Encoding for the Blackwell Channel with Reinforced Belief Propagation

Alfredo Braunstein
Institute for Scientific Interchange
Villa Gualino, viale S. Severo 65
10133, Turin, Italy
braunstein@isi.it

Farbod Kayhan
Institute for Scientific Interchange,
Politecnico di Torino,
10129, Turin, Italy
kayhan@isi.it

Guido Montorsi
Politecnico di Torino
Dipartimento di Elettronica
10129, Turin, Italy
Email: montorsi@polito.it

Riccardo Zecchina
International Center for Theoretical Physics,
Politecnico di Torino,
Institute for Scientific Interchange
Strada Costiera 11
I-34100, Trieste, Italy
Email: zecchina@ictp.it

Abstract—A key idea in coding for the broadcast channel (BC) is binning, in which the transmitter encode information by selecting a codeword from an appropriate bin (the messages are thus the bin indexes). This selection is normally done by solving an appropriate (possibly difficult) combinatorial problem. Recently it has been shown that binning for the Blackwell channel—a particular BC—can be done by iterative schemes based on Survey Propagation (SP). This method uses decimation for SP and suffers a complexity of $O(n^2)$. In this paper we propose a new variation of the Belief Propagation (BP) algorithm, named Reinforced BP algorithm, that turns BP into a solver. Our simulations show that this new algorithm has complexity $O(n \log n)$. Using this new algorithm together with a non-linear coding scheme, we can efficiently achieve rates close to the border of the capacity region of the Blackwell channel.

I. INTRODUCTION

Broadcast channels (BC) were first introduced and analyzed by Cover [7]. The general BC with $t$ receivers is depicted in Fig. 1. In a BC, a single transmitter sends simultaneously independent information to multiple receivers.

Coding for each receiver independently with a normal point-to-point code and sending the $t$ messages sequentially-by allocating proportions of time to each receiver- is known as time sharing strategy. It is shown in [7] that jointly optimized codes can have a larger capacity region for error-free communication than that of time sharing codes [1], [7], [8].

A key idea in coding for the BC is the binning strategy, which allows the transmitter to encode information by selecting a codeword from an appropriate bin. In this paper we deal with practical implementation of random binning for the BC. Existing practical binning schemes for BC are often based on structured codes and maximum likelihood algorithms. Martinian and Yedidia in [11] have used for the first time the random codes on graphs for quantization of a binary erasure source. Still their method works only for erasure sources and is not applicable to the general BC.

Recently, Wei Yu and M. Aleksic [18] showed that the binning problem for a particular BC, namely the Blackwell Channel (BC), when coding is performed by random low-density parity-check like codes, can be thought as a constraint satisfaction problem. They proposed an iterative encoder that works well at rates close to the border of the BC capacity region.

The main difference of this problem with that of decoding classical codes is that this combinatorial problem admits many solutions. In fact in these cases the application of BP allows to compute the cardinality of the solution space but fail to find a particular solution.

In [18] they use Survey Propagation (SP) algorithm for encoding, fixing one variable after each convergence (decimation). The main drawback of this method is the encoding complexity which grows as $O(n^2)$. Also the decimation works well only when the connectivity of XOR nodes are very small ($c = 2, 3$ and $4$).

In this paper we use a modified version of BP, called Reinforced Belief Propagation (RBP), originally proposed in the context of perceptron learning [3], which effectively turns BP into a solver. Experiments show that RBP does not converge for factor graphs with XOR function nodes. To overcome this, we propose a new class of sparse non-linear codes. These two modifications result in a more efficient encoding complexity (from $O(n^2)$ to $O(n \log n)$) and a lower Frame Error Rate (FER), i.e., the probability of not finding a solution to the
encoding problem.

This paper is organized as follows. In the next section we introduce the general framework of broadcast channels and their capacity regions. In section III we present the iterative updates for BP and RBP algorithms. Our scheme for coding for the BWC using non-linear nodes is explained in section IV. Our results are presented in section V. The final section is devoted to conclusions and outlooks.

