User Cooperation via Rateless Coding

Mahyar Shirvanimoghaddam, Yonghui Li and Branka Vucetic
School of Electrical and Information Engineering
The University of Sydney
Sydney, NSW, Australia
Emails: {mahyar.shirvanimoghaddam, yonghui.li, branka.vucetic}@sydney.edu.au

Abstract—This paper presents a new rateless coded cooperation (CC) scheme for the two-user cooperative multiple access channel (CMAC), where two users cooperatively communicate with a common destination. We consider two rateless CC strategies, a fully coded cooperation (FCC) scheme used in the conventional rateless cooperative schemes and a new partially coded cooperation (PCC) scheme. In FCC, each user starts coded cooperation process only after the whole block of the other user’s information symbols are fully recovered. In contrast, in PCC, each user starts cooperation as soon as it receives a fraction of new message sent from the other user. The degree distribution for the PCC scheme is designed to maximize the overall system throughput. Simulation results show that the proposed PCC scheme achieves a considerably higher throughput than the conventional scheme in various scenarios.

Index Terms—cooperative communication, MAC, rateless code, Raptor code.

I. INTRODUCTION

Due to the broadcast nature of wireless transmission, each user can overhear other users’ transmissions. As a result, each user can help other users in forwarding their messages. Therefore, each user’s messages are forwarded by multiple cooperative users and thus a cooperative diversity can be achieved for each user [1, 2]. So far, several user cooperation strategies have been proposed. In [3], a coded cooperation scheme was introduced. In this scheme, each user broadcasts its coded messages to other users and also to the destination. Each user then decodes the partners’ codeword and generates additional parity symbols before forwarding them to the destination. The coded cooperation achieves not only the cooperative diversity, but additional coding gains through the distributed coding process. Various coded cooperation schemes based on convolutional codes, turbo codes, space time codes and LDPC codes have been proposed in [4, 5, 6, 7] and their performances have been analyzed.

As a class of powerful codes originally designed for erasure channels, rateless codes have recently drawn a lot of attention. LT [8] and Raptor [9] codes are two common types of rateless codes. They are rateless in the sense that the total number of generated coded symbols is potentially limitless. Recently, rateless codes have been applied to the coded cooperation to further improve the transmission efficiency. As an example, Raptor codes have been applied to the simple three-nodes relay channel in [10, 11, 12].

In [13], a rateless coded cooperation scheme has been proposed for a practical half-duplex two user multiple access channel (MAC). In this scheme, only one user helps the other user in its transmission. Specifically, the cooperation starts only after one user has successfully transmitted its message to the destination and also completely decoded the other user’s message. In [14], a coded cooperation scheme based on rateless codes has been proposed for a 2-user CMAC. We refer to this scheme as the fully coded cooperation (FCC) scheme. Unlike [13], each user in FCC does not need to wait until its message is decoded at the destination before starting the cooperation phase. In fact, each user starts the cooperation process as soon as it fully recovers the other user’s message. The main drawback of the FCC scheme is that each user needs to successfully recover other users’ messages before it can help to forward them. This considerably decreases the transmission efficiency and increases the latency.

In this paper, we propose a partially coded cooperation (PCC) scheme based on rateless codes for the 2-user CMAC. In the proposed scheme, each user applies an LT code to encode its information symbols. The code for each user is designed in a way that they can be decoded partially by the partner and fully at the destination. Thus, at each transmission time frame (TF), a fraction of the new message from each user can be decoded by its partner. The partner can help to forward the message part decoded from the other user, by encoding it with its own message into a new rateless codeword. Simulation results show that the PCC scheme achieves a higher average system throughput than the FCC scheme in various simulated channel scenarios.

Current approaches on rateless coded 2-user CMAC use existing degree distributions designed for point-to-point transmission, which are not optimal for CMAC. In this paper, we will design and optimize the degree distribution for the PCC scheme to minimize the total overhead required to guarantee the successful decoding of both users’ messages at the destination. A linear programming optimization problem is then formulated to find the optimum degree distribution.

