smoothing parameter studied in [8]. The advantage of the flatness factor is two-fold:

- It gives a precise characterization $\epsilon_L(\sigma)$ by the theta series, so it is more precise than the bounds available in [8].
- The smoothing parameter is mostly concerned with small values of $\epsilon_L(\sigma)$, while the flatness factor can handle both small and large values. The latter is of interest in coding applications.

**B. Design Criterion**

We need that the flatness factor of $\mathcal{L}$ be as large as possible to ensure that the maximum likelihood metric will be as different as possible from the likelihood metrics of the other points of the lattice $\mathcal{L}$. Formally, we have the following design criterion of the lattice $\mathcal{L}$:

**Proposition 3.** [Goodness for network coding] In order to have good decoding performance in compute-and-forward, the theta series $\Theta_L(z)$ should be large.

This requires from the lattice $\mathcal{L}$ to be as “bad” as possible in the standard sense of coding (i.e., not dense at all). The generator matrix of $\mathcal{L}$ is $\sum_{j=1}^k a_j M_j U_j$. But, since $\sum_{j=1}^k a_j M_j U_j = 0$, from the HNF, we deduce that we have to align vector $a$ with vector $h$ as much as possible.

As an illustration, Figure 2 shows the values of some flatness factors as a function of the generalized signal to noise ratio $\mu$. For our purpose, the Leech lattice is worse than the $\mathbb{Z}^n$ lattice.

**V. DECODING ALGORITHM**

We are now interested in an algorithm which must maximize $\vartheta(\lambda)$ in an efficient way. To do this, we need the Fourier transform and Jacobi theta functions. In fact, we will not attempt to compute $\vartheta(\lambda)$ itself; instead, we will propose inhomogeneous Diophantine approximation as a solution to ML decoding.

**A. Fourier transform and Jacobi theta function**

Since $f_{\sigma,\mathcal{L}}(y)$ is obviously periodic on $\mathcal{L}$, its Fourier transform is defined on the dual lattice $\mathcal{L}^*$. The Fourier coefficients are given by

$$f_{\sigma,\mathcal{L}}(q^*) = \frac{1}{V(\mathcal{L})} e^{-2\pi i \sigma^2 ||q^*||^2}, \quad q^* \in \mathcal{L}^*. \quad (12)$$

The Fourier expansion of $f_{\sigma,\mathcal{L}}(y)$ is then

$$f_{\sigma,\mathcal{L}}(y) = \frac{1}{V(\mathcal{L})} \sum_{q^* \in \mathcal{L}^*} e^{-2\pi i \sigma^2 ||q^*||^2} e^{2\pi i q^* \cdot y}. \quad (13)$$

Now we express the Gaussian measure as the Jacobi theta function (sometime referred to as the Riemann theta function in the multi-dimensional case), via the Fourier transform (13). Let $\mathbb{H}_n$ be set of symmetric square matrices whose imaginary part is positive definite. For an $n \times n$ matrix $T \in \mathbb{H}_n$ and an $n$-dimensional complex vector $z$, the multi-dimensional theta function is defined as

$$\theta(z; T) = \sum_{q \in \mathbb{Z}^n} e^{\pi i q^T T q + 2\pi i q^T z}. \quad (14)$$

Comparing with (13), we recognize that

$$f_{\sigma,\mathcal{L}}(y) = \frac{1}{V(\mathcal{L})} \theta(z; T) \quad (15)$$
for $z = C^{-1}y$ and $T = 2\pi i\sigma^2(C^TC)^{-1}$. Known results of the Jacobi theta function may be applied to analyze the Gaussian measure over lattices.

B. The 1–D Real Case

Here,

$$\varrho(\lambda) = \sum_{k \in \mathbb{Z}} e^{-\frac{(\lambda-k\pi\sigma)^2}{2\pi^2}}$$

for some given values of $\alpha$ and $\beta$. So, we have to study the function $f_{\sigma, z}(y)$ since, obviously, we have

$$\varrho(\lambda) \propto f_{\sigma, z}(y - \alpha\lambda).$$

We will show now that that ML decoding is equivalent to Diophantine approximation

$$\arg\max_{\lambda \in \Lambda} \varrho(\lambda) = \arg\min_{\lambda, k \in \mathbb{Z}} |y - \alpha\lambda - \beta k|. \quad (16)$$

To do this, we specialize (15) to the 1-D case:

$$f_{\sigma, z}(y) = \frac{1}{\sqrt{2\pi\sigma}} \sum_{k \in \mathbb{Z}} e^{-\frac{(y-k\pi\sigma)^2}{2\pi^2}} = \sum_{k \in \mathbb{Z}} e^{-2\pi^2\sigma^2 k^2} e^{2\pi ik\sigma}. \quad (17)$$

In the 1-D case, the Jacobi theta function is a function defined for two complex variables $z$ and $\tau$:

$$\theta(z; \tau) = \sum_{k=-\infty}^{\infty} e^{\pi i k^2 + 2\pi ikz} = 1 + 2\sum_{k=0}^{\infty} e^{\pi i k^2} \cos(2\pi k z)$$

(18)

where $z$ can be any complex number and $\tau$ is confined to the upper half-plane. Comparing (17) and (18), we recognize that

$$f_{\sigma, z}(y) = \theta(z; \tau), \quad \text{for} \quad z = y, \quad \tau = 2\pi i\sigma^2. \quad (19)$$

By using the Jacobi theta function, we have,

**Proposition 4.** $f_{\sigma, z}(y)$ is monotonically decreasing for $y \in (0, 1/2)$.

**Proof:** Let $r = e^{\pi i r}$ and $z = y$. We fix $\tau$ so that $\theta(y; \tau) = \theta(y/r; \tau)$ is a function of $y$ for $r = e^{2\pi i\sigma^2}$. Using the triple product formula of the Jacobi theta function [10]

$$\theta(y; \tau) = \prod_{m=1}^{\infty} \left(1 - r^{2m} \right) \left(1 + e^{2\pi i \tau} r^{2m-1} \right) \left(1 + e^{-2\pi i \tau} r^{2m-1} \right),$$

we obtain an alternative expression of the Gaussian measure

$$f_{\sigma, z}(y) = \prod_{m=1}^{\infty} \left(1 - r^{2m} \right) \left(1 + 2 \cos(2\pi y) r^{2m-1} + r^{4m-2} \right). \quad (21)$$

Obviously, $f_{\sigma, z}(y)$ is periodic in $y$ with period 1, and its monotonicity is the same as that of $\cos(2\pi y)$.

**Corollary 5.** The Diophantine-approximation decoder (16), intuitively conjectured in [5], is in fact optimum in 1-D.

**Proof:** Follows directly from proposition 4. More precisely, $f_{\sigma, z}(y)$ is monotonically decreasing as $y$ gets away from an integer $a$, until it reaches the minimum $a \pm 0.5$.

C. The Multidimensional Case

The multidimensional case requires to be able to derive product formulas from Jacobi theta series and show that the function $f_{\sigma, L}(y)$ has a regular behavior (decreases when the distance between $y$ and a lattice point increases) inside a Voronoi cell, or inside a certain region centered at a lattice point. If it is true (and we conjecture it is), then we have

**Conjecture 6.** Maximizing the ML metric is equivalent or close to solving a multidimensional inhomogeneous Diophantine approximation problem. More precisely,

$$\arg\max_{\lambda \in \Lambda} \varrho(\lambda) \simeq \arg\min_{\lambda, \rho \in \mathbb{C}, \rho \in \mathbb{L}} \|\rho(\lambda) - q\|^2. \quad (22)$$

Even if we are unable to prove the conjecture for now, there are evidences supporting a decoder like (22). Since $f_{\sigma, L}(y)$ is the Fourier transform of a positive measure, we have

**Proposition 7.** $f_{\sigma, L}(y)$ is a positive-definite function. In particular, $f_{\sigma, L}(y) \leq f_{\sigma, L}(0)$.

Moreover, it is easy to see

**Proposition 8.** The first derivative $f'_{\sigma, L}(y)$ is 0, while the second derivative $f''_{\sigma, L}(y) < 0$, when $y \in \mathbb{L}$.

Therefore, $f_{\sigma, L}(y)$ is bell-shaped in the vicinity of a lattice point, which is a local (and also global) maximum. The Diophantine-approximation decoder is a good approximation with respect to this monotonic decoding metric (at least locally).

We close this investigation with a remark on the different roles played by Diophantine approximation in this paper and in [2]. Diophantine approximation was a detrimental effect in [2], while it plays a constructive role in this paper. We expect it will achieve a better performance in compute-and-forward.

**References**


Analysis of the Stochastic Channel Model by Saleh & Valenzuela via the Theory of Point Processes

Morten Lomholt Jakobsen, Troels Pedersen and Bernard Henri Fleury \{mlj,troels,bfl\}@es.aau.dk
Section Navigation and Communications, Dept. of Electronic Systems, Aalborg University
Fredrik Bajers Vej 7B, DK-9220 Aalborg East, Denmark

Abstract—In this paper we revisit the classical channel model by Saleh & Valenzuela via the theory of spatial point processes. By reformulating this model as a particular point process and by repeated application of Campbell’s Theorem we provide concise and elegant access to its overall structure and underlying features, like the intensity function of the component delays and the delay-power intensity. The flexibility and clarity of the mathematical instruments utilized to obtain these results lead us to conjecture that the theory of spatial point processes provides a unifying mathematical framework to define, analyze, and compare most channel models already suggested in literature and that the powerful tools of this framework have not been fully exploited in this context yet.

I. INTRODUCTION

Literature regarding channel models for (indoor) radio propagation dates back earlier than 1960, and most commonly the wireless multipath channel is characterized via its (time and space varying) impulse response [1]. Two classic and seminal contributions within channel modeling are those by Turin et al. [2] and Saleh & Valenzuela [3]. To some extent the (indoor) model by Saleh & Valenzuela can be seen as a generalization of the (urban) model by Turin. Specifically, the generalization aimed at mimicking cluster alike behavior since this effect was reported to have been observed experimentally.

Ever since the model by Saleh & Valenzuela (for short the S-V model) was proposed in 1987, many refined or marginally extended variants have appeared, see e.g. [4] and [5]. Unfortunately, these channel models have not been developed within any unifying mathematical framework. Instead their treatment is of rather ad-hoc nature and, as a result, their inherent features remain essentially veiled and any two different models are not easily comparable.

Recently the authors of [6] and [7] reformulated and outlined the S-V model in terms of marked point processes. The S-V model has also been revisited in [8] by use of shot-noise tools and point process theory. Among other things the analysis in [7] and [8] show that the overall intensity of the relative delays of multipath components grows linearly with the propagation delay. Unfortunately, the mathematical tools used in [7] to extract the features of the model are not directly associated with the general theory of point processes. On the other hand, the tools used in [8] are rather advanced and the derivations less transparent. Accordingly, the potential theoretical benefits arising through these point process reformulations are not immediately evident.

In this paper we showcase how the general theory of spatial point processes provides an insightful view upon the inherent structure and features of the classical S-V model. Like [7] and [8] we revisit the model and reformulate it as a particular point process. Aligned with [7] we show that the component delays consist of the union of a Poisson point process and a Cox point process and we derive the associated intensity function as an immediate consequence of Campbell’s Theorem. The derivation in [7] is similar but with no reference to Campbell’s Theorem. Furthermore, and in contrast to the involved proofs relying on shot-noise tools in [8], we obtain the delay-power intensity in a simple and direct way by invoking once more Campbell’s Theorem. These results demonstrate the potential of this well-known theorem from the theory of spatial point processes in the context of stochastic channel modeling. In view of this, our conclusion is that the theory of spatial point processes and its powerful tools have not been fully exploited yet to analyze the properties of most proposed stochastic channel models. This theory appears to provide the necessary unifying framework for which these models can be contrasted within.

II. POINT PROCESS FRAMEWORK

We assume familiarity with the basics of the theory of spatial point processes (see [9, Sec. 1.3, Chap. 2] and [10, Sec. 1.5, 6.2] for highly recommendable introductions). Concepts from abstract measure theory will be kept at a minimum.

A. Locally finiteness and simplicity

Denote by $Y$ a locally finite and simple point process defined on a $d$-dimensional space $S \subseteq \mathbb{R}^d$. For intuitive, practical and mathematical reasons, these two properties are convenient to impose since several technical aspects can then be disregarded. A point process is locally finite if the number of points falling in every bounded Borel set $B \subseteq S$ is almost surely finite. A point process is simple if, almost surely, no two points of the process coincide. Accordingly, any realization of the point process $Y$ can be identified as a countable set of points $\{y_1, y_2, y_3, \ldots\}$, $y_i \in S$, where the index $i$ of $y_i$ serves solely as a dummy label. Thus, the index is used only to distinguish points and to indicate countability. It does not indicate any ordering of the points.

B. The intensity function and Campbell’s Theorem

Consider the counting function defined, using a generic indicator function $1[\cdot] \in \{0, 1\}$, as

$$N_Y(B) := \sum_{y \in Y} 1[y \in B],$$

which equals the random number of points from $Y$ falling in the set $B$. For any fixed and bounded $B$, the count $N_Y(B)$ is

\[115\]
a non-negative integer-valued random variable. The expected value of the counting function \( \mu_y(B) := \mathbb{E}[N_y(B)] \) defines a measure on \( S \), the so-called intensity measure of \( Y \). If the intensity measure can be expressed as

\[
\mu_y(B) = \int_B \varphi_y(y) \, dy, \quad B \subseteq S,
\]

for a locally integrable function \( \varphi_y : S \to [0, \infty) \), then \( \mu_y \) is called the intensity function of \( Y \). The case when the intensity function exists is by far the most important for applications [11]. The importance of the intensity function is evident from the following result, often referred to as Campbell’s Theorem.

