

Article

# An Information-Spectrum Approach to the Capacity Region of the Interference Channel <sup>†</sup>

Lei Lin <sup>1,\*</sup>, Xiao Ma <sup>2</sup>, Chulong Liang <sup>3</sup>, Xiujie Huang <sup>4</sup> and Baoming Bai <sup>5</sup>

<sup>1</sup> School of Mathematics and Systems Science, Guangdong Polytechnic Normal University, Guangzhou 510665, China

<sup>2</sup> School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China; maxiao@mail.sysu.edu.cn

<sup>3</sup> Department of Electronic Engineering, City University of Hong Kong, Hong Kong 999077, China; chuliang@cityu.edu.hk

<sup>4</sup> Department of Computer Science, Jinan University, Guangzhou 510632, China; huangxj5@mail2.sysu.edu.cn

<sup>5</sup> State Laboratory of ISN, Xidian University, Xi'an 710071, China; bmbai@mail.xidian.edu.cn

\* Correspondence: linlei2@mail2.sysu.edu.cn

<sup>†</sup> This paper is an extended version of our paper published in the 2012 IEEE International Symposium on Information Theory, Cambridge, MA, USA, 1–6 July 2012.

Academic Editor: Raúl Alcaraz Martínez

Received: 8 April 2017; Accepted: 10 June 2017; Published: 13 June 2017

**Abstract:** In this paper, a general formula for the capacity region of a general interference channel with two pairs of users is derived, which reveals that the capacity region is the union of a family of rectangles. In the region, each rectangle is determined by a pair of spectral inf-mutual information rates. The presented formula provides us with useful insights into the interference channels in spite of the difficulty of computing it. Specially, when the inputs are discrete, ergodic Markov processes and the channel is stationary memoryless, the formula can be evaluated by the BCJR (Bahl-Cocke-Jelinek-Raviv) algorithm. Also the formula suggests that considering the structure of the interference processes contributes to obtaining tighter inner bounds than the simplest one (obtained by treating the interference as noise). This is verified numerically by calculating the mutual information rates for Gaussian interference channels with embedded convolutional codes. Moreover, we present a coding scheme to approach the theoretical achievable rate pairs. Numerical results show that the decoding gains can be achieved by considering the structure of the interference.

**Keywords:** capacity region; interference channel; information spectrum; limit superior/inferior in probability; spectral inf-mutual information rate

## 1. Introduction

In wireless communications, since the electromagnetic spectrum is limited, frequency bands are often simultaneously used by several radio links that are not completely isolated [1]. When several pairs of senders and receivers share a common communication medium, the transmission of information from one sender to the corresponding receiver interferes with communications between the other senders and their receivers. This communication model is called interference channel (IC), which was firstly defined by Shannon in [2] and furthered by Ahlswede in [3]. A basic problem for the IC is to determine the rate pairs at which information can be reliably transmitted over the channel, that is, the capacity region. However, the problem of characterizing this region has been open for over 40 years. The capacity regions are known only in some special cases, such as strong, very strong and deterministic interference channels [4–7]. In decades, the researchers provided various inner and outer bounds of the capacity region for the general IC. For instance, two outer bounds on the capacity region

of the Gaussian interference channel (GIFC) were derived in [8]. The first bound unifies and improves the outer bounds of Sato [9] and Carleial [10]. The second bound follows directly from the outer bounds in [11,12], which is deduced by considering a degraded GIFC and is even better than the first one for certain weak GIFCs. In 1981, Han and Kobayashi [5] proposed the best inner bound (the so-called HK region), which has been simplified by Kramer and Chong et al. in their independent works [13,14]. By introducing the idea of approximation, Etkin, Tse and Wang [15] showed that HK region [5] is within one bit of the capacity region for the GIFC.

In [16,17], a new computational model for the two-user GIFC was proposed, in which one pair of users (called *primary users*) are constrained to use a fixed encoder and the other pair of users (called *secondary users*) are allowed to optimize their code. The *accessible capacity* of the secondary users is defined as the maximum rate at which the secondary users can communicate reliably without degrading the performance of the primary users. Usually, the accessible capacity is higher than the maximum rate when treating the interference as noise, which is because the structure of the interference from the primary link has been taken into consideration in the computation, as is consistent with the spirit of [18,19]. However, in the computation of the accessible capacity [17], the primary link is allowed to have a non-neglected error probability. This makes the accessible capacity lower than the capacity region. As a result, the fixed-code constraints on the primary users will be relaxed in this paper. Namely, a pair of transmission rates at which both links can be asymptotically error-free will be calculated.

This paper is concerned with a more general IC, which is characterized by a sequence of transition probabilities (see Figure 1). By adopting the information spectrum approach [20,21], we present a general formula for the capacity region of the two-user general IC. From the formula, it can be seen that the capacity region is the union of a family of rectangles, in which each rectangle is determined by a spectral inf-mutual information rate pair. The information spectrum approach, which is based on the *limit superior/inferior in probability* of a sequence of random variables, has been proved to be powerful in characterizing the limit behavior of a general source/channel. For instance, in [20,22], Han and Verdú proved that the minimum compression rate for a general source equals its *spectral sup-entropy rate* and the maximum transmission rate for a general point-to-point channel equals its *spectral inf-mutual information rate* with an optimized input process. Also, the information spectrum approach can be used to derive the capacity region of a general multiple access channel [23]. For more applications of the information spectrum approach, see [21] and the references therein.

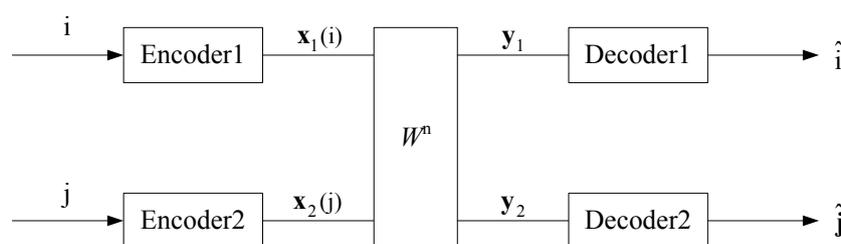


Figure 1. General interference channel  $W$ .

The structure of the paper is as follows. In Section 2, the definition of a general IC and the concept of the spectral inf-mutual information rate are introduced. Section 3.1 introduces the general formula for the capacity region proposed in [24]; while, in Section 3.2, a trellis-based algorithm is presented to compute the rate pairs for a stationary memoryless IC with discrete ergodic Markov sources. Section 3.3 presents the numerical results for a GIFC with binary-phase shift-keying (BPSK) modulation. Section 4 provides the detection and decoding algorithms for channels with structured interference. Section 5 concludes this paper.

In this paper, a random variable is denoted by an upper-case letter, say  $X$ , while its realization and sample space are denoted by  $x$  and  $\mathcal{X}$ , respectively. The sequence of random variables with length

$n$  are denoted by  $X^n$ , while its realization is denoted by  $\mathbf{x} \in \mathcal{X}^n$  or  $x^n \in \mathcal{X}^n$ . We use  $P_X(x)$  to denote the probability mass function (pmf) of  $X$  if it is discrete or the probability density function (pdf) of  $X$  if it is continuous.

## 2. Basic Definitions and Problem Statement

### 2.1. General IC

As shown in Figure 1, a general interference channel  $\mathbf{W}$  can be characterized by input alphabets  $\mathcal{X}_1, \mathcal{X}_2$ , output alphabets  $\mathcal{Y}_1, \mathcal{Y}_2$  and a sequence  $\mathbf{W} = \{W^n(\cdot, \cdot | \cdot, \cdot)\}_{n=1}^\infty$ , in which  $W^n : \mathcal{X}_1^n \times \mathcal{X}_2^n \rightarrow \mathcal{Y}_1^n \times \mathcal{Y}_2^n$  is a probability transition matrix. That is, for all  $n$ ,

$$\begin{aligned} W^n(\mathbf{y}_1, \mathbf{y}_2 | \mathbf{x}_1, \mathbf{x}_2) &\geq 0 \\ \sum_{\mathbf{y}_1 \in \mathcal{Y}_1^n, \mathbf{y}_2 \in \mathcal{Y}_2^n} W^n(\mathbf{y}_1, \mathbf{y}_2 | \mathbf{x}_1, \mathbf{x}_2) &= 1. \end{aligned}$$

The marginal distributions  $W_1^n, W_2^n$  of the  $W^n$  are given by

$$W_1^n(\mathbf{y}_1 | \mathbf{x}_1, \mathbf{x}_2) = \sum_{\mathbf{y}_2 \in \mathcal{Y}_2^n} W^n(\mathbf{y}_1, \mathbf{y}_2 | \mathbf{x}_1, \mathbf{x}_2), \tag{1}$$

$$W_2^n(\mathbf{y}_2 | \mathbf{x}_1, \mathbf{x}_2) = \sum_{\mathbf{y}_1 \in \mathcal{Y}_1^n} W^n(\mathbf{y}_1, \mathbf{y}_2 | \mathbf{x}_1, \mathbf{x}_2). \tag{2}$$

**Definition 1.** An  $(n, M_n^{(1)}, M_n^{(2)}, \varepsilon_n^{(1)}, \varepsilon_n^{(2)})$  code for the interference channel  $\mathbf{W}$  consists of the following essentials:

(a) message sets:

$$\begin{aligned} \mathcal{M}_n^{(1)} &= \{1, 2, \dots, M_n^{(1)}\}, & \text{for Sender 1} \\ \mathcal{M}_n^{(2)} &= \{1, 2, \dots, M_n^{(2)}\}, & \text{for Sender 2} \end{aligned}$$

(b) sets of codewords:

$$\begin{aligned} \{\mathbf{x}_1(1), \mathbf{x}_1(2), \dots, \mathbf{x}_1(M_n^{(1)})\} &\subseteq \mathcal{X}_1^n, & \text{for Encoder 1} \\ \{\mathbf{x}_2(1), \mathbf{x}_2(2), \dots, \mathbf{x}_2(M_n^{(2)})\} &\subseteq \mathcal{X}_2^n, & \text{for Encoder 2} \end{aligned}$$

For Sender 1 to transmit message  $i$ , Encoder 1 outputs the codeword  $\mathbf{x}_1(i)$ . Similarly, for Sender 2 to transmit message  $j$ , Encoder 2 outputs the codeword  $\mathbf{x}_2(j)$ .