The rest of the paper is organized as follows. Section II reviews LT codes and their encoding and decoding processes. The system model, and the FCC and PCC schemes are then represented in Section III. In Section IV, the degree distribution optimization problem is defined and the resulting degree distributions are presented. Simulation results are provided in Section V, and finally Section VI concludes the paper.
II. REVIEW OF LT CODES

Rateless codes were originally designed and optimized for erasure channels. The first practical realization of rateless codes is Luby Transform (LT) code [8], whose encoding process contains two important steps. First, an integer \( d \) between 1 and \( k \) is obtained according to a predefined probability distribution function, where \( k \) is the message length. We refer to this integer as the degree and the probability distribution function as the degree distribution. In the next step, \( d \) distinct information symbols are selected uniformly at random and XORed to generate one coded symbol. Let \( \gamma \) denote the probability that the degree of a coded symbol is \( i \), then the degree distribution of the LT code can be represented by its generator polynomial, \( \Omega(x) = \sum_{i=1}^{k} \Omega_i x^i \). Moreover, information and coded symbols can be seen as vertices of a bipartite graph (Fig. 1), where the information and coded symbols are the variable nodes and check nodes, respectively [9].

We use the decoder structure originally proposed in [8] to recover information symbols. The decoding process can be summarized as follows. First, the decoder finds all degree-one coded symbols and recovers one information symbol from each one of these coded symbols. In the next step, all recovered information symbols are removed from the bipartite graph leading to emerging more degree one coded symbols. This iterative decoding algorithm proceeds until no more degree one coded symbol can be found. Fig. 1 shows the decoding algorithm of an LT code, where three information symbols are recovered from four coded symbols, which have been generated from six information symbols.

The performance of LT codes is mainly determined by their degree distributions. The degree distribution of LT codes is usually tuned to minimize the total number of coded symbols required to recover all information symbols. Since the decoding process of LT codes is terminated when the number of degree one coded symbols is zero, to maximize the probability of recovering all information symbols, the number of degree one coded symbols needs to be large enough at each iteration. In [9], it has been shown that when the degree distribution of coded symbols is \( \Omega(x) \), the probability that an information symbol is not recovered after \( (i + 1) \) iterations, \( p_{i+1} \), can be calculated as follows

\[
p_{i+1} = e^{-\gamma \Omega'(1-p_i)}, \tag{1}
\]

where \( \gamma \) is the ratio of the number of coded symbols and that of information symbols (i.e., overhead) and \( \Omega'(x) \) is the derivative of \( \Omega(x) \). As a result, if an expected \( x \)-fraction of information symbols has been already recovered at some step of the algorithm, that fraction increases to \( 1 - e^{\gamma \Omega'(x)} \) in the next step. So, the expected fraction of information symbols recovered in this step is \( 1 - x - e^{\gamma \Omega'(x)} \) [9].

In [9], a linear programming optimization approach has been proposed to find the optimum degree distribution in order to keep the number of degree one coded symbols large enough at each iteration. To do so, for given \( \gamma, \delta \) and some constant \( c \), the degree distribution is found such that

\[
1 - x - e^{\gamma \Omega'(x)} \geq c \sqrt{\frac{1-x}{k}} \tag{2}
\]

or equivalently

\[
\Omega'(x) \geq -\ln \left( 1 - x - c \sqrt{\frac{1-x}{k}} \right) \gamma \tag{3}
\]

for \( x \in [0, 1-\delta] \). Later, we will use the same idea as in [9] to find optimum degree distributions in our proposed rateless coded cooperation schemes.

III. RATELESS CODED COOPERATION

A. System Model

In this paper, we consider a two-user cooperative multiple access channel (Fig. 2), where each user (U_1 or U_2) wants to transmit \( k \) information symbols to the destination, D, via the help of the other user. The channel between each pair of nodes is considered to be an erasure channel with a specific erasure probability. Let \( e_1 \) and \( e_2 \) denote the erasure probability of the channel between U_1 and D, U_2 and D, and that between U_1 and U_2, respectively. Also the inter-user channel is considered to be reciprocal. We further assume a time-division transmission, where in each time frame (TF), first U_1 transmits \( N \) coded symbols, and then U_2 broadcasts \( N \) coded symbols.