**Campbell’s Theorem.** Let \( Y \) be a point process on \( S \subseteq \mathbb{R}^d \) with intensity function \( \varphi_y \). Then for a real or complex-valued measurable function \( h : S \to \mathbb{R} \) (or \( \mathbb{C} \)), the random variable \( \sum_{y \in Y} h(y) \) has expected value

\[
\mathbb{E}\left[ \sum_{y \in Y} h(y) \right] = \int_S h(y) \varphi_y(y) \, dy, \quad (1)
\]

provided that the integral on the right exists.

Proofs with varying degrees of detail can be found in [9, Sec. 3.2], [11, Prop. 4.1] and [12, Thm. 2.2]. Often, the theorem is stated only for non-negative functions \( h \), since the equality in (1) is then unconditionally true, i.e. the integral is always well-defined but possibly divergent. When \( h \) is real-valued some care must be taken since the integral at the right hand side of (1) has no meaning if the positive and the negative part of \( h \) are not integrable. Similar care must be taken for complex \( h \).

**C. Poisson and Cox point processes**

We now define two classes of point processes which are particularly important for our treatment in the forthcoming section, namely Poisson point processes and Cox point processes. These definitions can be found in many text books covering the theory of spatial point processes. Our treatment is directly inspired by [11] and the interested reader may consult [10]–[12] for further details.

**Definition.** A point process \( Y \) on \( S \subseteq \mathbb{R}^d \) is called a Poisson point process with intensity function \( \varphi_y \) if:

(i) For any \( B \subseteq S \) with \( \mu_y(B) = \int_B \varphi_y(s) \, ds < \infty \) the count \( N_y(B) \) is Poisson distributed with mean \( \mu_y(B) \).

(ii) Given that \( N_y(B) = n \in \mathbb{N} \) where \( 0 < \mu_y(B) < \infty \), the distribution of \( Y \cap B \) is the same as that of \( n \) points drawn i.i.d. according to \( f_y \), where

\[
f_y(s) := \frac{\varphi_y(s) \mathbf{1}[s \in B]}{\mu_y(B)}.
\]

We write \( Y \sim \text{PoissonPP}(S, \varphi_y) \).

**Definition.** Let \( Z(s), \ s \in S \), be a non-negative random field such that, almost surely, every realization of \( Z \) is a locally integrable function on \( S \). If a point process \( Y \), conditioned on \( Z \), is a Poisson point process with intensity function \( Z \), then \( Y \) is called a Cox point process driven by \( Z \).

Cox point processes (also often referred to as doubly stochastic Poisson point processes [10]) are flexible models for clustered point patterns. Specifically, the two-level construction most commonly entails the Cox class to exhibit so-called over-dispersion compared to the Poisson class [11, Sec. 5.2].

**III. The Model by Saleh & Valenzuela**

In this section we analyze the impulse response of the classical S-V model within the framework of spatial point processes. The main purpose of this effort is to straightforwardly derive the features of this model through a flexible and powerful theory. Several relevant aspects of the model are revealed through this reformulation, e.g. its overall delay intensity, a concise and clear derivation of the average power gain and, a simple derivation of the delay-power intensity as well.

**A. Classical formulation**

Saleh & Valenzuela define the channel impulse response with cluster and within-cluster delays as [3, Eq. (25)]

\[
h(t) = \sum_{\ell=0}^{\infty} \sum_{k=0}^{\infty} \delta_{k,\ell} \exp(j\theta_{k,\ell}) \delta(t - (T_\ell + \tau_{k,\ell})), \quad (2)
\]

where \( \delta \) is the Dirac delta and \( j \) is the imaginary unit. The index \( \ell \) indicates a certain cluster and \( k \) is the within-cluster index. By definition in [3], \( T_0 = 0 \) and \( \tau_{0,\ell} = 0 \) for each \( \ell \in \mathbb{N}_0 := \{0\} \cup \mathbb{N} \). Beside these fixed delay components, a sequence of Poisson point processes are suggested such that

- \( \{T_\ell\}_{\ell \in \mathbb{N}} \sim \text{PoissonPP}(\mathbb{R}_+, \Lambda) \)
- \( \{\tau_{k,\ell}\}_{k \in \mathbb{N}} \sim \text{PoissonPP}(\mathbb{R}_+ , \lambda) \) for each \( \ell \in \mathbb{N}_0 \), with \( \Lambda, \lambda > 0 \) being two parameters. Moreover, conditional second-order moments are modeled such that [3, Eq. (26)]

\[
\mathbb{E}[\beta_{k,\ell}^2 | T_\ell, \tau_{k,\ell}] = Q \exp(-T_\ell / \Gamma) \exp(-\tau_{k,\ell} / \gamma), \quad (3)
\]

with \( \Gamma, \gamma > 0 \) and \( Q \) being the average power gain of the first component within the first cluster (i.e. corresponding to the fixed delay \( T_0 \)). Conditioned on all \( T_\ell \)'s and all \( \tau_{k,\ell} \)'s, the \( \beta_{k,\ell} \)'s are assumed to be mutually independent random variables. Specifically, each power gain \( \beta_{k,\ell}^2 \), conditioned on \( T_\ell \) and \( \tau_{k,\ell} \), should follow an exponential distribution with mean parameter decaying as described by (3). Fig. 1 illustrates the Poisson point processes involved in the S-V model.

Finally, it was mentioned in [3] that practically the doubly-infinite sum in (2) should "stop" whenever each of the exponentially decaying terms in (3) had become small enough. Through the insight gained via the forthcoming reformulation of this classical channel model we are able to motivate a less heuristic "stopping criterion".

**B. Point process formulation**

Naturally, we select the space \( S = \mathbb{R}_+ \) and let \( T_0 = 0 \) as above. In addition, we introduce the point processes:

- \( C := \{T_\ell\}_{\ell \in \mathbb{N}} \) (all cluster delays except \( T_0 \))
- \( W_\ell := \{T_\ell + \tau_{k,\ell}\}_{k \in \mathbb{N}} \) (delays within the \( \ell \)’th cluster)
- \( W := \bigcup_{\ell=0}^{\infty} W_\ell \) (all within-cluster delays)
- \( Y := C \cup W \) (all propagation delays except \( T_0 \)).
Notice that $C$ is the Poisson point process specified at first in the previous paragraph. Its intensity function has a simple form, namely $\varrho_w(t) = \Lambda$ for all $t \in S$. By conditioning, we immediately identify a sequence of Poisson point processes

$$W|T_k \sim \text{PoissonPP}(R_k, \lambda I[t > T_k]), \; \ell \in \mathbb{N}_0,$$

and since the Poisson class is stable with respect to countable superpositions [11, Prop. 3.6], we see that

$$W|C \sim \text{PoissonPP}(R_+, \bar{\varrho}_w),$$

with the staircase-like intensity function

$$\bar{\varrho}_w(t) = \lambda + \lambda \sum_{c \in C} I[t > c], \; t \in S. \tag{4}$$

Accordingly, we identify that the point process $W$, without conditioning on $C$, is a Cox point process driven by a stochastic process $Z$ having the same functional form as $\bar{\varrho}_w$ in (4) but with $C$ being random. The intensity function of the Cox point process $W$ is $\varrho_w(t) = \mathbb{E}[Z(t)]$ [11, Sec. 5.2], and by direct application of Campbell’s Theorem we get

$$\varrho_w(t) = \lambda + \lambda \mathbb{E} \left[ \sum_{c \in C} I[t > c] \right] = \lambda + \lambda \bar{t}, \; t \in S.$$

Since $Y = C \cup W$ is a union of almost surely disjoint point processes, its associated intensity function reads [10, Sec. 6.2.3]

$$\varrho_Y(t) = \varrho_c(t) + \varrho_w(t) = \lambda + \lambda \bar{t}, \; t \in S.$$

It is interesting to notice that the entire set of propagation delays (excluding the first component $T_0$) is the union of a Poisson point process and a Cox point process. Of course, the realization of $W$ depends upon the realization of $C$, i.e., these two point processes are not independent. In [7] this interpretation was inherently adopted, without being explicitly mentioned. Another interesting yet expected observation is that the intensity function $\varrho_w$ rises linearly with propagation delay, see Fig. 2. The jump of height $\lambda + \lambda \bar{t}$ at $T_0 = 0$ in the graph of $\varrho_w$ appears due to the cluster delays and the delays within the very first cluster. The term $\lambda \bar{t}$ result from the fact that, on average, a total of $\lambda \bar{t}$ additional clusters emerge during the interval $[0, \ell]$, with each and every one of them spawning further delay components at rate $\lambda$.

C. Multipath power gain

Analogous to the approach in [3], we consider the following non-negative random variable

$$G := \sum_{\ell=0}^{\infty} \sum_{k=0}^{\infty} \beta_{k,\ell}^2, \tag{5}$$

referred to as the total multipath power gain [3]. By splitting $G$ into three terms corresponding to $T_3$ and arrivals in $C$ and $W$, its expectation can be calculated as

$$\mathbb{E}[G] = \mathbb{E}[\beta_{0,0}^2] + \mathbb{E}\left[\sum_{k=1}^{\infty} \beta_{k,0}^2\right] + \mathbb{E}\left[\sum_{k=0}^{\infty} \sum_{\ell=1}^{\infty} \beta_{k,\ell}^2\right].$$

As in [3] we write $\beta(T_t, \tau_{k,\ell})$ in substitute for $\beta_{k,\ell}$ to facilitate a comprehensible notation in the following. For additional clarity we introduce the function

$$f(t, \bar{t}) := Q \exp \left(-t/\Gamma - \bar{t}/\gamma\right), \; t, \bar{t} \in S.$$

Notice that $f(T_t, \tau_{k,\ell}) = f(T_t, (T_t + \tau_{k,\ell}) - T_t)$ coincides with the expression in (3). Then, by intermediate conditioning on $C$, we calculate the expectation of the term $(\ast)$ as

$$\mathbb{E}[(\ast)] = \mathbb{E}\left[\sum_{c \in C} \beta^2(c, 0)\right] = \mathbb{E}\left[\sum_{c \in C} \mathbb{E}\left[\beta^2(c, 0) | C\right]\right] = Q \Gamma,$$

where the final step follows by application of Campbell’s Theorem. Next, by defining $C_0 := \{T_0\} \cup C$ and with a similar sequence of manipulations involving intermediate conditioning and Campbell’s Theorem, we find the expected value of $(\circ)$ to be

$$\mathbb{E}[(\circ)] = \mathbb{E}\left[\sum_{c \in C_0} \sum_{w \in W_c} \beta^2(c, w - c)\right] = \mathbb{E}\left[\sum_{c \in C_0} \sum_{w \in W_c} \mathbb{E}\left[\beta^2(c, w - c) | c, w\right]\right] = Q(1 + \Lambda \Gamma) \lambda \gamma. \tag{6}$$

Accordingly, the average total power gain is given by

$$\mathbb{E}[G] = \mathbb{E}[\beta_{0,0}^2] + \mathbb{E}[(\ast)] + \mathbb{E}[(\circ)] = Q + Q \Lambda \Gamma + Q(1 + \Lambda \Gamma) \lambda \gamma = Q(1 + \lambda \gamma) (1 + \Lambda \Gamma), \tag{7}$$

1Note that in (6) we abuse notation since the collections $W_c$ are not explicitly defined. We only defined these as $W_{t\ell}$ via the counting index $\ell$. 

---

Figure 1. Realization of Poisson point processes corresponding to the S-V model. Circle points indicate fixed delay components. The top process occurs with rate $\Lambda$ while each of the lower processes occurs with rate $\lambda$. A new point process is initialized whenever a new point emerges from the top process.

Figure 2. Intensity functions associated with the S-V model.
as was also reported in a footnote in [3]. Yet, the original sequence of arguments used to obtain this result may appear less instructive, see [3, Eq. (27), (31)] for comparison. Notice that, depending on how we choose to write out the product in (7), we end up with different interpretations of individual average power contributions.

### D. Delay-power intensity

Motivated by the definition of $G$ in (5) together with the relationship in (7), we consider

$$P(t) := \sum_{\ell=0}^{\infty} \sum_{k=0}^{\infty} \beta_{k,\ell} \delta(t - (T_\ell + \tau_{k,\ell})).$$

We wish to calculate how the average power gains are distributed across delay. From (7) we already know the mean total power gain, yet we seek to obtain further insight. The above definition of $P(t)$ is motivated by the fact that $\int_0^\infty p(t)dt = G$, and since $E[G]$ is finite, the non-negative random variable $G$ is itself finite almost surely. Accordingly, we define

$$P(t) := E[p(t)], \quad t \in S,$$

and we refer to this function as the delay-power intensity. By similar manipulations as in the previous paragraph (conditioning, Campbell’s Theorem, etc.) we find

$$P(t) = \frac{\delta(t)}{Q} + \left\{ k_1 \exp \left( -\frac{\alpha}{\gamma} t \right) \right\} \left\{ k_2 \exp \left( -\frac{\alpha}{\gamma} t \right) \right\}, \quad \Gamma \neq \gamma, \quad \Gamma = \gamma$$

where we have conveniently introduced the two constants

$$k_1 := \Lambda \left( 1 + \frac{\lambda}{\Gamma - \gamma} \right) \quad \text{and} \quad k_2 := \lambda \left( 1 - \Lambda \frac{\lambda}{\Gamma - \gamma} \right).$$

The same expression for $P(t)$ is obtained in [8, Chap. 2.3] using rather involved shot-noise tools with weighty notational overhead. Notice the particular relationship

$$E[G] = E \left[ \int_0^\infty p(t)dt \right] = \int_0^\infty E[p(t)]dt = \int_0^\infty P(t)dt.$$

The delay-power intensity of the S-V model is depicted in Fig. 3. Notice that $P(t)$ is not exponentially decaying, not even when $\Gamma = \gamma$ since $\phi_\gamma$ rises linearly (compare with the dotted line in Fig. 3).