(c) collections of decoding sets:

$$\begin{aligned} \mathcal{B}_1 &= \{\mathcal{B}_{1i} \subseteq \mathcal{Y}_1^n\}_{i=1, \dots, M_n^{(1)}}, & \text{for Decoder 1} \\ \mathcal{B}_2 &= \{\mathcal{B}_{2j} \subseteq \mathcal{Y}_2^n\}_{j=1, \dots, M_n^{(2)}}, & \text{for Decoder 2} \end{aligned}$$

where  $\mathcal{Y}_1^n = \bigcup_{i=1}^{M_n^{(1)}} \mathcal{B}_{1i}$ ,  $\mathcal{B}_{1i} \cap \mathcal{B}_{1i'} = \emptyset$  for  $i \neq i'$  and  $\mathcal{Y}_2^n = \bigcup_{j=1}^{M_n^{(2)}} \mathcal{B}_{2j}$ ,  $\mathcal{B}_{2j} \cap \mathcal{B}_{2j'} = \emptyset$  for  $j \neq j'$ . That is,  $\mathcal{B}_1$  and  $\mathcal{B}_2$  are the disjoint partitions of  $\mathcal{Y}_1^n$  and  $\mathcal{Y}_2^n$  determined in advance, respectively. After receiving  $\mathbf{y}_1$ , Decoder 1 outputs  $\hat{i}$  whenever  $\mathbf{y}_1 \in \mathcal{B}_{1\hat{i}}$ . Similarly, after receiving  $\mathbf{y}_2$ , Decoder 2 outputs  $\hat{j}$  whenever  $\mathbf{y}_2 \in \mathcal{B}_{2\hat{j}}$ .

(d) probabilities of decoding errors:

$$\begin{aligned} \varepsilon_n^{(1)} &= \frac{1}{M_n^{(1)}M_n^{(2)}} \sum_{i=1}^{M_n^{(1)}} \sum_{j=1}^{M_n^{(2)}} W_1^n(\mathcal{B}_{1i}^c | \mathbf{x}_1(i), \mathbf{x}_2(j)), \\ \varepsilon_n^{(2)} &= \frac{1}{M_n^{(1)}M_n^{(2)}} \sum_{i=1}^{M_n^{(1)}} \sum_{j=1}^{M_n^{(2)}} W_2^n(\mathcal{B}_{2j}^c | \mathbf{x}_1(i), \mathbf{x}_2(j)), \end{aligned}$$

where “c” denotes the complement of a set. Here we have assumed that each message of  $i \in \mathcal{M}_n^{(1)}$  and  $j \in \mathcal{M}_n^{(2)}$  is produced independently with uniform distribution.

**Remark 1.** It is optimal to minimize the probability of errors so that the decoding sets  $\mathcal{B}_{1i}$  and  $\mathcal{B}_{2j}$  are defined according to the the maximum likelihood decoding [25]. Namely,

$$\hat{i} = \arg \max_i \Pr\{\mathbf{y}_1 | \mathbf{x}_1(i)\}$$

and

$$\hat{j} = \arg \max_j \Pr\{\mathbf{y}_2 | \mathbf{x}_2(j)\}$$

are selected as the estimates of the transmitted messages by the two receivers, respectively.

**Definition 2.** A rate pair  $(R_1, R_2)$  is achievable if there exists a sequence of  $(n, M_n^{(1)}, M_n^{(2)}, \varepsilon_n^{(1)}, \varepsilon_n^{(2)})$  codes such that

$$\begin{aligned} \lim_{n \rightarrow \infty} \varepsilon_n^{(1)} = 0 \quad \text{and} \quad \lim_{n \rightarrow \infty} \varepsilon_n^{(2)} = 0, \\ \liminf_{n \rightarrow \infty} \frac{\log M_n^{(1)}}{n} \geq R_1 \quad \text{and} \quad \liminf_{n \rightarrow \infty} \frac{\log M_n^{(2)}}{n} \geq R_2. \end{aligned}$$

**Definition 3.** The set of all achievable rates is called the capacity region of the interference channel  $\mathbf{W}$ , which is denoted by  $\mathcal{C}(\mathbf{W})$ .

### 2.2. Preliminaries of Information-Spectrum Approach

We introduce the notions in [21] as follows.

**Definition 4** (liminf in probability). For a sequence of random variables  $\{Z^n\}_{n=1}^\infty$ ,

$$p\text{-}\liminf_{n \rightarrow \infty} Z^n \triangleq \sup\{\beta | \lim_{n \rightarrow \infty} \Pr\{Z^n < \beta\} = 0\}.$$

**Definition 5.** If two random variables sequences  $\mathbf{X}_1 = \{X_1^n\}_{n=1}^\infty$  and  $\mathbf{X}_2 = \{X_2^n\}_{n=1}^\infty$  satisfy that

$$P_{X_1^n X_2^n}(\mathbf{x}_1, \mathbf{x}_2) = P_{X_1^n}(\mathbf{x}_1)P_{X_2^n}(\mathbf{x}_2) \tag{3}$$

for all  $\mathbf{x}_1 \in \mathcal{X}_1^n, \mathbf{x}_2 \in \mathcal{X}_2^n$  and  $n$ , they are called independent and denoted by  $\mathbf{X}_1 \perp \mathbf{X}_2$ .

Similar to [20], we give

**Definition 6.** Let  $S_I \triangleq \{(\mathbf{X}_1, \mathbf{X}_2) | \mathbf{X}_1 \perp \mathbf{X}_2\}$ . Given an  $(\mathbf{X}_1, \mathbf{X}_2) \in S_I$ , for the interference channel  $\mathbf{W}$ , we define the spectral inf-mutual information rate by

$$\underline{I}(\mathbf{X}_1; \mathbf{Y}_1) \equiv p\text{-}\liminf_{n \rightarrow \infty} \frac{1}{n} \log \frac{P_{Y_1^n | X_1^n}(Y_1^n | X_1^n)}{P_{Y_1^n}(Y_1^n)}, \tag{4}$$

$$\underline{I}(\mathbf{X}_2; \mathbf{Y}_2) \equiv p\text{-}\liminf_{n \rightarrow \infty} \frac{1}{n} \log \frac{P_{Y_2^n | X_2^n}(Y_2^n | X_2^n)}{P_{Y_2^n}(Y_2^n)}, \tag{5}$$

where

$$P_{Y_1^n | X_1^n}(y_1 | x_1) = \sum_{x_2, y_2} P_{X_2^n}(x_2) W^n(y_1, y_2 | x_1, x_2), \tag{6}$$

$$P_{Y_2^n | X_2^n}(y_2 | x_2) = \sum_{x_1, y_1} P_{X_1^n}(x_1) W^n(y_1, y_2 | x_1, x_2). \tag{7}$$

### 3. The Capacity Region of General IC

In this section, we firstly present without proof the formula for the capacity region  $\mathcal{C}(\mathbf{W})$  of the general IC derived in [24]. Also the algorithm to compute achievable rate pairs is presented.

#### 3.1. The Main Theorem

**Theorem 1.** *The capacity region  $\mathcal{C}(\mathbf{W})$  of the interference channel  $\mathbf{W}$  is given by*

$$\mathcal{C}(\mathbf{W}) = \bigcup_{(\mathbf{X}_1, \mathbf{X}_2) \in \mathcal{S}_I} \mathcal{R}_{\mathbf{W}}(\mathbf{X}_1, \mathbf{X}_2), \tag{8}$$

where  $\mathcal{R}_{\mathbf{W}}(\mathbf{X}_1, \mathbf{X}_2)$  is defined as the collection of all  $(R_1, R_2)$  satisfying that

$$0 \leq R_1 \leq \underline{I}(\mathbf{X}_1; \mathbf{Y}_1), \tag{9}$$

$$0 \leq R_2 \leq \underline{I}(\mathbf{X}_2; \mathbf{Y}_2). \tag{10}$$

#### 3.2. The Algorithm to Compute Achievable Rate Pairs

Theorem 1 provides a general formula for the capacity region of a general IC. However, it is usually difficult to compute the spectral inf-mutual information rates given in (9) and (10). In order to get insights into the interference channels, we make the following assumptions:

- (1) the channel is stationary and memoryless, that is, the transition probability of the channel can be written as

$$W^n(y_1, y_2 | x_1, x_2) = \prod_{i=1}^n W(y_{1,i}, y_{2,i} | x_{1,i}, x_{2,i});$$

- (2) sources are restricted to be stationary and ergodic discrete Markov processes.

With the above assumptions, the spectral inf-mutual information rates are reduced as

$$\underline{I}(\mathbf{X}_1; \mathbf{Y}_1) = \lim_{n \rightarrow \infty} \frac{1}{n} I(X_1^n; Y_1^n), \tag{11}$$

$$\underline{I}(\mathbf{X}_2; \mathbf{Y}_2) = \lim_{n \rightarrow \infty} \frac{1}{n} I(X_2^n; Y_2^n), \tag{12}$$

which can be evaluated by the Monte Carlo method [26–28] using BCJR algorithm [29] over a trellis. Actually, any stationary and ergodic discrete Markov source can be depicted by a time-invariant trellis. That is, a trellis section can uniquely specify the source. A trellis section is composed of left (or starting) states and right (or ending) states, which are connected by branches in between. For example, Source  $x_1$  can be specified by a trellis  $\mathcal{T}_1$  as follows.

