Balanced Cooperative Linear Data Exchange with Physical Layer Network Coding

Nyan Lin
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa Prefecture
(+81)80 4254 3948, 923-1211
nyanlin@jaist.ac.jp

Brian M. Kurkoski
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa Prefecture
(+81)761-51-1266, 923-1292
kurkoski@jaist.ac.jp

Yuto Lim
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa Prefecture
(+81)761-51-1285, 923-1292
ylim@jaist.ac.jp

Yasuo Tan
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa Prefecture
(+81)761-51-1246, 923-1292
ytan@jaist.ac.jp

ABSTRACT

In this paper, we consider a group of nearby wireless end devices such as mobile phones cooperating to exchange their received packets from a common base station in order to fulfill the lost packets in other members. This problem can be solved by the linear coding in which each client transmits linear combinations of the received packets to minimize total number of transmissions. As the end devices have the limited energy resources, fairness should be a desirable property for each of them to support each other until all members in the group satisfy their needs. We introduce a cooperative algorithm that will maintain the fairness among nodes by distributing the number of transmissions they make. In this algorithm, each client changes the role of transmitter and receiver based on the information they keep in each round of data transmission. Moreover, we also incorporate physical layer network coding into this problem domain in order to further reduce the required transmission time slots and accomplish the data exchange process as quickly as possible.

CCS Concepts

•Human-centered computing → Ubiquitous and mobile devices; •Networks → Ad hoc networks; *Link-layer protocols;* **•Mathematics of computing → Coding theory;**

ICC '16, March 22-23, 2016, Cambridge, United Kingdom

DOI: http://dx.doi.org/10.1145/2896387.2896413

Keywords

Cooperative data exchange; Linear network coding; Physical layer network coding

1. INTRODUCTION

A group of nearby wireless devices downloads a large file, which is divided into n packets, from a common base station which may exist at a far distance. Some devices successfully receive only some subset of packets and they cannot receive some other packets because of the wireless link imperfection such as the noisy channel. Instead of requesting the lost packets to the base station from each client device, they can utilize the advantage of short-distance communication links among them by the built-in Bluetooth or Wi-Fi interfaces [3, 4, 7, 12]. They only need to collectively possess n packets within the group and cooperate to share the packets they have already received. With this approach, clients will possess a faster and reliable short-range communication and the base station will be able to serve other users after n transmissions of packets. Cooperation between neighboring devices can give benefits such as energy saving of a device by means of sharing the load among them and bandwidth saving and low delay services because the expensive long-distance cellular links are free for another users. Moreover, the overall capacity of the network will be increased by exploiting the broadcast nature of the wireless medium and the interaction (cooperation) between the neighboring nodes. Therefore, cooperative communication will become one of the major features for the emerging fourth and fifth generation wireless systems [6, 8].

With the application of linear coding, the benefits of cooperation can be further considered for this problem because many devices can simultaneously gain from one linearly coded packet (linear combination) transmission. We assume that mobile devices are within transmission range of each other and they can hear the broadcast transmission from each other successfully. In this linear approach, each packet is considered to be an element of a finite field $\mathbb{F}$. A client device creates linear combination of its received packets and choose one combination that will benefit the other peers and transmits the selected linear combined packet. As other peer nodes in the cooperative group may already have some independent packets, they will receive a new one after performing Gaussian elimination of the received packets. This work is similar to the index coding problem in [1], where a central base station performs the transmissions of linear combined packets to make other clients satisfy their requirements. The difference is that, in cooperative approach, most of the clients involve in the transmission process. In this category, many research works such as [2, 7, 9, 10] studied mainly the optimum number of transmissions to reduce the complexity, overhead and delay and all clients ultimately recover the lost packets. Most of the works formulate the problem into integer programming and prove that it is NP-hard. Some other works find the theoretical upper and lower bounds on the required number of transmissions and then propose some efficient algorithms such as polynomial time algorithm. They mainly focused on the minimum cost as a whole group without considering much on the individual.