Finally, as mentioned in the beginning of this section, we are now able to motivate a simple “stopping criterion” suitable, e.g. for simulation purposes. Specifically, select a delay threshold $\tau_{\text{stop}}(\alpha)$ such that

$$\int_0^{\tau_{\text{stop}}(\alpha)} P(t)dt = \alpha E[G],$$

for a relevant choice of $\alpha \in (0, 1)$, e.g. $\alpha = 0.99$.

### IV. Conclusion

In this contribution we have revisited the radio channel model by Saleh & Valenzuela (the S-V model) within the framework of spatial point processes. We have shown that the component delays in the S-V model emerge from the union of a Poisson point process and a Cox point process. Furthermore, we have demonstrated that the intensity function of the component delays and the delay-power intensity can be derived in a straightforward and rigorous manner as an immediate consequence of Campbell’s Theorem.

The above results indicate that the theory of spatial point processes yields a natural, unifying theoretical framework for dealing with stochastic channel models. This applies in particular to most channel models already suggested in literature, including the models by Turin et al. [2], Spencer et al. [4], and Chong et al. [5]. Our results also reveal that the powerful tools available in this framework, like Campbell’s Theorem, have not been exploited to their full extent in this context yet. Overall the considered application to the S-V model and to some extent the work in [6]–[8] show that the resulting mathematical treatments inherit clarity and conciseness, in addition to rigorousness, in contrast to the traditionally used ad-hoc and heuristic arguments.

### ACKNOWLEDGMENT

This work has been funded by the project ICT-248894 Wireless Hybrid Enhanced Mobile Radio Estimators 2 (WHERE2), the 4GMCT cooperative research project co-financed by Intel Mobile Communications, Agilent Technologies, AAU and the Danish National Advanced Technology Foundation, and by the Danish Agency for Science, Technology and Innovation.

### REFERENCES

Low-Complexity Soft-Output Detection for Impulse-Radio Ultra-Wideband Systems Via Combining Multiple Observations

Andreas Schenk\textsuperscript{a} and Robert F.H. Fischer\textsuperscript{b}
\textsuperscript{a}Lehrstuhl für Informationsübertragung, Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany,
\textsuperscript{b}Institut für Nachrichtentechnik, Universität Ulm, Ulm, Germany,
Email: andreas.schenk@lnt.de, robert.fischer@uni-ulm.de

Abstract—Avoiding explicit channel estimation, autocorrelation-based receivers enable low-complexity detection of impulse-radio ultra-wideband (IR-UWB) signals. In this paper, we present power-efficient IR-UWB detection schemes, which process the receive symbol stream partitioned into overlapping blocks. Properly combining the individual observations gained from each blockwise detection, a final (soft-)decision with improved reliability is obtained. By means of numerical simulations we show that the presented approach enables a reduction of the hardware complexity of the autocorrelation receiver at no degradation in performance and only slightly increased computational complexity, and is perfectly suited especially for coded transmission.

I. INTRODUCTION

Noncoherent receivers are widely regarded as the key to low-complexity detection in impulse-radio ultra-wideband (IR-UWB) systems, which are especially well-suited for short-range low-data-rate communications, such as, e.g., in wireless-sensor networks [1]. The inherent loss in performance of conventional noncoherent autocorrelation-based differential detection (DD), as compared to coherent detection based on explicit channel estimation, can be alleviated by advanced autocorrelation-based detection schemes, which process blocks of receive symbols jointly, i.e., perform multiple-symbol differential detection (MSDD) [2]–[4] or decision-feedback differential detection (DF-DD) [5]. However, significant challenges for implementation of the autocorrelation receiver (ACR) front-end are induced, as delaying the (analog) receive signal over several symbol durations is required to compute the correlation coefficients required for these detection schemes.

In this paper, we propose to obtain multiple observations of the same symbol by processing overlapping blocks of symbols. The resulting information gained from these individual observations is then combined to obtain a final decision with improved reliability. Depending on the choice of the blockwise detection scheme (we consider MSDD and DF-DD) and the method, how the individual observations are combined, the presented scheme is able to deliver hard and soft decisions, and is thus suited for coded and uncoded transmission. The benefit of the presented scheme is to reduce the hardware implementation complexity of the ACR front-end at no degradation in performance.

The proposed scheme can be seen as a development of subset MSDD introduced for noncoherent detection of differential phase-shift keying (DPSK, or more general differential space-time modulation) [6], [7], where similarly multiple observations are generated with an overlapping block-structure. However, as opposed to our approach, in [6], [7] most of the observations are simply treated as unreliable, thus discarded, and only a single observation is processed further.

After defining the IR-UWB system model and briefly reviewing autocorrelation-based detection in Sec.II, multiple-observations combining based on blockwise detection is introduced in Sec.III. The effectiveness of the approach is validated in Sec.IV by means of numerical simulations of uncoded and coded transmission; final remarks conclude the paper.

II. AUTOCORRELATION-BASED DETECTION OF IR-UWB

A. IR-UWB System Model

We consider differentially encoded binary pulse-amplitude-modulated IR-UWB transmission. Assuming no inter-symbol-interference (i.e., the symbol duration $T$ is chosen sufficiently large, such that each pulse has decayed before the next pulse is received) and a time-invariant multipath channel, the sampled receive signal in the $k$th symbol interval may compactly be written as an $(N_s = f_s T)$-dimensional vector

$$\tilde{r}_k = b_k \bar{p} + \tilde{n}_k$$

where $f_s$ is the sampling frequency, $b_k$ are differentially encoded information symbols $a_k \in \{\pm 1\}$ (i.e., $b_k = b_{k-1} a_k$ with $b_0 = 1$), and $\bar{p}$ is the (sampled) receive pulse shape, resulting from the convolution of the transmit pulse, multipath channel and receive filter impulse response. Neglecting the reference symbol, after suitable normalization, the energy per symbol is given as $E_s = 1$. $\tilde{n}_k$ collects the samples of the zero-mean additive Gaussian noise, each with variance $\sigma_n^2 = f_s N_0 / 2$, where $N_0 / 2$ is the two-sided power-spectral density of the underlying additive white Gaussian noise.

B. Autocorrelation Receiver Front-End

Avoiding explicit channel estimation, autocorrelation-based detection computes the correlation of the current symbol with up to $L$ preceding symbols, cf., e.g., [2]–[5], [8]. Significant gains are achieved by adapting the correlation interval $T_i$ to the channel characteristics at hand [1], i.e., considering only
the relevant part $r_k$ of $\tilde{r}_k$ for autocorrelation by selecting $N = T f_s$ successive elements (typically the first) out of $N_s$. This effectively reduces the time-bandwidth product to $N = f_s T$. Thus, the output of the $L$-branch ACR for the $k$th symbol is given as, for $l = 1, \ldots, L$,

$$z_{k-l,k} = r_{k-l}^T r_k = b_{k-l} | b_k + \eta_{k-l,l}. \tag{2}$$

Assuming that the correlation interval captures the entire pulse energy (i.e., due to normalization $p^T p = p^2 = 1$), $z_{k-l,k}$ is composed of the phase transition from $b_{k-l}$ to $b_k$, and “information × noise” and “noise × noise” terms, summarized in the equivalent noise term $\eta_{k-l,l} = b_{k-l}^T p n_k + b_k n_{l+1}^T p + n_{l+1}^T n_k$. It has been shown that already for moderate time-bandwidth products $N$ this equivalent noise is well approximated as uncorrelated zero-mean Gaussian random variables with variance $\sigma_n^2 = 2\sigma_n^2 + N(\sigma_n^2)^2$ [8]. For clarity, we note that, given the assumptions above, this Gaussian-approximated discrete-time equivalent model is independent of the actual pulse shape, i.e., the channel realization [9]. In this paper, all simulations of the IR-UWB system use this equivalent model.

The correlation coefficients serve as input for various detection schemes [1]–[5], [8], the most simple being symbolwise differential detection (DD), evaluating only the sign of the correlation. The channel realization [9]. In this paper, all simulations of the IR-UWB system use this equivalent model.

The correlation coefficients serve as input for various detection schemes [1]–[5], [8], the most simple being symbolwise differential detection (DD), evaluating only the sign of the correlation coefficient of the current symbol and its predecessor, i.e., $a_k = \text{sign}(z_{0,k})$ with $L = 1$.

Finally, we point out that, due to the large signal bandwidth, implementation of the ACR is typically envisioned fully in the analog domain, instead of digital processing based on a sufficiently sampled version. This imposes a major drawback of ACR-based detection, namely the required accurate (analog) delay lines. In this paper, we aim to mitigate this drawback by means of detection schemes based on an ACR with less branches, i.e., reduced, $L$, at no degradation in performance.

III. Soft-Output Detection Based on Multiple Observations

Based on the output of an $L$-branch ACR, low-complexity, yet powerful detection schemes process the receive symbol stream partitioned into blocks of $L$ symbols. Before discussing the generation and combination of multiple observations obtained from this blockwise processing, we briefly review two variants of block-based detection schemes, namely (soft-output) MSDD and DD-DF [2]–[5], which deliver individual (soft) decisions for the symbols within each block.

A. Obtaining an Individual Observation

Keeping notation simple, we consider an arbitrary block of information symbols grouped into a vector $\mathbf{a} \in \{-1,1\}^L$ with all indices to be understood as intra-block indices.

1) Soft-Output Multiple-Symbol Differential Detection: MSDD delivers the blockwise-optimal sequence [3], [5], i.e., the sequence with minimum decision metric

$$\mathbf{a}_{\text{MSDD}} = \arg\min_{\mathbf{a} \in \{-1,1\}^L} \Lambda(\mathbf{a}) \tag{3}$$

where the decision metric is defined as

$$\Lambda(\tilde{\mathbf{a}}) = \sum_{k=1}^L \left( \sum_{l=0}^{L-1} |z_{l,k} - \tilde{b}_l z_{l,k}| \right) \tag{4}$$

and $\tilde{b}_l$, $0 \leq k \leq L$, and $z_{l,k}$, correspond to the differentially encoded hypothesis $\tilde{a}$, and the correlation coefficients (cf., (2)) of the considered block, respectively.

Soft decisions, i.e., reliability information on the decision of each symbol, is obtained by extending MSDD to so-called soft-output MSDD (SO-MSDD) [4]. In terms of (max-log approximated) log-likelihood ratios (LLRs), soft output for the $k$th (binary) information symbol is given as

$$\text{LLR}_k = \frac{1}{\sigma_n^2 (L+1)} \left( a_{k,\text{MSDD}} (\Lambda_{\text{MSDD}} - \Lambda) \right) \tag{5}$$

where $\Lambda_{\text{MSDD}} = \Lambda(\mathbf{a}_{\text{MSDD}})$ denotes the metric associated with the MSDD estimate (3), and

$$a_{k,\text{MSDD}} = \min_{\tilde{a} \in \{-1,1\}} \Lambda(\tilde{\mathbf{a}}) \tag{6}$$

the decision metric of the corresponding counter hypothesis, i.e., the minimum metric with the restriction $a_k = -a_{k,\text{MSDD}}$.

Utilizing the triangular structure of the decision metric, MSDD can be viewed as a tree-search-problem, which in turn can efficiently be solved employing the single-tree-search soft-output sphere decoder (SO-SD) [10], [4]. To further reduce the computational complexity of the SO-SD in particular at low SNR, LLR clipping during the SO-SD search process (clipping level $\text{LLR}_{\text{clip}}$) [10], [4] is applied. In addition, we employ an optimized decision order within the search process. This sorting of the SD input is based on the same optimization criterion as of DF-DD (cf. Sec. III-A2 and [5]), and requires some modifications to the original SO-SD algorithm [4]. Minimum throughput of the MSDD device, despite the varying complexity of the SD, can effectively be guaranteed by introducing a cumulative maximum complexity limit (e.g., per codeword) similar to [10]. Due to space limitations, these techniques are not described in greater detail.

2) Decision-Feedback Differential Detection: A closely-related detection scheme is blockwise DD-DD [5], which decides the symbols within each block in a successive manner taking into account the feedback from already decided symbols within the block. In [5] it has been shown that DD-DD achieves close-to-optimum performance at very low, and in particular constant computational complexity, if the decision order is optimized, such that in each step the most reliable symbol is decided next (taking into account the actual correlation coefficients and the previous decisions, see [5] for details).

B. Obtaining Multiple Observations

In the related context of noncoherent detection of DPSK (and more general unitary differential space-time modulation), in [6], [7] it has been proposed to employ MSDD in combination with an overlapping block-structure. Since multiple blocks thus contain the same symbol, processing of each block delivers (possibly different) beliefs on the same symbol, i.e., multiple observations are available. Observing that the reliability of the estimates degrades at the block edges for
DPSK [6], [7], only the middle part of each block is processed further, so-called subset MSDD, as illustrated in the left part of Fig. 1. Already available observations of symbols at the block edges are treated as unreliable and are simply discarded.

Avoiding to discard already available observations, we propose to suitably combine all observations obtained from processing of each block, resulting in a (possibly more reliable) final decision. The basic concept is illustrated on the right hand side of Fig. 1. There are different options how to combine multiple observations to deliver a final (soft) decision for the respective symbol, as discussed in Sec. III-C. In addition note that, in the context of MSDD of IR-UWB, significantly different reliabilities within each block cannot be observed, and, as we employ an optimized decision order within each block (cf., [5] and Sec. III-A), the positions of the most reliable symbols are no longer limited to the block center.