II. NOTATIONS AND BASIC CONCEPTS

In this section we first introduce the basic concepts and then briefly review some results on capacity region for deterministic broadcast channels.

Definition 2.1: A broadcast channel consists of an input alphabet \( \mathcal{X} \), two output alphabets \( \mathcal{Y}_1 \) and \( \mathcal{Y}_2 \) and a probability transition function \( P(y_1, y_2 | x) \). The channel is said to be memoryless if

\[
P(y_1, y_2 | x) = \prod_{i=0}^{n} P(y_{1i}, y_{2i} | x_i).
\]

A \((2^n R_1, 2^n R_2, n)\) code for a BC with independent information consists of an encoder

\[
E : 2^n R_1 \times 2^n R_2 \rightarrow \mathcal{X}^n,
\]

and two decoders

\[
D_1 : Y_1^n \rightarrow 2^n R_1, \quad D_2 : Y_2^n \rightarrow 2^n R_2.
\]

We assume that the transmitted message pair \((W_1, W_2)\) is uniformly distributed over the set \(2^n R_1 \times 2^n R_2\). The probability of error \( P^c \) is defined to be

\[
P^c = P(W_1 \neq \hat{W}_1 \ or \ W_2 \neq \hat{W}_2).
\]

Definition 2.2 (Capacity Region): A rate pair \( (R_1, R_2) \) is called achievable for the BC if there is a sequence \( \{(2^n R_1, 2^n R_2, n)\}_n \) codes with \( P^c \to 0 \) as \( n \to \infty \). The capacity region of the broadcast channel is the closure of the set of achievable rates.

A broadcast channel is deterministic if the channel transition probabilities are deterministic, i.e., \( P(y_1, y_2 | x) \) is a \( 0 \times 1 \) function. The largest achievable rate region for a general BC using the binning strategy is known as the Marton's region [12]. This region is proved to be the capacity region for a discrete deterministic channels [13].

A well-known example of a deterministic BC is the BWC (see Fig. 2). The BWC has one input with three symbols and two outputs each one with two symbols. Given two messages \( W_1 \) and \( W_2 \), the goal is to find the codewords \( y_1 \in 2^n R_1 \) and \( y_2 \in 2^n R_2 \) such that \( f(y_{1i}, y_{2i}) \neq (1, 1) \) for \( i = 1, 2, ..., n \). The other three combinations are allowed and they can be reached by selecting one of the three input symbols of the channel. Even though this channel is not realistic, it is a non-trivial BC which illustrates the conflict between transmitting information to first receiver and transmitting to second receiver [10].

Since the channel is deterministic we have \( H(X) = H(Y_1, Y_2) \). In the rest we assume a uniform probability distribution over \( X \). With this input distribution the capacity region for BWC becomes

\[
\begin{align*}
R_1 & \leq H(Y_1) \\
R_2 & \leq H(Y_2) \\
R_1 + R_2 & \leq \log |\mathcal{Y}_1| + \log |\mathcal{Y}_2|
\end{align*}
\]

This capacity region is shown in Fig. 3.

III. BP AND RBP ALGORITHMS

Let \( g : \mathcal{S} \subseteq \mathbb{R}^n \rightarrow \mathbb{R} \) be a real valued function over the domain \( \mathcal{S} \) and

\[
g(x_1, x_2, ..., x_n) \sim \prod_{j \in \mathcal{M}} f_j(x_j) \quad (1)
\]

where \( X_j \) is a subset of the set of variables.

Definition 3.1: A factor graph of a function \( g \) factorized as in (1) is a bipartite graph with \( n \) vertex in one part (variable nodes) and \( \mathcal{M} \) vertex in the second part (factor nodes). An edge connects variable node \( x_i \) to factor node \( f_j \) if and only if \( x_i \) is an argument of the local function \( f_j \), i.e., \( x_i \in X_j \).

We show the \( i \)-th marginal function associated with \( g(x_1, x_2, ..., x_n) \) by

\[
g_i(x_i) \sim \sum_{x \sim \{x_i\}} g(x_1, x_2, ..., x_n)
\]

where the symbol \( \sim \{x_i\} \) indicates the set of all variable configurations with the \( i \)-th variable fixed to \( x_i \).

