Title	マルチソースマルチリレーネットワークにおける計算と転送のオーバーラップチャンクコード
Author(s)	Ngeth, Rithea
Citation	
Issue Date	2019-03
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/15792
Rights	
Description	Supervisor:リム　勇仁，情報科学研究科，博士

Doctoral Dissertation

Overlapped Chunked Codes over Compute-and-Forward in Multi-Source Multi-Relay Networks

Rithea NGETH

Supervisor: Associate Professor Yuto LIM

School of Information Science
Japan Advanced Institute of Science and Technology

March, 2019

Abstract

With the growing network density and the limited channel bandwidth, the interference between nodes would degrade the performance in throughput, energy consumption and latency which is expected suitable for the application based on the information sharing between nodes. To solve this problem, this dissertation considers the solution using physical-layer network coding (PNC) which takes use of interference instead of dealing with interference. On the other hand, straightforward network coding (SNC) including intra-flow network coding (IANC) and inter-flow network coding (IENC) has shown its role in improving the network performance. PNC and SNC are employed in the different layer of the protocol stack, i.e., PNC is applied at the physical-layer, and SNC is applied at layer-2 or upper. IANC mainly is employed to improve network performance by reducing the protocol overhead via the feature of overlapped chunked codes (OCC). PNC can improve the network through significantly via IENC approach. Up to the present, there are few studies on the application of IANC with PNC where a big message is divided into blocks, and they are grouped into chunks. This dissertation aims to study the performance of the application of OCC with PNC.

In order to achieve this purpose, this dissertation considers a scenario of multi-source multi-relay network where a PNC approach, compute-and-forward (CF) based on nested lattice code(NLC), is employed for the simultaneous transmissions from the sources to the relays. A popular network coding technique, random linear network coding (RLNC), is employed within each chunk before encoding with NLC at each source. This dissertation provides a design of overlapped chunked code (OCC) for this scenario, called OCC/CF, and an OCC-based retransmission scheme called RLNC/CF.

OCC is an IANC approach where the feedback about the reception state information at the destination can be avoided, hence the transmission scheme employing OCC might have better performance than a feedback-based transmission scheme when the protocol overhead such as the transmission time of feedback and the loss of feedback is considered. The key to design OCC is the decodability condition which is provided in this dissertation,

and an OCC with contiguously overlapping fashion is applied for the design for the performance observation and evaluation. The decoding scheme with this OCC is provided as well. In addition, the estimations of the decodability, the performance in term of channel efficiency (corresponding to network throughput) and the decoding complexity are given in order to provide the option to obtain the different desired term of performance. The design is done by using the empirical probability distributions where a new term, the probability distribution of the participation factor of a source to the forwarded data for a chunk, is introduced and plays an important role in the design of OCC/CF.

On the other hand, for the transmissions in lossy channel, in order to have data blocks received on time, the retransmission is made if these packets could not be recovered by the previous transmissions. An efficient retransmission is needed for data transmission in multi-source multi-relay network while CF is employed, because the destination could not ensure all blocks could be recovered after a retransmission round. The other advantage of contiguously OCC is that the unrecoverable transmitted blocks of a chunk could become recoverable by the transmission of the blocks of the next chunk. By taking use this feature, in RLNC/CF scheme, the determination of the number of overlapped blocks for each source by using the knowledge of the reception state of the previous chunk from the aid of feedback, empirical probability distributions and for an expected reception state of the transmission of the next chunk to obtain the highest expected decodability is given. In addition, different selections of expected reception state is studied to figure how to obtain the desired performance.

From the numerical results obtained by doing simulation in MATLAB, the improvement gained from employing OCC/CF depends on the level of the protocol overhead. The channel efficiency gained from employing OCC/CF in multi-source multi-relay network can approach the channel efficiency employing the OCC with the same overlapping fashion, contiguously overlapping fashion, in single flow transmission when the probability distributions of the participation factor of all source are dense near the chunk size. Although the applied OCC causes high design overhead for OCC/CF, but this design might provide a reference for the future design of OCC for this scenario with better performance. On the other hand, RLNC/CF scheme can provide some improvement in channel efficiency over a cooperative CF scheme, showing its advantage in reducing the protocol

overhead and improving the efficiency of retransmission. On the other hand, selection of expected reception state does a trade-off between channel efficiency and reception delay.
Keywords: Network coding, Overlapped chunked code, Retransmission, Compute-and-Forward, Multi-source multi-relay network

Acknowledgments

This work would have not been possible without the support of many people, whom I would like to thank here.

First of all, I am ineffably indebted to my supervisor, Professor Yuto Lim for his invaluable guidance, encouragement and kind help. I owe my profound gratitude to my second supervisor, Professor Yasuo Tan for his inspiring instructions and comments throughout this research endeavor. Their brilliant ideas and suggestions helped me to improve this dissertation.

I am extremely thankful to my adviser for the Minor Research Project, Associate Professor Brian Michael Kurkoski for his patient guidance, warm encouragement and useful suggestions to accomplish the Minor Research project and this dissertation.

I also would like to convey my sincere gratitude to the members of examination committee: Associate Professor Hidekazu Murata from Kyoto University and Associate Professor Razvan Florin Beuran from JAIST, for their patience in reading this dissertation and contributing to its content to improve the quality of the dissertation.

In addition, I am thankful to the other current members and former members in Tan and Lim laboratories for their help and support and for the valuable discussions on various topics.

On the other hand, I am extremely thankful to Government of Japan for the financial support by the MEXT scholarship program offered to my country.

Finally, I would like to appreciate my family, relatives and friends from Cambodia for their support.

Contents

List of Figures

List of Tables

Acronyms

ACK	Acknowledgement
AF	Amplify-and-Forward
AWGN	Additive White Gaussian Noise
b.s	back-substitution
BACK	Block Acknowledgement
BATs	Batched sparse
BER	Block Error Rate
BSR	Block Success Rate
ch.cs	chain of chunks
co.cs	combination of chunks
CF	Compute-and-Forward
CF/PO	Compute-and-Forward with Protocol Overhead
CR	Cognitive Radio
CSI	Channel State Information
CW	Codeword
CWC	Codeword Combination
d.s	decodability state
f.b.s.s	full back-substituted state
FC/CF	Fountain Code over Compute-and-Forward
FDI	Forwarding Decision Information
FER	Frame Error Rate
h.b.s.s	half back-substituted state
IANC	Intra-flow Network Coding
IENC	Inter-flow Network Coding
l.b.s	left back-substitution
l.b.s.s	left back-substitution state

LNC	Linear Network Coding
LT	Luby Transform
MMSE	Minimum Mean Square Error
n.b.s.s	not back-substituted state
N/A	Not Applicable
NACK	Negative Acknowledgement
NC	Network Coding
NLC	Nested Lattice Code
NOMA	Non-Orthogonal Multiple Access
OCC	Overlapped Chunked Code
OCC/CF	Overlapped Chunked Code over Compute-and-Forward
PNC	Physical-layer Network Coding
PU	Primary User
q.b.s.s	quasi-half back-substituted state
r.b.s	right back-substitution
r.b.s.s	right back-substitution state
RLNC	Random Linear Network Coding
RLNC/CF	Random Linear Network Coding over Compute-and-Forward
RLNC/OC	Random Linear Network Coding via Orthogonal Channel
RSI	Reception State Information
RRI	Retransmission Request Information
SACK	Selective Acknowledgement
SIC	Successive Interference Cancellation
SINR	Signal-to-Interference-plus-Noise Ratio
SNC	Straightforward Network Coding
SNR	Signal-to-Noise Ratio
SUs	Secondary Users
TWRC	Two-Way Relay Channels
XOR	Exclusive OR

List of Symbols

q	size of finite field (q is a prime number in this work)
$\mathbb{F}_q$	the finite field with size q
$\mathbf{b}_d$	d-th information block or input block
D	number of input blocks for applying RLNC
d	index of a input block, $d \in \{1, 2, \cdots, D\}$
n	number symbols per block, or the number of dimensions of the lattice
$\mathbf{w}$	a coded block from applying RLNC
$\mathbf{w}_d$	d-th coded block from applying RLNC
$\bigoplus$	summation operation over $\mathbb{F}_q$
$\cdot$	multiplication operation
$(\ ) \bmod q$	operation mapping a value inside ( ) into finite field
^{T}	transposition operation of a vector or a matrix
$^{-1}$	inverse operation of a matrix
$\boldsymbol{\chi}$	$= [\chi_1, \chi_2, \cdots, \chi_D]^T$, the coding coefficient vector of $\mathbf{w}$
$\boldsymbol{\chi}_d$	the coding coefficient vector of $\mathbf{w}_d$
p	the message success rate of the link from a sender to a receiver
D_t	the number of time slots needed to transmit a chunk
Λ	a lattice
$\mathbf{x}$	a lattice point
$\mathbf{G}$	the generator matrix of a lattice
$\mathbf{b}$	the information block inside a lattice point with maximum number symbol n
$\mathbb{R}$	field of real values
$\mathbb{Z}$	field of integer values
Λ_ξ	a scaled lattice

ξ	the scaling ratio of the scaled lattice Λ_ξ
$\mathbf{G}_\xi$	the generator matrix of the scaled lattice Λ_ξ
Λ_c	coding lattice
Λ_s	shaping lattice
$\mathbf{G}_c$	the generator matrix of the coding lattice Λ_c
$\mathcal{V}_s$	the fundamental Voronoi region of the shaping lattice Λ_s
$[\,]\bmod\mathcal{V}_s$	operation mapping a lattice point inside $[\,]$ into inside $\mathcal{V}_s$
Λ_c/Λ_s	a nested lattice code formed by Λ_c and Λ_s
R	the coding rate of a nested lattice code
n'	number of symbol inside the information block $\mathbf{b}$
$P_{\max}$	an assigned maximum transmit power of a NLC codeword per dimension
K	total number of senders or sources
k	index of source, $k \in \{1, 2, \cdots, K\}$
$\mathbf{x}_k$	NLC codeword sent from sender or source k
$\mathbf{z}$	AWGN with zero mean at a receiver
$\sigma_{\mathbf{z}}^2$	variance of AWGN $\mathbf{z}$
$\mathbf{y}$	received signal (accumulative codeword) at a receiver
h_k	a real channel coefficient or channel gain of the link from sender k to the receiver
ψ_k	an instantaneous received signal-to-noise ratio of the link from sender k to the receiver
P	the mean value of transmit power of a codeword per dimension
$\mathbf{v}$	a desired linear combination of K codewords, computed at a receiver
$\mathbf{a}$	a desired linear combination integer coefficient vector corresponding to $\mathbf{v}$ (or to $\mathbf{h}$)
a_k	k-th element of $\mathbf{a}$
α	a gain to do multiplication with the received signal to obtain $\mathbf{v}$
$\mathbf{h}$	a channel coefficient vector of links from K senders to a receiver
$\mathcal{R}(\mathbf{h}, \mathbf{a})$	the computation rate region for the pair $(\mathbf{h}, \mathbf{a})$

α_{MMSE}	the gain α which provides the minimum mean square error
R_k	the message rate of source k while employing NLC
$\mathbf{u}$	the codeword combination obtained by mapping $\mathbf{v}$ into $\mathcal{V}_s$
$\boldsymbol{\beta}$	the combination coefficient vector corresponding to $\mathbf{u}$
L	total number of the relays
L'	number of stages of relays
l	index of a relay, $l \in \{1, 2, \cdots, L\}$
M	total number of the destinations
m	index of a destination, $m \in \{1, 2, \cdots, M\}$
$h_{\text{SR}}^{(k,l)}$	a channel coefficient of the link from source k to relay l
$h_{\text{RD}}^{(l,m)}$	a channel coefficient of the link from relay l to destination m
$h_{\text{SD}}^{(k,m)}$	a channel coefficient of the link from source k to relay m
$\text{SNR}_{\text{SR}}^{(k,l)}$	average SNR of the link from source k to relay l
$\text{SNR}_{\text{RD}}^{(l,m)}$	average SNR of the link from relay l to destination m
$\text{SNR}_{\text{SD}}^{(k,m)}$	average SNR of the link from source k to destination m
i	index of chunk
$d_k^{(i)}$	the number of input blocks inside chunk i for source k
d_{MAX}	the maximum number of input blocks of a chunk among all sources and all chunks
$\mathbf{B}_k^{(i)}$	$= \left[\mathbf{b}_{k1}^{(i)}, \mathbf{b}_{k2}^{(i)}, \cdots, \mathbf{b}_{kd_k^{(i)}}^{(i)}\right]$, blocks of chunk i for source k
$D_k^{(i)}$	the number of coded blocks from RLNC, generated for chunk i at source k
d_t	index of time slot of D_t time slots for transmitting a chunk
$\omega_{kd_t}^{(i)}$	channel access allocation state for source k at time slot d_t while transmitting chunk i
$\Omega_k^{(i)}$	channel access allocation state for source k while transmitting chunk i
$D^{(i)}$	total number of input blocks of all sources for chunk i
$\mathbf{w}_{kd_t}^{(i)}$	the coded block of source k to be transmitted at time slot d_t for chunk i