Of course, the generation of multiple observations, i.e., processing of overlapping blocks, leads to an increased computational complexity and processing delay; however, the analog ACR front-end remains unchanged. As we aim for a flexible tradeoff between performance and computational complexity, we vary the mutual shift $s = 1, \ldots, L$ of adjacent blocks (i.e., overlap of $L - s$ symbols) to adjust the number of multiple observations for each symbol. Thus, we obtain on average $L/s$ observations per symbol (for finite symbol streams this value can be ensured by reducing the blocksize at the beginning and at the end), with maximum overlap for $s = 1$, and the traditional single observation per symbol for $s = L$. Note that blocksize and overlap are defined based on the information symbols; the respective blocks of $L + 1$ differentially encoded symbols overlap by at least one symbol.

C. Combining Multiple Observations

Depending on the employed blockwise detection scheme, different options can be considered how multiple observations are combined to a final estimate of the information symbol.

1) Combining Multiple Soft Decisions: If soft decisions are available for each individual observation, as obtained from SO-MSDD, we may consider these observations as independent (though independence cannot be assured), and obtain the resulting final estimate according to the principle of parallel information combining [11]. In terms of LLRs, the resulting LLR for each symbol is then directly given as the sum of the corresponding individual LLRs.

In the case of coded transmission, the resulting LLR can directly be passed to the subsequent (soft-input) channel decoder. Since LLR clipping affects each SO-SD run, the maximum absolute input value is $L \cdot \text{LLR}_{\max}$. Final hard-decisions can be obtained by evaluating only the sign of the sum of the LLRs, corresponding to combining the individual hard decisions according to their reliability.

2) Combining Multiple Hard Decisions: Based on multiple hard decisions, here obtained from DF-DD (of course, hard-output MSDD is also possible, but not considered for brevity), a final hard decision is directly given following the majority of the individual decisions (i.e., evaluating the sign of the sum of the individual estimates), which effectively treats each observation with equal reliability (in case of a draw, we choose $\pm 1$ at random). Clearly, this strategy leads to improved performance only if more than two observations are available per symbol, i.e., only if $s < L/2$.

However, the most interesting option is to combine multiple hard decisions of the same symbol to form a single soft decision. Exemplarily, consider the case that three (possibly different) hard decisions are available. If these coincide, the reliability of this symbol is three times as large as the reliability of only a single observation, whereas if only two coincide and one differs, the reliability is reduced to the case of a single observation. This method can be implemented by using the sum of the individual hard-decisions as (quantized and scaled) “soft-output”. Depending on the employed channel decoder, scaling of these soft-output values may be neglected if the decoding metric is scale-invariant (e.g., in a convolutional coded system employing the Viterbi algorithm for soft-decision decoding). However, e.g., in an LDPC-coded system, the soft output should be scaled to match the decoder input range.

IV. PERFORMANCE AND COMPLEXITY

Modeling a realistic IR-UWB scenario [1], for numerical simulations we set the time-bandwidth product to $N = 400$, corresponding, e.g., to an ACR correlation interval of $T_s = 33$ ns and a Nyquist frequency of 6 GHz ($f_s = 12$ GHz). Typically, the symbol duration is chosen larger (e.g., $T = 75$ ns) to preclude inter-symbol interference.

We evaluate the effectiveness of the presented approach based on the key figures performance (i.e., the required energy per bit $E_b$ vs. spectral noise density $N_0$) and computational complexity at a desired bit error rate (BER) operating point. As a measure for the computational complexity of MSDD we use the average number of visited nodes per symbol of the SO-SD tree-search process. Due to LLR clipping and the optimized decision order, its variance is relatively low and can effectively be upper-bounded by introducing a cumulative maximum complexity limit. On the contrary, DF-DD has constant complexity, which, using its relation to SD-based tree search [5], corresponds to a single node per symbol.

First, we consider uncoded transmission ($E_b = E_s$), i.e., multiple observations are combined to yield a final hard decision. Fig. 2 depicts the tradeoff performance vs. (average) complexity at an operating point of $\text{BER} = 10^{-3}$, obtained by adjusting the mutual shift of the blocks ($s = 1, \ldots, L$) for a blocksize $L = 3, 5$ and 10. We consider the two detection schemes combining multiple soft decisions obtained from SO-
MSDD (sign of sum of LLRs), and combining multiple hard decisions obtained from DF-DD (sign of sum of estimates).

For SO-MSDD, the LLR clipping level is set to $LLR_{\text{max}} = 15$. For both cases the upper-left point corresponds to the maximum block-overlap (maximum number of $L$ observations, $s = 1$), and the lower-right point to traditional detection with only a single observation per symbol (no overlap, $s = 1$).

The performance of SO-MSDD improves with increasing number of combined individual soft-decisions, with gains of up to 0.8 dB compared to traditional MSDD; the increase in complexity is proportional to the average number of observations per symbol, $L/s$. For comparison, tradeoff-curves employing the SO-SD without LLR clipping are also given, which prove that LLR clipping is crucial to achieve low complexity at only slight performance degradation [4], [10].

Combining multiple hard decisions (DF-DD) improvements are only achieved for $s < L/2$, as otherwise up to two observations are available. However, as opposed to the varying complexity of SD-based MSDD, DF-DD achieves even larger gains at lower and, in particular, fixed complexity.

Finally, the performance-complexity tradeoff is evaluated for coded transmission at BER $= 10^{-6}$ in Fig. 3. We consider a bit-interleaved coded modulation scheme employing maximum-free-distance convolutional codes (non-recursive, non-systematic, zero-padded encoder [12]) with constraint length $\nu = 4$, a random block-interleaver (size 1500 symbols), and, following the reasoning in [9], a code-rate of $R_c = 2/3$ ($E_b = E_a R_c$). As we aim for soft-decision decoding using the Viterbi algorithm, the final soft decision is obtained by combining either multiple soft decisions (sum of LLRs) in case of SO-MSDD as blockwise detection scheme, or multiple hard decisions (sum of estimates) in case of DF-DD, cf. Sec. III-C.

It can be observed that for all blocksizes each additional observation leads to notable gains. Combining of multiple observations is especially effective, if multiple hard decisions are combined to a single soft decision (DF-DD); the difference between the traditional single observation and $L$ observations, e.g., amounts to almost 2 dB for $L = 10$. Consequently, this scheme enables to effectively tradeoff low-cost computational complexity against costly implementation complexity of the ACR front-end at no degradation in performance by processing shorter, but overlapping blocks (thus, lower maximum delay required in the ACR) and combining the multiple observations.

V. CONCLUSIONS

In this paper, we have proposed a power-efficient, yet low-complexity detection scheme for autocorrelation-based IR-UWB receivers, which is based on combining multiple observations for the same symbol obtained from blockwise detection using an overlapping block structure. Our approach is able to deliver hard and soft decisions with improved reliability at only marginal increase in computational complexity, by, simplified speaking, repeating very-low-complexity steps multiple times and combining the individual observations. It is perfectly suited especially for coded transmission and enables to reduce the blocksize, and thus also the implementation complexity of the analog autocorrelation receiver front-end, at no degradation in performance.

REFERENCES

Amplify-and-Forward Versus Decode-and-Forward Relaying: Which is Better?

Georgy Levin and Sergey Loyka

Abstract—Performance of multi-hop MIMO relay channels under the amplify-and-forward and decode-and-forward protocols are compared via the capacity and SNR gains. In an N-hop channel of linear topology with multi-antenna source and destination and single-antenna relay nodes, the capacity gain of the DF relaying over the AF one does not exceed \( \log_2 N \) bit/s/Hz and its SNR gain does not exceed \( N \), for any channel realization. This conclusion also applies to selection relaying, to the outage probability/capacity and the ergodic capacity in an arbitrary block-fading channel, and can be further extended to hybrid relaying. The conditions under which the DF and AF relaying have nearly identical performance are identified.

I. INTRODUCTION

Due to its numerous advantages (e.g. improved coverage, throughput, system capacity, power/battery life etc.), multi-hop relaying has recently attracted significant attention in both academia [1] and industry. Most common relaying strategies are decode-and-forward (DF) and amplify-and-forward (AF). While a DF relay decodes, re-modulates and retransmits the received signal, an AF one simply amplifies and retransmit the signal without decoding. Compared to an AF relay, the complexity of a DF one is significantly higher due to its full processing capability. The DF protocol also requires a sophisticated media access control layer, which is unnecessary in the AF protocol. Overall, a DF relay is nearly as complex as a base station. Does the performance improvement of DF relaying outweigh its complexity burden? To answer this question (in part), the present paper will quantify this performance improvement.

There seems to be no consensus in the literature as to the question: What relaying protocol is better, AF or DF? While some studies find the DF protocol to be superior, others find the other way around. Indeed, it has been shown in [2] based on numerical simulations, that the AF multi-hop relaying outperforms the DF one under uncoded BPSK modulation in terms of outage probability and bit error rate (BER), which was explained by the error propagation effect in the DF relaying outweighing the noise amplification of the AF relaying. An analysis of a maximum likelihood demodulation presented in [3] for coherent cooperative diversity in uncoded BPSK systems shows that the DF relaying with more than one relay loses about half of the diversity of the AF relaying.

These studies, however, are limited to uncoded systems. Most modern communications systems rely on powerful channel codes and the uncoded results do not extrapolate directly to those systems. Indeed, it was shown in [4] that single-antenna multi-hop Rayleigh-fading relay channels under the DF protocol achieve higher ergodic (mean) capacity than under the AF one. A similar conclusion was obtained in [5] for the ergodic capacity of MIMO multi-hop relay systems. The study in [6] shows that the outage probability in multi-hop relay channels is higher under the AF protocol, which automatically transforms into smaller outage capacity. Therefore, one has to conclude that while the AF protocol is better for uncoded systems (where the error propagation effect outweighs the noise amplification), the opposite is true for systems using powerful capacity-approaching codes. Finally, the popular diversity-multiplexing framework (DMT, see e.g. [7]) has been also applied to relay channels [8][9]. We caution the reader that this framework, due to its asymptotic nature \( \left( SNR \to \infty \right) \), is known to provide misleading conclusions in many scenarios (including relay ones) as far as the finite-SNR performance is concerned [6][10].

In this paper, we consider coded systems via their capacity analysis (i.e. the fundamental limit to error-free data transmission) and quantify the performance superiority of the DF relaying over the AF one, for a fixed channel (or, alternatively, for a given realization of an arbitrary-fading channel), via two performance metrics: capacity and SNR gains. The main result is that, in an N-hop channel with multi-antenna source and destination and single-antenna relay nodes, the capacity gain of the DF relaying does not exceed \( \log_2 N \) bit/s/Hz and its SNR gain does not exceed \( N \). This conclusion also applies to selection relaying, to the outage probability/capacity and the ergodic capacity in an arbitrary block-fading channel, and can be further extended to hybrid relaying. The conditions under which the DF and AF relaying have nearly identical performance and, thus, the DF relaying is not worth the effort due to its higher complexity, are identified.

II. RELAY CHANNEL MODEL AND CAPACITY

Let us consider a multi-hop relay channel when the source (transmitter) and the destination (receiver) are equipped with multiple antennas, and the full-duplex relay nodes have a...
single antenna, as shown in Fig. 1. Throughout the paper, we assume that the channel is of linear topology, i.e. only adjacent relays can "see" each other [8] and there is no direct (source-destination) link (this is the case when relay nodes are needed most), and that the full CSI is available at the source and the destination \(^1\).

A. Amplify-and-Forward Protocol

The signal received by i-th relay is \(y_i = h_i x_{i-1} + \xi_i\), \(i \geq 1\), where \(x_{i-1}\) is the signal transmitted by (i-1)-th relay, \(h_i\) is the i-th hop channel (between (i-1) and i-th relays), and \(\xi_i\) is i-th relay AWGN noise, and we assume the baseband, discrete-time, frequency-flat channel model. The signal transmitted by i-th relay in the AF mode is \(x_i = \sqrt{K_i} y_i\), where \(K_i\) is its power gain. Thus, the input-output relationship of the whole N-hop AF relay channel is

\[
y = \sqrt{K_{N-1}} h_N y_1 + \sqrt{K_{N-2}} h_{N-1} y_2 + \ldots + \sqrt{K_1} h_2 y_N + \xi_N
\]

where \(x\) and \(y\) are the transmitted (source) and received (destination) signal vectors, \(h_1, h_N\) are the 1-st and last hop channels (bold and regular characters denote vectors and scalars); note that 1-st and N-th hops are vector channels, all others are scalars, \(\xi_i \sim CN(0, \sigma_i^2)\), \(i = 1, \ldots, N-1\), \(\xi_N \sim CN(0, \sigma_N^2)\), \(I\) is identity matrix, and \(\ast\) denotes Hermitian conjugation. Without loss of generality, the relay power gains \(K_i\) are further absorbed into \(h_2 \cdots h_N\) via the substitution \(h_i \to \sqrt{K_{i-1}} h_i\).