Calculating the marginal functions in general is a hard task. BP is an efficient and exact algorithm to calculate all marginal functions \( g_i(x_i) \) when the factor graph of \( g \) is cycle-free. It is possible to use BP also in the presence of loops. The resulting algorithm will be iterative and calculates the
beefficientwhenthe
binatorial
Unfortunately
themselves
reviewthe
These
nodeswhichtheir
marginalsapproximately. In the rest of this section, first we
review the BP update rules and then present a generalization
of BP called Reinforced BP algorithm (RBP) [4].
Let \( \mu_{x \rightarrow f}(x) \) denotes the message sent form variable node
\( x \) to factor node \( f \) at the \( 6 \)th iteration. Similarly, \( \mu_{f \rightarrow x}(x) \)
denotes the message sent from factor node \( f \) to variable node
\( x \) at the iteration \( \ell \). Also, let
\[
\mathcal{N}(x_i) \triangleq \{ j | x_i \in X_j \},
\]
\[
\mathcal{M}(f_j) \triangleq \{ i | x_i \in X_j \},
\]
then the BP algorithm messages can be expressed as follows:

Local Function to Variable:
\[
\mu_{f \rightarrow x}(x_i) \propto \sum_{\sim \{x_i\}} \left( f_j(X_j) \prod_{i \in \mathcal{M}(f_j) \backslash \{i\}} \mu_{x \rightarrow f}(x_i) \right) \tag{2}
\]

Variable to Local Function:
\[
\mu_{x \rightarrow f}(x_i) \propto \prod_{i \in \mathcal{N}(x_i) \backslash \{j\}} \mu_{f \rightarrow x}(x_i) \tag{3}
\]

For \( \ell = 1 \), we initialize the messages \( \mu_{x \rightarrow f}(x) \) randomly.
These updating rules tell us how to produce locally outgoing
messages from incoming messages. We define the marginal function of variable \( x_i \) at iteration \( \ell + 1 \) as
\[
g_{x_i}^{\ell+1}(x_i) \propto \prod_{i \in \mathcal{N}(x_i)} \mu_{x \rightarrow f}(x_i) \tag{4}
\]
The algorithm converges after \( \ell \) iterations if and only if for
all variables \( x_i \) and all function nodes \( f_j \)
\[
\mu_{f \rightarrow x}^{\ell+1}(x_i) = \mu_{f \rightarrow x}(x_i)
\]
In practice we need to predefine maximum number of iterations \( \ell_{\text{max}} \) and a precision parameter \( \epsilon \) as the input to the algorithm.

BP has been generalized/modifed in many ways [2], [4],
[5], [15], [19]. BP and its generalizations have proven to
be efficient when the variables are biased toward a solution.
Unfortunately when this condition is not fulfilled marginal
themselves are not sufficient to find a solution to the
combinatorial problem and one has to resort to some decimation
techniques ( [2], [18]), resulting in high computational
complexity.

We will show here the RBP equations [3] that turn BP
into an efficient solver. The idea is to introduce a new set
of reinforcement messages which drive the equations toward
a single solution. First we modify the original factor graph by
adding to each variable node a new function node. In Fig. 4
these new function nodes are depicted by black squares. These
function nodes are dynamic and at the \( \ell \)th iteration take the
value \( (g_{\ell-1}(x_i))^{\gamma(\ell-1)} \), i.e., a power of the marginal
of the variable \( x_i \) at the preceding iteration. \( \gamma(\ell) \) is a non decreasing
function in \([0, 1]\) with \( \gamma(0) = 0 \). While the updating rule (2)
at function nodes does not change for RBP, the variable to
function messages should be modified as below.