$\boldsymbol{\chi}_{kd_t}^{(i)}$	$= \left[\chi_{kd_t1}^{(i)}, \chi_{kd_t2}^{(i)}, \cdots, \chi_{kd_td_k^{(i)}}^{(i)}\right]^T$, the coding coefficient vector of the coded block $\mathbf{w}_{kd_t}^{(i)}$
$\mathbf{x}_{kd_t}^{(i)}$	the NLC codeword generated from $\mathbf{w}_{kd_t}^{(i)}$
$\mathbf{X}_k^{(i)}$	$= \left[\mathbf{x}_{k1}^{(i)}, \mathbf{x}_{k2}^{(i)}, \cdots, \mathbf{x}_{kD_k^{(i)}}^{(i)}\right]$, generated NLC codewords of source k for chunk i
$r_{ld_t}^{(i)}$	the number of codeword combinations satisfying Condition (2.13) for relay l at time slot d_t while transmitting chunk i
ι	index of a codeword combination satisfying Condition (2.13) for relay l at time slot d_t while transmitting chunk i for $\iota \in \{1, 2, \cdots, r_{ld_t}^{(i)}\}$
$\mathbf{u}_{ld_t\iota}^{(i)}$ (also for $\mathbf{u}_{ld_t}^{(i)}$)	ι-th codeword combination computed at relay l from codewords transmitted at time slot d_t while transmitting chunk i. $\mathbf{u}_{ld_t}^{(i)}$ is for $\iota = 1$
$\boldsymbol{\beta}_{ld_t\iota}^{(i)}$ (also for $\boldsymbol{\beta}_{ld_t}^{(i)}$)	the combination coefficient vector corresponding to $\mathbf{u}_{ld_t\iota}^{(i)}$. $\boldsymbol{\beta}_{ld_t}^{(i)}$ is for $\iota = 1$
$\beta_{kld_t\iota}^{(i)}$ (also for $\beta_{kld_t}^{(i)}$)	the k-th element of $\boldsymbol{\beta}_{ld_t\iota}^{(i)}$. $\beta_{kld_t}^{(i)}$ is for $\iota = 1$
$\mathbf{c}_{ld_t\iota}^{(i)}$ (also for $\mathbf{c}_{ld_t}^{(i)}$)	the combined coding coefficient vector of $\mathbf{u}_{ld_t\iota}^{(i)}$. $\mathbf{c}_{ld_t}^{(i)}$ is for $\iota = 1$
D'_t	the number of re-encoded codeword combinations needed to be forwarded
$\mathbf{u}_{ld_t}^{\prime(i)}$	d_t-th re-encoded codeword combinations generated at relay l for chunk i
$\boldsymbol{\chi}_{ld_t}^{(i)}$	re-encoding coefficient vector of $\mathbf{u}_{ld_t}^{\prime(i)}$
$\mathbf{c}_{ld_t}^{\prime(i)}$	the combined coding coefficient vector of $\mathbf{u}_{ld_t}^{\prime(i)}$
$r_m^{(i)}$ (also for r_m, $r^{(i)}$, r)	the total number of linearly independent codeword combinations filtered by Condition (2.13) at destination m for chunk i. r_m is for any chunk, and $r^{(i)}$ is for $m = 1$, and r is for $m = 1$ and any chunk
$\mathbf{C}_m^{(i)}$ (also for $\mathbf{C}^{(i)}$)	the matrix formed from $r_m^{(i)}$ linearly independent codeword combinations filtered by Condition (2.13) at destination m for chunk i. $\mathbf{C}^{(i)}$ is for $m = 1$
$\rho_m(r_m)$ (also for $\rho(r)$)	the empirical probability distribution of $r_m^{(i)}$. $\rho(r)$ is for $m = 1$

$D_{\max}$	maximum of $D^{(i)}$ of all chunks
$\theta_{km}^{(i)}$ (also for $\theta_k^{(i)}$)	the participation factor of source k to the codeword combinations (to $\mathbf{C}_m^{(i)}$) received at destination m for chunk i. θ_{km} is for any chunk, and $\theta_k^{(i)}$ is for $m = 1$
$\lambda_{km}(\theta_{km})$ (also for $\lambda_k(\theta_k)$)	the empirical probability distribution of the participation factor $\theta_{km}^{(i)}$. $\lambda_k(\theta_k)$ is for $m = 1$
$d_{k\max}$	maximum of $d_k^{(i)}$ of all chunks
$\nu^{(i)}$ (also for ν)	number of blocks in chunk i, which also belong to the other chunks. ν is for any chunk
p_{km} (also for p_k)	the probability that a chunk is decodable while employing an OCC in single flow transmission, which is designed by using $\theta_{km}^{(i)}$. p_k is for $m = 1$
p_d	the probability that a chunk is decodable while employing an OCC in single flow transmission, which is designed by using $\rho_m(r_m)$, by using $\rho(r)$ for $m = 1$
p_{dec}	the probability that a chunk is decodable while an OCC is employed at each source in multi-source multi-relay networks (multiple flows transmission), used in considering some special cases
p_{deff}	the probability that a chunk is decodable while an OCC is employed at each source in multi-source multi-relay networks (multiple flows transmission), used in considering for general cases
η	the channel efficiency corresponding to p_d
η_{eff}	the effective channel efficiency corresponding to p_{deff}
$\bar{r}_m$ (also for $\bar{r}$)	the mean value of $r_m^{(i)}$ of all chunks, and $\bar{r}$ is for $m = 1$
$\bar{\eta}_m$ (also for $\bar{\eta}$)	the channel capacity defined by the ratio of $\bar{r}_m$ to D_t, and $\bar{\eta}$ is for $m = 1$
$\mu_k^{(i)}$ (also for μ_k)	the number of innovative blocks of each chunk comparing to the previous chunks, i.e., chunk $i - 1$, while employing contiguously OCC at source k. μ_k is for when it is constant for all chunks

$\gamma_k^{(i)}$	the number of blocks taken from the previous chunk, i.e., chunk $i-1$, while employing contiguously OCC at source k
μ	$\sum_{k=1}^{K} \mu_k$
$\gamma^{(i)}$ (also for γ)	$\sum_{k=1}^{K} \gamma_k^{(i)}$. $\gamma^{(1)} = 0$, and γ refers to the case $i > 1$
N_k	the number of original blocks at source k
N	$\sum_{k=1}^{K} N_k$
t	index of chunk
ϕ	length of chain of chunks
$\phi_{\max}$	maximum of ϕ
$\bar{\phi}$	the mean value of ϕ
ϕ'	the length of chain of chunks inside a chain of chunks with length ϕ, i.e., $\phi' \in \{1, 2, \cdots, \phi\}$
d.s(t)	the decodability state of chunk t
d.s(ch.cs)	the decodability state of the considered chain of chunks
l.b.s.s(t)	left back-substitution state of chunk t
r.b.s.s(t)	right back-substitution state of chunk t
$\mathbf{C}_c$	the combined coding coefficient matrix of chain of chunks with length ϕ
r_{ch}	the total number of original blocks inside a chain of chunks
r_{th}	the minimum number of linearly independent coded blocks for a chain of chunks that can be decoded. If the number of linearly independent coded blocks is lower than r_{th}, then the chain of chunks is considered as undecodable directly
$r_p^{(i)}$	the minimum number of linearly independent coded blocks for chunk i that chunk i can participate the process of combination of chunks
N_k	the total number of blocks for source k
Ψ_k	the degree (number of input blocks per chunk) distribution for source k while considering the application of batched sparse code
r'	a new version of r where $r \in \{1, 2, \cdots, D_{\max}\}$, and $r' \in \{1, 2, \cdots, D\}$ while $D \leq D_{\max}$

$\rho'(r')$	new version of $\rho(r)$ according to the new version of r, i.e., r'
n.b.s.s(r')	a chunk that has r' linearly independent coded blocks, and its state is non back-substituted state
h.b.s.s(r')	a chunk that has r' linearly independent coded blocks, and its state is haft back-substituted state
f.b.s.s(r')	a chunk that has r' linearly independent coded blocks, and its state is full back-substituted state
q.b.s.s(r')	a chunk that has r' linearly independent coded blocks, and its state is quasi-half back-substituted state
$\varrho_n(r')$	the probability that there are chunks have r' linearly independent coded blocks with n.b.s.s
$\varrho_h(r')$	the probability that there are chunks have r' linearly independent coded blocks with h.b.s.s
$\varrho_q(r')$	the probability that there are chunks have r' linearly independent coded blocks with q.b.s.s
$\varrho_f(r')$	the probability that there are chunks have r' linearly independent coded blocks with f.b.s.s
$\varrho_d(r')$	the probability that a chunk has r' linearly independent coded blocks after conducting the updating process
p_s	the probability that a combination can make the corespondent ch.cs decodable
q_s	$1 - p_s$
$\varrho_b(r')$	the probability that a chunk that has r' linearly independent coded blocks can play the role of the beginning chunk
$\varrho_i(r')$	the probability that a chunk that has r' linearly independent coded blocks can play the role of the intermediate chunk
$\varrho_e(r')$	the probability that a chunk that has r' linearly independent coded blocks can play the role of the ending chunk
p_c	the probability that a combination of chunks exists and is decodable

q_c	the probability that a possible combination of chunks is not decodable
p_t	the transfer probability
$r_{\mathrm{bs}}^{(t)}$	the rank of the matrix $\mathbf{C}^{(t)}$ with ν rows eliminated
$r_{\mathrm{bs}}^{\prime(t)}$	new version of r_{bs}
$r_{\max}^{\prime(t)}$	the maximum value of r_{bs} and $r_{\mathrm{bs}}^{\prime(t)}$
$r_{\min}^{\prime(t)}$	the minimum value of $r_{\mathrm{bs}}^{\prime(t)}$
$\rho_{\mathrm{bs}}\left(r_{\mathrm{bs}}\right)$	the probability distribution of $r_{\mathrm{bs}}^{(t)}$
ρ_{bs}	the probability distribution of $r_{\max}^{\prime(t)}$
$\rho_{\mathrm{bs}}^{\prime}\left(r_{\mathrm{bs}}^{\prime}\right)$	the probability distribution of $r_{\mathrm{bs}}^{\prime(t)}$
$r_{\mathrm{in}}^{(t)}$	the rank increment from the aid of ν overlapped blocks to undecodable chunk t
$\rho_{\mathrm{in}}\left(r_{\mathrm{in}}\right)$	the probability distribution of $r_{\mathrm{in}}^{(t)}$
p_{bs}	the fraction (probability) of the decoded chunks that can be used to conduct back-substitution
q_{bs}	$1-p_{\mathrm{bs}}$
$\varrho_{\phi}\left(\phi\right)$	the probability (fraction) that a chain of chunks with length ϕ is successfully decoded
N_{ch}	the number of chains of chunks with length $\phi_{\max}$
ϵ_{thr}	the minimum increment of p_d to terminate the process of updating probabilities
p_{thr}	the threshold probability to select an allocation $\left[\left(\mu_1,\gamma_1\right),\left(\mu_2,\gamma_2\right),\cdots,\left(\mu_K,\gamma_K\right)\right]$
τ_f	the transmission time of a feedback, normalized by a slot time for transmitting a block
p_f	reception success rate of feedback
c_{fc}	a parameter of fountain code (Luby transform code)
δ_{fc}	a parameter of fountain code (Luby transform code)
S_{fc}	the number of received coded blocks such that the receiver can recover all blocks with probability $1-\delta_{\mathrm{fc}}$
N_{dec}	the total number of decoded blocks