In our work, we focus on the fairness among the clients based on the number of transmissions they make because energy saving is important not only for the whole data exchange but also for each individual participant. For example, in the problem we considered, we assume that clients collectively possess all packets from the base station and if a client with independent packets spends all its energy quickly before the data exchange process finishes, other clients will never satisfy their needs. By its principle, the client with the maximum number of packets possesses the best combination as the dependency between the received packets is low in earlier iteration. The best combination is defined as a combination that will increase the rank of the subspace of as many clients as possible. The authors in [7] formulated a lower and upper bound of the number of transmissions needed to satisfy the linear data exchange problem and showed that their algorithm performs closer to the lower bound by a numerical simulation. However, they did not consider fairness on the number of transmissions each client makes. They just choose the candidate client which possesses the maximum number of packets as a transmitter for next round. If there are more than one candidate client, their scheme choose the next transmitter randomly.

As a consequence, a node with the maximum number of packets in the cooperative group has to utilize its energy and computing resources more than the others do. This is the problem we want to focus on in our algorithm to share the fairness among as many client devices as possible. For example, a scenario where only one client device receives most of the packets transmitted from the base station. As an example, in Figure 1, client c_3 has received five packets out of six and is the maximum. Client c_2 has received the least number of packets. As an encoded packet is a combination of multiple packets, many other client devices that missed different packets can benefit from the same encoded packet. Therefore, choosing the client with the maximum number of packets as transmitter is usually useful.

In this example, the data exchange among four clients accomplished with three coded packets transmission: two transmissions from the maximum client, c_3 and one from the second maximum client, c_1. However, with this kind of selection of next transmitter, the problem appears in the later iteration of data exchange. As more clients gradually increase the number of received packets, the dependency between the packets becomes higher. In this situation, choosing the clients with the maximum packets will not always provide significant effect. Choosing another client, which has less participation in the data exchange process, in later iteration will keep fairness among the clients.

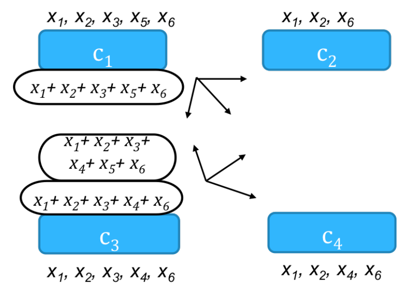


Figure 1: **Data exchange among four clients.**

The rest of the paper is organized as follows. We describe the proposed balanced coding scheme and the algorithm in Section 2. In Section 3, we discuss the trade off between the computational time and the best selection of the linear combination. Section 4 presents the introduction of physical-layer network coding (PNC) into the data exchange problem. Simulation and discussions of the results are presented in Section 5 and 6. Finally, the conclusion follows in Section 7.

2. BALANCED CODING SCHEME

In our proposed balanced coding scheme, we propose an algorithm which will choose the transmitter for next round of iteration based on the number of packets each client node possesses. Moreover, each node should also take into account the number of transmissions it made so far in order to maintain the fairness among the nodes. Our approach is to keep an information table like Table 1 in every client. At the start of data exchange, all clients should broadcast the index information of their packet receiving status. Then the client with the maximum number of packets calculates the linear combination of its received packets and starts to transmit a combination. Deciding on a best combination is described in next section. After each round of linear coded packet transmission, every node updates its information table and can easily decide who should take turn for transmission in next round based on this information. This is possible because all nodes are within the transmission range and every node can hear the transmission from each other. In [11], the authors used the unparking method by sending the access request message to turn a client device in master role to slave, which they call a node that is in transmitter or receiver role respectively, for the Piconet-based distributed cooperative approach, which is similar to the data exchange problem here. In our approach, we do not use control messages to change a client device from one role to another from time to time. We just need a simple MAC mechanism to start transmission.