B. Fully Coded Cooperation (FCC) Scheme

In FCC scheme, each user needs to wait until it can fully recover other user’s message before starting the cooperation process with the other user. The overall transmission of the FCC scheme is divided into two phases, a broadcast phase and a cooperation phase. In the broadcast phase, each user encodes its \( k \) information symbols into a codeword of length \( N \) by using an LT code with the degree distribution \( \Omega_1(x) \) and broadcast them in its allocated time slot. Also, each user overhears the other user’s transmission and tries to decode its message. If both users’
messages are successfully decoded at the destination before they are recovered by the other user, the destination will send an acknowledgment to both users. Then they start broadcasting new messages. Otherwise, each user that has successfully decoded the other user’s message, starts the cooperation phase and generates $N$ coded symbols from both users’ information symbols using another degree distribution $\Omega_2(x)$ and transmits them to the destination. In fact, each user remains in the broadcast phase if it cannot decode the other user’s message before receiving an acknowledgement from the destination. Fig. 3-a shows the FCC scheme for a 2-user CMAC.

Since the overall codeword received at the destination consists of the coded symbols transmitted in the broadcast phase with the degree distribution $\Omega_1(x)$ and that in the cooperative phase with the degree distribution $\Omega_2(x)$, the degree distribution of the overall codeword at the destination are different from that in the point-to-point transmission. Therefore, the conventional degree distribution designed for point-to-point transmission, such as those in [9], are not optimal anymore. New degree distributions need to be optimized for cooperative scenarios.

In [14], a linear programming optimization problem has been formulated to find the optimum degree distribution in the broadcast phase and the cooperative phase. The key property of the degree distribution is that when one user’s information symbols are known at the destination, the other user’s information symbols should be decoded at a minimal overhead. Also, when one user’s information symbols are known at the destination, they can be removed from the bipartite graph at the destination. Then, the degree distribution of the overall codeword will be updated. This degree distribution should be found in a way that the decoding process can proceed and all information symbols of the other user can be recovered. This can be guaranteed when the condition in (3) is satisfied. Further details of the FCC scheme and its degree distribution optimization problem can be found in [14].

C. Partially Coded Cooperation (PCC) Scheme

Rateless codes have the property that when the number of coded symbols is less than that of information symbols, the receiver is still able to recover some of information symbols. As an example, Fig. 1 shows that three information symbols are recovered from four coded symbols which are generated from six information symbols. It has been shown in [15, 16] that conventional degree distributions which have been originally designed for full decoding do not perform well in partial decoding when the number of coded symbols is less than that of information symbols. Therefore, new degree distributions should be found to achieve a higher recovery rate in partial decoding. The PCC scheme tries to efficiently use this property of rateless codes to achieve a higher system throughput in CMACs.

In the PCC scheme, each user first generates $N$ coded symbols from its own information symbols using a LT code with the degree distribution $\Omega(x)$, and transmits them to the other user and the destination in its allocated time slot in the first time frame (TF$_1$). Upon receiving coded symbols in TF$_1$, each user starts the decoding process to recover other user’s information symbols. We assume that $U_i$ has recovered $s_i^{(1)}$ information symbols from the other user in TF$_1$, where $i \in \{1, 2\}$ and $0 \leq s_i^{(1)} \leq k$. In the next time frame, TF$_2$, $U_i$ generates $N$ coded symbols using its $k$ information symbols as well as $s_i^{(1)}$ information symbols form the other user, and transmits these coded symbols to the other user and the destination. More generally, in each time frame, each user generates coded symbols from its own information symbols and those from the other user which have been recovered in previous time frames, and broadcast them. At a same time, each user tries to recover more information symbols from the other user by overhearing its transmission and performing LT decoding process in each time frame. Finally, when the destination successfully decodes both users’ information symbols, it sends an acknowledgment and users start broadcasting new messages. Fig. 3-b shows the PCC scheme for a 2-user CMAC.