Variable to Local Function for RBP:
\[
\mu_{x \rightarrow f}^{\ell+1}(x_i) \propto (g_{\ell}^{(x_i)})^{\gamma(\ell)} \prod_{i \in \mathcal{N}(x_i) \backslash \{j\}} \mu_{f \rightarrow x}^{\ell}(x_i). \tag{5}
\]

In this paper we deal only with binary constraint satisfaction
problems, where \( x \in \{0, 1\}^n \) and the local functions \( f_j(X_j) \)
are 0-1 indicator functions. A vector \( (x_1, x_2, \ldots, x_n) \) satisfies
\( f_j(X_j) \) if \( f_j(X_j) = 1 \). \((x_1, x_2, \ldots, x_n) \) is called a solution of
the constraint satisfaction problem if all local functions are
satisfied, i.e., \( \prod_{j \in \mathcal{M}} f_j(X_j) = 1 \). It is easy to show that if
RBP converges, it converges to a solution of our problem (all
messages completely polarized to delta functions). This simple
modification provides us with a solver with complexity \( \mathcal{O}(n) \)
(assuming roughly constant convergence time). Note that
the number of iteration of RBP depends also on the choice of \( \gamma(\ell) \)
in (5). As our experiments show, choosing an optimal \( \gamma \) can
dramatically decrease the number of iterations of RBP at least
for the binning problem. For the rest of this paper we will set
\[
\gamma(\ell) = 1 - \gamma_0 \gamma_1^{\ell}, \tag{6}
\]
where \( \gamma_0, \gamma_1 \) are in \([0, 1] \).

IV. CODING FOR THE BLACKWELL CHANNEL USING
NON-LINEAR NODES

One of the main coding strategies for deterministic broadcast channel is binning. The idea is to generate \( 2^{nH(y_1)} \) codewords \( y_1 \) and \( 2^{nH(y_2)} \) codewords \( y_2 \) and randomly assign them into \( 2^{nR_1} \) and \( 2^{nR_2} \) bins. To transmit a particular pair of
bin indices \((i,j)\), the transmitter looks for a pair of codeword
\((y_1, y_2) \in (i,j)\) such that they are jointly typical.

For the BWC, the joint typicality of \( y_1 \) and \( y_2 \) is equivalent
to being consistent with the channel constraints. Therefore,
we are looking for efficient ways to finding a pair \((y_1, y_2)\)
such that \((y_{1i}, y_{2i}) = (1,1)\) does not occur for \( i = 1, 2, \ldots, n \).

Wei Yu and Marko Aleksić in [18] have suggested a
random binning method for BWC based on low-density parity
check like codes. In this section we first review their results
and then modify their scheme using non-linear nodes and
RBP algorithm. As we will see in the next section, these
modifications imply a better encoding complexity and a lower
FER for large function node connectivity.
Fig. 5. Factor graph for LDPC like encoding for Blackwell channel.

Fig. 6. Entropy as a function of rate ($R_1 = R_2$) for different function node connectivity $c$. At any given rate the entropy of codes with non-linear factor nodes increases with $c$ and approaches the entropy of linear codes.

<table>
<thead>
<tr>
<th>Rate</th>
<th>0.5</th>
<th>0.6</th>
<th>0.7</th>
<th>0.72</th>
<th>0.73</th>
<th>0.74</th>
<th>0.75</th>
</tr>
</thead>
<tbody>
<tr>
<td>FER</td>
<td>0</td>
<td>0</td>
<td>0.03</td>
<td>0.1</td>
<td>0.35</td>
<td>0.825</td>
<td>0.975</td>
</tr>
<tr>
<td>BER</td>
<td>0</td>
<td>0</td>
<td>0.00011</td>
<td>0.0013</td>
<td>0.00425</td>
<td>0.0119</td>
<td>0.0347</td>
</tr>
<tr>
<td>$\gamma_1$</td>
<td>0.99</td>
<td>0.995</td>
<td>0.999</td>
<td>0.9995</td>
<td>0.99999</td>
<td>0.999995</td>
<td></td>
</tr>
</tbody>
</table>

TABLE I
BER AND FER OF NON-LINEAR LDPC LIKE ENCODERS AT A GIVEN RATE ($R_1 = R_2$) AND CONNECTIVITY $c = 6$.

V. RESULTS
Table I shows the FER and BER of our constructed non-linear codes for the BWC with $n = 1000$ and constant connectivity $c = 6$ at different rates. The last line reports the values we chose for $\gamma_1$.