N_{fb}	the total number of feedback
N_{ts}	the total number of time slots taken excluding the transmission time of feedback
ε_t	the transmission efficiency
$\hat{r}^{(i)}$ (also $\hat{r}$)	the expected or estimated value of $r^{(i)}$. $\hat{r}$ is for any i
$\vartheta\left(\gamma^{(i)}\right)$	the probability that the blocks of chunk $i-1$ and chunk i can be recovered after transmitting chunk i with $\rho\left(r\right)$ while considering in single flow transmission with the assigned number of blocks taken from the previous chunk, $\gamma^{(i)}$
$\vartheta_k\left(\gamma_k^{(i)}\right)$	the probability that the blocks of chunk $i-1$ and chunk i can be recovered after transmitting chunk i with $\lambda_k\left(\theta_k\right)$ while considering in single flow transmission with the assigned number of blocks taken from the previous chunk, $\gamma_k^{(i)}$
$r_{\gamma\min}^{(i)}$	the minimum number of coded blocks to be received at the transmission of chunk i such that the coded blocks of chunks $i-1$ and i can be recovered while $\gamma^{(i)}$ blocks are taken from the previous chunk
$p_s\left(r\right)$	the probability that the chain of chunks with combination $\left[r^{(i-1)}, r\right]$ has $2\cdot K\cdot D_t-\gamma^{(i)}$
ϑ_{eff}	the effective probability that the blocks of all source in chunk $i-1$ and chunk i can be recovered with the assigned $\gamma^{(i)}$ and $\gamma_k^{(i)}$ for all k
$r_{\max}$	the total number of blocks of all sources per chunk, $r_{\max}=K\cdot D_t$
$\bar{\eta}$	the upper bound of η_{eff} for RLNC/CF scheme
$\bar{\varepsilon}_t$	the upper bound of ε_t for RLNC/CF scheme
p_{RD}	forwarding success rate
τ_{rcv}	reception delay, the period of time from when a block is added into the transmission window to when it is received
η_d	$\vartheta_{\text{eff}}\cdot\left(r^{(i-1)}+\hat{r}\right)$
r_d	selection of $\hat{r}$ such that η_d is maximum
τ_ϕ	the bound of reception delay by an assigned $\phi_{\max}$

r_j	selection of $\hat{r}$ for the joint of selection $\hat{r} = \bar{r}$ and selection $\hat{r} = r_d$ with an assigned $\phi_{\max}$
ϵ_{eff}	failure rate

Chapter 1

Introduction

1.1 Research Background

With the development of science and technology, the network nodes can be found everywhere, and the network density, i.e., the network connections becomes higher and higher [1]. For example, in houses, they can be electric appliances (smart home networks) [2]. On roads, they could be vehicles, road side access point, smart phones carried by pedestrians, etc (vehicular networks) [3]. Sharing information between these network nodes can provide useful application such as monitoring and controlling home appliance via wireless networks by near or far distance, improving driving safety and traffic management in vehicular networks, etc. However, with the increase in the required connections in wireless networks and limited channel bandwidth, especially, for the devices using unlicensed frequency band, the message transmission of each node might be suffered by the interference from the other nodes.

The impact from interference can cause collision at the receiving node (hence, message loss), high energy consumption due to increasing receiving signal-to-interference-plus-noise ratio (SINR), long latency to complete the data transmission, etc. The network performance in term of throughput, energy consumption and latency would be significant to decide whether to employ the application using information sharing between network nodes. For example, the latency constraint required for intelligent transport systems is $10 - 100$ ms [4]. In addition, if energy consumption of the application is high to obey the latency constraint, then this application might not satisfy well the applications with network devices using battery power supply, and it might become unwelcome.

To deal with the interference, the interference management such as resource allocation and access allocation (scheduling) [5], for example, can be employed to improve the network performance in term of throughput, energy consumption and latency. Scheduling is designed in the application using the same frequency band (spectral channel). Resource allocation operates in the different spectral channel for frequency channel allocation, and it operates in the same spectral channel for transmit power allocation. When only single frequency channel is applied, successive interference cancellation (SIC) [6, 7] can be applied at the receiver which receives the collided signals via transmit power allocation. The source message that has high SINR could be decoded and extracted from the collided signals, and it is used then to extract the other source message. This approach can allows the multiple sender to transmit simulateneously. ZigZag decoding [8] can be employed to extract the message from the collided messages of two messages with the same content from the same source where there is a time gap between the receiving time of two signals. If considering on multiple access technique, allowing the multiple users to transmit their simultaneously to a common receiver is called non-orthogonal multiple access (NOMA). NOMA has been used in term of multi-user superposition transmission by long term evolution-advanced in the fourth generation of cellular mobile communications [9].

On the other hand, physical-layer network coding (PNC) [10] can generate the collided signal into functions of the signals transmitted from the different senders. For example, PNC maps the collided signal into the linear combinations of the transmitted signals of different sources [11]. SIC can be also done by using PNC, for example, in the work of [12], if a receiver can generate the required linear combinations of the transmitted signals correctly. However, if a receiver could not ensure the recovery of all the original messages from all senders with SIC, it can cooperate with the other receivers, e.g., combining all linear combinations of the transmitted signals as in multi-source multi-relay channels [11]. The final destination can recover all the original messages from the different senders if there are enough functions (linear combinations in [11]) of the transmitted signals of different senders. Hence, PNC also enables multiple senders to transmit their message simultaneously to a common receiver via the same frequency channel, i.e., non-orthogonal channel, and the issues caused by interference can be solved more efficiently by using

PNC, i.e., even if SIC could not be done.

Considering the wireless transmissions via non-orthogonal spectral channel is to solve the issue of spectrum crunch. However, there is still the other spectrum bandwidth available, for example, millimeter wave [13]. However, the transmissions via non-orthogonal channel or PNC can improve the efficiency of using the acquired spectrum bandwidth since the network connections is growing exponentially [14]. In addition, PNC can also be employed in optical communication [10, 15].

1.2 Network Coding

Network coding (NC) was introduced by R. Ahlswede et al. [16] for packet switching network where a big message file is divided into small packets. The concept of network coding is that the outgoing packet of the sender is a function of incoming packets. By considering the flow of message transmission, i.e., the sources and the destination, network coding approaches could be divided into intra-flow (or intra-session) network coding (IANC) and inter-flow (or inter-session) network coding (IENC) [17, 18], where a flow is a pair of a source and a destination.

1.2.1 Intra-Flow Network Coding

IANC is applied with the packets from the same source. It is employed in the purpose to achieve the min-cut capacity [19]. In addition, it could be used to reduce the protocol overhead caused from feedback overhead since a feedback or an acknowledgement (ACK) is needed from the receiver when a packet is correctly received[20]. The popular network coding technique for IANC is linear network coding (LNC)[19] which linearly combines the input packets in a finite field $\mathbb{F}_q$, i.e., the linear combination coefficient vector (coding coefficient vector) and the input packets are in the finite field, where q is the size of the finite field. In case of lossy network, random linear network coding (RLNC)[20, 21] where the coding coefficient vectors are randomly drawn from the finite field. The benefit of LNC and RLNC is that the receiver only waits to receive enough linearly independent (innovative) coded packets such that the original packets can be recovered (decoded), and only an ACK is needed to send back to the source, thus the transmission time needed to complete the message transmission can be reduced. In order to reduce the encoding and decoding computational complexity, input packets are grouped into disjoint generations or chunks before encoding [22–24], and LNC or RLNC is applied among each chunk. This grouping fashion is called chunked code, and a feedback (ACK) is needed when each chunk can be decoded.

If re-encoding is employed at the intermediate nodes (relays) while LNC or RLNC is applied in wireless multi-hop network, high linear independence between the coded packets from the different senders, i.e., q is enough high, can improve the network throughput, for example, in a scenario of a two-hop single-flow where the direct link transmissions from the source to the destination are also considered [25]. The intermediate node re-encodes (or decodes and re-encodes) the correct coded packets with RLNC before forwarding to the destination. The number of time slots (or transmissions) needed for forwarding can be reduced because there might be coded packets already received by the destination via the direct link. These coded packets are linearly independent of the re-encoded packets forwarded by the intermediate nodes. The benefit would be more significant when the intermediate nodes are dense between the source and the destination. The well-known of the application of IANC is in the work of [24].

On the other hand, some network codes (or erasure codes) that can ensure the decodability at the receiver without need of feedback in single-flow transmission, i.e., the transmission from a source to a destination via intermediate nodes if needed, were proposed such as in the works of [26–31]. The grouping fashion in these works can be called overlapped chunked code (OCC), where a packet can belong to more than one chunk. Hence, a decoded chunk can help the other undecodable chunks which have common packets by back-substituting the already recovered packets to the undecodable chunks. These works are to deal with the protocol overhead caused by the transmission delay of feedback and the loss of feedback. This protocol overhead would be more significant, for example, in the transmission between satellite and ground stations (the transmission delay of feedback is long) or in multi-hop transmission where feedback might be lost [32]. Some of these works such as in [30, 31] tried to obtain the performance of the proposed network codes reach the channel capacity.

1.2.2 Inter-Flow Network Coding

IENC is applied with the packets from the different flows, i.e., from the different sources and normally for different destinations. The most studied scenarios is two-ways relay channels (TWRC) [33]. The most employed network coding technique is exclusive OR (XOR) which is the special case of linear network coding (when taking $q = 2$). The total number of time slots needed (the total number of transmission needed) are reduced from four time slots (transmissions) to three time slots (transmissions) if considering in TWRC with perfect channel condition. Hence, the network throughput and the energy efficiency are improved. The application can be extended to the scenario of cross topology (or x-atom topology) for the general case. The growing density of network nodes and many data transmission flows between the network nodes can increase the chance to conduct inter-flow network coding, and hence more benefit from inter-flow network coding [34].

Although it seems that IENC has higher potential to improve the network performance than IANC, however, the success improvement of network performance by IENC depends on the nature of broadcast transmission and the existence of the overheard packet at each source, where in TWRC, the packet overheard by a source is the packet sent by that source itself, and it is the packet sent by the other source in cross topology. Hence, the overheard packet reception success rate is perfect for TWRC, but it might be imperfect for cross-topology if considering the transmission in lossy channel. Hence, in order to obtain the advantage from IENC, an additional interaction between nodes is needed to check the overheard packet reception success rate at each source and whether the appropriate topology (TWRC, cross topology, etc.) exists or not to determine the opportunity to conduct IENC. Hence, the protocol overhead would be high.

1.2.3 Joint of IANC and IENC

Using the joint of IANC and IENC, i.e., employing IANC and IENC in the multiple-flow transmission (typically two-flow transmission) with common intermediate node (relay), was studied in the works of [25, 35–37] to obtain the benefit from both approaches. The cross-topology was often taken for the case of study. The advantage of IANC with re-encoding with RLNC at relay was studied in [25]. How to decide which packets to be done with IENC at relay (or how many redundant coded packets needed to be generated by relay) to efficiency obtain the benefit of IENC (or to adapt the overheard packet loss rate of the overhearing channel link) was discussed in the work of [35]. The selection of number of input packets for each chunk at each source was considered in the work of [36] to support different channel conditions. The joint of IENC and IANC with an erasure code (an OCC), fountain code, was proposed in the work of [37] for a two-way multi-hop line network to obtain the benefit of OCC and IENC.

1.3 Physical-layer Network Coding

As the research in network coding became deeper and deeper, the concept of network coding was applied at the physical-layer, and it was called analog network coding [18, 38] or PNC [10]. PNC takes use the fact that the electromagnetism wave of the receiving signal at a receiver is the superposition of those of transmitting signals from the different sources. The network coding approaches described in the previous section is called straightforward network coding (SNC) for this dissertation. SNC is conducted with information symbol, normally in a finite field $\mathbb{F}_q$ while PNC is conducted with the information signal, in the real field $\mathbb{R}$.