2.1 Algorithm for Balanced Coding Scheme

At each iteration of the algorithm, one of the clients broadcasts a linear combination of the packets in the set $X =$

Table 1: **Example information table : received packets and frequency of transmissions**

Clients	Packets						Transmissions
	P_1	P_2	P_3	P_4	P_5	P_6	
c_1	1	1	1	0	1	1	1
c_2	1	1	0	0	0	1	0
c_3	1	1	1	1	0	1	2
c_4	1	1	0	1	0	1	0

$\{x_1, \cdots, x_n\}$ of n packets. For a coded packet x we denote by $C_x \in F^n$ the corresponding vector of linear coefficient, i.e., $x = C_x.(x_1, \cdots, x_n)^T$. We also denote by Y_i the subspace spanned by vector corresponding to the linear combinations available at client c_i. In the beginning of the algorithm, Y_i is equal to the subspace spanned by vectors that correspond to the packets in X_i, i.e., $Y_i = \langle\{C_x|x \in X_i\}\rangle$. The goal of the algorithm is to simultaneously increase the dimension of the subspaces $Y_i, i = 1, \cdots, k$, for as many clients as possible. At each iteration, the algorithm identifies a client $c_i \in C$ whose subspace Y_i is of maximum dimension. Then, client c_i selects a vector $b \in Y_i$ in a way that will increase the dimension of Y_j for each client $c_j \neq c_i$, and transmits the corresponding packet $b.(x_1, \cdots, x_n)^T$. At some iteration, the subspaces associated with a number of clients may become identical. We merge this group of clients into a single client with the same subspace.

Algorithm 1 Algorithm for balanced coding scheme

```
for i = 1 to k do
  Y_i = ⟨{C_x | x ∈ X_i}⟩
end for
while there is a client i with dim Y_i < n do
  while ∃c_i, c_j ∈ C i ≠ j, such that Y_i = Y_j do
    C = C \ {c_i}
  end while
  Find a client c_i with a subspace Y_i of maximum dimension
  if there is only one c_i then
    Select a vector b_i ∈ Y_i such that b_i ∉ Y_j for each i ≠ j
    Let client c_i broadcast packet x = b.(x_1, ··· , x_n)^T .
    Store c_i as transmitter in the table
  else
    Find a client c_j with a subspace Y_j of maximum dimension
    Select the client that has less previous transmissions
    if c_i is chosen then
      Select a vector b_i ∈ Y_i such that b_i ∉ Y_j for each i ≠ j
    else
      Select a vector b_j ∈ Y_j such that b_j ∉ Y_i for each j ≠ i
    end if
  end if
  Let client c_i or c_j broadcast packet x = b.(x_1, ··· , x_n)^T .
  for l = 1 to k do
    Y_i ← Y_i + ⟨{b}⟩
  end for
end while
```

Algorithm 2 Algorithm for selecting best combination

```
if (number of other clients with rank = max_rank) ≥ (number of other clients - 1) then
  Set control variable = 1
  Calculate best combination
else if (number of other clients with rank ≥ max_rank − 1) ≥ half of number of other clients then
  Set control variable = 2
  Calculate best combination
else
  Set control variable = number of other clients
  Calculate best combination
end if
```

3. SELECTION OF BEST COMBINATION

In the linear data exchange scheme, each client has to calculate the receiving information to choose a suitable combination for the transmission. Choosing a best combination that can increase the dimension of the subspace of as many clients as possible is one of the main tasks. The main challenge for this job is the computational time required to find a best combination. To overcome this challenge, we introduce a variable in our algorithm, which will control the number of clients that should increase their ranks. We derive the value of this variable by studying the relationship between the ranks of each client after running the simulation many times. The chosen combination might not be an optimum one for some cases. The client with the maximum number of received packets is usually chosen for this task. The condition and the value of control variable for picking the best combination is described in Algorithm 2.

4. PHYSICAL LAYER NETWORK CODING (PNC)

Let us refer to the explanation of PNC in [5]. The authors described that the basic idea of PNC is to exploit the mixing of signals that occurs naturally when electromagnetic (EM) waves are superimposed on one another. In particular, at a receiver, the simultaneous transmissions by several transmitters result in the reception of a weighted sum of the signals. This weighted sum is a form of network coding operation by itself.