- Both the left and right states are selected from the set  $\mathcal{S}_1 = \{0, 1, \dots, |\mathcal{S}_1| - 1\}$ ;

- Each branch is represented by a three-tuple  $b = (s_1^-(b), x_1(b), s_1^+(b))$ , where  $s_1^-(b)$  is the left state,  $s_1^+(b)$  is the right state, and the symbol  $x_1(b) \in \mathcal{X}_1$  is the associated label. We also assume that a branch  $b$  is uniquely determined by  $s_1^-(b)$  and  $x_1(b)$ ;
- At time  $t = 0$ , the source starts from state  $s_{1,0} \in \mathcal{S}_1$ . If at time  $t - 1$  ( $t > 0$ ), the source is in the state  $s_{1,t-1} \in \mathcal{S}_1$ , then at time  $t$  ( $t > 0$ ), the source generates a symbol  $x_{1,t} \in \mathcal{X}_1$  according to the conditional probability  $P(x_{1,t}|s_{1,t-1})$  and goes into a state  $s_{1,t} \in \mathcal{S}_1$  such that  $(s_{1,t-1}, x_{1,t}, s_{1,t})$  is a branch. Obviously, when the source runs from time  $t = 0$  to  $t = n$ , a sequence  $x_{1,1}, x_{1,2}, \dots, x_{1,n}$  is generated. The Markov property says that

$$P(x_{1,t}|x_{1,1}, \dots, x_{1,t-1}, s_{1,0}) = P(x_{1,t}|s_{1,t-1}).$$

So the probability of a given sequence  $x_{1,1}, x_{1,2}, \dots, x_{1,n}$  with the initial state  $s_{1,0}$  can be factored as

$$P(x_{1,1}, x_{1,2}, \dots, x_{1,n}|s_{1,0}) = \prod_{t=1}^n P(x_{1,t}|s_{1,t-1}).$$

Similarly, we can represent  $x_2$  by a trellis  $\mathcal{T}_2$  with the state set  $\mathcal{S}_2 = \{0, 1, \dots, |\mathcal{S}_2| - 1\}$ . Each branch is denoted by  $b = (s_2^-(b), x_2(b), s_2^+(b))$ , where  $s_2^-(b)$  is the left state,  $s_2^+(b)$  is the right state and the symbol  $x_2(b) \in \mathcal{X}_2$  is the associated label. Assume that source  $x_2$  starts from the state  $s_{2,0} \in \mathcal{S}_2$ . If at time  $t - 1$  ( $t > 0$ ), the source is in the state  $s_{2,t-1} \in \mathcal{S}_2$ , then at time  $t$  ( $t > 0$ ), the source generates a symbol  $x_{2,t} \in \mathcal{X}_2$  according to the conditional probability  $P(x_{2,t}|s_{2,t-1})$  and goes into a state  $s_{2,t} \in \mathcal{S}_2$  such that  $(s_{2,t-1}, x_{2,t}, s_{2,t})$  is a branch. The probability of a given sequence  $x_{2,1}, x_{2,2}, \dots, x_{2,n}$  can be factored as

$$P(x_{2,1}, x_{2,2}, \dots, x_{2,n}|s_{2,0}) = \prod_{t=1}^n P(x_{2,t}|s_{2,t-1}).$$

For simplicity, the initial states have been fixed as  $s_{1,0} = 0$  and  $s_{2,0} = 0$ , which can be removed from the equations.

Next we focus on the evaluation of  $\lim_{n \rightarrow \infty} \frac{1}{n} I(X_1^n; Y_1^n)$ , while  $\lim_{n \rightarrow \infty} \frac{1}{n} I(X_2^n; Y_2^n)$  can be estimated similarly. Specifically, we can express the limit as

$$\lim_{n \rightarrow \infty} \frac{1}{n} I(X_1^n; Y_1^n) = \lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n) - \lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n | X_1^n), \tag{13}$$

where  $\lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n)$  and  $\lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n | X_1^n)$  can be estimated by similar methods (For continuous  $\mathbf{y}_1$ , the computations of  $\lim_{n \rightarrow \infty} \frac{1}{n} h(Y_1^n)$  and  $\lim_{n \rightarrow \infty} \frac{1}{n} h(Y_1^n | X_1^n)$  can be implemented by substituting pdf for pmf). As an example, we show how to compute  $\lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n)$ . According to the Shannon-McMillan-Breiman theorem [30], it can be seen that, with probability 1,

$$\lim_{n \rightarrow \infty} -\frac{1}{n} \log P(y_1^n) = \lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n),$$

where  $y_1^n$  stands for  $(y_{1,1}, y_{1,2}, \dots, y_{1,n})$ . Then evaluating  $\lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n)$  is converted to computing

$$\lim_{n \rightarrow \infty} -\frac{1}{n} \log P(y_1^n) \approx -\frac{1}{n} \log \left( \prod_{t=1}^n P(y_{1,t} | y_1^{t-1}) \right) = -\frac{1}{n} \sum_{t=1}^n \log P(y_{1,t} | y_1^{t-1})$$

for a sufficiently long typical sequence  $y_1^n$ . Here, the key is to compute the conditional probabilities  $P(y_{1,t} | y_1^{t-1})$  for all  $t$ . Since both  $\mathbf{y}_1$  and  $\mathbf{y}_2$  are hidden Markov sequences, this can be done by performing the BCJR algorithm over the following product trellis.

- The product trellis has the state set  $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2$ , where “ $\times$ ” denotes Cartesian product.

- Each branch is represented by a four-tuple  $b = (s^-(b), x_1(b), x_2(b), s^+(b))$ , where  $s^-(b) = (s_1^-(b), s_2^-(b))$  is the left state,  $s^+(b) = (s_1^+(b), s_2^+(b))$  is the right state. Then  $x_1(b) \in \mathcal{X}_1$  and  $x_2(b) \in \mathcal{X}_2$  are the associated labels in branch  $b$  such that  $(s_1^-(b), x_1(b), s_1^+(b))$  and  $(s_2^-(b), x_2(b), s_2^+(b))$  are branches in  $\mathcal{T}_1$  and  $\mathcal{T}_2$ , respectively.
- At time  $t = 0$ , the sources start from state  $s_0 = (s_{1,0}, s_{2,0}) \in \mathcal{S}$ . If at time  $t - 1$  ( $t > 0$ ), the sources are in the state  $s_{t-1} = (s_{1,t-1}, s_{2,t-1}) \in \mathcal{S}$ , then at time  $t$  ( $t > 0$ ), the sources generate symbols  $(x_{1,t} \in \mathcal{X}_1, x_{2,t} \in \mathcal{X}_2)$  according to the conditional probability  $P(x_{1,t}|s_{1,t-1})P(x_{2,t}|s_{2,t-1})$  and go into a state  $s_t = (s_{1,t}, s_{2,t}) \in \mathcal{S}_2$  such that  $(s_{t-1}, x_{1,t}, x_{2,t}, s_t)$  is a branch.

The following description shows how to compute  $P(y_{1,t}|y_1^{t-1})$  by performing the BCJR algorithm. Given the received sequence  $\mathbf{y}_1$ , we define

- *Branch metrics:* To each branch  $b_t = \{s_{t-1}, x_{1,t}, x_{2,t}, s_t\}$ , we assign a metric

$$\rho(b_t) \triangleq P(b_t|s_{t-1})P(y_{1,t}|x_{1,t}x_{2,t}) \tag{14}$$

$$= P(x_{1,t}|s_{1,t-1})P(x_{2,t}|s_{2,t-1})P(y_{1,t}|x_{1,t}x_{2,t}), \tag{15}$$

In the computation of  $\lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n | X_1^n)$ , the metric is replaced by  $P(b_t|s_{t-1}, x_{1,t})P(y_{1,t}|x_{1,t}x_{2,t})$ .

- *State transition probabilities:* The transition probability from  $s_{t-1}$  to  $s_t$  is defined as

$$\gamma_t(s_{t-1}, s_t) \triangleq P(s_t, y_{1,t} | s_{t-1}) \tag{16}$$

$$= \sum_{b_t: s^-(b_t)=s_{t-1}, s^+(b_t)=s_t} \rho(b_t). \tag{17}$$

- *Forward recursion variables:* We define the *a posteriori* probabilities

$$\alpha_t(s_t) \triangleq P(s_t | y_1^t), \quad t = 0, 1, \dots, n. \tag{18}$$

Then

$$P(y_{1,t} | y_1^{t-1}) = \sum_{s_{t-1}, s_t} \alpha(s_{t-1}) \gamma_t(s_{t-1}, s_t), \tag{19}$$

where the values of  $\alpha_t(s_t)$  can be computed recursively by

$$\alpha_t(s_t) = \frac{\sum_{s_{t-1}} \alpha_{t-1}(s_{t-1}) \gamma_t(s_{t-1}, s_t)}{\sum_{s_{t-1}, s_t} \alpha_{t-1}(s_{t-1}) \gamma_t(s_{t-1}, s_t)}. \tag{20}$$

In summary, the algorithm to estimate the entropy rate  $\lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n)$  is described as follows.