By assuming that the identities of the transmitted signals, i.e., which sources transmit, are known by the receiver, the receiver tries to map the superimposed signals into a function of the transmitted signals. Excluding SIC technique based on power allocation, which is mostly employed in NOMA [39], only PNC approach that maps the superimposed signal into a function of transmitted signals from different senders are mentioned here. In the work of [18, 38], the superimposed signal is the sum of transmitted signals from different senders, and the approach is called amplify-and-forward (AF). The application of AF was studied in TWRC and in line networks (multi-hop networks with single intermediate node for each hop) [38].

For TWRC, PNC is done as IENC approach. The relay amplifies the superimposed signals and then forwards to the destinations. Each destination tries to extract their desired packets from the forwarded signal by subtracting it by their transmitted signal. The assigned transmit power of each sender is determined by the channel gain or channel coefficient of the link from that sender to the relay. This channel coefficient is estimated by the relay and informed the correspondent sender via feedback. The number of time slots needed is reduced from three time slots to two time slots by comparing with the IENC approach mentioned in the previous section. However, the number of transmissions are still three. It means that the efficiency of using channel in time domain, called *channel efficiency* in this work, can be improved from $\frac{2}{3}$ packet/time slot to $\frac{2}{2} = 1$ packet/time slot in ideal case. However, the efficiency of transmitting a packet (by counting the total number of transmissions of all sources), called *transmission efficiency* in this work, has no change in the ideal case. The ideal case is the case that there is no packet loss, and decoding the superimposed signal is perfectly achieved as desired. On the other hand, for line networks, PNC is done as IANC approach. PNC is done at the node while its both neighboring nodes, i.e., its previous hop and its next hop, transmit the new packet and the former packet, respectively.

The shortcoming of AF is that the receiver needs to amplify the superimposed signal before forwarding, hence the noise built inside the superimposed signal is also amplified [11]. Then, the forwarded signal could not provide high correctness of the desired packets for each destination after decoding. A reliable PNC [40] or Compute-and-forward (CF) based on nested lattice codes (NLC) [11] is also a PNC approach proposed by Nazer and Gastpar. CF decodes a superimposed signal to obtain linear combinations of transmitted signals, i.e., linear combination of NLC codeword, where the linear combination coefficient vectors are in a finite field $\mathbb{F}_q$. The correctness of the codeword combinations depends on the strength of noise built naturally in the superimposed signals and the noise built from forcing the channel coefficient (channel gain) of each link to integer value. The final destination can recover the original information from all senders if there are enough codeword combinations. Therefore, CF plays a role as RLNC [41]. An implementation of a two-way relay network with compute-and-forward in GNU radio was given in the work of [42]. In addition to TWRC, CF are mainly employed in multi-source multi-relay channels. In this work, channel efficiency reflects the network throughput and latency, and transmission efficiency reflects the energy efficiency.

1.4 Vision and Objectives

The works previously done by the other researchers shows the advantage of applying SNC in the data transmission. Network coding can improve the the network throughput and energy efficiency with IENC in multiple-flow transmission, and can also improve network throughput and the network throughput with IANC in multi-hop wireless network transmissions. Furthermore, the network throughput can be improved against protocol overhead with IANC in single-flow transmission. The joint of IANC and IENC can bring the benefit of IANC and IENC. In addition, PNC can further improve the network throughput, i.e., the channel efficiency.

The vision of this work is to apply network coding (PNC and SNC with both IANC and IENC approaches) in multiple-flow multi-hop wireless network transmissions to improve the network throughput, the energy efficiency and to reduce the transmission latency when the network density is high.

The work in [40] showed the advantage of applying nested lattice codes over analog network coding based on amplify-and-forward in TWRC. This made CF which is based on nested lattice code [11] become a promising network coding approach, i.e., PNC approach, [41, 43, 44]. Furthermore, CF can be employed for SIC [12] and can support multiple-input multiple-output (MIMO) transmission mode [45]. Therefore, CF is employed as the PNC approach for this work. In addition, different from NOMA where a relay is used in many cases, this work considers the employment of multiple relays to support multiple-flow transmission while there might be some flow transmissions do not affect each other at all although the same channel in frequency domain and time domain is employed. By taking only a part of networks as the case of study, multi-source multi-relay networks with one or many destinations is considered for this work. From now on, the term *blocks* are used instead of packets in this work.

On the other hand, there are rare work that employs PNC and both SNC approaches. An existing work can be found in [46] where the probability that a chunk of blocks of all users can be recovered, i.e., the decodability of a chunk, is analyzed with the activity of users, i.e., the probability that users access the channel. Blocks of a chunk is encoded with IANC approach, i.e., is applied before encoding with NLC. However, because the considered channel condition is additive white Gaussian noise channel (AWGN) without fading, hence it might not satisfy all channel conditions.

In addition, only the case that the blocks can be recovered or the case that a chunk of blocks can be recovered is counted as successful transmission, and the case that a chunk are undecodable is counted as unsuccessful transmission. The same consideration is done when IANC is not applied, i.e., the number of blocks per chunk is one, such as in the work of [41, 45, 47]. There would be waste of channel use and energy consumption if not trying to make undecodable chunk become decodable, i.e., recoverable. A retransmission scheme was proposed in the work of [48] while PNC with IENC approach is applied in cross-topology. However, an addition interaction between sources is needed to manage the retransmission, thus the protocol overhead is increased, and it might be not practical if this interaction is not realizable, for example, both sources are not in the communication range of each other. In addition, for multi-source multi-relay networks, since all relays might need to cooperate with each other in order to forward the linear combinations of signals of all sources to the next hop [41, 47], hence it would not be efficient to wait for a relay having all required codeword combinations before forwarding, as it was done with IANC in single-flow transmission [25]. On the other hand, at the source side, the sent blocks should be recovered at the destination as soon as possible such that they are not expired before the deadline because the retransmission only could be done from the sources while applying PNC [48]. Since an undecodable chunk can become decodable by the aid of the decoded chunks while OCC is employed, this work aims to investigate the performance of OCC in this considered scenario by considering two solutions employing OCC to improve the network performance: (i) employing OCC as IANC approach with PNC while the feedback can be avoided; (ii) employing a feedback-based and OCC-based retransmission scheme which is efficient and practical with low protocol overhead. The detailed objectives of this dissertation are as below:

- improving the network throughput, i.e., channel efficiency, by designing an OCC to be applied with CF where the feedback about the reception state can be avoided. The design is done using the empirical probability distributions which are related to channel state to support all channel conditions. Furthermore, the computational complexity of encoding and decoding, i.e., how many blocks sent such that the destinations need to wait to receive for starting decoding, is considered in the design in order to support the network nodes with low specification hardware. The considered protocol overhead is mainly the transmission time of feedback and the loss of feedback;

- proposing a retransmission scheme by using block ACK (BACK) scheme and by using the feature of OCC where an undecodable chunk can become decodable by the aid of a decodable chunk or the transmitted chunk with a determined probability;
- improving the energy efficiency by reducing the total number of transmissions (including the transmission of feedback) to complete the data transmission.

The novelty of this work is applying OCC over CF, i.e., blocks are grouped into overlapped chunks and encoded with RLNC before passing NLC encoder at each source, in multi-source multi-relay networks.

1.5 Contribution

Regarding the objectives above, this work provides contributions as below:

- proposed a design of OCC which is applied before employing NLC at the sources, where RLNC is applied within each chunk. The feedback about the reception state of each chunk can be avoided. For the start, only the transmissions from the sources to the relays are considered. The design are based on the empirical probability distributions which are related to the channel state, the probability distribution of the number of linearly independent codeword combinations and the probability distributions of the participation factor of each source into linearly independent codeword combinations all provided by the relays for the transmission of a chunk. The condition of decodability for the design was provided for the design that could be applied with any kind of OCC, i.e., any overlapping fashion. This decodability condition turns the design of OCC for multi-source multi-relay transmissions into multiple designs of OCC in single-flow transmission with the unique applied overlapping fashion. An application with a conservatively overlapped chunked code, i.e., contiguously overlapping fashion to the design was given. In addition, the decodability was estimated with limited number of received chunks, i.e., the number of received chunks that the destination needs to receive to start decoding. It is useful for the case that destination is with low hardware specification. The performance in terms of channel efficiency and transmission efficiency are considered while channel efficiency corresponds to the network throughput, and transmission efficiency corresponds to energy consumption. These two terms could be estimated via the estimated decodability. The Transmission scheme employing this OCC is called OCC/CF in this dissertation. This work corresponds to the published works in [2,4,5];

- proposed a transmission scheme called RLNC/CF where RLNC is employed within each contiguously overlapped chunk with the same size at each source for multi-source multi-relay network, and the feedback from the destination is needed to manage the retransmission. BACK scheme is applied for the transmission of each chunk and each hop to provide more feasibility when the number of sources is large. The determination of the number of overlapped blocks for the next transmitting chunk for each source for an expected reception state using the empirical probability distributions mentioned above was given. The blocks which are not selected for retransmission are assumed to be recovered by the destination after the transmission of next chunks, and they could be released from the transmission window. The performance for different selections of expected reception state was studied with the expected decodability of the blocks of the previous undecodable chunks and the transmitting chunk. The time that a block needs to be successfully received by the destination, i.e., reception delay and the rate that its reception time is beyond the deadline or it could not recovered, i.e., are further considered in addition to channel efficiency and transmission efficiency. The studied selections of the expected reception state can be used to obtain the desired performance while employing RLNC/CF scheme. This work corresponds to the published works in [5][8], in the submitted work [3] and in the work to be submitted [1].

1.6 Dissertation Organisation

The remainder of this dissertation is organised as follows:

- Chapter 2. System Model

  Chapter 2 starts with a brief introduction of the network coding technique employed in this work such as random linear network coding, chunked codes, overlapped codes, nested lattice code and compute-and-forward. Then, the system model for this work is described then. The considered scenario, the channel model and the approaches used in this work such as encoding process, forwarding process, empirical probability distributions are presented then.

- Chapter 3. A Design of Overlapped Chunked Codes over Compute-and-Forward
  Chapter 3 describes a design of overlapped chunked codes applied before nested lattice code, called OCC/CF, for multi-source multi-relay networks where compute-and-forward is employed. The decodability condition is provided with the analysis of decodability. The description of an application with contiguously overlapped chunked code for the design of OCC/CF is given then. It includes the decoding scheme, the estimation of decodability and the performance observation. The numerical results are given then with some discussions.

- Chapter 4. A Retransmission Scheme based on Overlapped Chunked Code over Compute-and-Forward
  Chapter 4 describes a feedback-based transmission scheme employing overlapped chunked code at each source before nested lattice code, called RLNC/CF, for multi-source multi-relay network where compute-and-forward and BACK scheme are employed, and end-to-end transmissions are considered. The encoding and forwarding scheme is described. The determination of the number of overlapped blocks for each chunk and each source is given then. The performance of different selections of expected reception state is studied. Then, the numerical results are given with some discussions.

- Chapter 5. Conclusion and Future Works
  This chapter summarizes the contents described and lists the future works that are planned to be done.

Chapter 2

Network Coding Techniques and System Models

In this chapter, the network coding technique, RLNC and CF based on NLC, are briefly described. The system model for this work are described then.

Notation: Boldface letters are used for vectors, e.g., $\mathbf{a}$. The capital boldface letters are for matrices, e.g., $\mathbf{G}$. Superscripts ^{T} and $^{-1}$ refer to matrix transposition operation and inverse operation, respectively. $\mathbb{R}$ and $\mathbb{Z}$ denote the field of real values and the field of integer values, respectively. In addition, sign $\cdot$ refers to the multiplication operation, and sign $\times$ is used to express the size of the matrix. On the other hand, $\log^+ x$ refers to the operation $\max\{\log_2 x, 0\}$.