It is worth noting that the PCC scheme does not require the other user’s message to be completely decoded at each user before starting the cooperation process. Cooperation starts as soon as the users are able to recover at least one information symbol from the other user. Also in PCC, the users use the same degree distribution in the whole transmission. Thus compared to the FCC scheme which uses separate degree distributions in each transmission phase, the PCC scheme has a lower encoding complexity since the users do not need to
change the degree distribution during their transmissions.

IV. DEGREE DISTRIBUTION OPTIMIZATION OF THE PCC SCHEME

Since in PCC, each user transmits $N$ coded symbols in each time frame and some of them may be erased by the channel, each user receives on average $N(1 - e)$ coded symbols from the other user. Also, due to symmetry and the assumption that the inter-user channel is reciprocal, the average number of recovered information symbols form the other user is the same at both users. Thus, we consider that each user can recover $s_i$ information symbols in TF$_i$. In the next time frame, each user encodes the partially decoded symbols of the other user, together with its own information symbols, to generate a new packet of length $N$, and transmits it to the destination.

More specifically, in TF$_{i+1}$, the received coded symbols at each user have been generated from $k+s_i$ information symbols using $\Omega(x)$ as the degree distribution. Since, $s_i$ information symbols of each user are already known at the other user in TF$_{i+1}$, all edges connected to the known symbols can be removed from the bipartite graph at users. To generate a coded symbol of degree $d+l$, $d+l$ information symbols are selected uniformly at random from $k+s_i$ information symbols. The probability that this coded symbol is connected to $l$ known information symbols and $d$ unknown information symbols is

\[
\Omega_d^{i+1} = \sum_{l=0}^{s_i} \Omega_{d+l} \frac{s_i}{s_i \choose l} \frac{k}{k + s_i} \frac{d}{d + l}, \quad d = 0, 1, \ldots, k.
\]  

(4)

Since each user tries to decode the other user message from the received coded symbols in TF$_{i+1}$ and the previous time frames, the number of coded symbols in TF$_{i+1}$ will be $N^{i+1} = (i+1)N(1-e)$. The probability that a coded symbol is of degree $d$ in TF$_{i+1}$ can be calculated as follows.

\[
\Delta_d^{i+1} = \frac{1}{i+1} \sum_{j=1}^{i+1} \Omega_d^{j}, \quad d = 1, 2, \ldots, k,
\]  

(5)

which arises from the fact that a specific coded symbol is received in TF$_j$ with probability $N(1 - e)/\Omega^{i+1}$ and it is of degree $d$ with probability $\Omega_d^{j}$. By inserting (4) into (5) we have

\[
\Delta_d^{i+1} = \frac{1}{i+1} \sum_{j=1}^{i+1} \sum_{l=0}^{s_i} \Omega_{d+l} \frac{s_i}{s_i \choose l} \frac{k}{k + s_i} \frac{d}{d + l}, \quad d = 0, 1, 2, \ldots, k.
\]  

(6)

As said before, if the degree distribution of coded symbols is $\Omega(x)$, the probability that an information symbol is not recovered after $l$ iterations is $p_l = e^{-\omega(1-p_i)}$, where $\omega(x) = \Omega'(x)/\Omega(1)$. In the PCC scheme, the degree distribution of coded symbols in TF$_i$ is $\Delta(i)(x)$ at each user. So, the probability that an information symbol is not recovered after $l$ iterations, denoted by $p_l$, can be calculated as follows

\[
p_l = e^{-\alpha(i)\delta(i)(1-p_i)},
\]  

(7)

where $\delta(i) = \Delta(i)(x)/\Delta(i)'(1)$ and $\alpha(i) = N(i)\Delta(i)'(1)/k$. As a result, $s_i$ can be calculated as $k(1-p_i)$ when $l$ and $k$ go to infinity. Fig. 4 shows the fraction of recovered information symbols in each TF for the case that $N = 100$, $k = 1000$ and $\Omega(x) = 0.05x+0.55x^2+0.25x^3+0.05x^6+0.1x^8$. Analytical results are also depicted in Fig. 4, which show an excellent agreement with the simulation results.