To find the capacity of the channel in \((1)\), consider the sufficient statistics for \(y\), which is (see e.g. [6][7]),

\[
z = |h_N|^{-1} h_N^\ast y = |h_N|^2 h_N \ast x + |h_N| \cdots |h_2| \ast y_2 + \ldots + |h_N| \cdots |h_{i-1}| \ast y_{i-1} + |h_N| \cdots |h_2| \ast \xi_2 + \ldots + |h_{N-1}| \cdots |h_2| \ast \xi_2 + \ldots + |h_N| \cdots \cdots |h_1| \ast \xi_1
\]

where \(|h|^2 = h^\ast h\). The 1st term represents the signal received at the destination, and the other terms represent the relay noise propagated to the destination and the destination own noise. Note that sufficient statistics preserves the mutual information and capacity, as well as error rate under maximum-likelihood decoding, so that \((2)\) is a scalar channel equivalent to \((1)\). From this, the SNR at the destination can be expressed as

\[
g_N \cdots g_2 h_i^\ast R_s h_i / \sigma_i^2 \leq \sigma_s^2 \prod_{k=1}^N g_k / \sigma_k^2 = \gamma_{AF}
\]

where \(R_s = E[x^\ast x]\) is the covariance matrix of the source transmitted signal, \(\sigma_s^2 = tr R_s = E[x^\ast x]\) is the total power, \(g_k = |h_k|^2\) is the power gain of k-th hop channel, and \(\sigma_k^2 = \sigma_N^2 + \sum_{k=1}^{N-1} |h_k|^2 \gamma_k\) is the total noise power at the destination. The inequality in \((3)\) follows from \(h_i^\ast R_s h_i \leq |h_i|^2 \sigma_s^2\), and the equality is achieved when \(R_s = \sigma_s^2 h_i^\ast h_i / |h_i|^2\), i.e. beamforming from the source to the 1-st relay \(^2\), \(x = s \cdot h_1 / |h_1|\), where \(s\) is the scalar transmitted symbol of the total power \(\sigma_s^2\). From \((2)\), achieving the upper bound in \((3)\) also requires maximum ratio combining (MRC, also known as spatial matched filtering) at the destination.

The AF relay channel capacity, i.e. the ultimate bound to error-free data transmission under the AF protocol, can now be compactly expressed from the upper bound in \((3)\) as

\[
C_{AF} = \log(1 + \gamma_{AF}) \text{[bit/sHz]}
\]

where \(\gamma_{AF} = (\sum_{i=1}^{N} \gamma_i^{-1})^{-1}\) is the equivalent SNR, i.e. an SNR in the AWGN channel that has the same capacity as the N-hop AF relay channel above, and \(\gamma_i = g_1 \cdots g_i \sigma_i^2 / \sigma_s^2\) is the SNR at i-th relay if all previous relays were noiseless \((i = N\) corresponds to the destination), which is also the SNR at the destination when all the relays but i-th, and also the destination are noiseless. The capacity in \((4)\) is achieved by the beamforming at the source (towards 1-st relay), the MRC at the destination and using a temporal capacity-achieving code on the equivalent scalar channel. Since \(\gamma_{AF}\) is an increasing function of \(\gamma_i\), each AF relay has to use the maximum available power/gain.

B. Decode-and-Forward Protocol

The system model is in this case,

\[
y_1 = h_1 x_1 \cdots x_{i-1} + \cdots + \xi_N
\]

where \(x_1 = \xi_1, y_2 = h_2 x_1, \ldots\), \(y = h_N x_{N-1} + \xi_N\), \((5)\). The AF relay channel capacity, i.e. the ultimate bound to error-free data transmission under the AF protocol, can now be compactly expressed from the upper bound in \((3)\) as

\[
C_{DF} = \min_{i=1, \ldots, N} \{C_i\} = \log(1 + \gamma_{DF})
\]

where \(C_i = \log(1 + \gamma_i)\) is the capacity of i-th hop, \(\gamma_i = g_1 \cdots g_i \sigma_i^2 / \sigma_s^2\) is the SNR at i-th relay, and \(\gamma_{DF} = \min_{i=1, \ldots, N} \{\gamma_i\}\) is the effective SNR of an AWGN channel of the same capacity as the DF relay channel. This is also the relay channel capacity for the linear topology in Fig. 1. To make a fair comparison, we have set the transmitted signal power at each relay node to be equal to that in the AF protocol, i.e. \(g_1 \cdots g_i \sigma_i^2\) for i-th relay \(^3\). Note that the capacity in \((6)\) is achieved by beamforming at the source and MRC at the destination, in combination with a temporal capacity-achieving code for each hop (unlike the AF protocol, where a single temporal code is required for the whole channel).

III. COMPARISON OF AF AND DF PROTOCOLS

We are now in a position to compare the capacities in \((4)\) and \((6)\), which represent the ultimate rate bounds for coded systems under these protocols.

\(^2\)When no source CSI is available, the best transmission strategy is isotropic signaling, \(R_s = \sigma_s^2 I / n_s\) (see e.g. [6][7]), where \(n_s\) is the number of source antennas, which achieves \(\gamma_{AF} / n_s\) at the destination, so that all the results below will hold with appropriately scaled SNR.

\(^3\)The signal power constraint is the standard one for the RF/microwave amplifier design [11]. Some authors set the same total (signal + noise) power. Our analysis can be extended to that scenario as well. In particular, our lower bounds will hold and also the upper bounds in the high SNR regime.
Proposition 1: The capacity gain $\Delta C = C_{DF} - C_{AF}$ of the DF relaying over the AF one in a N-hop relay channel in Fig. 1 is bounded as follows:

$$0 \leq \Delta C \leq \log N \quad (7)$$

for any channel realizations $h_1, h_2, \ldots, h_N$. When $C_{DF} < \infty$, the lower bound is achieved if and only if all hops are equally strong and the SNR is sufficiently high, $\gamma_1 = \gamma_2 = \ldots = \gamma_N \to \infty$.

Proof: Since $\Delta C = \log [(1 + \gamma_{DF})(1 + \gamma_{AF})^{-1}]$, the lower bound follows from,$$
\gamma_{AF} = (\gamma_1^{-1} + \ldots + \gamma_N^{-1})^{-1} \leq \min(\gamma_1, \ldots, \gamma_N) = \gamma_{DF}
$$
where the equality is achieved if and only if $\gamma_k < \infty$ for some $k$ and $\infty$ otherwise, provided that $\gamma_{DF} < \infty$. To prove the upper bound, let $\gamma_{[1]} \leq \gamma_{[2]} \leq \ldots \leq \gamma_{[N]}$ be the ordered SNRs. Note that $\gamma_{AF} \geq \gamma_{[1]}/N$ and $\gamma_{DF} = \gamma_{[1]}$, so that

$$\Delta C \leq \log N + \log[(1 + \gamma_{[1]})(N + \gamma_{[1]}^{-1})] \leq \log N$$

The 1st inequality becomes equality when $\gamma_1 = \gamma_2 = \ldots = \gamma_N$ and the 2nd one – when $\gamma_{[1]} \to \infty$.

It follows from Proposition 1 that $C_{DF} \geq C_{AF}$, i.e. the AF relaying is sub-optimal in general (for arbitrary channel realization). In practical terms, $C_{DF} \approx C_{AF}$, i.e. no significant gain is provided by the DF relaying over AF one, when one relay (or the destination) is much weaker than the others, $\gamma_{[1]} \ll \gamma_{[2]}$. Under such condition, the AF relaying is a preferable solution since DF one incurs a significant complexity penalty in the relay design and implementation, and its capacity $C_{DF} = \log(1 + \gamma_{[1]}) \approx C_{AF}$ is almost the same as that of the AF one (dominated by the weakest hop).

The gain is maximum when all hops are equally strong and, for a two-hop channel, it never exceeds 1 bit/s/Hz.

The advantage of the DF relaying over the AF one can also be cast in terms of an SNR gain. Define the SNR gain $G$ of the DF relaying from the following:

$$C_{AF}(G\gamma_0) = C_{DF}(\gamma_0) \quad (8)$$

where $\gamma_0 = \sigma_2^2/\sigma_N^2$ is the source SNR. It shows how much more SNR (or, equivalently, the source power) is required for the AF relaying to achieve the same capacity as the DF relaying. This gain can be characterized as follows.

Proposition 2: The SNR gain of DF relaying over AF one for the N-hop relay channel in Fig. 1 is bounded as follows:

$$1 \leq G \leq N \quad (9)$$

and the lower bound is achieved when the destination and all the relays but one are noiseless. The upper bound is achieved when all hops are equally strong, $\gamma_1 = \gamma_2 = \ldots = \gamma_N$.

Proof: Follows along the same line as that of Proposition 1 using $G = \gamma_{DF}/\gamma_{AF}$.

We note that an equivalent form of (9) is

$$C_{AF}(\gamma_0) \leq C_{DF}(\gamma_0) \leq C_{AF}(N\gamma_0) \quad (10)$$

and the condition for achieving the upper bound in (9) can be re-written as follows,

$$\sigma_N^2 = g_N\sigma_{k-1}^2 = \ldots = g_N g_{N-1} \ldots g_2 \sigma_1^2 \quad (11)$$

i.e. the destination own noise power equals to the noise power of any relay propagated to the destination. This balance of all noise powers is required to exploit the full advantage of the DF relaying. When all noise powers are the same, i.e. $\sigma_N^2 = \ldots = \sigma_1^2$, this reduces to $g_N = \ldots = g_2 = 1$, which is equivalent to $|h_i|^2 K_i = 1$, i.e. each relay compensates for the path loss of the hop following it.

An approximate (practical) condition for achieving the lower bound is $\gamma_{[1]} \ll \gamma_{[2]}$. For a two-hop channel, this reduces to $\sigma_N^2 \gg g_2 \sigma_1^2$ or $\sigma_N^2 \ll g_2 \sigma_1^2$, i.e. either the destination is significantly more noisy than the relay (as seen at the destination) or the other way around \footnote{e.g. when there is a line-of-sight (LOS) link between the base station and the relay, but no LOS between the relay and a mobile user.}, which is illustrated in Fig. 2. This is the case when the DF relaying is not worth the complexity effort. In any case, the DF relaying outperforms the AF relaying by not more than 1 bit/s/Hz or 3 dB in a two-hop channel.

To capture the conditions when the upper and lower bounds are achieved in Propositions 1 and 2, we introduce the following definition.

Definition 1: The relay channel in Fig. 1 is balanced when $\gamma_1 = \gamma_2 = \ldots = \gamma_N$ (the upper bounds are achieved in (9), and in (7) when the SNR is sufficiently high). The relay channel is misbalanced when $\gamma_1/\gamma_{[2]} \to 0$ (the lower bounds are achieved in (7) and (9)).

While Propositions 1 and 2 compare the DF and AF relaying for any given channel realization (or a fixed channel), the corresponding comparison can also be made for randomly-block-fading channels in terms of their two main performance metrics, outage probability and outage capacity [7]. In particular, the SNR gain bounds in (9) also apply to the latter metrics, and also to the ergodic capacity. The outage probability is
defined as [7],

$$P_{\text{out}}(\gamma_0) = \Pr \{ C(\gamma_0) < R \}$$ (12)

i.e. the probability that the channel is not able to support the target rate $R$, where $C(\gamma_0)$ is the instantaneous channel capacity. We further comment that the channel outage probability is also the best achievable codeword error probability. It follows from (10) that, for any fading distribution,

$$P_{\text{AF}}(N\gamma_0) \leq P_{\text{DF}}(\gamma_0) \leq P_{\text{out}}(\gamma_0)$$ (13)

From this, the DF relaying has at most $N$-fold SNR gain over the AF relaying in terms of the outage probability, i.e. as in (9), and the DF and AF relaying have the same diversity gain (under arbitrary fading distribution). The outage capacity is the largest rate such that the outage probability does not exceed a given threshold $\varepsilon$ [7],

$$C_\varepsilon(\gamma_0) = \max \{ R : P_{\text{out}}(\gamma_0, R) \leq \varepsilon \}$$ (14)

where $P_{\text{out}}(\gamma_0, R)$ denotes the outage probability as a function of the SNR $\gamma_0$ and the target rate $R$. Using this definition and (13), it follows that

$$C_\varepsilon^{\text{DF}}(\gamma_0) \leq C_\varepsilon^{\text{AF}}(\gamma_0) \leq C_\varepsilon^{\text{AF}}(N\gamma_0)$$ (15)

so that the SNR gain of the DF relaying in terms of the outage capacity does not exceed $N$ either. From (10), the same inequality also holds in terms of the ergodic (mean) capacity $C(\gamma_0)$: $C_{\text{AF}}(\gamma_0) \leq C_{\text{DF}}(\gamma_0) \leq C_{\text{AF}}(N\gamma_0)$.

Note that the results above hold for arbitrary fading distribution. They can also be extended to more complicated relay channel topologies under selection relaying, which is done in the next section.

IV. SELECTION RELAYING

Let us now consider a two-hop relay channel under selection relaying, where only the best relay node (out of $N$) is used at any time. It is motivated by its low complexity and also by the fact that little interference is created to other users since only one relay is transmitting [9]. Below, we compare the DF and AF protocols for this selection relaying scheme.

**Proposition 3:** The capacity gain $\Delta C = C_{\text{DF}} - C_{\text{AF}}$ and the SNR gain $G = \gamma_{\text{DF}}/\gamma_{\text{AF}}$ of the DF relaying over the AF relaying in a two-hop selection relaying channel are bounded as follows:

$$0 \leq \Delta C \leq 1 \text{ bit/s/Hz}, \quad 1 \leq G \leq 2$$ (16)

for any channel realization. The lower bounds are achieved when either the destination or the best DF relay is noiseless, i.e. the channel is misbalanced. The upper bounds are achieved when the best AF and DF relays are the same, their two hops are equally strong, i.e. the channel is balanced, $\gamma_1 = \gamma_2$ and, for the capacity gain, the SNR is sufficiently high, $\gamma_1 = \gamma_2 \rightarrow \infty$.

Note that the bounds in (16) are independent of the total number of nodes $N$ (out of which the best one is selected) and are the same as in (7) and (9) for $N = 2$, i.e. selection relaying in a two-hop channel does not improve the maximum possible gain of the DF over AF relaying.

We note that these selection relaying results can also be extended to the $N$-hop selection relaying, as in Fig. 3, using the same reasoning as above.