2.1 Straightforward Network Coding

2.1.1 Random Linear Network Coding

If $[\mathbf{b}_1, \mathbf{b}_1, \cdots, \mathbf{b}_D]$ are the D input blocks, where $\mathbf{b}_d \in \mathbb{F}_q^n$, n is the number of symbols inside each input block, and $d \in \{1, 2, \cdots, D\}$. The process of RLNC to generate an output (coded) block $\mathbf{w} \in \mathbb{F}_q^n$ can be expressed as below:

$$\mathbf{w} = \left(\bigoplus_{d=1}^{D} \chi_d \cdot \mathbf{b}_d \right) \mod q, \tag{2.1}$$

where $\boldsymbol{\chi} = [\chi_1, \chi_2, \cdots, \chi_D]^T \in \mathbb{F}_q^D$ is called coding coefficient vector of the coded block $\mathbf{w}$. The sign $\bigoplus$ refers to the summation operation over finite field $\mathbb{F}_q$, and $(\ ) \mod q$ is the operation mapping the values inside parentheses ( ) or inside brackets [ ] into finite field $\mathbb{F}_q$ in this dissertation. $\boldsymbol{\chi}$ is randomly drawn from the finite field $\mathbb{F}_q^D$. For LNC, $\boldsymbol{\chi}$ might be exactly determined in order to achieve a purpose, for example, to generate D linearly independent coded blocks at the sender with D input blocks, or to generate $D_t \geq D$ coded blocks with sparse coding coefficient matrix $\left[\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \cdots, \boldsymbol{\chi}_{D_t}\right] \in \mathbb{F}_q^{D \times D_t}$. Two coded blocks are linearly independent or innovative if their coding coefficient vector is linearly independent to each other in the employed finite field.

At the receiver, if there are D linearly independent and correctly received coded blocks $[\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_D]$, then the D original blocks can be correctly recovered by

$$[\mathbf{b}_1, \mathbf{b}_1, \cdots, \mathbf{b}_D] = \left([\mathbf{w}_1, \mathbf{w}_1, \cdots, \mathbf{w}_D] \cdot [\boldsymbol{\chi}_1, \boldsymbol{\chi}_1, \cdots, \boldsymbol{\chi}_D]^{-1}\right) \mod q, \tag{2.2}$$

where $\boldsymbol{\chi}_d$ is the coding coefficient vector correspondent to the received coded block $\mathbf{w}_d$.

The decoding process can be done by using Gaussian elimination, especially, in case that the received coding coefficient matrix $[\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \cdots, \boldsymbol{\chi}_D]$ is a dense matrix. To generate a coded block, $\mathcal{O}(n \cdot D)$ finite field operations are needed [30]. A finite field operation refers to the addition or multiplication of two elements. For decoding process, $\mathcal{O}(D^2 + D \cdot n)$ finite field operations per block are needed. On the other hand, if $[\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \cdots, \boldsymbol{\chi}_D]$ is a sparse matrix, then the decoding process can be done by using back-substitution or inactivation decoding method [49, 50] to reduce the computational complexity.

2.1.2 Chunked Codes and Overlapped Chunked Codes

From the previous section, the computational complexity of encoding and decoding process depends on the number of input blocks, D, i.e., if D is large, then it becomes high, especially, for the decoding process. If a big file message are divided into big number of blocks, in order to reduce the computational complexity, the divided blocks are grouped into chunks (or generations, batches) [19, 22], and LNC or RLNC are applied within each chunk. This way of encoding is called chunked code. There are two types of chunked codes: non-overlapped chunked code and overlapped chunked code (OCC).

Non-overlapped chunked code is to group the blocks into disjoint chunks, i.e., there are no common blocks between all chunks. An ACK is usually needed for each chunk after the receiver can decode the received coded blocks of that chunk to inform the sender such that it stops transmitting the coded block of the current chunk and prepares the transmission of next chunk. The feedback can be avoided if the estimated block success rate (BSR) for each chunk is exact, thus the sender can generate $D_t = \frac{D}{p}$ coded blocks per chunk where there are D blocks per chunk, and p is the BSR of the link from the sender to the receiver, such that the receiver can have D linearly independent coded blocks to extract the original blocks, for example. However, it is hard to be achieved since the channel state varies and cannot be exactly estimated in practice. On the other hand, the feedback can be avoided by compensating for the loss in the worst case, but it might consume high cost of channel use and energy consumption. The computational complexity of encoding and decoding process depends on the number of blocks in each chunk.

For OCC, an original block might belong to more than one chunk. Hence, in the decoding process, back-substitution can be employed, i.e., the original blocks which have already been recovered and belong to more than one chunk can be substituted into the chunks that contain those blocks and are not decodable (undecodable), i.e., the chunks that do not have enough linearly independent coded blocks. Hence, the number of linearly independent blocks (including coded blocks and recovered blocks) is increased in those chunks, then the undecodable chunks can become decodable. In addition, with suitable design of OCC, the feedback from the receiver for each chunk reception can be avoided, all original blocks can be recovered without knowledge of reception states of the transmitted chunks. However, the computational complexity of decoding process depends on the total number of original blocks in the worst case when there are no any decodable chunks, for example in the work of [27, 28].

2.2 Compute-and-Forward based on Nested Lattice Code

2.2.1 Nested Lattice Codes

An n-dimensional lattice Λ is a linear additive subgroup of $\mathbb{R}^n$, i.e., if $\mathbf{x}_1, \mathbf{x}_2 \in \Lambda$, then $\mathbf{x}_1 + \mathbf{x}_2 \in \Lambda$ and $-\mathbf{x}_1 \in \Lambda$. A lattice point $\mathbf{x} \in \Lambda$ is generated by the generator matrix $\mathbf{G} \in \mathbb{R}^{n \times n}$ and an integer vector $\mathbf{b} \in \mathbb{Z}^n$ by:

$$\mathbf{x} = \mathbf{G} \cdot \mathbf{b}. \tag{2.3}$$

The fundamental Voronoi region of Λ, $\mathcal{V}$, is the space that is closer to the origin $\mathbf{x}_o$ ($\mathbf{x}_o = \mathbf{0}$) than to the other lattice points. A scaled lattice $\Lambda_\xi = \xi \cdot \Lambda$ is obtained by scaling the generator matrix of Λ, i.e., $\mathbf{G}_\xi = \xi \cdot \mathbf{G}$, for $\xi \neq 0$. A lattice Λ_ξ is nested in Λ if $\Lambda_\xi \subseteq \Lambda$. If ξ is a non-zero positive integer, then $\Lambda_\xi = \xi \cdot \Lambda$ is nested in Λ.

NLC is formed by a coding lattice Λ_c and a shaping lattice Λ_s, where $\Lambda_s \subseteq \Lambda_c$. The codebook of NLC is the coset leaders of Λ_c/Λ_s, i.e., the lattice points (codewords) of Λ_c that are inside the fundamental Voronoi region of Λ_s, $\mathcal{V}_s$. If taking $\Lambda_s = q \cdot \Lambda_c$, where q is a prime number, and the generator matrix of Λ_c, $\mathbf{G}_c$, is full rank, then the coding rate of NLC is $R = \log_2 q$. The number of codewords is q^n. The feature of NLC is that the linear combination of two codewords is still a codeword. The encoding process of NLC can be done as below.

$$\mathbf{x} = [\mathbf{G}_c \cdot \mathbf{b}] \mod \mathcal{V}_s, \tag{2.4}$$

where $\mathbf{b} \in \mathbb{F}_q^{n'}$ is the information where $1 \leq n' \leq n$, and $\mathbf{x}$ is the NLC codeword corresponding to $\mathbf{b}$, and $[\,] \mod \mathcal{V}_s$ is the operation mapping a lattice point of Λ_c into $\mathcal{V}_s$. This operation restricts the transmit power of a sending codeword by an assigned maximum transmit power per dimension, $P_{\max}$, i.e., $\|\mathbf{x}\|^2 = x_1^2 + x_2^2 + \cdots + x_n^2 \leq n \cdot P_{\max}$ with $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T$. The message rate is $\frac{n'}{n} \log_2 q$. However, this dissertation only considers the case $n' = n$ for convenience. The consideration for the case $0 \leq n' < n$ can be found in the work of [11].

The decoding process to extract $\mathbf{b}$ from $\mathbf{x}$ can be done as below:

$$\mathbf{b} = \left[\mathbf{G}_c^{-1} \cdot \mathbf{x}\right] \mod q. \tag{2.5}$$

2.2.2 Compute-and-Forward

If K senders transmit their codeword simultaneously to a common receiver, the accumulative codeword at the receiver can be expressed by:

$$\mathbf{y} = \sum_{k=1}^{K} h_k \cdot \mathbf{x}_k + \mathbf{z}, \tag{2.6}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ is an n-dimensional NLC codeword, which is transmitted from sender k for $k \in \{1, 2, \cdots, K\}$. $h_k \in \mathbb{R}$ is a real channel coefficient or channel gain of the link from sender k to the receiver. For the case of complex channel coefficient, the derivation can be done as in the work of [11]. On the other hand, $\mathbf{z} \in \mathbb{R}^n$ is AWGN with zero mean and variance $\sigma_{\mathbf{z}}^2$ per dimension. If ψ_k is an instantaneous received signal-to-noise ratio (SNR) of the link from sender k to the receiver, then $|h_k|$ can be obtained by:

$$|h_k| = \sqrt{\frac{\psi_k \cdot \sigma_{\mathbf{z}}^2}{P}}, \tag{2.7}$$

where $P = \frac{\overline{\|\mathbf{x}\|^2}}{n}$ is the mean transmit power of a codeword per dimension, and the sign of h_k depends on the phase of the received codeword $\mathbf{x}_k$.

The receiver computes $\mathbf{y}$ to obtain the linear combinations of the codewords of K senders, where one of these codeword combinations, $\mathbf{v}$, can be expressed by

$$\mathbf{v} = \sum_{k=1}^{K} a_k \cdot \mathbf{x}_k, \tag{2.8}$$

where $\mathbf{a} = [a_1, a_2, \cdots, a_K]^T \in \mathbb{Z}^K$ is a desired linear combination integer coefficient vector at the receiver, and a_k is called the k-th element of $\mathbf{a}$. To obtain $\mathbf{v}$, the receiver determines a gain α to multiply both side of (2.6) as below

$$\begin{aligned}\alpha \cdot \ \mathbf{y} &= \alpha \cdot \left(\sum_{k=1}^{K} h_k \cdot \mathbf{x}_k + \mathbf{z} \right) \\ &= \sum_{k=1}^{K} a_k \cdot \mathbf{x}_k + \left[\sum_{k=1}^{K} (\alpha \cdot h_k - a_k)\, \mathbf{x}_k + \alpha \cdot \mathbf{z} \right] \\ &= \mathbf{v} + \left[\sum_{k=1}^{K} (\alpha \cdot h_k - a_k)\, \mathbf{x}_k + \alpha \cdot \mathbf{z} \right].\end{aligned} \tag{2.9}$$

The term in bracket, $\left[\sum_{k=1}^{K} (\alpha \cdot h_k - a_k)\, \mathbf{x}_k + \alpha \cdot \mathbf{z}\right]$, is called effective noise, where the first term is the noise produced from forcing the channel coefficient vector $\mathbf{h} = [h_1, h_2, \cdots, h_K]^T \in \mathbb{R}^K$ to an integer coefficient vector $\mathbf{a} = [a_1, a_2, \cdots, a_K]^T \in \mathbb{Z}^K$, and the second term is from AWGN.

From the work in [11], the computation rate region related to the pair $(\mathbf{h}, \mathbf{a})$, defined by $\mathcal{R}(\mathbf{h}, \mathbf{a})$, can be obtained by

$$\mathcal{R}(\mathbf{h}, \mathbf{a}) = \max_{\alpha \in \mathbb{R}} \frac{1}{2} \log^{+} \left(\frac{P}{\alpha^2 \cdot \sigma_{\mathbf{z}}^2 + P \cdot \| \ \alpha \cdot \mathbf{h} - \mathbf{a}\|^2} \right). \tag{2.10}$$

Hence, the maximum $\mathcal{R}(\mathbf{h}, \mathbf{a})$ can be obtained by choosing α as the minimum mean square error(MMSE) coefficient α_{MMSE} which is expressed as below

$$\alpha_{\text{MMSE}} = \frac{P \cdot \mathbf{h}^T \cdot \mathbf{a}}{\sigma_{\mathbf{z}}^2 + P \cdot \|\mathbf{h}\|^2}, \tag{2.11}$$

and the maximum $\mathcal{R}(\mathbf{h}, \mathbf{a})$ is

$$\mathcal{R}(\mathbf{h}, \mathbf{a}) = \frac{1}{2} \log^{+} \left(\left[\|\mathbf{a}\|^2 - \frac{P \cdot \left(\mathbf{h}^T \cdot \mathbf{a}\right)^2}{\sigma_{\mathbf{z}}^2 + P \cdot \|\mathbf{h}\|^2} \right]^{-1} \right). \tag{2.12}$$

According to the work in [11], $\mathcal{R}(\mathbf{h}, \mathbf{a})$ is achievable for any large enough n and for the existing encoders and decoders such that the receiver can recover the desired codeword combination with $\mathbf{a} \neq \mathbf{0}$ with the average probability of error $\epsilon > 0$ if the maximum message rate of all sources satisfies the condition:

$$\mathcal{R}(\mathbf{h}, \mathbf{a}) > \max\{R_1, R_2, \cdots, R_K\}, \tag{2.13}$$

where R_k is the message rate for source k while employing NLC.