We estimated the algorithmic complexity of the presented coding scheme in a series of experiments described below. In particular, we will show how the convergence time changes as a function of $n$ and $\gamma_1$. The RBP algorithm was run with an estimated optimal value of $\gamma_1$, and we have chosen a cutoff time of $\frac{1}{(1-\gamma_1)}$ to measure the bit and frame error rates.

Fig. 7 shows the average number of iterations needed (in the case of success) for a rate $R_1 = R_2 = 0.70$ as a function of
thenumberofiterationsfor
bound
Table
less
Fig.
with
first200iterations
0.001
Lw
L
1e-05
L
1
valuesof
0.9!
Error
results
and
bars
are
Bio.
Error
as
BP,
and
performance.
approximatelythe
one can
avoid
considerably
non-linear
problems
even when they have a large solution
space. The algorithm have the same complexity of BP and thus
considerably smaller than the decimation approach applied to
BP/SP proposed in [18].

Using RBP we have constructed a general and rather
efficient encoding scheme for the BWC. Our codes
provide good encoding performances for rates up to 0.72.
This result can be possibly improved by optimizing the
function \( \gamma(\ell) \) and the
degree distributions of the code.

Our scheme compares well with existing ones: for linear
codes with \( R = R_1 = R_2 = 0.75 \) and
decimation, as it was
reported also in [18], one can get the bit error rate of
5.10^{-3}.
Still, simulations show that the FER in this case is 0.9
and it does not improve for smaller rates like \( R = 0.72 \) with
the same connectivity. On the other hand it works only for low
function node connectivity. Our scheme is much more flexible
and provides a comparable FER and BER at \( R = 0.75 \) with
lower computational complexity. For smaller rates our codes
outperform the existing linear encoding schemes.

\section{VI. Conclusion and Outlooks}

We have introduced a novel variation of the BP algorithm,
called reinforced BP, that turns it into an efficient solver for

\[ n \text{ and } \gamma_1 \text{ and for 160 encoding operations. These simulations indicate that the number of iterations increase as } O(\log n). \]
Although the number of iterations increase (exponentially)
with \( \gamma_1 \), both the BER and FER decrease (exponentially) as
it can be seen in Fig. 8. Note that for rates closer to the capacity
bound (\( R_1 = R_2 \approx 0.785 \)) a value of \( \gamma_1 \) closer to 1 (and larger
number of iterations) is needed.

Although the results depicted in Fig. 7 indicate a logarithmic
increase in the number of iterations as a function of \( n \), this
result may be due to a not optimized choice of \( \gamma(\ell) \). For
example by choosing \( \gamma_0 = 0.8 \) in (6) it is possible to reduce the
number of iterations for \( n = 4000 \) and \( \gamma_1 = 0.999 \) by
nearly 25%. In other words, one can avoid approximately the
first 200 iterations of RBP without loosing in performance.

\section{References}

[1] P. P. Bergmans, Random Coding Theorem for Broadcast Channels with

Algorithm for Satisfiability, Random Structures and Algorithms, 27, 201-
226, 2005.


P11016, 2005.

BF-Based Decoding Algorithms of LDPC codes, IEEE Communications
Letters, vol. 6, no. 5 May 2002.


1, Jan. 1972.

[8] T. M. Cover and J. A. Thomas, Elements of Information Theory, Wiley,


analytic solution to an efficient algorithm, Phys. Rev. E66, 056126, 2002

[15] T. Muroyan, Theless-Anderson-Palmer Approach for Lossy Compress-

[16] K. Nakamura, Y. KabaShima, R. M. Zangendo and D. Saad, Statistical
Mechanics of Broadcast Channels Using Low Density Parity Check

[17] T. J. Richardson, A. Shokrollahi and R. Urbanke, Design of Capacity
Approaching Irregular Low-Density Parity Check Codes, IEEE Trans.

[18] Wei Xu and M. Aleksee, Coding for the Blackwell Channel: A Survey
Propagation Approach, International Symposium on Information Theory
(ISIT), 2005.

Advances in Neural Information Processing Systems, vol 13, MIT Press,