Since a fraction of each user’s message is decoded at the other user in each time frame, we can divide each user’s message into different parts, where the $i^{th}$ part of each user’s message is decoded in TF$_i$ at the other user. Let $P_i$ and $Q_i$ denote the $i^{th}$ part of $U_1$ and $U_2$, respectively. Due to symmetry, the length of $P_i$ is the same as the length of $Q_i$ and it equals to $s_i - s_{i-1}$. If $s_i$ information symbols of each user are already known at the destination, which means that parts 1, 2, ..., and $i$ of each user’s message are recovered at the destination, the destination needs to recover remaining parts of two users’ messages.

When the destination already knows $s_i$ information symbols of each user, it removes all edges connected to these symbols. As a result the degree distribution of coded symbols will be

\[
\Delta_d^{i} = \sum_{l=0}^{2s_i} \Omega_{d+l} \frac{2s_i}{s_i \choose l} \frac{2k}{d} \frac{d}{d + l}, \quad d = 0, 1, \ldots, k,
\]  

(8)

Eq. (8) can be easily obtained by using a similar approach as for (4). Similar to [9, 14], to ensure that the destination can decode the remaining information symbols, the following
TABLE I
Degree distributions for various values of $N$; $\mu$ is the average degree of a coded symbol, and $k = 10000$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>1000</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Omega_1$</td>
<td>0.0069</td>
<td>0.0069</td>
</tr>
<tr>
<td>$\Omega_2$</td>
<td>0.4898</td>
<td>0.4898</td>
</tr>
<tr>
<td>$\Omega_3$</td>
<td>0.1656</td>
<td>0.1691</td>
</tr>
<tr>
<td>$\Omega_4$</td>
<td>0.0883</td>
<td>0.0743</td>
</tr>
<tr>
<td>$\Omega_5$</td>
<td>0.0224</td>
<td></td>
</tr>
<tr>
<td>$\Omega_6$</td>
<td>0.1109</td>
<td>0.1050</td>
</tr>
<tr>
<td>$\Omega_7$</td>
<td>0.0666</td>
<td>0.0693</td>
</tr>
<tr>
<td>$\Omega_8$</td>
<td>0.0207</td>
<td>0.0187</td>
</tr>
<tr>
<td>$\Omega_9$</td>
<td>0.0451</td>
<td></td>
</tr>
<tr>
<td>$\mu$</td>
<td>5.95</td>
<td>5.95</td>
</tr>
</tbody>
</table>

condition needs to be satisfied.

$$\Delta^{(i)}(x) \geq -r_j \ln \left( 1 - x - c \sqrt{\frac{1 - x}{2(k - s_j)}} \right),$$

(9)

for $x \in [0, 1 - \delta]$ and some constant $r_i$, $c$ and $\delta$. Therefore, to find the optimum degree distribution, (9) needs to be satisfied for all $i$’s. Therefore, the optimization problem can be summarized as follows.

$$\text{maximize } \sum_{j=1}^{L} r_j$$

subject to

$$\sum_{d=1}^{k} \sum_{l=0}^{s_j} d \Omega_{d+l} \left( \frac{2s_j}{l} \right) \left( \frac{2k - s_j}{d} \right) x^{d-1}$$

$$\geq -r_j \ln \left( 1 - x - c \sqrt{\frac{1 - x}{2(k - s_j)}} \right), \quad x \in [0, 1 - \delta]$$

(2).

$$\sum_{d=1}^{k} \Omega_d = 1 ,$$

(3).

$$0 \leq \Omega_d \leq 1 , \quad d = 1, 2, ..., k$$

For given $s_j$’s, the objective function and all constraints are linear in terms of $\Omega(x)$ and $r_i$, so the optimization problem can be solved by means of linear programming. In order to find the optimum degree distribution using a linear programming algorithm, we take an initial value for $s_j$ as depicted in Fig. 4. Table I shows several optimized degree distributions obtained by this method.