**Algorithm 1.**

1. **Initializations:** Choose a sufficiently large number  $n$ . Set the initial state of the trellis to be  $s_0 = 0$ . The forward recursion variables are initialized as  $\alpha_0(s) = 1$  if  $s = 0$  and otherwise  $\alpha_0(s) = 0$ .
2. **Simulations for Sender 1:** Generate a Markov sequence  $\mathbf{x}_1 = (x_{1,1}, x_{1,2}, \dots, x_{1,n})$  according to the trellis  $\mathcal{T}_1$  of source  $\mathbf{x}_1$ .
3. **Simulations for Sender 2:** Generate a Markov sequence  $\mathbf{x}_2 = (x_{2,1}, x_{2,2}, \dots, x_{2,n})$  according to the trellis  $\mathcal{T}_2$  of source  $\mathbf{x}_2$ .
4. **Simulations for Receiver 1:** Generate the received sequence  $\mathbf{y}_1$  according to the transition probability  $W^n(\mathbf{y}_1, \mathbf{y}_2 | \mathbf{x}_1, \mathbf{x}_2)$ .
5. **Computations:**
  - (a) For  $t = 1, 2, \dots, n$ , compute the values of  $P(y_{1,t} | y_1^{t-1})$  and  $\alpha_t(s_t)$  recursively according to Equations (19) and (20).

(b) Evaluate the entropy rate

$$\lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n) = -\frac{1}{n} \sum_{t=1}^n \log P(y_{1,t} | y_1^{t-1}).$$

Similarly, we can evaluate the entropy rate  $\lim_{n \rightarrow \infty} \frac{1}{n} H(Y_1^n | X_1^n)$ . Therefore, we obtain the achievable rate  $I(\mathbf{X}_1; \mathbf{Y}_1) = \lim_{n \rightarrow \infty} \frac{1}{n} I(X_1^n; Y_1^n)$ .

### 3.3. Numerical Results

We consider the GIFC as shown in Figure 2, where the channel inputs  $\mathbf{x}_1(i)$  and  $\mathbf{x}_2(j)$  are BPSK sequences with power constraints  $P_1$  and  $P_2$ , respectively; the additive noises  $\mathbf{n}_1$  and  $\mathbf{n}_2$  are sequences of independent and identically distributed (i.i.d.) standard Gaussian random variables, which are assumed to be independent of the the channel inputs  $\mathbf{x}_1(i)$  and  $\mathbf{x}_2(j)$ ; constant  $\sqrt{a}$  represents the gain of the interference link; the channel outputs  $\mathbf{y}_1$  and  $\mathbf{y}_2$  are

$$\mathbf{y}_1 = \mathbf{x}_1(i) + \sqrt{a}\mathbf{x}_2(j) + \mathbf{n}_1, \tag{21}$$

$$\mathbf{y}_2 = \mathbf{x}_2(j) + \sqrt{a}\mathbf{x}_1(i) + \mathbf{n}_2. \tag{22}$$

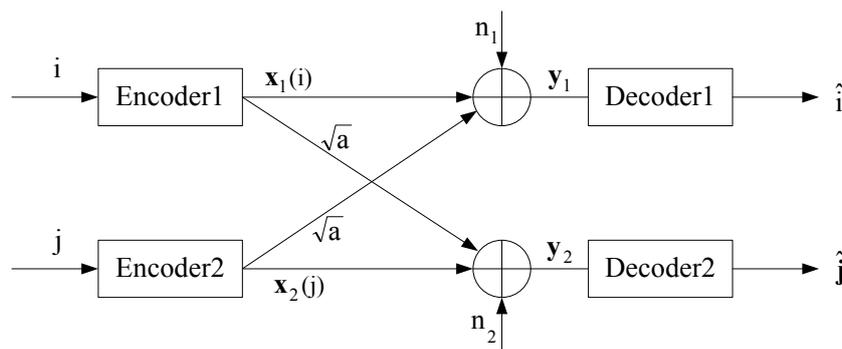


Figure 2. Symmetric Gaussian interference channel.

We assume that  $\mathbf{x}_1$  and  $\mathbf{x}_2$  are the outputs from two (possibly different) generalized trellis encoders driven by independent and uniformly distributed (i.u.d.) input sequences, as proposed in [16]. As examples, we consider two input processes. One is referred to as “UnBPSK”, standing for an i.u.d. BPSK sequence; the other is referred to as “CcBPSK”, standing for an output sequence from the convolutional encoder with the generator matrix  $G(D) = [1 + D + D^2 \ 1 + D^2]$  driven by an i.u.d. input sequence.

When Sender 1 uses CcBPSK and Sender 2 uses UnBPSK, the trellis representation of the scheme can be seen in Figure 3. The numerical results are presented in Figure 4. There are three rectangles, OECH, ODBG and OFAI, each of which is determined by a pair of spectral inf-mutual information rates. Specifically, the rectangle OECH corresponds to the case when both senders use UnBPSK as inputs; the rectangle ODBG corresponds to the case when Sender 1 uses UnBPSK as input and Sender 2 uses CcBPSK as input; and the rectangle OFAI corresponds to the case when Sender 1 uses CcBPSK as input and Sender 2 uses UnBPSK as input. The point “A” can be achieved by a coding scheme, in which Sender 1 uses a binary linear (coset) code concatenated with the convolutional code and Sender 2 uses a binary linear code, while the point “B” can be achieved similarly. By time-sharing scheme, the points on the line “AB” can be achieved. The point “C” states the limit when the two senders use binary linear codes but take the interference as an i.u.d. additive (BPSK) noise. It is obvious that the area of the pentagonal region ODBAI is greater than that of the rectangle OECH, which hints that the bandwidth-efficiency can be potentially improved by knowing the structure of the interference.

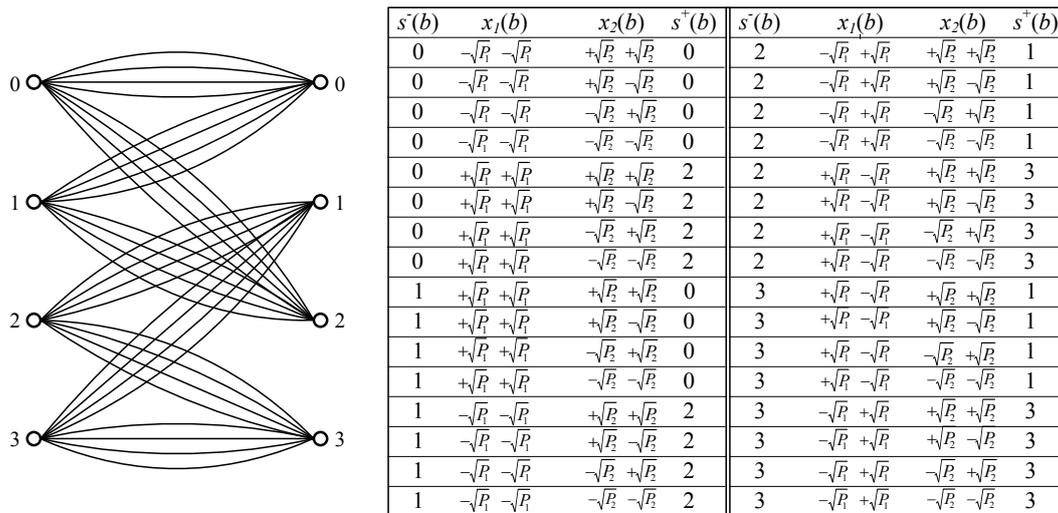


Figure 3. The trellis section of (CcBPSK, UnBPSK) with 32 branches. For each branch  $b$ ,  $s^-(b)$  and  $s^+(b)$  are the left state and the right state, respectively; while the associated symbols  $x_1(b)$  and  $x_2(b)$  are the transmitted signals at Sender 1 and Sender 2, respectively.

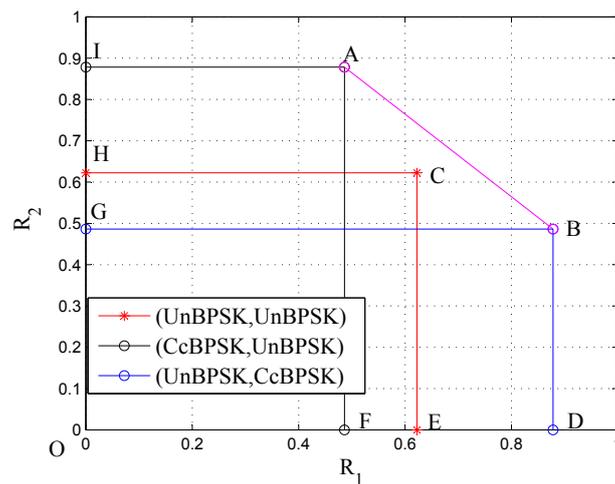


Figure 4. The evaluated achievable rate pairs of a specific Gaussian interference channel (GIFC), where  $P_1 = P_2 = 7.0$  dB and  $a = 0.5$ . The rectangle OECH with legend “(UnBPSK, UnBPSK)” corresponds to the case when both senders use UnBPSK as inputs; the rectangle ODBG with legend “(UnBPSK, CcBPSK)” corresponds to the case when Sender 1 uses UnBPSK as input and Sender 2 uses CcBPSK as input; and the rectangle OFAI with legend “(CcBPSK, UnBPSK)” corresponds to the case when Sender 1 uses CcBPSK as input and Sender 2 uses UnBPSK as input.

#### 4. Decoding Algorithms for Channels with Structured Interference

The purpose of this section has two-folds. The first is to show the decoding gain achieved by taking into account the structure of the interference. The second is to present a coding scheme to approach the point “B” in Figure 4.