In the second part of our work, we try to apply the concept of PNC to data exchange algorithm. We consider transmitting the addition of two linear combinations from two clients. First, our algorithm chooses a linear combination from the client with maximum number of packets. Then it finds another linear combination from the second maximum-rank client. These two combinations are transmitted simultaneously and allowed to add in the air naturally when electromagnetic (EM) waves are superimposed on one another. We assume a relay node in a PNC system to deal with the mapping of the mixed signal to the desired network-coded signal $S_R = S_1 \bigoplus S_2$. Relay broadcasts S_R to the other clients in the second time slot. In the algorithm, we add two combinations first and then transmit the addition. If there is more than one client with the maximum number of packets, the two transmitters are carefully designated based on the information of their previous transmissions as in the balanced coding scheme.

One of the advantages of this transmission scheme is that the two transmitters can themselves benefit from the addition of coded packets. This is different from the previous approach in which the transmitter node cannot increase its rank by its transmitted coded packet in the present round. Another advantage is that two clients can transmit in the same time slots and the number of required time slots is reduced by the PNC. The relay is simply an intermediate node that receives the combinations transmitted from the clients and transmits the coded packet back to the clients. However, it also has its own drawback. Sometimes, the addition results in no increment of the rank of other clients. This is because the selected combinations from two clients have some linearly dependency. For that situation, transmitting only one combination will guarantee the benefit.

Algorithm 3 Algorithm for PNC coding scheme

for $i = 1$ to k **do**
  $Y_i = \langle\{C_x | x \in X_i\}\rangle$
end for
while there is a client i with dim $Y_i < n$ **do**
  while $\exists c_i, c_j \in C i \neq j$, such that $Y_i = Y_j$ **do**
    $C = C \setminus \{c_i\}$
  end while
  Find a client c_i with a subspace Y_i of maximum dimension
  if there is only one c_i **then**
    Find a client c_j with a subspace Y_j of smaller maximum dimension than Y_i
    Select a vector $b_i \in Y_i$ such that $b_i \notin Y_j$ for each $i \neq j$
    Select a vector $b_j \in Y_j$ such that $b_j \notin Y_i$ for each $j \neq i$
    Add two vectors, $b = b_i + b_j$
    Let client c_i and c_j broadcast packet $x = b.(x_1, \cdots, x_n)^T$.
  else
    Find a client c_i with a subspace Y_i of maximum dimentsion
    if dimension dim Y_i = number of packets **then**
      Select a vector $b \in Y_i$ such that $b \notin Y_j$ for each $i \neq j$
    else
      Select a vector $b_i \in Y_i$ such that $b_i \notin Y_j$ for each $i \neq j$
      Select a vector $b_j \in Y_j$ such that $b_j \notin Y_i$ for each $j \neq i$
      Add two vectors, $b = b_i + b_j$
    end if
    Let client c_i and c_j broadcast packet $x = b.(x_1, \cdots, x_n)^T$.
  end if
  for $l = 1$ to k **do**
    $Y_i \leftarrow Y_i + \langle\{b\}\rangle$
  end for
end while

5. SIMULATION

We created a computer simulation in MATLAB to study the performance of our algorithm and transmission in PNC scheme. We used the finite elements from $GF(2)$ to index each received and lost packet of each client for the less complexity. A comparison of transmission in balanced scheme compared to the transmission without balanced scheme (i.e., nominating the next transmitter randomly), is studied by varying the number of client nodes involved in the data exchange. We also studied the effect of initial packet receiving probability P_{init} on the system performance by using different probability values in each simulation with total 10 packets. By theory, the number of time slots required by the PNC scheme is less than that does not include the PNC as two users transmit in the same time slot. From the standpoint of users, they still need to involve in the transmission and have to utilize their energy resources. In this simulation, we studied how PNC can help the data exchange finish earlier than the scheme without PNC while reducing the total number of time slots required.