V. SIMULATION RESULTS

We assume that the transmitter has no knowledge of channel state information of any channel, either inter-user channel or the user to the destination channel. Also we consider that $N = 1000$ and $k = 10000$ for our simulations. In our simulations, we compare the proposed PCC and FCC schemes with conventional no-cooperation and perfect-cooperation schemes. In the no-cooperation scheme, each user only generates coded symbols from its own information symbols and transmits them to the destination without any cooperation with the other user. In the perfect-cooperation scheme, we assume that each user knows perfectly the other user’s message before its transmission. So each user generates coded symbols from both users’ information symbols and sends them to the destination.

We first consider the symmetric case, where the erasure probability of the channel between the users and the destination are the same ($e_1 = e_2$). In the no-cooperation scheme, the destination requires at least $k(1 + \delta_k)$ coded symbols from each user to ensure that it can completely decode both users’ messages, where $\delta_k$ is the average overhead required for a successful decoding of $k$ information symbols at the destination. So, each user needs to send at least $k(1 + \delta_k)/(1 - e_1)$ coded symbols to the destination. As a result, in the no-cooperation scheme, users should send overall $2k(1 + \delta_k)/(1 - e_1)$ coded symbols to the destination.

On the other hand, in the perfect-cooperation scheme where each user already knows the other user’s message before transmission, the destination needs to receive at least $2k(1 + \delta_{2k})$ coded symbols from both users to be able to recover both users’ information symbols, where $\delta_{2k}$ is the average overhead required for a successful decoding of $2k$ information symbols at the destination. Therefore, overall $2k(1 + \delta_{2k})/(1 - e_1)$ coded symbols should be sent by the users. Since $\delta_k$ and $\delta_{2k}$ go to zero when $k$ goes to infinity [17], the total number of transmitted coded symbols in the no-cooperation and perfect-cooperation schemes will be the same when $k$ goes to infinity. This means that the average system throughput for both the no-cooperation and perfect-cooperation scheme is the same. Thus, in the symmetric case, the cooperation between the users cannot improve error performance at the destination when $k$ is a relatively large number.

Fig. 5 shows the average system throughput for the asymmetric case when the erasure probability of the channel between $U_1$ and $D$ is 0.3 and that from $U_1$ and $D$ is 1. It can be noted from the figure that the PCC scheme outperforms the FCC scheme. Note that, this case is similar to the relay channel scenario, because information symbols of $U_2$ are only sent by $U_1$.

The average system throughput is depicted in Fig. 6 for the case that $e_1 = 0.2$ and $e_2 = 0.8$. It is clear that in the FCC scheme, when the erasure probability of the inter-user channel is larger than that of between the users and the destination, each user’s message can be recovered at the destination earlier than at the other user. As a result, the FCC scheme performs similar to the no-cooperation scheme. However, since in the PCC scheme, users can cooperate even if they can decode a fraction of the other user’s message, the PCC scheme considerably outperforms the FCC and no-cooperation schemes. It also approaches the perfect-cooperation scheme when the erasure probability of the inter-user channel is low.

The advantages of the PCC scheme is clearer in Fig. 7, where the PCC scheme brings about 10% throughput gains compared to the FCC scheme at inter-user erasure probability of 0.6. In fact, the performance of the FCC scheme is more similar to the no-cooperation scheme in the case of high inter-
user erasure probability. However, the PCC scheme performs partial cooperation between the users and thus results in a higher average throughput. It is worth noting that when the erasure probability of the inter-user channel is very low, the performance of both the PCC and FCC scheme are very close to the perfect-cooperation scheme.

VI. CONCLUSION

A new coded cooperation scheme based on rateless codes was proposed. Two cooperation strategies, a partially coded cooperation (PCC) scheme and fully coded cooperation (FCC) scheme, were considered. The degree distributions for the proposed PCC scheme have been optimized using a linear programming algorithm. Simulation results show that the PCC scheme with an optimized degree distribution outperforms the FCC scheme in terms of the average system throughput.

REFERENCES