##### 4.1. A Coding Scheme

We design a coding scheme using Kite codes (The main reason that we choose Kite codes is that it is convenient to set up the code rates. Actually, given data length, the code rates of Kite codes can be “continuously” varying from 0.1 to 0.9 with satisfactory performance, as shown in [31,32].). Kite codes are a class of low-density parity-check (LDPC) codes, which can be decoded using the sum-product

algorithm (SPA) [33,34]. As shown in Figure 5, Sender 1 uses a Kite code (with a parity-check matrix  $\mathbf{H}_1$ ) and Sender 2 uses a Kite code (with a parity-check matrix  $\mathbf{H}_2$ ) concatenated with the convolutional code with the generator matrix

$$G(D) = [1 + D + D^2 \ 1 + D^2].$$

*Encoding:* For Sender 1, a binary sequence  $\mathbf{u}_1 = (u_{1,1}, u_{1,2}, \dots, u_{1,L_1})$  of length  $L_1$  is encoded by a Kite code into a coded sequence  $\mathbf{c}_1 = (c_{1,1}, c_{1,2}, \dots, c_{1,N})$  of length  $N$ . For Sender 2, a binary sequence  $\mathbf{u}_2 = (u_{2,1}, u_{2,2}, \dots, u_{2,L_2})$  of length  $L_2$  is firstly encoded by a Kite code into a sequence  $\mathbf{v}_2 = (v_{2,1}, v_{2,2}, \dots, v_{2,N'})$  of length  $N'$  and then the sequence  $\mathbf{v}_2$  is encoded by the convolutional code with the generator matrix  $G(D)$  into a coded sequence  $\mathbf{c}_2 = (c_{2,1}, c_{2,2}, \dots, c_{2,N})$  of length  $N$ .

*Modulation:* The codewords  $\mathbf{c}_k$  are mapped into the bipolar sequences  $\mathbf{x}_k = (x_{k,1}, x_{k,2}, \dots, x_{k,N})$  with  $x_{k,i} = \sqrt{P_k}(1 - 2c_{k,i})$  ( $k = 1, 2$ ), where  $P_k$  is the power. Then we transmit  $\mathbf{x}_k$  for  $k = 1, 2$  over the interference channel.

*Decoding:* After receiving  $\mathbf{y}_1$ , Receiver 1 tries to recover the transmitted message  $\mathbf{u}_1$ . Similarly, after receiving  $\mathbf{y}_2$ , Receiver 2 tries to recover the transmitted message  $\mathbf{u}_2$ . We will consider several decoding algorithms in the next subsection to recover the transmitted messages.

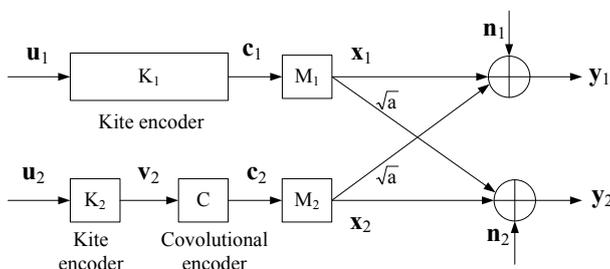


Figure 5. A coding scheme for the two-user GIFC.

#### 4.2. Decoding Algorithms

In this subsection, depending on the knowledge about the interference, we design four decoding schemes, including “knowing only the power of the interference”, “knowing the signaling of the interference”, “knowing the CC” and “knowing the whole structure”. We focus on the decoding of Receiver 1, while the decoding of Receiver 2 can be implemented similarly (There is no decoding scheme “Knowing the CC” for User 2 because User 1 has no convolutional structure). All these decoding algorithms will be described as *message processing/passing* algorithms over normal graphs [35].

##### 4.2.1. Message Processing/Passing Algorithms over Normal Graphs

Figure 6 shows a normal graph consisting of edges and vertices, which represent variables and subsystem constraints, respectively. Let  $\mathbf{Z} = \{Z_1, Z_2, \dots, Z_n\}$  be  $n$  distinct random variables that form a subsystem  $S^{(0)}$ . A normal subgraph with edges representing  $\mathbf{Z}$  and a vertex  $S^{(0)}$  representing the constraints can be used to depict the subsystem. Each *half-edge* (ending with a dongle) may potentially be coupled to some half-edge in other subsystems. For instance,  $Z_1$  and  $Z_m$  are displayed to be connected to subsystems  $S^{(1)}$  and  $S^{(m)}$ , respectively. We call the corresponding edge *full-edge*. A *message* is associated with each edge, which is defined as the pmf/pdf of the corresponding variable. As in [36], we use the notation  $P_{Z_i}^{(S^{(i)} \rightarrow S^{(0)})}(z)$  to denote the message from  $S^{(i)}$  to  $S^{(0)}$ . In particular, we use the notation  $P_{Z_i}^{(I \rightarrow S^{(0)})}(z)$  to represent the initial messages “driving” the subsystem  $S^{(0)}$ . For example, such initial messages can be the a priori probabilities from the source or the a posteriori probabilities computed from the channel observations. Assume that all messages to  $S^{(0)}$  are

available. The vertex  $S^{(0)}$ , as a *message processor*, delivers the outgoing message with respect to any given  $Z_i$  by computing the likelihood function

$$P_{Z_i}^{(S^{(0)} \rightarrow S^{(i)})}(z) \propto \Pr \left\{ S^{(0)} \text{ is satisfied} \mid Z_i = z \right\}, z \in \mathcal{Z} \tag{23}$$

by considering all the available messages and the system constraints. Since the computation of the likelihood function is irrelevant to the incoming message  $P_{Z_i}^{(S^{(i)} \rightarrow S^{(0)})}(z)$ , we claim that  $P_{Z_i}^{(S^{(0)} \rightarrow S^{(i)})}(z)$  is exactly the so-called *extrinsic message*.

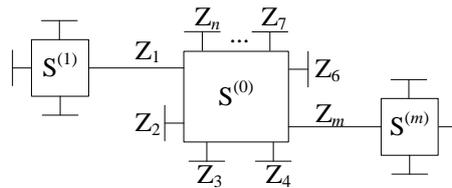


Figure 6. A normal graph of a general (sub)system.

#### 4.2.2. Knowing Only the Power of the Interference

The decoding scheme for “knowing only the power of the interference” is the simplest one, which can be described as a message processing/passing algorithm over the normal graph as shown in Figure 7a. In this scheme, the interference from Sender 2 is treated as a Gaussian distribution with mean zero and variance  $aP_2$ , where “ $P_2$ ” is the power and “ $a$ ” is the square of interference coefficient. That is, Receiver 1 assumes that  $X_{2,j} \sim \mathcal{N}(0, aP_2)$  for  $j = 1, 2, \dots, N$ . Since  $N_{1,j} \sim \mathcal{N}(0, 1)$  for  $j = 1, 2, \dots, N$ , the decoding algorithm is initialized by the initial messages as follows

$$P_{C_{1,j}}^{(\Sigma_1 \rightarrow K_1)}(c) = \Pr \left\{ C_{1,j} = c \mid \mathbf{y}_1, X_{2,j} \sim \mathcal{N}(0, aP_2), j = 1, 2, \dots, N \right\} \tag{24}$$

$$\propto \frac{1}{\sqrt{2\pi(1+aP_2)}} \exp \left\{ -\frac{[y_{1,j} - \sqrt{P_1}(1-2c)]^2}{2(1+aP_2)} \right\}, c \in \mathbb{F}_2$$

for  $j = 1, 2, \dots, N$ . Then the decoding algorithm uses SPA to compute iteratively the extrinsic messages  $P_{U_{1,i}}^{(K_1 \rightarrow |)}$  and  $P_{C_{1,j}}^{(K_1 \rightarrow \Sigma_1)}$ . Once these are done, we make the following decisions:

$$\hat{u}_{1,i} = \begin{cases} 0, & \text{if } P_{U_{1,i}}^{(| \rightarrow K_1)}(0)P_{U_{1,i}}^{(K_1 \rightarrow |)}(0) > P_{U_{1,i}}^{(| \rightarrow K_1)}(1)P_{U_{1,i}}^{(K_1 \rightarrow |)}(1), \\ 1, & \text{otherwise.} \end{cases} \tag{25}$$

$$\hat{c}_{1,j} = \begin{cases} 0, & \text{if } P_{C_{1,j}}^{(\Sigma_1 \rightarrow K_1)}(0)P_{C_{1,j}}^{(K_1 \rightarrow \Sigma_1)}(0) > P_{C_{1,j}}^{(\Sigma_1 \rightarrow K_1)}(1)P_{C_{1,j}}^{(K_1 \rightarrow \Sigma_1)}(1), \\ 1, & \text{otherwise.} \end{cases} \tag{26}$$

for  $i = 1, 2, \dots, L_1$  and  $j = 1, 2, \dots, N$ . The details about the decoding algorithm are shown as below.

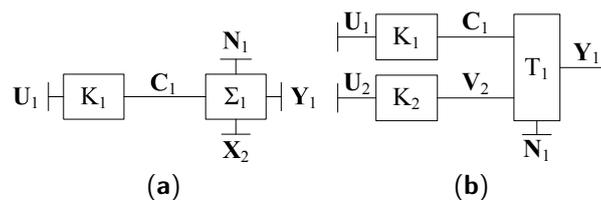


Figure 7. The normal graphs: (a) stands for the normal graph of “knowing only the power of the interference” and “knowing the signaling of the interference” for Decoder 1; (b) stands for the normal graph of “knowing the CC” and “knowing the whole structure” for Decoder 1.