Table 2: **Simulation parameters**

Parameter	**Value**
Hardware specification	IEEE 802.11a OFDM
Simulation environment	MATLAB 2014b
P_{init}	0.7, 0.6, 0.5
Number of packets	10
Number of clients	4, 6, 8
Antenna type	Omnidirectional Antenna
Number of experiments	100

6. DISCUSSION OF RESULTS

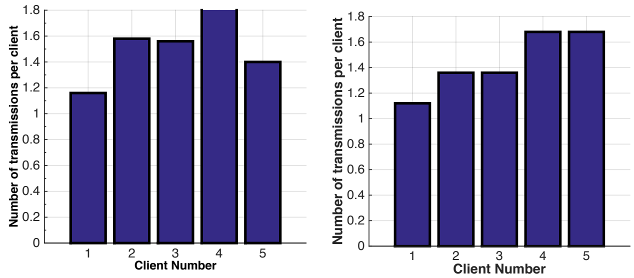
This section presents the performance of the proposed scheme based on the results obtained from the simulation. We will discuss on the number of transmissions from each client in the balanced coding scheme, the number of clients whose ranks increase in each round of iteration in PNC scheme and the total number of transmissions and time slots applied for the entire data exchange process. The results for each scenario are the values averaged over the 100 experiments for 5 clients downloading 10 packets from a base station with the initial receiving probability of 0.6.

6.1 Transmissions from Each Client

Figure 2 shows that our proposed scheme can significantly maintain the fairness on the transmission among the clients participating in cooperative data exchange. In Figure 2 (a), client c_4 is the only one with the maximum number of transmissions. This condition disappears in the proposed balanced coding scheme of Figure 2 (b). The algorithm tries to balance the transmissions of the clients who possess similar number of packets. This can be easily proved that in the transmissions of clients c_2 and c_3; and clients c_4 and c_5. But the client c_1 shows no significant change in the balanced scheme. This is because client c_1 is the client with the minimum number of packets received initially and it gives least participation in the data exchange process. Client c_1 is receiver in most of the iterations. This figure shows that the total transmissions in proposed balanced scheme can even decrease a few.

6.2 Number of Increased Clients per Round

In the transmission with the PNC scheme, most of the



(a) (b)

Figure 2: **Number of transmissions produced (a) Without balancing scheme (b) With balanced scheme.**

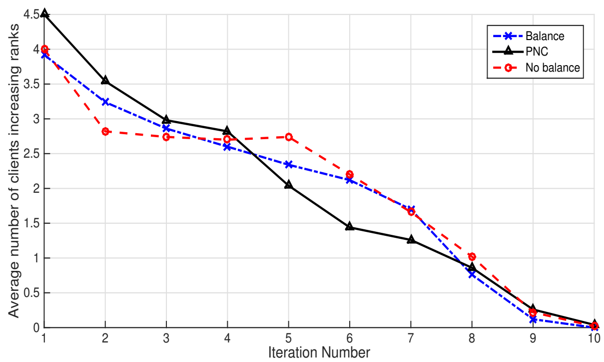


Figure 3: **Number of increased clients with PNC in earlier iterations**

clients devices increase their ranks after receiving the PNC-coded packet in the earlier round of iteration as depicted in Figure 3. This helps the data exchange process to complete quickly with fewer transmission time slots than the traditional transmission scheme while also ensuring that all the devices in the cooperative group recover their lost packets. Therefore, bandwidth and energy resources are saved. The number of transmissions from each clients is still high as described in Figure 4 as they involve in simultaneous transmission, but at the same time, it also reduces the total number of transmissions because the PNC coded packet can help increase the rank of the subspace of both the transmitter itself and the receivers. This is different from the original transmission scheme of linear combinations, where only the receivers can benefit from the coded packet in each round.

6.3 Transmissions for One Data Exchange Process

The total number of transmissions required by the proposed schemes for the completion of one data exchange process is within the upper and lower bound of the reference paper [7]. The scheme with PNC shows fewer number of transmissions than the one with the random coding scheme and the balanced coding scheme as depicted in Figure 5.