**Proposition 4:** The capacity gain $\Delta C$ and the SNR gain $G$ of the DF relaying over the AF relaying in an $N$-hop selection relaying channel in Fig. 3 are bounded as follows:

$$0 \leq \Delta C \leq \log N, \quad 1 \leq G \leq N$$ (17)

The lower bounds are achieved when the destination and all the relays but one of the best DF relaying path are noiseless (misbalanced channel). The upper bounds are achieved when the best DF and AF relaying paths are the same and all the hops of the best path are equally strong (balanced channel) and, for the capacity gain, the SNR is sufficiently high, $\gamma_1 = \gamma_2 = \cdots = \gamma_N \rightarrow \infty$.

Finally, we point out that similar results also hold for a hybrid relaying scheme, where some of the relay nodes are DF ones and the rest are AF, and that all the bounds also extend to the outage probability/capacity and the ergodic capacity under arbitrary fading distribution, as discussed in Section III.

REFERENCES


Abstract—This work identifies the computational reserves required for the maximum likelihood (ML)-based sphere decoding solutions that achieve, in the high-rate and high-SNR limit, a vanishing gap to the error-performance of the optimal brute force ML decoder. These error performance and complexity guarantees hold for most multiple-input multiple-output scenarios, all reasonable fading statistics, all channel dimensions and all full-rate lattice codes. The analysis also identifies a rate-reliability-complexity tradeoff establishing concise expressions for all full-rate lattice codes. The analysis also identifies a rate-costs, all reasonable fading statistics, all channel dimensions and guarantees hold for most multiple-input multiple-output scenarios.

I. INTRODUCTION

For multiple-input multiple-output (MIMO) systems, error probability and encoding-decoding complexity are widely considered to be two limiting and interrelated bottlenecks (cf. [1], [2]). Specifically, if a small gap to the brute-force maximum likelihood (ML) performance is acceptable then different branch-and-bound algorithms such as the sphere decoder (SD) have been known to provide a complexity-performance tradeoff. Albeit suboptimal in terms of the error-performance, these SD based solutions might result in a preferable rate-costs, although this could be relaxed. Combining (1) and (2) yields the equivalent system model

\[ y = Ms + w, \]

where \( M \triangleq \rho \hat{R}^T \Sigma H \in \mathbb{R}^{n \times k}. \)

B. Sphere Decoder

Let QR = M be the thin QR factorization of the code-channel matrix M and r \( \triangleq Q^H y, \) then (3a) yields \( r = Rs + Q^H w \) and the ML decoder for this system takes the form

\[ \hat{s}_{ML} = \arg \min_{\hat{s} \in \mathcal{C}} \| r - R\hat{s} \|^2. \]

We use SD to implement the decoder in (4), which identifies as candidates the vectors \( \hat{s} \in \mathcal{C} \) that for some search radius \( \xi > 0, \) satisfy \( \| r - R\hat{s} \|^2 \leq \xi^2. \) The algorithm specifically uses the upper-triangular nature of R to recursively identify partial symbol vectors \( \hat{s}_k, k = 1, \ldots, \kappa, \) for which

\[ \| r_k - R_k \hat{s}_k \|^2 \leq \xi^2, \]

where \( \hat{s}_k \) and \( r_k \) respectively denote the last \( k \) components of \( \hat{s} \) and \( r, \) and where \( R_k \) denotes the \( k \times k \) lower-right submatrix of R.

We note that the error performance and the total number of visited nodes is a function of the search radius \( \xi. \) We use fixed search radius \( \xi = \sqrt{\log \rho} \) for some \( z > kd(r) \) such that

\[ P(\| Q^H w \|^2 > \xi^2) \leq P(\| w \|^2 > \xi^2) \leq \rho^{-d(r)}, \]

which implies a vanishing probability of excluding the transmitted information vector from the search. We use \( \tilde{z} \) to denote the exponential equality, i.e., we write \( f(\rho) \leq \rho^{\tilde{z}} \) to denote

\[ \lim_{\rho \to \infty} \log \frac{f(\rho)}{\log \rho} = B, \]

and \( \leq, \geq \) are defined similarly.

Complexity analysis for ML-based sphere decoder achieving a vanishing performance-gap to brute force ML decoding

Arun Singh and Petros Elia
Mobile Communications Department
EURECOM, Sophia Antipolis, France
Email: {singhak, elia}@eurecom.fr

Joakim Jaldén
ACCESS Linnaeus Center, Signal Processing Lab
School of Electrical Engineering, KTH, Stockholm, Sweden
Email: jalden@kth.se
C. Rate-reliability-complexity tradeoff in outage-limited MIMO communications

In the high SNR regime, a given encoder $X$ and decoder $D$, are said to achieve a multiplexing gain $r$ and diversity gain $d_{O}(r)$ if (cf. [1])

$$\lim_{\rho \to \infty} R(\rho) = r, \quad \text{and} \quad -\lim_{\rho \to \infty} \frac{\log P_{e}}{\log \rho} = d_{O}(r)$$

(7)

where $P_{e}$ denotes the probability of codeword error with a ML-based sphere decoder $D$, employing time-out policies.

We characterize complexity in terms of the complexity exponent (cf. [4], [6]). Let $N_{\text{max}}$ denote the amount of computational reserves, in floating point operations (flops) per $T$ channel uses, that the transceiver is endowed with, in the sense that after $N_{\text{max}}$ flops, the transceiver must simply terminate, potentially prematurely and before completion of its task. The complexity exponent then takes the form

$$c(r) := \lim_{\rho \to \infty} \frac{N_{\text{max}}}{\log \rho}.$$

(8)

We note that the complexity exponent is intimately intertwined with the achievable error performance and that any attempt to reduce $c(r)$ may be at the expense of a substantial degradation in error-performance.

For ML-based SD a vanishing performance gap to ML can, in the high SNR regime, be quantified as

$$P(\hat{s}_{\text{ML}} \neq s) \leq \rho^{-d(c)}$$

where $P(\hat{s}_{\text{ML}} \neq s) \leq \rho^{-d(c)}$ describes the error probability of the brute force ML decoder, and where $c$ is the complexity exponent that describes the computational resources required to achieve this performance gap. Generally a smaller performance gap requires a larger complexity exponent.

At this point a natural question to ask would be - how large computational reserves are required to achieve a vanishing gap to the brute force ML performance. While this question was first addressed and partially answered in [4] for the specific settings of i.i.d. Rayleigh fading quasi-static channels with specific channel dimensions, specific codes, and specific permutation orderings, we here provide answers for the most general MIMO settings, i.e., for all reasonable fading statistics, all channel dimensions, all MIMO scenarios and all full-rate lattice codes.

II. COMPLEXITY OF ML-BASED SPHERE DECODING

The total number of visited nodes is commonly taken as a measure of the sphere decoder complexity which is given by

$$N_{SD} = \sum_{k=1}^{\kappa} N_{k},$$

(10)

where $N_{k}$ denotes the number of visited nodes at layer $k$ that corresponds to the $k$th component of the transmitted symbol vector $s$ and is given by $N_{k} \triangleq |N_{k}|$ where $N_{k} \triangleq \{|s_{k} \in \mathbb{C}^{l} \mid \|s_{k} - R_{k}s_{k}\|^{2} \leq \xi^{2}\}$.

At this point we want to clarify that the analysis presented here is specific to sphere decoding, and that it does not account for any other ML based solutions that could, under some (arguably rare) circumstances, be more efficient. A classical example of such rare circumstances would be a MIMO scenario, or equivalently a set of fade statistics, that always generate diagonal channel matrices.

We are interested in the ML-based SD complexity required to achieve a vanishing gap to brute force ML performance. We recall that a ML-based SD with run-time constraints, in addition to making the ML errors $(\hat{s}_{\text{ML}} \neq s)$, also makes errors when the run-time limit of $\rho^{*}$ flops for $x > c(r)$ becomes active, as well as when the fixed search radius $\kappa$ causes $N_{k} = 0$. Consequently the corresponding performance gap to the brute force ML decoder, takes the form (cf. (9))

$$g(x) = \lim_{\rho \to \infty} \frac{P(\{N_{SD} \geq \rho^{*}\} \cup \{N_{k} = 0\})}{P(\hat{s}_{\text{ML}} \neq s)} = 0.$$
we have that \( \hat{c}(r) = \sum_{i=1}^{n} \min \left( \frac{rT}{\kappa} - \frac{1}{2}(1 - \mu^*_i), \frac{rT}{\kappa} \right) \). Furthermore given the monotonicity of the rate function \( I(\mu) \), and the fact that the objective function in (12) does not increase in \( \mu_i \) beyond \( \mu_i = 1 \), we may also assume without loss of generality that \( \mu^*_i \leq 1 \) for \( i = 1, \ldots, \kappa \). It then follows that

\[
\tau(r) \leq \hat{c}(r) = \sum_{i=1}^{q} \left( \frac{rT}{\kappa} - \frac{1}{2}(1 - \mu^*_i) \right) . \tag{13}
\]

### A. Universal Lower Bound on Complexity

In this section we establish that \( \hat{c}(r) = \hat{c}(r) \), i.e., the sphere decoder visits a total number of nodes that is close to \( \rho^{\hat{c}(r)} \) with a probability that is large compared to the probability of decoding error \( P(\hat{s}_{ML} \neq s) \) to lead.

Let \( q \in [1, \kappa] \) be the largest integer for which \( \frac{rT}{\kappa} - \frac{1}{2}(1 - \mu^*_i) > 0 \), in which case (13) takes the form

\[
\hat{c}(r) = \sum_{i=1}^{q} \left( \frac{rT}{\kappa} - \frac{1}{2}(1 - \mu^*_i) \right) . \tag{14}
\]

We quickly note that without loss of generality we can assume that \( q \geq 1 \) as otherwise \( \tau(r) = c(r) = 0 \). Consequently it is the case that \( \mu^*_i > 0 \) for \( i = 1, \ldots, q \).

We proceed to define three events \( \Omega_1, \Omega_2 \) and \( \Omega_3 \) which we will prove to be jointly sufficient so that the total number of nodes visited by the sphere decoder, employing a channel dependent fixed decoding order, is close to \( \rho^{\hat{c}(r)} \). These events are given by

\[
\Omega_1 \triangleq \{ \mu^*_i - 2\delta < \mu_i < \mu^*_i - \delta, j = 1, \ldots, q \},
\]

for a given small \( \delta > 0 \),

\[
\Omega_2 \triangleq \left\{ \|w\|^2 < \frac{\epsilon^2}{\kappa} \right\}, \tag{16}
\]

\[
\Omega_3 \triangleq \left\{ \|s\| < \frac{1}{2} \rho^{\epsilon T} \right\}. \tag{17}
\]

Note also that by choosing \( \delta \) sufficiently small, and using the fact that \( \mu^*_i > 0 \) for \( i = 1, \ldots, q \), we may without loss of generality assume that \( \Omega_1 \) implies that \( \mu_i > 0 \) for all \( i = 1, \ldots, \kappa \).

Following the footsteps of [4, Lemma 2] it can be shown that in the presence of events \( \Omega_1, \Omega_2 \) and \( \Omega_3 \) which we can remove the ML-based SD boundary constraints \( Z^p \) (cf.(4)). This removal allows us to lower bound the number of nodes visited at layer \( k \) as \( \kappa \) (cf. (4), Lemma 1))

\[
N_k \geq \sum_{i=1}^{k} \left\lfloor \frac{2\epsilon}{\sqrt{k} \sigma_i(R_k)} - \sqrt{k} \right\rfloor . \tag{18}
\]

In the following, and up until (28), we will work toward upper bounding \( \sigma_i(R_k) \) for the case of \( q \in [1, \kappa \cdot 1] \), the case of \( q = \kappa \) is treated separately later on. Towards this we first consider a Greedy QR decomposition. The diagonal elements of \( \tilde{R} \) satisfy \( \tilde{r}_{11} \geq \cdots \geq \tilde{r}_{pp} \). Let \( M_{ip} \in \mathbb{R}^{p \times p} \) contains the first \( p \) columns of \( M \). It then follows that

\[
\tilde{M}_{ip} = \tilde{M}_{ii} = \tilde{Q} \tilde{R}_i p , \tag{19}
\]

where, \( \Pi_i \) and \( \tilde{R}_i p \) denote the sub matrices consisting of the first \( p \) columns of \( \Pi \) and \( \tilde{R} \) respectively. Now let \( R_k \) be the \( p \times p \) upper triangular matrix consisting of the first \( p \) rows of \( R_k \), then we get that \( \sigma_i(M_{ip}^p(M_{ip})) = \sigma_i(R_k^p R_k) \) for \( i = 1, \ldots, p \). For \( R_k^p R_k \), having diagonal entries \( \tilde{r}_{22}^p \geq \cdots \geq \tilde{r}_{pp}^p \) and singular values \( \sigma_i(M_{ip}^p(M_{ip})) \leq \cdots \leq \sigma_p(M_{ip}^p(M_{ip})) \), we have that for \( k = 1, \ldots, \rho \) then ([8, Theorem 2.3])

\[
\prod_{i=1}^{k} \tilde{r}_{ii}^p \leq \prod_{i=1}^{p} \sigma_{p-i+1}(M_{ip}^p M_{ip}). \tag{20}
\]

From [7, Lemma 4.3] regarding the Greedy QR decomposition, we have that \( \tilde{r}_{kk}^p \geq \frac{\sigma_{k-1+p}(M_{ip}^p M_{ip})}{\kappa - i + 1} \).