**Algorithm 2 (“knowing only the power of the interference”).**

- Initialization:
  1. Initialize  $P_{U_{1,i}}^{(I \rightarrow K_1)}(u) = \frac{1}{2}$  for  $i = 1, 2, \dots, L_1$  and  $u \in \mathbb{F}_2$ .
  2. Compute  $P_{C_{1,j}}^{(\Sigma_1 \rightarrow K_1)}(c)$  for  $j = 1, 2, \dots, N$  and  $c \in \mathbb{F}_2$  according to (24).
  3. Set a maximum iteration number  $J$  and iteration variable  $j = 1$ .
- Repeat while  $j \leq J$ :
  1. Compute extrinsic messages  $P_{U_{1,i}}^{(K_1 \rightarrow I)}$  and  $P_{C_{1,j}}^{(K_1 \rightarrow \Sigma_1)}$  for  $i = 1, 2, \dots, L_1$  and  $j = 1, 2, \dots, N$  using SPA.
  2. Make decisions according to (25) and (26). Denote  $\hat{\mathbf{u}}_1 = (\hat{u}_{1,1}, \hat{u}_{1,2}, \dots, \hat{u}_{1,L_1})$  and  $\hat{\mathbf{c}}_1 = (\hat{c}_{1,1}, \hat{c}_{1,2}, \dots, \hat{c}_{1,N})$ .
  3. Compute the syndrome  $\mathbf{S}_1 = \hat{\mathbf{c}}_1 \cdot \mathbf{H}_1^T$ . If  $\mathbf{S}_1 = \mathbf{0}$ , output  $\hat{\mathbf{u}}_1$  and  $\hat{\mathbf{c}}_1$  and exit the iteration.
  4. Set  $j = j + 1$ . If  $\mathbf{S}_1 \neq \mathbf{0}$  and  $j > J$ , report a decoding failure.
- End decoding.

4.2.3. Knowing the Signaling of the Interference

The decoding algorithm for this scheme is almost the same as Algorithm 2, see Figure 7a. The difference is that  $X_{2,j} \sim \mathcal{B}(1/2)$  (Bernoulli-1/2 distribution. Strictly speaking,  $X_{2,j}$  is a shift/scaling version of  $\mathcal{B}(1/2)$ .) for  $j = 1, 2, \dots, N$ . So the computation of  $P_{C_{1,j}}^{(\Sigma_1 \rightarrow K_1)}(c)$  is changed into

$$\begin{aligned}
 P_{C_{1,j}}^{(\Sigma_1 \rightarrow K_1)}(c) &= \Pr \{C_{1,j} = c | \mathbf{y}_1, X_{2,j} \sim \mathcal{B}(1/2), j = 1, 2, \dots, N\} \\
 &\propto \frac{1}{2} \frac{1}{\sqrt{2\pi}} \exp \left\{ -\frac{[y_{1,j} - \sqrt{P_1}(1 - 2c) - \sqrt{aP_2}]^2}{2} \right\} \\
 &+ \frac{1}{2} \frac{1}{\sqrt{2\pi}} \exp \left\{ -\frac{[y_{1,j} - \sqrt{P_1}(1 - 2c) + \sqrt{aP_2}]^2}{2} \right\}, \\
 & c \in \mathbb{F}_2
 \end{aligned} \tag{27}$$

Then the decoding algorithm of “knowing the signaling of the interference” can be shown as below.

**Algorithm 3 (“knowing the signaling of the interference”).**

- Initialization:
  1. Initialize  $P_{U_{1,i}}^{(I \rightarrow K_1)}(u) = \frac{1}{2}$  for  $i = 1, 2, \dots, L_1$  and  $u \in \mathbb{F}_2$ .
  2. Compute  $P_{C_{1,j}}^{(\Sigma_1 \rightarrow K_1)}(c)$  for  $j = 1, 2, \dots, N$  and  $c \in \mathbb{F}_2$  according to (27).
  3. Set a maximum iteration number  $J$  and iteration variable  $j = 1$ .
- Repeat while  $j \leq J$ :
  1. Compute extrinsic messages  $P_{U_{1,i}}^{(K_1 \rightarrow I)}$  and  $P_{C_{1,j}}^{(K_1 \rightarrow \Sigma_1)}$  for  $i = 1, 2, \dots, L_1$  and  $j = 1, 2, \dots, N$  using SPA.
  2. Make decisions according to (25) and (26), respectively.
  3. Compute the syndrome  $\mathbf{S}_1 = \hat{\mathbf{c}}_1 \cdot \mathbf{H}_1^T$ . If  $\mathbf{S}_1 = \mathbf{0}$ , output  $\hat{\mathbf{u}}_1$  and  $\hat{\mathbf{c}}_1$  and exit the iteration.
  4. Set  $j = j + 1$ . If  $\mathbf{S}_1 \neq \mathbf{0}$  and  $j > J$ , report a decoding failure.
- End decoding.

#### 4.2.4. Knowing the CC

“Knowing the CC” means that Decoder 1 knows the structure of the convolutional code. This scheme can be described as a message processing/passing algorithm over the normal graph as shown in Figure 7b. Actually, the vertex  $T_1$  is a combination of three subsystems, convolutional encoder, modulation and GIFC constraint, which can be specified by a trellis  $\mathcal{T}$  with parallel branches [16]. Therefore, the BCJR algorithm can be used to compute the extrinsic messages  $P_{C_{1,j}}^{(T_1 \rightarrow K_1)}(c)$  for  $j = 1, 2, \dots, N$  over the trellis  $\mathcal{T}$ . Since the structure of Kite code for Sender 2 is unknown, the constraint of vertex  $K_2$  is inactive. In this case, the pmf of variable  $V_{2,k}$  ( $k = 1, 2, \dots, N'$ ) is assumed to be Bernoulli-1/2 distribution. There are two strategies to implement the BCJR algorithm. One is called “BCJR-once”, in which the BCJR algorithm is performed only once. The other strategy is called “BCJR-repeat”, in which the BCJR algorithm is performed more than once. In this scheme, the decoding decisions on  $C_{1,j}$  are modified into

$$\hat{c}_{1,j} = \begin{cases} 0, & \text{if } P_{C_{1,j}}^{(T_1 \rightarrow K_1)}(0)P_{C_{1,j}}^{(K_1 \rightarrow T_1)}(0) > P_{C_{1,j}}^{(T_1 \rightarrow K_1)}(1)P_{C_{1,j}}^{(K_1 \rightarrow T_1)}(1), \\ 1, & \text{otherwise,} \end{cases} \quad (28)$$

for  $j = 1, 2, \dots, N$ . These two decoding procedures are described in Algorithms 4 and 5, respectively.

#### Algorithm 4 (BCJR-once).

- Initialization:
  1. Initialize pmf  $P_{C_{1,j}}^{(K_1 \rightarrow T_1)}(c) = \frac{1}{2}$  and  $P_{C_{2,j}}^{(l \rightarrow T_1)}(c) = \frac{1}{2}$  for  $j = 1, 2, \dots, N, c \in \mathbb{F}_2$  and  $P_{V_{2,k}}^{(l \rightarrow T_1)}(v) = \frac{1}{2}$  for  $k = 1, 2, \dots, N', v \in \mathbb{F}_2$ .
  2. Compute extrinsic messages  $P_{C_{1,j}}^{(T_1 \rightarrow K_1)}(c)$  for  $j = 1, 2, \dots, N, c \in \mathbb{F}_2$  using BCJR algorithm over the parallel branch trellis  $\mathcal{T}$ .
  3. Set a maximum iteration number  $J$  and iteration variable  $j = 1$ .
- Repeat while  $j \leq J$ :
  1. Compute extrinsic messages  $P_{U_{1,i}}^{(K_1 \rightarrow |)}$  and  $P_{C_{1,j}}^{(K_1 \rightarrow T_1)}$  for  $i = 1, 2, \dots, L_1$  and  $j = 1, 2, \dots, N$  using SPA.
  2. Make decisions according to (25) and (28).
  3. Compute the syndrome  $\mathbf{S}_1 = \hat{\mathbf{c}}_1 \cdot \mathbf{H}_1^T$ . If  $\mathbf{S}_1 = \mathbf{0}$ , output  $\hat{\mathbf{u}}_1$  and  $\hat{\mathbf{c}}_1$  and exit the iteration.
  4. Set  $j = j + 1$ . If  $\mathbf{S}_1 \neq \mathbf{0}$  and  $j > J$ , report a decoding failure.
- End Decoding

#### Algorithm 5 (BCJR-repeat).

- Initialization:
  1. Initialize pmf  $P_{C_{1,j}}^{(K_1 \rightarrow T_1)}(c) = \frac{1}{2}$  and  $P_{C_{2,j}}^{(l \rightarrow T_1)}(c) = \frac{1}{2}$  for  $j = 1, 2, \dots, N, c \in \mathbb{F}_2$  and  $P_{V_{2,k}}^{(l \rightarrow T_1)}(v) = \frac{1}{2}$  for  $k = 1, 2, \dots, N', v \in \mathbb{F}_2$ .
  2. Set a maximum iteration number  $J$  and iteration variable  $j = 1$ .
- Repeat while  $j \leq J$ :
  1. Compute extrinsic messages  $P_{C_{1,j}}^{(T_1 \rightarrow K_1)}(c)$  for  $j = 1, 2, \dots, N, c \in \mathbb{F}_2$  using BCJR algorithm over the parallel branch trellis  $\mathcal{T}$ .
  2. Compute extrinsic messages  $P_{U_{1,i}}^{(K_1 \rightarrow |)}$  and  $P_{C_{1,j}}^{(K_1 \rightarrow T_1)}$  for  $i = 1, 2, \dots, L_1$  and  $j = 1, 2, \dots, N$  using SPA.