7. CONCLUSIONS

In this paper, we proposed a scheme for the balanced lin-

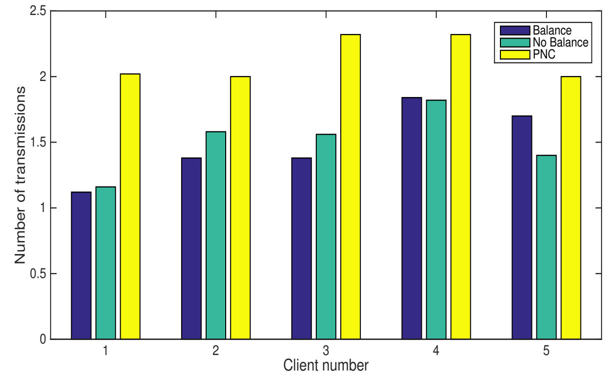


Figure 4: **Number of transmissions from each client with PNC.**

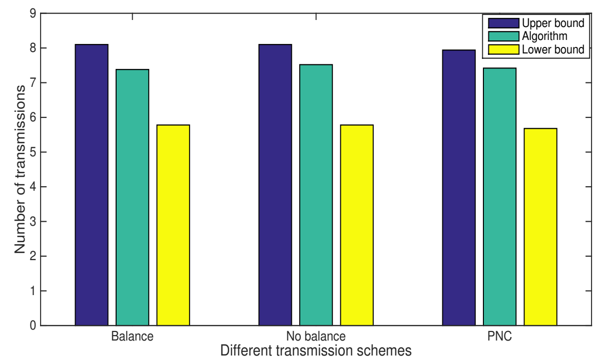


Figure 5: **Comparison of total number of transmissions for 3 schemes**

ear data exchange problem to maintain the fairness among the client devices to ensure that a certain client does not leave the group and lose the independent packets it stores. We used an information table that keeps the number of transmissions each client makes in each round of data exchange process to be taken into account in deciding the next transmitter client. By this approach, the total number of transmissions decreases while distributing the work load among the clients. Moreover, by allowing two clients to simultaneously transmit the linear combinations of their received packets and allowing them to add in the air, our simulation results show that physical layer network coding helps other clients receive the missed messages mostly in the early iteration and leads to the quick completion of data exchange process while also keeping the number of transmission time slots required less than the traditional scheme.

8. REFERENCES

[1] M. Chaudhry and A. Sprintson. Efficient algorithms for index coding. In *INFOCOM Workshops 2008, IEEE*, pages 1–4, April 2008.

[2] T. Courtade and R. Wesel. Coded cooperative data exchange in multihop networks. *Information Theory, IEEE Transactions on*, 60(2):1136–1158, Feb 2014.

[3] A. Heidarzadeh and A. Sprintson. Cooperative data exchange with unreliable clients. *CoRR*, abs/1508.03871, 2015.

[4] Y. Keshtkarjahromi, H. Seferoglu, R. Ansari, and A. A. Khokhar. Network coding for cooperative mobile devices with multiple interfaces. *CoRR*, abs/1503.02266, 2015.

[5] S. C. Liew, S. Zhang, and L. Lu. Physical-layer network coding: Tutorial, survey, and beyond. *CoRR*, abs/1105.4261, 2011.

[6] S. Mohammad. Opportunistic cooperative communication using buffer-aided relays. *Qatar Foundation Annual Research Forum Proceedings*, page ICTP 025, November 2013.

[7] S. Y. E. Rouayheb, A. Sprintson, and P. Sadeghi. On coding for cooperative data exchange. *CoRR*, abs/1002.1465, 2010.

[8] ScienceDaily. Cooperative driving will become common: Data exchange between vehicles and road network. Science news from research organizations, Technical Research Centre of Finland (VTT), www.sciencedaily.com/releases/2015/07/150702073740.htm, July 2015.

[9] Y. Sui, X. Wang, J. Wang, L. Wang, and S. Hou. Deadline-aware cooperative data exchange with network coding. *Computer Networks*, 97:88 – 97, 2016.

[10] S. Tajbakhsh and P. Sadeghi. Coded cooperative data exchange for multiple unicasts. In *Information Theory Workshop (ITW), 2012 IEEE*, pages 587–591, Sept 2012.

[11] Q. Zhang, F. Fitzek, and M. Katz. Cooperative power saving strategies for ip-services supported over dvb-h networks. In *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, pages 4107–4111, March 2007.

[12] Q. Zhang, J. Heide, M. V. Pedersen, and K. Rikkinen. *Network Coding: Fundamentals and Applications*, chapter 5 Network Coding and User Cooperation for Streaming and Download Services in LTE Networks, pages 115–140. Academic Press, December 2012.