After determining a suitable α and obtaining $\mathbf{v}$, $\mathbf{v}$ is then mapped into $\mathcal{V}_s$ to obtain $\mathbf{u}$ before forwarding to obey the transmit power constraint as at the senders side, i.e., $\|\mathbf{u}\|^2 \leq P_{\max}$, as below.

$$\mathbf{u} = [\mathbf{v}] \mod \mathcal{V}_s = \left[\sum_{k=1}^{K} \beta_k \cdot \mathbf{x}_k\right] \mod \mathcal{V}_s, \tag{2.14}$$

where $\boldsymbol{\beta} = [\beta_1, \beta_2, \cdots, \beta_K]^T \in \mathbb{F}_q^K$ and $\boldsymbol{\beta} = [\mathbf{a}] \mod q$. $\boldsymbol{\beta}$ is a combination coefficient vector for codeword combination $\mathbf{u}$.

2.3 System Model

2.3.1 Scenario and Assumption

This dissertation considers a scenario of multi-source multi-relay multi-destination. There are K sources, L relays and M destinations. Each node is equipped with a single omni-directional antenna. In case of multiple receiving antenna, computing approach at each receiving node, i.e., relay or destination, can be considered as in the work of [45] to obtain better performance with the gained diversity of the codeword combinations. The direct links between the sources and the destinations are also considered to fit the general case, for example, in case of two-way relay channel, cross topology, etc. An example of the scenario with two-source two-relay two-destinations network is shown in Figure 2.1.

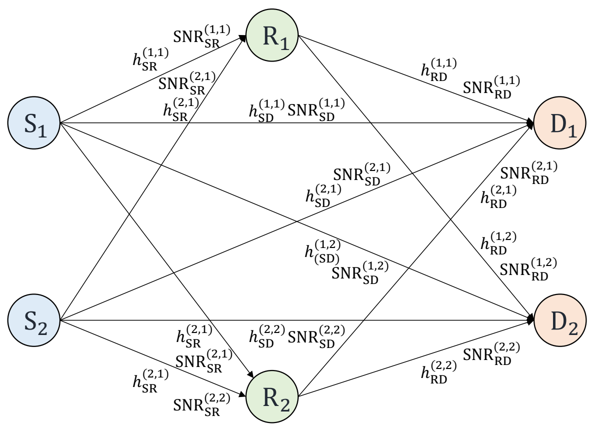


Figure 2.1: Scenario for the case of a two-source two-relay two-destination network.

In Figure 2.1, $h_{\text{SR}}^{(k,l)}$, $h_{\text{RD}}^{(l,m)}$ and $h_{\text{SD}}^{(k,m)}$ are the channel coefficients corresponding to the instantaneous received SNRs of the link from source k to relay l, from relay l to destination m and from source k to destination m, respectively, where $k \in \{1, 2, \cdots, K\}$, $l \in \{1, 2, \cdots, L\}$ and $m \in \{1, 2, \cdots, M\}$. $\text{SNR}_{\text{SR}}^{(k,l)}$, $\text{SNR}_{\text{RD}}^{(l,m)}$ and $\text{SNR}_{\text{SD}}^{(k,m)}$ denote the average received SNR of the link from source k to relay l, from relay l to destination m and from source k to destination m, respectively. The considered scenario can be extended to the case that there are L' stages of relays.

In this work, a codeword combination can be forwarded or accepted by the destinations if its computation rate satisfies Condition (2.13). In addition, each receiver can generate multiple linearly independent codeword combinations if they satisfy Condition (2.13). On the other hand, only the receivers have knowledge of channel coefficients because the channel coefficients are assumed varied quickly such that the channel coefficient could not be correctly estimated. In case that the estimation of channel coefficient at the senders is efficient, the higher computation rate can be obtained, hence, lower block error rate (BER) can be achieved. This work assumes that the estimation of the channel coefficient at receiver are with acceptable error. On the other hand, this work assumes that time is slotted and synchronized. The study of performance with asynchronized transmission was described in the work of [10].

All sources apply the same NLC with the same message rate which is $R_1 = R_2 = \cdots = R_K = R = \log_2 q$, i.e., symmetric message rate is considered. Asymmetric message rate that has been studied, for example, in the work of [11] is considered as the future work for the extension of this work. However, employing symmetric message rate (lower than or equal to $\log_2 q$) might be feasible for actual channel state because of Condition (2.13), i.e., the fact that the maximum of message rates of all sources is the message rate of all sources is more desired. In addition, if the received SNRs is not sufficient, the transmit powers at all sources should adapt the channel state to obtain the desired reception state at the receivers (relays or destinations). Employing asymmetric transmit power that has been studied in the work [44, 51, 52] can be considered for this situation as the future work of this work. On the other hand, erasure correction code can be applied before NLC encoder, e.g., employing Reed-Solomon code as in the work [42], and after RLNC encoder for this work. However, employing erasure correction code is also considered as the future work.

The scenario considered in this work can be a part of wireless multi-hop networks. It is considered as data collection in wireless sensor networks or data back-hauling in ultra-dense networks when $M = 1$. On the other hand, if its application in cognitive radio (CR) network is considered, the primary user (PU) is one of K sources, and the other sources are secondary users (SUs). Alternatively, all sources can be assigned as SUs.

2.3.2 Channel Model

Only real channel coefficients are considered, and the block channel fading is assumed, i.e., the channel coefficient for a whole block signal, i.e., codeword, within a time slot along a channel link is constant. In addition, Rayleigh fading is considered, and the channel coefficient is independently and identically distributed for each channel link. Hence, the real channel coefficient is normally distributed [45, 53]. On the other hand, AWGN has zero mean and unit variance, i.e., $\sigma_{\mathbf{z}}^2 = 1$ (which can be achieved by doing the normalization with P).

2.3.3 Computing Combination Coefficient Vector

In this work, $\mathbf{a}$ is determined by applying the approach proposed by Fincke and Pohst [54] as in the work of [55] to obtain the maximum $\mathcal{R}(\mathbf{h}, \mathbf{a})$. To suit the case that the hardware specification of nodes is low, this work exploits Condition (2.13) to reduce the computational overhead by reducing the number of candidates of $\mathbf{a}$ in searching for which $\mathbf{a}$ can provide the highest $\mathcal{R}(\mathbf{h}, \mathbf{a})$. In addition, Condition (2.13) is also used to filter codeword combination for forwarding to the destinations at each relay. Since $R = \log_2 q$, the higher value q results in a high BER. In this work, only small value of q is considered. Reducing the number of candidates of $\mathbf{a}$, i.e., reducing the bounds of the value of the elements of $\mathbf{a}$, can be done as in the works of [55, 56] by replacing the condition $\mathcal{R}(\mathbf{h}, \mathbf{a}) > 0$ with $\mathcal{R}(\mathbf{h}, \mathbf{a}) > R$.

However, this modification causes some decrease in performance in the BER because codeword combination might be correctly received without satisfying Condition (2.13). By comparing with the case that applies condition $\mathcal{R}(\mathbf{h}, \mathbf{a}) > 0$, the number of candidates, BER and computational latency are shown in Figure 2.2. The specification of the

Table 2.1: Specification of the simulation platform.

Term	Description
Processor	2.5-GHz Intel Core i7
Memory	16-GB 1600-MHz DDR3
Operating System	Mac OS
Software Tool	MATLAB

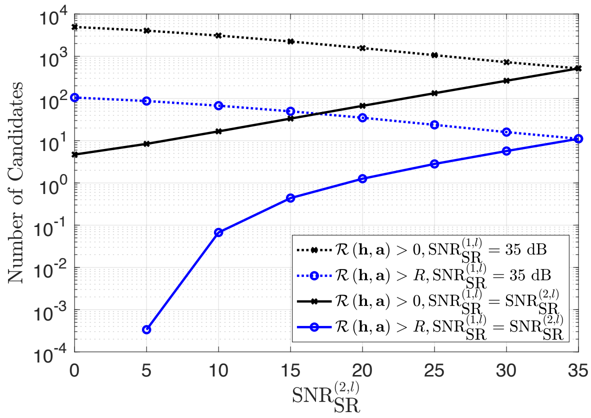
employed platform is shown in Table 2.1. $E_8/7E_8$ is used for NLC in this comparison, where $q = 7$, and E_8 is a well-known $n = 8$ lattice.

The results are obtained by considering the codeword combinations of two sources at relay l and taking $\text{SNR}_{\text{SR}}^{(2,l)} = 0 \to 35\,\text{dB}$ and $\text{SNR}_{\text{SR}}^{(1,l)}$ with two cases: $\text{SNR}_{\text{SR}}^{(1,l)} = 35\,\text{dB}$ and $\text{SNR}_{\text{SR}}^{(1,l)} = \text{SNR}_{\text{SR}}^{(2,l)} = 0 \to 35\,\text{dB}$. Relay l only computes to obtain only a codeword combination with the highest computation rate. The BER for condition $\mathcal{R}(\mathbf{h}, \mathbf{a}) > 0$ was obtained by comparing the codeword combination with the combination of the original codewords. For the case with Condition (2.13), the codeword combination is filtered with Condition (2.13) first before comparing with the combination of the original codewords. From Figure 2.2, this setting performs the trade-off between the computational latency and the performance in term of BER.

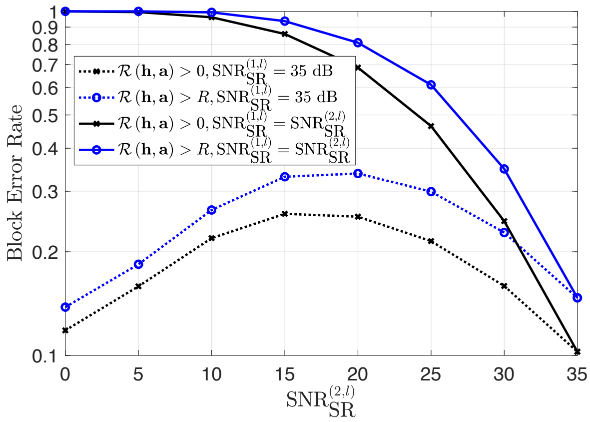
However, the approach above is only used for the case that $K \leq 2$. For the case $K \geq 3$, this work considers the simple approach, rounding approach, instead. For rounding approach, the value of α is varied and the combination coefficient vector $\mathbf{a}$ is obtaining by rounding $\alpha \cdot \mathbf{h}$, while $\mathbf{a}$ is determined first, and then α is obtained by (2.11) for Fincke-Pohst approach. It is obvious that rounding approach could not perfectly provide $\mathbf{a}$ for the maximum $\mathcal{R}(\mathbf{h}, \mathbf{a})$, but the number of candidates or the computation latency is constant and depends the variation gap of α. The computation latency of rounding approach might be lower than Fincke-Pohst approach when $K \geq 3$, especially, when the average SNRs are large.