3. Make decisions according to (25) and (28).
  4. Compute the syndrome  $\mathbf{S}_1 = \hat{\mathbf{c}}_1 \cdot \mathbf{H}_1^T$ . If  $\mathbf{S}_1 = \mathbf{0}$ , output  $\hat{\mathbf{u}}_1$  and  $\hat{\mathbf{c}}_1$  and exit the iteration.
  5. Set  $j = j + 1$ . If  $\mathbf{S}_1 \neq \mathbf{0}$  and  $j > J$ , report a decoding failure.
- End Decoding

#### 4.2.5. Knowing the Whole Structure

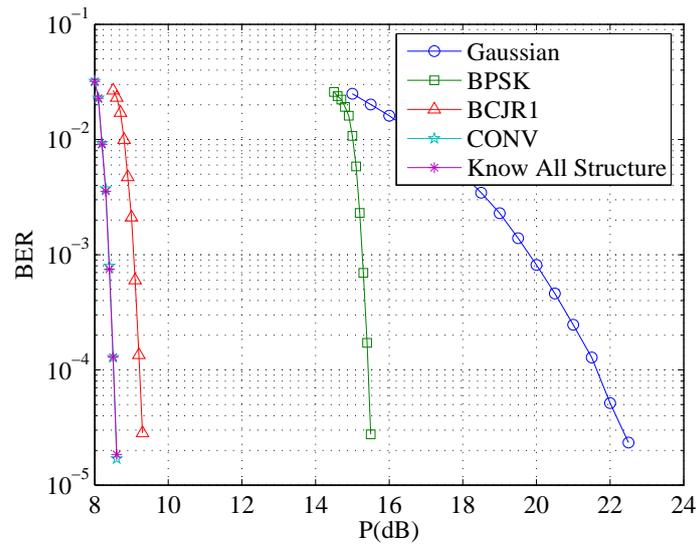
The scheme “knowing the whole structure” for Receiver 1 can also be described as a message processing/passing algorithm over the normal graph shown in Figure 7b. Since knowing the whole structure of the interference, Receiver 1 can decode iteratively utilizing the structure of both users. Using the BCJR algorithm,  $P_{C_{1,j}}^{(T_1 \rightarrow K_1)}(c)$  and  $P_{V_{2,k}}^{(T_1 \rightarrow K_2)}(v)$  are computed simultaneously over the parallel branch trellis  $\mathcal{T}$ . The iterative decoding algorithm is presented in Algorithm 6.

#### Algorithm 6 (“knowing the whole structure”).

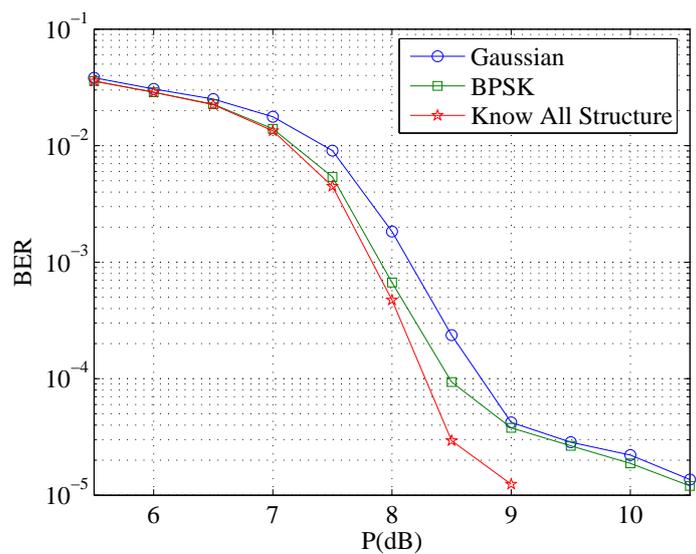
- Initialization:
  1. Initialize pmf  $P_{C_{1,j}}^{(K_1 \rightarrow T_1)}(c) = \frac{1}{2}$  and  $P_{C_{2,j}}^{(I \rightarrow T_1)}(c) = \frac{1}{2}$  for  $j = 1, 2, \dots, N, c \in \mathbb{F}_2$  and  $P_{V_{2,k}}^{(K_2 \rightarrow T_1)}(v) = \frac{1}{2}$  for  $k = 1, 2, \dots, N', v \in \mathbb{F}_2$ .
  2. Set a maximum iteration number  $J$  and iteration variable  $j = 1$ .
- Repeat while  $j \leq J$ :
  1. Compute extrinsic messages  $P_{C_{1,j}}^{(T_1 \rightarrow K_1)}(c)$  for  $j = 1, 2, \dots, N, c \in \mathbb{F}_2$  and  $P_{V_{2,k}}^{(T_1 \rightarrow K_2)}(v)$  for  $k = 1, 2, \dots, N', v \in \mathbb{F}_2$  using BCJR algorithm over the parallel branch trellis  $\mathcal{T}$ .
  2. Compute extrinsic messages  $P_{U_{1,i}}^{(K_1 \rightarrow I)}$  and  $P_{C_{1,j}}^{(K_1 \rightarrow T_1)}$  for  $i = 1, 2, \dots, L_1$  and  $j = 1, 2, \dots, N$  using SPA.
  3. Compute extrinsic messages  $P_{V_{2,k}}^{(K_2 \rightarrow T_1)}(v)$  for  $k = 1, 2, \dots, N', v \in \mathbb{F}_2$  using SPA.
  4. Make decisions according to (25) and (28).
  5. Compute the syndrome  $\mathbf{S}_1 = \hat{\mathbf{c}}_1 \cdot \mathbf{H}_1^T$ . If  $\mathbf{S}_1 = \mathbf{0}$ , output  $\hat{\mathbf{u}}_1$  and  $\hat{\mathbf{c}}_1$  and exit the iteration.
  6. Set  $j = j + 1$ . If  $\mathbf{S}_1 \neq \mathbf{0}$  and  $j > J$ , report a decoding failure.
- End Decoding

#### 4.3. Numerical Results

In this subsection, simulation results of the decoding algorithms are shown and analyzed. Simulation parameters of Figures 8 and 9 are presented in Table 1. In these two figures, we let the power constraints of two senders be same, that is,  $P_1 \equiv P_2 = P$ . Here, “Gaussian” stands for the scheme “knowing only the power of the interference”, “BPSK” stands for the scheme “knowing the signaling of the interference”, “BCJR1” stands for the scheme “BCJR-once”, “CONV” stands for the scheme “BCJR-repeat” and “Know All Structure” stands for the scheme “knowing the whole structure”. Figure 8 shows the error performance of Receiver 1. From Figure 8, we can see that, for Receiver 1, more details of the structure of the interference are known, better performance is obtained, that is, the decoding gains get larger. Similarly, Figure 9 gives the error performance of Receiver 2, where the scheme “knowing the whole structure” still has the best performance and the decoding gain can be achieved by taking into account the structure of the interference.



**Figure 8.** The error performance of Receiver 1. “Gaussian” stands for the scheme “knowing only the power of the interference”, “BPSK” stands for the scheme “knowing the signaling of the interference”, “BCJR1” stands for the scheme “BCJR-once”, “CONV” stands for the scheme “BCJR-repeat” and “Know All Structure” stands for the scheme “knowing the whole structure”.



**Figure 9.** The error performance of Receiver 2. “Gaussian” stands for the scheme “knowing only the power of the interference”, “BPSK” stands for the scheme “knowing the signaling of the interference” and “Know All Structure” stands for the scheme “knowing the whole structure”.

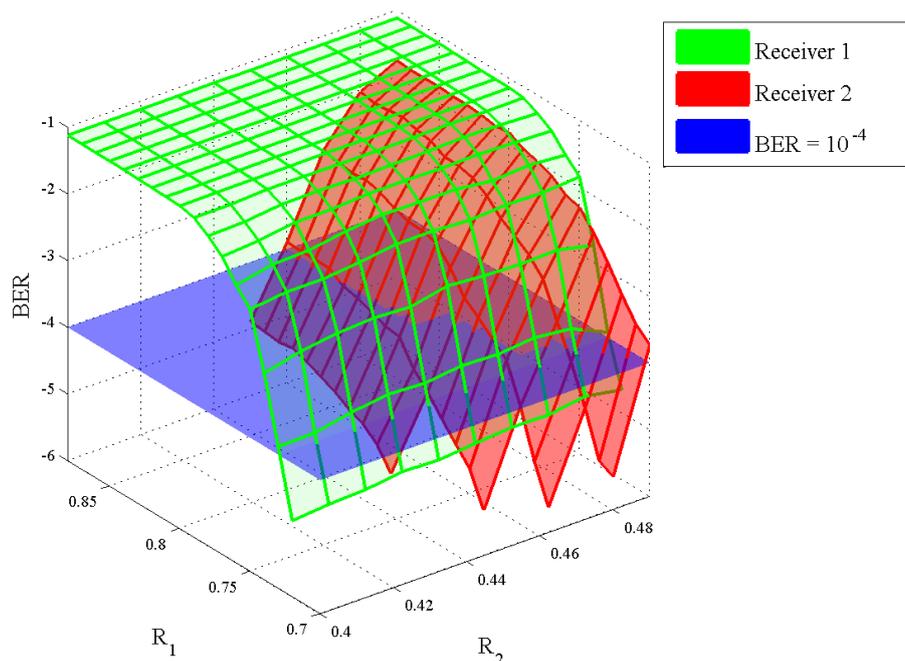
**Table 1.** Parameters of the bit error rate (BER) performance simulations.