Consequently we have that for \( k = 1, \ldots, p \)

\[
\sigma_{p-k+1}(M_{ip}^p M_{ip}) \geq \sigma_{k-1+p}(M_{ip}^p M_{ip}) \prod_{i=1}^{k} \frac{1}{2m_i - i + 1}. \tag{22}
\]

We then have \( \sigma_{p-k+1}(M_{ip}^p M_{ip}) \leq \sigma_{k+1-p}(M_{ip}^p M_{ip}) \), \( k = 1, \ldots, p \) (cf. [9, Theorem 4.3.15]), and it follows that for \( k = 1, \ldots, p \)

\[
\sigma_{p-k+1}(M_{ip}^p M_{ip}) \leq \sigma_{k-1+p}(M_{ip}^p M_{ip}) \tag{23}
\]

Recalling that \( \sigma_1(M_{ip}^p M_{ip}) \leq \cdots \leq \sigma_p(M_{ip}^p M_{ip}) \), we have that

\[
\sigma_i(M_{ip}^p M_{ip}) \leq \sigma_i(M_{ip}^p M_{ip}) \tag{24}
\]

for \( i = 1, \ldots, q \), where exponential equality follows from (23). From (3b) for \( i = 1, \ldots, \kappa \), we have that

\[
\sigma_i(M) \leq \rho^{-\frac{\epsilon T}{\kappa} + \frac{1}{2}(1 - \mu_i)} . \tag{25}
\]

Furthermore (15) gives that

\[
\sigma_i(M) \leq \rho^{-\frac{\epsilon T}{\kappa} + \frac{1}{2}(1 - \mu_i)} \quad \text{for} \quad i = 1, \ldots, q \tag{26}
\]

\[
\sigma_i(M) \leq \rho^{-\frac{\epsilon T}{\kappa} + \frac{1}{2}(1 - \mu_i)} \leq \rho^{-\frac{\epsilon T}{\kappa} + \frac{1}{2}}, \tag{27}
\]

\[
\sigma_{q+1}(M) \leq \rho^{-\frac{\epsilon T}{\kappa} + \frac{1}{2}(1 - \delta)} . \tag{28}
\]

Substituting (27) in (28) gives that

\[
\sigma_i(R_k) \leq \rho^{-\frac{\epsilon T}{\kappa} + \frac{1}{2}(1 - \mu_i)} , \quad i = 1, \ldots, q . \tag{29}
\]
Consequently, going back to (18), we have that
\[
\left[ \frac{2\varepsilon}{\sqrt{k\sigma_s(R)}}, \sqrt{k} \right]^+ \geq \rho \left( \frac{\varepsilon}{2}, \frac{\varepsilon}{2} (1-\mu) \right),
\]
(29)
As a result, for \( k = q \) with \( q \in [1, K-1] \) we have that
\[
N_q \geq \rho \sum_{i=1}^q \left( \frac{\varepsilon}{2} - \frac{\varepsilon}{2} (1-\mu) \right) = \rho \varepsilon \sum_{i=1}^q (1-\mu) = \rho \varepsilon \theta q,
\]
where the last equality follows from (14). For the case of \( q = \kappa \), from (18) and (27a) we have that
\[
N_q \geq \rho \sum_{i=1}^\kappa \left( \frac{\varepsilon}{2} - \frac{\varepsilon}{2} (1-\mu) \right) = \rho \varepsilon \kappa.
\]
Consequently for \( q \in [1, \kappa] \) we have that \( N_{SD} \geq \rho \varepsilon (r-Kd) \) for small \( \delta > 0 \), where \( K \in (\frac{q}{2}, q, \kappa) \).

We note that (15)-(17) jointly imply that \( N_{SD} \geq \rho \varepsilon (r-Kd) \) for some \( \delta' \neq Kd + \delta_1 \), where \( \delta > \delta_1 > 0 \), it follows that
\[
P \left( N_{SD} \geq \rho \varepsilon (r-Kd) \right) \geq P(\Omega_1 \cap \Omega_2 \cap \Omega_3) = P(\Omega_1) \] (32)
where exponential equality follows from the independence of the events \( \Omega_1, \Omega_2 \) and \( \Omega_3 \) and from the fact that \( P(\Omega_2) \approx \rho \theta \) (cf.(6)) and \( P(\Omega_3) \approx \rho \theta \). With \( \Omega_1 \) being an open set, we have that
\[
\lim_{\rho \to \infty} \frac{P(\Omega_1)}{\log \rho} \leq \inf_{\mu \in \Omega_1} I(\mu) = I(\tilde{\mu}) < I(\mu^*) = d(r),
\]
(33)
where \( \tilde{\mu} = \{ \mu^*_1 - 2\delta, \ldots, \mu^*_\kappa - 2\delta, 0, \ldots, 0 \} \), the last inequality follows from the monotonicity of the rate function \( I(\mu) \) and where the last equality follows from the fact that, by definition, \( I(\mu^*) = d(r) \).

Consequently (32) and (33) along with the definition of the lower bound in (11b) imply that \( \varepsilon (r) = \tilde{\varepsilon}(r) \), for arbitrarily small \( \delta > 0 \). The following lemma directly holds corresponding to a vanishing performance gap.

**Lemma 1:** Irrespective of channel fading statistics and of the full-rate code applied, for every realization of channel \( M \) there exists a channel dependent column permutation matrix \( \Pi \) such that the ML-based sphere decoder with decoding order \( \Pi \) has the complexity exponent \( c(r) = \tilde{c}(r) \).

To show the dependence of \( \Pi \) on \( M \), we henceforth use \( \Pi_M \) instead of \( \Pi \). Under the assumption that each column permutation matrix ‘appears’ with non-zero probability, then for every column permutation matrix \( \Pi_\kappa \in \mathbb{R}^{K \times \infty} \) we have that \( P(\Pi_M = \Pi_\kappa) \approx \rho \theta \), where probability is taken over random \( M \). Then the following theorem is a consequence of Lemma 1.

**Theorem 1:** For any full-rate code and fading distribution such that \( P(\Pi_M = \Pi_\kappa) > \varepsilon \) \( \forall \ \Pi_\kappa \), for some \( \varepsilon > 0 \), the complexity exponent of the ML-based sphere decoder with any fixed decoding order is given by \( c(r) = \tilde{c}(r) \).

**III. RATE-RELIABILITY-COMPLEXITY TRADEOFF**

In this section we present a rate-reliability-complexity tradeoff for ML-based SD, which identifies the optimal diversity gain achievable in the presence of any run-time constraint imposed due to the unavailability of enough computational resources required to achieve a vanishing gap. The proofs are omitted from this writeup due to the lack of space.

**Theorem 2:** For any full-rate code with ML-based diversity gain \( d(r) \) and any fading distribution such that \( P(\Pi_M = \Pi_\kappa) > \varepsilon \) \( \forall \ \Pi_\kappa \), for some \( \varepsilon > 0 \), the achievable diversity performance \( d_P(r) \) for ML-based SD with any fixed decoding order and a run-time constraint \( \rho d_P(r) \) flops, is uniquely described by
\[
d_P(r) = \min \{ d(r), m(\overline{d}(r), x), c_P(r) \} \leq \rho d_P(r) \geq 0,
\]
(34)
where \( m(\overline{d}(r), x) \) is the infimum of \( I(\mu) \)
\[\text{s.t.} \sum_{i=1}^K \left( \frac{r}{K} - \frac{1}{2} (1 - \mu_i) \right)^+ \geq c_P(r) + \varepsilon, \quad 1 \geq \mu_1 \geq \cdots \geq \mu_K \geq 0. \]

Example: For a \( 2 \times 2 \) i.i.d. Rayleigh channel with the \( 2 \times 2 \) Perfect code [5] and a ML-SD with run-time constraint of \( \rho \delta \) flops, the achievable diversity gain is depicted in Fig.1.

![Fig. 1. Achievable diversity gain for 2x2 Perfect codes.](image)

**IV. CONCLUSIONS**

The presented performance guarantees hold for the most general MIMO settings, i.e., for all reasonable fading statistics, all channel dimensions, all MIMO scenarios and all full-rate lattice codes. Such guarantees may be utilized for the practical implementation of telecommunication systems.

**REFERENCES**


Symbol-by-symbol APP Decoding based on Supercode Decoding

Jürgen Freudenberger
and Uwe Kaiser
HTWG Konstanz
University of Applied Sciences, Konstanz, Germany
Institute for System Dynamics (ISD)
Email: juergen.freudenberger@htwg-konstanz.de

Abstract—An efficient sub-optimum symbol-by-symbol APP decoding algorithm for linear binary block codes is presented. The three-step decoding procedure consists of: 1) list decoding of two supercodes, 2) calculating the intersection of these two lists, and 3) symbol-by-symbol APP decoding for the intersection. The presented results indicate that the proposed decoding procedure obtains good estimates of the ideal soft-output values and reduces the decoding complexity significantly. It may be an attractive alternative to conventional implementations of APP decoding.

I. INTRODUCTION

Most commonly, symbol-by-symbol a posteriori probability (APP) decoding of block codes is implemented by representing the code as a graph, the so-called trellis, and applying the well-known BCJR algorithm [1]. Alternative implementations of soft in/soft-out decoding are based on the soft-output Viterbi algorithm [2] or sub-optimum versions of the BCJR algorithm [3].

In this paper we consider a different approach which is based on the concept of supercode decoding introduced by Barg et al. [4]. The algorithm from [4] actually combines several previously known decoding strategies for discrete memoryless channels with the new idea of supercode decoding. A generalization of all employed decoding steps to channels with continuous output alphabets appears to be difficult. Therefore, we concentrate on the problem of supercode decoding. Consequently the intermediate decoding steps differ from that described in [4]. In particular we present efficient solutions to the problems: 1) list decoding of supercodes and 2) calculating the intersection \( L = \cap_i L_i \). The list \( L_i \) is represented as a trellis and the final decoding step 3) can be performed using, for example, Viterbi decoding in this trellis. Such a trellis based supercode decoding was devised and investigated for the binary symmetrical channel (BSC) in [5].

In this work we generalize trellis based supercode decoding to symbol-by-symbol soft in/soft-out a posteriori probability (APP) decoding for linear binary block codes, i.e. in the last decoding step we decode the trellis representing the intersection with the BCJR algorithm. Due to the trellis based approach we are not limited to the Hamming metric. Hence, the presented approach is suitable for soft in/soft-out decoding. It is inherently sub-optimum, because the APP decoding does not consider the complete code. Nevertheless, the presented simulation results indicate that the proposed decoding procedure obtains good estimates of the ideal soft-output values (similar to the well known Max-Log-MAP approximation [3]) and reduces the decoding complexity significantly.

In Section II, we briefly discuss the basic concept of supercode decoding. Sections III and IV describe a novel implementation of the trellis based supercode decoding-algorithm presented in [5]. When implementing an algorithm there is always a trade off between memory usage and speed. The new implementation reduces the runtime complexity compared to the approach presented in [5]. Finally, some simulation results are presented in section V.

II. BASIC CONCEPT

A supercode \( C_i \) of the binary block code \( C \) is a code containing all codewords of \( C \). For a linear code \( C \) with parity-check matrix \( H \), we can construct two supercodes \( C_1 \) and \( C_2 \) such that \( C = C_1 \cap C_2 \). Let

\[
H = \begin{pmatrix} H_1 \\ H_2 \end{pmatrix}
\]

be the parity-check matrix of the code \( C \), this means that \( H_1 \) and \( H_2 \) are two sub-matrices of \( H \). Then the sub-matrices \( H_1 \) and \( H_2 \) define the supercodes \( C_1 \) and \( C_2 \), respectively.

Example 1: Consider for example the code \( C = \{(0000), (1110), (1011), (0101)\} \) with parity-check matrix

\[
H = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix}
\]

We obtain

\[
H_1 = \begin{pmatrix} 1 & 1 & 0 & 1 \end{pmatrix}
\]

\[
C_1 = \{(0000), (1100), (1110), (0010), (1011), (1001), (1011), (0101)\}
\]

and

\[
H_2 = \begin{pmatrix} 0 & 1 & 1 & 1 \end{pmatrix}
\]

\[
C_2 = \{(0000), (1000), (0110), (1110), (1011), (1101), (0011), (0101)\}
\]
where the underlined vectors are the codewords of the code $C$.

Supercode decoding is performed in three steps:

1) List decoding of each supercode $C_i$ such that the transmitted codeword is in the list $L_i$, provided that the weight of the error pattern is less or equal $\rho$.

2) Calculating the intersection $L = \cap_i L_i$. These are the codewords of the code $C$ that are in the ball around the received sequence.

3) Applying symbol-by-symbol soft in/soft-out a posteriori probability decoding to the list $L_i$.

The idea here is that list decoding of the supercodes is less complex than a decoding approach which is based on the code itself. The supercodes are represented by their minimal trellises [6]. The first decoding step is performed by a marking algorithm which labels all state transitions in a supercode trellis that corresponds to a codeword in the list $L_i$.

The list $L$ is also represented by a trellis. This trellis is constructed using the merging algorithm. The merging algorithm calculates the intersection $L = \cap_i L_i$ by merging the labeled state transitions from both supercode trellises.

Before we describe the algorithm’s implementation we introduce the data structures which the trellis is composed of.

III. TRELLIS REPRESENTATION

A trellis $T = (S, W)$ is a directed graph with a set of nodes $S = \{\sigma\}$ and a set of branches $W = \{w\}$ which connects the nodes. The set $S$ can be split into $n + 1$ subsets $S = S_0 \cup S_1 \cup \cdots \cup S_n$, where $n$ denotes the length of the code. Such a subset is also called a level $S_i$ of the trellis. Each branch $w$ has a label $b \in \{0, 1\}$ which represents the code symbol $v_i(w_i)$.

To represent the trellis, three data structures are used. One for the node $\sigma$, a list of nodes for a level $S_i$, and a list of the list of nodes for the complete trellis $S$. The branches $W$ are references which are stored in the nodes.