2.3.4 Encoding and Computing

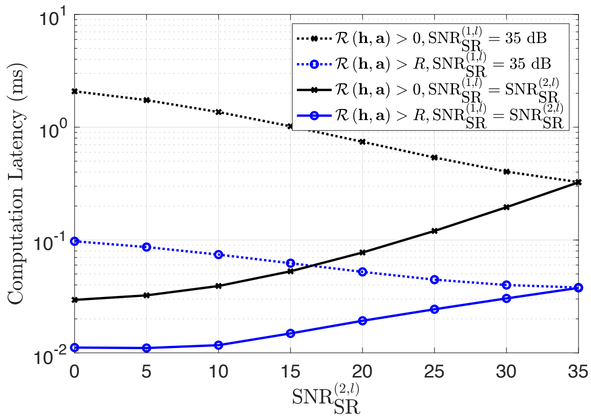
For source k where $k \in \{1, 2, \cdots, K\}$, the i-th chunk consists of $d_k^{(i)}$ blocks which are expressed by $\mathbf{B}_k^{(i)} = \left[\mathbf{b}_{k1}^{(i)}, \mathbf{b}_{k2}^{(i)}, \cdots, \mathbf{b}_{kd_k^{(i)}}^{(i)}\right]$. There are D_t time slots in maximum for transmitting each chunk for each sourc, and source k only transmits $D_k^{(i)} \leq D_t$ coded



(a)



(b)



(c)

Figure 2.2: The performance applying condition $\mathcal{R}(\mathbf{h},\mathbf{a}) > R$, compared with condition $\mathcal{R}(\mathbf{h},\mathbf{a}) > 0$. (**a**) Number of candidates; (**b**) block error rate (BER); (**c**) computational latency.

blocks, i.e., only uses $D_k^{(i)}$ time slots for chunk i. $\Omega_k^{(i)} = \left\{\omega_{k1}^{(i)}, \omega_{k2}^{(i)}, \cdots, \omega_{kD_t}^{(i)}\right\}$ denotes the channel access allocation of source k during transmitting chunk i, where $\omega_{kd_t}^{(i)} \in \{0, 1\}$ for $d_t \in \{1, 2, \cdots, D_t\}$. If $\omega_{kd_t}^{(i)} = 1$, source k is active at d_t-th time slot while transmitting chunk i. Otherwise, source k is idle. $D^{(i)}$ and $d_{\max}$ denote $\sum_{k=1}^{K} d_k^{(i)}$ and $\max\{d_k^{(i)}, 1 \leq k \leq K, \forall i\}$, respectively. RLNC is applied among chunk to generate $D_k^{(i)}$ coded blocks, $\mathbf{W}_k^{(i)} = \left[\mathbf{w}_{k1}^{(i)}, \mathbf{w}_{k2}^{(i)}, \cdots, \mathbf{w}_{kD_k^{(i)}}^{(i)}\right]$. The d_t-th coded block, $\mathbf{w}_{kd_t}^{(i)}$ for $d_t \in \{1, 2, \cdots, D_k^{(i)}\}$, is obtained by:

$$\mathbf{w}_{kd_t}^{(i)} = \left[\bigoplus_{d=1}^{d_k^{(i)}} \chi_{kd_td}^{(i)} \cdot \mathbf{b}_{kd}^{(i)}\right] \mod q, \tag{2.15}$$

where $\boldsymbol{\chi}_{kd_t}^{(i)} = \left[\chi_{kd_t1}^{(i)}, \chi_{kd_t2}^{(i)}, \cdots, \chi_{kd_td_k^{(i)}}^{(i)}\right]^T$ is a zero vector if $\omega_{kd_t}^{(i)} = 0$, and randomly drawn from $\mathbb{F}_q^{d_k^{(i)}}$ if $\omega_{kd_t}^{(i)} = 1$. For better linear independence of $\boldsymbol{\chi}_{kd_t}^{(i)}$ to the others, the coding coefficient matrix $\left[\boldsymbol{\chi}_{k1}^{(i)}, \boldsymbol{\chi}_{k2}^{(i)}, \cdots, \boldsymbol{\chi}_{kD_k^{(i)}}^{(i)}\right]$ can be psudo-randomly generated such that the rank of this matrix is equal to $\min\left\{d_k^{(i)}, D_k^{(i)}\right\}$, and $\boldsymbol{\chi}_{kd_t}^{(i)}$ is successively taken from this matrix when $\omega_{kd_t}^{(i)} = 1$. Superscript $^{(i)}$ is sometimes omitted when any chunk is considered. The computational complexity of the encoding process depends on chunk size $d_k^{(i)}$ for source k.

For source k and chunk i, the coded blocks of each chunk are then NLC encoded before transmitting to generate $D_k^{(i)}$ NLC codewords, $\mathbf{X}_k^{(i)} = \left[\mathbf{x}_{k1}^{(i)}, \mathbf{x}_{k2}^{(i)}, \cdots, \mathbf{x}_{kD_k^{(i)}}^{(i)}\right]$, as shown in Figure 2.3. All sources transmit their generated codewords according to their channel access allocation. Hence, there might be some sources transmitting their codeword simultaneously at a slot time.

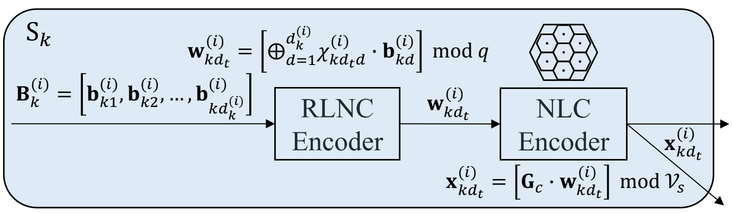


Figure 2.3: Encoding process at source k for chunk i.

The coding vector $\boldsymbol{\chi}_{kd_t}^{(i)}$ and the information of chunk i for source k, such as source ID (k), $d_k^{(i)}$, etc., can be attached to the transmitting data, e.g., at the header of the frame. However, the location of the attached information for a source should not overlap with those of the other sources, as shown in Figure 2.4. Hence, the small chunk size is preferred for the header with a limited length. This work assumes that the length of the attached information is negligible compared with the length of the sending block. Alternatively, this information can be known by the receivers (relays or destinations) by broadcasting from each source and forwarding from relays, for example. This work assumes that the content of this information is correctly received.

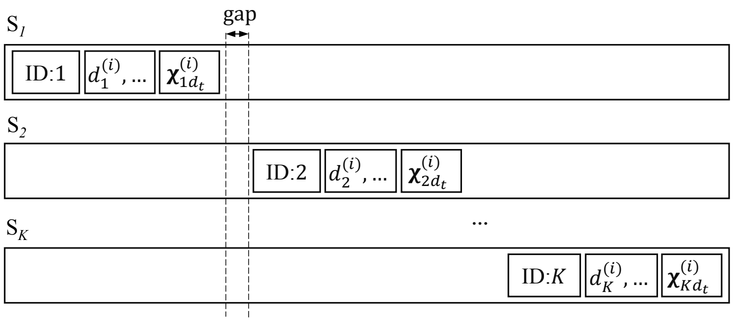


Figure 2.4: Locations of the attached information in the header of the sending frame of all sources for chunk i, where "$d_k^{(i)}, \ldots$" refers to the information about chunk i for source k such as chunk size $d_k^{(i)}$, information about the overlapped blocks, etc.

Relay l for $l \in \{1, 2, \cdots, L\}$ computes the superposition of $\sum_{k=1}^{K} \omega_{kd_t}^{(i)}$ codewords, $\left[\omega_{1d_t}^{(i)} \cdot \mathbf{x}_{1d_t}^{(i)}, \omega_{2d_t}^{(i)} \cdot \mathbf{x}_{2d_t}^{(i)}, \cdots, \omega_{Kd_t}^{(i)} \cdot \mathbf{x}_{Kd_t}^{(i)}\right]$, at slot d_t for chunk i to obtain their linear combinations $\left[\mathbf{u}_{ld_t1}^{(i)}, \mathbf{u}_{ld_t2}^{(i)}, \cdots, \mathbf{u}_{ld_tr_{ld_t}^{(i)}}^{(i)}\right] \in \mathbb{R}^{n\times\ r_{ld_t}^{(i)}}$ to forward to the destinations, where $r_{ld_t}^{(i)}$ is the number of codeword combinations that satisfy Condition (2.13) at relay l at slot d_t for chunk i and $0 \leq r_{ld_t}^{(i)} \leq\ K$. $\mathbf{u}_{ld_t\iota}^{(i)}$ for $\iota \in \{1, 2, \cdots, r_{ld_t}^{(i)}\}$ is expressed by

$$\mathbf{u}_{ld_t\iota}^{(i)} = \left[\sum_{k=1}^{K} \beta_{kld_t\iota}^{(i)} \cdot \mathbf{x}_{kd_t}^{(i)}\right] \mod \mathcal{V}_s, \tag{2.16}$$

where $\boldsymbol{\beta}_{ld_t\iota}^{(i)} = \left[\beta_{1ld_t\iota}^{(i)}, \beta_{2ld_t\iota}^{(i)}, \cdots, \beta_{Kld_t\iota}^{(i)}\right]^T \in \mathbb{F}_q^K$ is the ι-th combination coefficient vector computed at relay l at the d_t-th slot for chunk i, while $\beta_{kld_t\iota}^{(i)}$ would be zero if $\omega_{kd_t}^{(i)} = 0$ for all l. Then, the combined coding coefficient vector of $\mathbf{u}_{ld_t\iota}^{(i)}$, $\mathbf{c}_{ld_t\iota}^{(i)} \in \mathbb{F}_q^{D^{(i)}}$, is:

$$\mathbf{c}_{ld_t\iota}^{(i)} = \left(\left[\beta_{1ld_t\iota}^{(i)} \cdot \boldsymbol{\chi}_{1d_t}^{(i)}, \beta_{2ld_t\iota}^{(i)} \cdot \boldsymbol{\chi}_{2d_t}^{(i)}, \cdots, \beta_{Kld_t\iota}^{(i)} \cdot \boldsymbol{\chi}_{Kd_t}^{(i)}\right]^T\right) \mod q. \tag{2.17}$$

2.3.5 Forwarding

Each relay can work in haft-duplex mode by sharing the spectral channel with the sources, or in full-duplex mode by using the same spectral channel or the different spectral channels with the sources. However, this work does not focus on the mode that relays use, and haft-duplex is assumed. Since the codeword combination computed at each relay is not completely always correct, relay l can computes $r_{ld_t}^{(i)}$ linearly independent codeword combinations at the d_t-th slot for chunk i by selecting another computation rate threshold for Condition (2.13), i.e., the message rate R to filter the codeword combination for forwarding to the destinations. In this case, each relay would send more additional information such as the values of computation rate and the combined coding coefficient vector of each codeword combination to the destinations before forwarding the codeword combinations, such that the destinations can decide which relays and which codeword combinations should conduct forwarding and should be forwarded, respectively, to obtain the highest sum computation rate [41, 45, 47]. The interaction between the relays and the destinations can be done via control messages. The relays forward the codewords combinations separately, i.e., via orthogonal channel, or simultaneously, i.e., via non-orthogonal channel, according to the different applications.

In case that the codeword combinations which satisfy Condition (2.13) are perfectly correct at each relay, then re-encoding process can be done at relay before forwarding. The interaction between the relays and the destinations might be avoided when re-encoding process is conducted at each relay. It might be useful, especially, when there are multiple stages of relays. By assuming that there are $r_l^{(i)}$ codeword combinations filtered with Condition (2.13) at relay l for chunk i, if D_t' re-encoded codeword combinations are needed to forward to the next hop, then the re-encoding process to generate a forwarding codeword combination, $\mathbf{u}_{ld_t}'^{(i)}$ for $d_t \in \{1, 2, \cdots, D_t'\}$, can be done as below.

$$\mathbf{u}_{ld_t}'^{(i)} = \left[\sum_{\iota'=1}^{r_l'^{(i)}} \chi_{ld_t\iota'}^{(i)} \cdot \mathbf{u}_{l\iota'}^{(i)} \right] \mod \mathcal{V}_s, \tag{2.18}$$

where $\boldsymbol{\chi}_{ld_t}^{(i)} = \left[\chi_{ld_t1}^{(i)}, \chi_{ld_t2}^{(i)}, \cdots, \chi_{ld_tr_l'^{(i)}}^{(i)} \right]^T \in \mathbb{F}_q^{r_l'^{(i)}}$ is the re-encoding coefficient vector, and $\left[\mathbf{u}_{l1}^{(i)}, \mathbf{u}_{l2}^{(i)}, \cdots, \mathbf{u}_{lr_l'^{(i)}}^{(i)} \right]$ with the correspondent combined coding coefficient vectors

$\left[\mathbf{c}_{l1}^{(i)}, \mathbf{c}_{l2}^{(i)}, \cdots, \mathbf{c}_{lr_l'^{(i)}}^{(i)}\right]$ are the codeword combinations stored at relay l for chunk i. The combined coding coefficient vector of $\mathbf{u}_{ld_t}'^{(i)}$, $\mathbf{c}_{ld_t}'^{(i)} \in \mathbb{F}_q^{D^{(i)}}$, is

$$\mathbf{c}_{ld_t}'^{(i)} = \left[\sum_{\iota'=1}^{r_l'^{(i)}} \chi_{ld_t\iota'}^{i} \cdot \mathbf{c}_{l\iota'}^{(i)}\right] \mod q. \tag{2.19}$$

From now on, the codeword combinations received at the relays and the destinations are the codeword combinations that satisfy Condition (2.13), and their correctness is assumed for this work.