Parameters	Values
Square of interference coefficient $a$	0.5
Maximum iteration number $J$	200
Kite Code of Sender 1	$N = 10000, L_1 = 8782$
Kite Code of Sender 2	$N' = 5000, L_2 = 4862$
Generator matrix $G(D)$	$[1 + D + D^2 \quad 1 + D^2]$
Code rate pair $(R_1, R_2)$	$(0.8782, 0.4862)$

As we said before, another objective in this section is to find out a code rate pair nearest to the point “B” in Figure 4 with bit error rate (BER) performance of  $10^{-4}$ . So we do the simulations with different code rate pairs. In the simulations, we adopt the scheme “knowing the whole structure” and gradually decrease the code rates from the point “B” with a step length 0.01. Simulation parameters for different code rate pairs are listed in Table 2, while the simulation results are presented using a 3D graph in Figure 10. From the figure, it is obvious that as the code rates of two users are decreasing, the BER also decreases. Finally, we find that the “best” code rate pair is (0.71, 0.48) for User 1 and User 2. The theoretical value of the point “B” is about (0.878, 0.486). So we can see that the gap between the result using our decoding scheme and the theoretical value is small.

**Table 2.** Parameters of the simulations for different code rate pairs.

Parameters	Values
Square of interference coefficient $a$	0.5
Maximum iteration number $J$	200
Code length $N$ of Kite Code of Sender 1	10000
Code length $N'$ of Kite Code of Sender 2	5000
Generator matrix $G(D)$	$[1 + D + D^2 \ 1 + D^2]$
Step length	100
Range of message length $L_1$	7100 ~ 8800
Range of message length $L_2$	4000 ~ 4900



**Figure 10.** Error performance of two users with different code rate pairs  $(R_1, R_2)$ . Blue plane represents BER level, green surface stands for the error performance of Receiver 1 and red surface stands for the error performance of Receiver 2.

## 5. Conclusions

The paper showed that the capacity region of the two-user general IC is the union of a family of rectangles. Each rectangle is defined by a pair of spectral inf-mutual information rates associated with two independent input processes. When the channel is stationary memoryless and the inputs are discrete Markov, we can calculate the defined pair of rates. We can also conclude that taking into account the structure of the interference processes can improve the simplest inner bounds (obtained by

treating the interference as noise). Also, a concrete coding scheme to approach the theoretical achievable rate pairs was presented, which showed that the decoding gain can be achieved by considering the structure of the interference.

**Acknowledgments:** This paper was presented in part at the 2012 IEEE International Symposium Information Theory. This work was partially supported by the China NSFs (Nos. 61601131 and 61401177), the Guangdong NSFs (Nos. 2016A030313727, 2016A030308008 and 2014A030310183) and the Foundation for Distinguished Young Talents in Higher Education of Guangdong (No. 2015KQNCX086). This work was also supported by the Science and Technology Project of Guangzhou (No. 201510010193).

**Author Contributions:** Xiao Ma conceived the generalized scheme and contributed to the writing of manuscript; Lei Lin provided the proof and completed the paper writing; Chulong Liang performed the experiments and numerical analysis. Xiujie Huang and Baoming Bai helped analyze the data and contributed to the writing of manuscript. All authors have read and approved the final manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Carleial, A.B. Interference channels. *IEEE Trans. Inf. Theory* **1978**, *24*, 60–70.
2. Shannon, C.E. Two-way communication channels. In Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Berkely, CA, USA, 20 June–30 July 1960; Neyman, J., Ed.; University of California Press: Berkely, CA, USA, 1961; Volume 1, pp. 611–644.
3. Ahlswede, R. The capacity region of a channel with two senders and two receivers. *Ann. Probab.* **1974**, *2*, 805–814.
4. Carleial, A.B. A case where interference does not reduce capacity. *IEEE Trans. Inf. Theory* **1975**, *21*, 569–570.
5. Han, T.S.; Kobayashi, K. A new achievable rate region for the interference channel. *IEEE Trans. Inf. Theory* **1981**, *27*, 49–60.
6. El Gamal, A.A.; Costa, M.H.M. The capacity region of a class of deterministic interference channels. *IEEE Trans. Inf. Theory* **1982**, *28*, 343–346.
7. Costa, M.H.M.; El Gamal, A.A. The capacity region of the discrete memoryless interference channel with strong interference. *IEEE Trans. Inf. Theory* **1987**, *33*, 710–711.
8. Kramer, G. Outer bounds on the capacity of Gaussian interference channels. *IEEE Trans. Inf. Theory* **2004**, *50*, 581–586.
9. Sato, H. Two-user communication channels. *IEEE Trans. Inf. Theory* **1977**, *23*, 295–304.
10. Carleial, A.B. Outer bounds on the capacity of interference channels. *IEEE Trans. Inf. Theory* **1983**, *29*, 602–606.
11. Sato, H. On degraded Gaussian two-user channels. *IEEE Trans. Inf. Theory* **1978**, *24*, 637–640.
12. Costa, M.H.M. On the Gaussian interference channel. *IEEE Trans. Inf. Theory* **1985**, *31*, 607–615.
13. Kramer, G. Review of rate regions for interference channels. In Proceedings of the 2006 IEEE International Zurich Seminar on Communications (IZS), Zurich, Switzerland, 22–24 February 2006; pp. 162–165.
14. Chong, H.F.; Motani, M.; Garg, H.K.; Gamal, H.E. On the Han-Kobayashi region for the interference channel. *IEEE Trans. Inf. Theory* **2008**, *54*, 3188–3195.
15. Etkin, R.H.; Tse, D.N.C.; Wang, H. Gaussian interference channel capacity to within one bit. *IEEE Trans. Inf. Theory* **2008**, *54*, 5534–5562.
16. Huang, X.; Ma, X.; Lin, L.; Bai, B. Accessible Capacity of Secondary Users over the Gaussian Interference Channel. In Proceedings of the IEEE International Symposium on Information Theory, St. Petersburg, Russia, 31 July–5 August 2011; pp. 811–815.
17. Huang, X.; Ma, X.; Lin, L.; Bai, B. Accessible Capacity of Secondary Users. *IEEE Trans. Inf. Theory* **2014**, *60*, 4722–4738.
18. Baccelli, F.; El Gamal, A.A.; Tse, D. Interference Networks with Point-to-Point Codes. *IEEE Trans. Inf. Theory* **2011**, *57*, 2582–2596.
19. Moshksar, K.; Ghasemi, A.; Khandani, A.K. An Alternative To Decoding Interference or Treating Interference As Gaussian Noise. In Proceedings of the IEEE International Symposium on Information Theory, St. Petersburg, Russia, 31 July–5 August 2011; pp. 1176–1180.
20. Han, T.S.; Verdú, S. Approximation theory of output statistics. *IEEE Trans. Inf. Theory* **1993**, *39*, 752–772.

21. Han, T.S. *Information-Spectrum Methods in Information Theory*; Springer: Berlin/Heidelberg, Germany, 2003.
22. Verdú, S.; Han, T.S. A general formula for channel capacity. *IEEE Trans. Inf. Theory* **1994**, *40*, 1147–1157.
23. Han, T.S. An information-spectrum approach to capacity theorems for the general multiple-access channel. *IEEE Trans. Inf. Theory* **1998**, *44*, 2773–2795.
24. Lin, L.; Ma, X.; Huang, X.; Bai, B. An information spectrum approach to the capacity region of general interference channel. In Proceedings of the IEEE International Symposium on Information Theory, Cambridge, MA, USA, 1–6 July 2012; pp. 2261–2265.
25. Etkin, R.H.; Merhav, N.; Ordentlich, E. Error Exponents of Optimum Decoding for the Interference Channel. *IEEE Trans. Inf. Theory* **2010**, *56*, 40–56.
26. Kavčić, A. On the capacity of Markov sources over noisy channels. In Proceedings of the IEEE 2001 Global Telecommunications Conference, San Antonio, TX, USA, 25–29 November 2001; Volume 5, pp. 2997–3001.
27. Arnold, D.M.; Loeliger, H.A. On the information rate of binary-input channels with memory. In Proceedings of the IEEE International Conference on Communications, Helsinki, Finland, 11–14 June 2001; Volume 9, pp. 2692–2695.
28. Pfister, H.D.; Soriaga, J.B.; Siegel, P.H. On the achievable information rates of finite state ISI channels. In Proceedings of the IEEE Global Telecommunications Conference, San Antonio, TX, USA, 25–29 November 2001; Volume 5, pp. 2992–2996.
29. Bahl, L.R.; Cocke, J.; Jelinek, F.; Raviv, J. Optimal decoding of linear codes for minimizing symbol error rate. *IEEE Trans. Inf. Theory* **1974**, *20*, 284–287.
30. Cover, T.M.; Thomas, J.A. *Elements of Information Theory*; John Wiley & Sons: Hoboken, NJ, USA, 1991.
31. Ma, X.; Zhao, S.; Zhang, K.; Bai, B. Kite codes over Groups. In Proceedings of the IEEE Information Theory Workshop, Paraty, Brazil, 16–20 October 2011; pp. 481–485.
32. Zhang, K.; Ma, X.; Zhao, S.; Bai, B.; Zhang, X. A New Ensemble of Rate-Compatible LDPC Codes. In Proceedings of the IEEE International Symposium on Information Theory, Cambridge, MA, USA, 1–6 July 2012.
33. Wiberg, N.; Loeliger, H.A.; Kötter, R. Codes and iterative decoding on general graphs. *Eur. Trans. Commun.* **1995**, *6*, 513–526.
34. Kschischang, F.R.; Frey, B.J.; Loeliger, H.A. Factor graphs and the sum-product algorithm. *IEEE Trans. Inf. Theory* **2001**, *47*, 498–519.
35. Forney, G.D., Jr. Codes on graphs: Normal realizations. *IEEE Trans. Inf. Theory* **2001**, *47*, 520–548.
36. Ma, X.; Zhang, K.; Chen, H.; Bai, B. Low Complexity X-EMS Algorithms for Nonbinary LDPC Codes. *IEEE Trans. Commun.* **2012**, *60*, 9–13.



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).