A node can have up to two predecessors $N_0^B(\sigma)$ and successors $N_0^S(\sigma)$, $b \in \{0, 1\}$. The references to the predecessors and successors are assigned to zero if there is no corresponding node. Each incoming branch $w_i^0$, which connects $\sigma$ with $N_0^B(\sigma)$ and outgoing branch $w_i^1$ which connects $\sigma$ and $N_0^S(\sigma)$ has a metric. We differentiate between two metric values. The forward metric $\Lambda^F(\sigma)$, which is the metric between $\sigma$ and $N_0^S(\sigma)$, and the backward metric $\Lambda^B(\sigma)$, which is the metric between $\sigma$ and $N_0^B(\sigma)$. Furthermore a node stores the markings $M_b(\sigma)$ for the outgoing branches $w_i^b$. Hence, a node of the super code trellises can be described as a 10-tupel:

$$\sigma = (\Lambda^B_0(\sigma), \Lambda^B_1(\sigma), \Lambda^F_0(\sigma), \Lambda^F_1(\sigma), N_0^B(\sigma), N_0^S(\sigma), N_1^B(\sigma), M_0(\sigma), M_1(\sigma))$$

These nodes are used to construct the trellises of the supercodes. For the trellis representing the list $L$ we have to extend the node with four attributes. Two references $N^{(1)}(\sigma)$ and $N^{(2)}(\sigma)$ which point to the corresponding nodes in the supercode trellises. Additionally the level $L(\sigma)$ of the node and a visited flag $V(\sigma)$ were added to implement the breath first search efficiently. Thus, a node of the trellis can be described as a 14-tupel:

$$\sigma = (\Lambda^B_0(\sigma), \Lambda^B_1(\sigma), \Lambda^F_0(\sigma), \Lambda^F_1(\sigma), N_0^B(\sigma), N_0^S(\sigma), N_1^B(\sigma), N^{(1)}(\sigma), N^{(2)}(\sigma), \lambda, M_0(\sigma), M_1(\sigma), V(\sigma), L(\sigma))$$

The implementation uses three trellises for decoding. The two supercode trellises $T^{(1)}$ and $T^{(2)}$ for the supercodes $C_1$ and $C_2$, and the trellis $T$ for the code $C$. Each node $\sigma$ in $T$ has a reference to a node $\sigma^{(1)}$ in $T^{(1)}$ and $\sigma^{(2)}$ in $T^{(2)}$, such that

$$\sigma = \left( \begin{array}{c} \sigma^{(1)} \\ \sigma^{(2)} \end{array} \right)$$

IV. SUPERCODE DECODING

The first step of supercode decoding is list decoding of each supercode $C_i$ such that the transmitted codeword is in the list $L_i$ provided that the weight of the error pattern is less or equal $\rho$. With trellis based supercode decoding the list decoding is performed in the trellis of the supercodes and each list is represented by a marked trellis. Hence, the first decoding step is called the marking algorithm.

a) The marking algorithm: The marking algorithm is applied to the two supercode trellises.

Step 1: Set $\Lambda^F_b(\sigma_0) = \Lambda^F_b(\sigma_1) = 0$ and $M_0(\sigma_0) = M_1(\sigma_0) = 0$

Step 2: Calculate the forward metric for each node $\sigma_t$, $t = 1, \ldots, n$ when $N_b^B(\sigma_t) \neq 0, b = 0, 1$:

$$\Lambda^F_b(\sigma_t) = \Lambda^F_b(N_b^B(\sigma_t)) + \lambda(\omega_{t-1})$$

with

$$\Lambda^F_b(\sigma) = \begin{cases} \min(\Lambda^F_b(\sigma), \Lambda^F_b(\sigma)), & \text{if } N_0^B(\sigma) \neq 0 \land N_1^F(\sigma) \neq 0 \\ \Lambda^F_b(\sigma), & \text{if } N_0^B(\sigma) \neq 0 \land N_1^F(\sigma) = 0 \\ \Lambda^F_b(\sigma), & \text{if } N_1^B(\sigma) \neq 0 \land N_0^F(\sigma) = 0 \\ \Lambda^F_b(\sigma), & \text{if } N_1^B(\sigma) \neq 0 \land N_0^F(\sigma) \neq 0 \\ \lambda, & \text{if } N_0^B(\sigma) = 0 \land N_1^F(\sigma) = 0 \\ \lambda, & \text{if } N_1^B(\sigma) = 0 \land N_0^F(\sigma) = 0 \\ \lambda, & \text{if } N_0^B(\sigma) \neq 0 \land N_1^F(\sigma) \neq 0 \land N_0^B(\sigma) \neq 0 \land N_1^F(\sigma) = 0 \end{cases}$$

Reset all markings: $M_b(\sigma_t) = 0$

Step 2a: If $\Lambda^F_b(\sigma_n) \leq \rho$, stop decoding.

Step 3: Set $\Lambda^B_b(\sigma_0) = \Lambda^B_b(\sigma_1) = 0$

Step 4: Calculate the backward metric for each node $\sigma_t$, $t = n - 1, \ldots, 0$ when $N_b^B(\sigma_t) \neq 0, b = 0, 1$:

$$\Lambda^B_b(\sigma_t) = \Lambda^B_b(N_b^B(\sigma_t)) + \lambda(\omega_{t-1})$$

with

$$\Lambda^B_b(\sigma) = \begin{cases} \min(\Lambda^B_b(\sigma), \Lambda^B_b(\sigma)), & \text{if } N_0^B(\sigma) \neq 0 \land N_1^B(\sigma) \neq 0 \\ \Lambda^B_b(\sigma), & \text{if } N_0^B(\sigma) \neq 0 \land N_1^B(\sigma) = 0 \\ \Lambda^B_b(\sigma), & \text{if } N_1^B(\sigma) \neq 0 \land N_0^B(\sigma) = 0 \\ \Lambda^B_b(\sigma), & \text{if } N_1^B(\sigma) \neq 0 \land N_0^B(\sigma) \neq 0 \land N_1^B(\sigma) \neq 0 \land N_0^B(\sigma) = 0 \end{cases}$$

132
Set markings for $b = 0, 1$:

$$M_b(\sigma_t) = M(N^R_b(\sigma_t)) \land ((\Lambda^L(\sigma) + \Lambda^R(\sigma)) \leq \rho)$$

with $M(\sigma) = M_0(\sigma) \lor M_1(\sigma)$

Considering that each node of both supercode trellises has to be visited twice the runtime complexity of the marking algorithm is given by the size of the supercode trellises.

The second step of supercode decoding is the calculation of the intersection $L = \cap \mathcal{L}_i$. Again, the intersection $L = \cap \mathcal{L}_i$ is represented by a trellis. The intersection is calculated with the merging algorithm.

b) The merging algorithm (Breath first Viterbi):

Before we introduce the merging algorithm we have to define a queue. This queue is needed for the breath first search implementation. A queue is a list of $m$ nodes $Q = \langle \sigma_1, \sigma_{i+1}, \ldots, \sigma_{i+m-1} \rangle$ which can shrink and grow dynamically. The queue has two operations. One to add a node at the end $Q = enqueue(Q, \sigma) = \langle \sigma_1, \ldots, \sigma_{i+m-1}, \sigma \rangle$ and a second to remove the first node $Q = dequeue(Q) = \sigma_{i+1}, \ldots, \sigma_{i+m-1}, \sigma_{i+m}$.

Step 1: Set $\Lambda_b^L(\sigma_0) = \Lambda_b^L(\sigma) = 0$ and $Q = enqueue(Q, \sigma_0)$

Step 2: Breath first search

```
while(\( \sigma = dequeue(Q) \))
{
    if(\( L(\sigma) > 0 \))
    {
        \[ \Lambda_b^L(\sigma) = \Lambda_b^L(N^R_b(\sigma)) + \lambda(\omega_L(\sigma) - 1) \]
        \[ \Lambda_b^L(\sigma) = \Lambda_b^L(N^L_b(\sigma)) + \lambda(\omega_L(\sigma) - 1) \]
    }

    \[ M_0(\sigma) = M_0(N^{(1)}(\sigma)) \land M_0(N^{(2)}(\sigma)) \]
    \[ i \in M_0(\sigma) \land \neg V(N^R_0(\sigma)) \]
    \[ V(N^R_0(\sigma)) = 1 \]
    \[ Q = enqueue(Q, N^S_0(\sigma)) \]

    \[ M_1(\sigma) = M_1(N^{(1)}(\sigma)) \land M_1(N^{(2)}(\sigma)) \]
    \[ i \in M_1(\sigma) \land \neg V(N^L_1(\sigma)) \]
    \[ V(N^L_1(\sigma)) = 1 \]
    \[ Q = enqueue(Q, N^S_1(\sigma)) \]
}
```

Step 3: Find best path from $\sigma_{n-1}$ to $\sigma_0$ considering only the marked nodes.

Step 4: Reset markings and visited flags. (Second breath first search run, with inverted visited flag)

This algorithm avoids iterating over the supercode trellis nodes. In order to reduce the runtime complexity we introduce a pairing of supercode trellis nodes which can be done once before the actual decoding. The simplest way to do this is to create the trellis of the code $C$. To achieve that a node of the trellis is implicitly marked we add two references to the corresponding nodes in the supercode trellises. As a result, a node in the trellis is marked if both corresponding nodes in the supercode trellises are marked. To calculate the metrics of the marked nodes we cannot iterate over all trellis nodes to check if a node is marked. This would mean that the merging algorithm has the same runtime complexity as Viterbi decoding. Instead of iterating over all nodes we use a breath first search [7] to visit only the marked nodes. The breath first search was chosen because it traverses the trellis level by level. Consequently, the metrics can be calculated as in the marking algorithm.

V. SIMULATION RESULTS

In this chapter we present some simulation results for binary Reed-Muller code. We compare the trellis based supercode decoder with the Viterbi and BCJR algorithms.

The simulation results for APP decoding are presented in Fig. 1. The left diagram shows the bit error rates for the $R_M(3,6)$ code (with code length $n = 64$ and dimension $k = 42$). With $\rho = 5$ the APP decoding based on supercode decoding achieves almost BCJR-performance. With $\rho = 4$ the decoding performance is close to that of the Max-Log-MAP algorithm. In the right diagram we present the quality of the soft output values as the error compared to the bit error rate $P_b$. We estimate the bit error rate from the soft output values as

$$\hat{P}_b = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{1 + e^{|L_5|}}$$

where $L$ is a soft output value [8], [9]. The right diagram in Fig. 1 presents the relative error

$$\frac{|\hat{P}_b - P_b|}{P_b}$$

It can be seen that with $\rho = 5$ the quality of the soft output is comparable to the Max-Log-MAP algorithm.

Table I shows the number of nodes visited by the supercode decoding algorithm for different boundaries. Additionally the number of nodes visited by the Viterbi algorithm is presented. As said before, the complexity of the marking algorithm is independent of the input sequence and the bounded distance. Hence we can divide the complete decoding complexity in a static part, caused by the marking, and a dynamic part, caused by the merging. For $R_M(2,5)$ (with code length $n = 32$ and dimension $k = 16$) and $\rho = 4$ the number of visited nodes by the Viterbi and supercode decoding algorithm are almost similar. But for $R_M(3,6)$ and $\rho = 4$ the Viterbi algorithm has to visit eight times more nodes than the supercode decoding algorithm.

VI. CONCLUSIONS

In this work we have presented an efficient sub-optimum symbol-by-symbol soft in/soft-out a posteriori probability (APP) decoding algorithm for linear binary block codes. The new method is based on trellis based supercode decoding
which was introduced in [5]. The new implementation presented in this paper reduces the runtime complexity compared to the approach presented in [5].

The presented simulation results indicate that the proposed APP decoding procedure obtains good estimates of the ideal soft-output values and reduces the decoding complexity significantly compared to decoding in the complete syndrome trellis. It is therefore an attractive alternative to conventional implementations of APP decoding for linear binary block codes.

LDPC codes or turbo codes that are based on binary block codes usually employ high rate Hamming codes [10], [11], [12] that allow conventional low complexity APP decoding. The new algorithm may enable other code constructions that are based on block codes with higher error correcting capabilities.

REFERENCES

# Author Index

<table>
<thead>
<tr>
<th>Abbe, E.</th>
<th>Jaldén, J.</th>
<th>Schenk, A.</th>
<th>Shah, D.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altuğ, Y.</td>
<td>Kaiser, U.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Amir, E.</td>
<td>Karzand, M.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Belfiore, J.-C.</td>
<td>Khisti, A.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Blasco-Serrano, R.</td>
<td>Kim, Y.-H.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bossert, M.</td>
<td>Koch, T.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boston, N.</td>
<td>Kontoyiannis, I.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bross, S. I.</td>
<td>Kubin, G.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bustin, R.</td>
<td>Land, I.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conti, D.</td>
<td>Lapidoth, A.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cuff, P.</td>
<td>Levin, G.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dimakis, A. G.</td>
<td>Ling, C.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dörpinghaus, M.</td>
<td>Li, X.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Elia, P.</td>
<td>Loyka, S.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Erez, U.</td>
<td>Lucani, D. E.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Etezadi, F.</td>
<td>Macris, N.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feizi, S.</td>
<td>Mathar, R.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fischer, R. F. H.</td>
<td>Matsumoto, R.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fleury, B. H.</td>
<td>Médard, M.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Freudenberg, J.</td>
<td>Nazer, B.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gastpar, M.</td>
<td>Orlitsky, A.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Geiger, B.</td>
<td>Pedersen, T.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Goela, N.</td>
<td>Permuter, H.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Grant, A.</td>
<td>Rooij, S. de</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Huber, M.</td>
<td>Rüngeler, M.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jakobsen, M. L.</td>
<td>Saber Tehrani, A.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

135