2.3.6 Empirical Probability Distributions

This work employs the empirical probability distributions to design the proposed schemes and to take as a reference of the network performance. The employed empirical probability distributions for this work are related to channel state, i.e., the reception state of the transmission of all chunks. In this work, the empirical probability distributions at the destinations are only considered and employed. This work assumes that there are D_t channel uses (time slots) for the transmission stage from the sources to the relays and D_t' channel uses for the transmission stage from the relays to the destinations, excluding the time uses for the interaction between nodes (sources, relays and destinations) via control messages or feedback when collecting the data for the empirical probability distributions. In addition, the allocations of channel use for all sources are full without idle within D_t, i.e., $\omega_{kd_t}^{(i)} = 1, \forall k, i$, and d_t.

$r_m^{(i)}$ (r_m for any chunk) denotes the total number of linearly independent codeword combinations correctly received at destination m for chunk i, including the transmission stage from the sources to the relays since the direct link transmissions from the sources to the destinations might be also considered within this stage. Hence, $r_m^{(i)}$ is the rank of matrix $\mathbf{C}_m^{(i)} \in \mathbb{F}_q^{D^{(i)} \times r_m^{(i)}}$ which is formed by $r_m^{(i)}$ linearly independent combined coding coefficient vectors of the codeword combinations received at destination m for chunk i. As in the works of [30, 31], in this work, $\rho_m(r_m)$ denotes the empirical probability distribution of r_m where $r_m \in \{0, 1, \cdots, D_{\max}\}$, and $D_{\max} = \max\{D^{(i)}, \forall i\}$. Hence, $\sum_{r_m=0}^{D_{\max}} \rho_m(r_m) = 1$.

On the other hand, in this work, $\theta_{km}^{(i)}$ (θ_{km} for any chunk) denotes the rank of the part of the matrix $\mathbf{C}_m^{(i)}$ from row $1+\sum_{k'=1}^{k-1} d_{k'}^{(i)}$ to row $\sum_{k'=1}^{k} d_{k'}^{(i)}$. $\theta_{km}^{(i)}$ is defined as the participation factor of source k in $\mathbf{C}_m^{(i)}$, i.e., in the received codeword combinations for chunk i at destination m. In addition, $\lambda_{km}(\theta_{km})$ denotes the empirical probability distribution of θ_{km}, where $\theta_k \in \{0, 1, \cdots, d_{k\max}\}$, and $d_{k\max} = \max\{d_k^{(i)}, \forall i\}$. Hence, $\sum_{\theta_{km}=0}^{d_{k\max}} \lambda_{km}(\theta_{km}) = 1$. In addition, because $r_m^{(i)}$ and $\theta_{km}^{(i)}$ for all k must satisfy the relationships $\sum_{k=1}^{K} \theta_{km}^{(i)} \geq r_m^{(i)}$ and $r_m^{(i)} \geq \max_{k=1}^{K} \left\{\theta_{km}^{(i)}\right\}$, thus there should be correlation between $\rho_m(r_m)$ and $\lambda_{km}(\theta_{km})$ for all k.

In addition, exploiting the empirical probability distributions for this work might enable the application with the other channel fading models, hence, the robustness of the proposed schemes can be ensured.

2.4 Summary

This chapter presented a brief introduction of the network coding technique employed in this work, the system model such as the considered scenario and the channel model and the approaches used in this work such as encoding process, forwarding process, empirical probability distributions.

Chapter 3

A Design of Overlapped Chunked Codes Over Compute-and-Forward

3.1 Introduction

This chapter proposes a design of overlapped chunked codes (OCC) for multi-source multi-relay single destination networks where a physical-layer network coding approach, compute-and-forward (CF) based on nested lattice codes (NLC), while the simultaneous transmissions from the sources to the relays are only studied, and the forwarding transmissions from the relays to the destination are assumed lossless. This code is called OCC/CF for this work. In this chapter, the direct link transmissions from the sources to the destinations are not considered. OCC is applied before NLC before transmitting for each source. Random linear network coding is applied within each chunk. The purpose of OCC/CF is to improve the network performance, especially, in term of channel efficiency while the protocol overhead such as the transmission time of feedback, the loss of feedback are considered. In this design, a decodability condition to design OCC/CF is provided. The design is done by using the empirical probability distributions of the number of innovative codeword combinations and the empirical probability distribution of the participation factor of each source to the codeword combinations received for a chunk transmission. In addition, an OCC with a contiguously overlapping but non-rounded-end fashion is employed for the design. An estimation is done to select an allocation, i.e., the number of innovative blocks per chunk and the number of blocks taken from the previous chunk for all sources, that is expected to provide the desired performance, for example, maximum channel efficiency, or high decodability, or low computational complexity. The numerical results obtained from the simulations showed that the design of OCC/CF not only depends on the empirical rank probability distribution, but also on the empirical probability distribution of the participation factor, and the design overhead of OCC/CF is low when the probability distribution of the participation factor of each source is dense at the chunk size. In addition, the performance improvement of the transmission scheme employing OCC/CF comparing to a feedback-based transmission scheme depends on the size of feedback and feedback success rate. The lower design overhead of OCC/CF is to be the future work of this dissertation.

This chapter is organized as follows. The problems are stated in Section 3.2. Related works are described in Section 3.3. Section 3.4 describes the scenario of this chapter. Then, Section 3.5 describes the design of OCC/CF, mainly the condition of decodability. Section 3.6 talks about the application of an OCC to the design of OCC/CF by the applied decoding scheme. The estimation of the decodability for the designed OCC/CF is given in Section 3.7. The performance observation and the reference schemes are described in Section 3.8. Section 3.9 provides the numerical results and discussion. At the end, Section 3.10 gives the summary of this chapter.

3.2 Problem Statement

When CF is applied in multi-source multi-relay networks, in case that each source transmits a codeword simultaneously to the relays for each chunk, i.e., $D_t = 1$ and $\omega_{k1} = 1$ for all k, and each relay computes the superimposed codeword to obtain the codeword combinations to forward to the destination, the destination can recover the original blocks of all sources from the codeword combinations forwarded from the relays if it receives enough linearly independent correct codeword combinations. The cooperation between the destination and the relays can provide the opportunity for the destination to collect or to select linearly independent codeword combinations with desired purposes such as obtaining the highest sum rate [41, 45] or obtaining the highest throughput [51] while the sources might also take part into, e.g., transmit power allocation . However, in some cases, codeword combinations are not qualified to be forwarded to the destination, i.e., considered as unsuccessfully received, and the codeword combinations at the different relays might be linearly dependent on each other since all relays compute the superimposed codewords independently. Retransmission and feedback sending back to the sources will be needed if the relays cannot provide enough linearly independent codeword combinations to the destination to recover the original blocks of all sources. However, feedback might be lost, and it takes some delay in reaching the sources, hence, it causes some wasted channel use and energy consumption.

In the traditional communication network, when a block or a packet is correctly received by the destination, a feedback, i.e., an ACK, is sent back to the source to manage the next transmission. Negative ACK (NACK) is used to inform about the unsuccessful reception of a sent block. However, if feedback is lost or its transmission time is significant, e.g., the transmissions between ground stations and satellite, then the protocol overhead can affect the end-to-end network performance, especially for the lossy multi-hop multi-source wireless networks because feedback needs to be forwarded via the intermediate nodes to the sources.

On the other hand, RLNC can be applied to reduce the protocol overhead since an ACK is needed when the receiver can decode the received coded blocks. However, if the number of input blocks is large, then the encoding and decoding computational complexity which depends on the number of input blocks, especially decoding complexity, will be high and not practical. The large number of blocks is grouped into disjoint generations or chunks [22, 23], then the computational complexity can be reduced. Nevertheless, when the number of blocks per chunk, i.e., size of the chunk, is too small, the protocol overhead is still significant. If the BSR of the transmission link is known, then the sources can transmit each chunk with an expected number of coded blocks, and feedback can be avoided [24]. However, if BSR is not constant, then there will be some chunks that are not decodable. To deal with this problem, the works in [27, 57] proposed OCC, where a block can belong to more than one chunk. A decoded chunk can be used to help to decode the other undecodable chunks by back-substitution, i.e., blocks from decoded chunks are substituted into the undecodable chunks that also have them as input blocks. The other designs of OCCs and those of the codes similar to OCC were proposed then such as in the works of [28–31]. These designed codes are mainly for single flow transmission or multicast transmission, i.e., the transmission of a source data. Up to the present, there is no design of OCC for the data transmission in multi-source multi-relay networks.

This work considers the design and the application of OCC to the data transmissions in multi-source multi-relays networks where CF based on NLC is employed. The designed OCC is denoted by OCC/CF in this chapter. The aim is to investigate the advantage of OCC/CF over a feedback-based transmission scheme. In addition, low computational complexity is considered such that the proposed work is applicable to low specification wireless nodes, e.g., wireless sensor nodes. This work considers varying channel states where only receivers have knowledge of channel coefficients. The blocks of each source message are grouped into chunks. RLNC is done within each chunk before encoding with NLC. Only the transmissions from the sources to the relays are considered. The challenge to apply OCC in a multi-source multi-relay network is how to design OCC/CF such that the decodability of each chunk of all sources at the destination is ensured or the desired network performance, for example, maximum channel efficiency, is achieved.

3.3 Related Works

To ensure that the destination can recover the original blocks of all sources, applying different message rates were studied in the works of [11], and applying different transmit powers were studied in the works of [44, 51, 52]. However, these works mostly rely on the correctness of the estimation of the channel states at the senders. In case of fast varying channel states, the channel states for transmitting a chunk would be not steady, then there will still be the blocks of some chunks could not be recovered. If applying the worst case, then the network performance would not be optimal due to high compensation, although the feedback might not be needed.

By considering the case of symmetric message rate and symmetric transmit power, in the single flow transmission, to complete the message transmission without the need for feedback, the code to be mentioned would be rateless code where the number of coded blocks is unlimited, and the transmitter keeps sending the coded blocks until the receiver can recover all original blocks. Fountain code [26] is an erasure code and a rateless code. The feature of fountain code is low computational complexity in encoding and decoding processes since they are done in the binary field, i.e., $\mathbb{F}_2$. Fountain codes includes Luby

Transform (LT) code [58], Raptor code [59] and online code [60]. The decodability for LT code depends on the degree distribution, which is determined based on the soliton distribution. Degree is the number of input blocks to generate a coded block. The input blocks for each coded block are randomly selected. Raptor code applied the precoding process before encoding such that while a fraction of coded blocks are received, then all original blocks are recoverable. Online codes applied a precoding process for the distributed networks. The decoding process, while employing fountain codes, starts when at least an one-degree coded block, i.e., plain block, exists and stops when there are no more one-degree coded blocks. The decoded blocks are back-substituted into the new received coded blocks which also have them as input blocks. The application of the inactivation decoding method [49] was studied in the work [50] for the decoding process of LT code and Raptor code to reduce the decoding complexity because the transfer matrix of the received coded blocks, i.e., the coding coefficient matrix of the received coded blocks, is a sparse matrix.

For RLNC, each element of the coding coefficient matrix of the sending coded blocks are randomly drawn from a finite field $\mathbb{F}_q$ (normally, q is enough large, e.g., $q = 2^8$). The linear independence between coded blocks with RLNC is higher than with sparse network coding (the generated coding coefficient matrix of the coded blocks is a sparse matrix) especially in lossy communication networks, but the computational complexity of RLNC is higher. RLNC was employed within each chunk for OCC proposed in the work of [57] where two overlapping fashion were given: rectangular grid code and diagonal grid code. The number of chunks is finite, but the decodability of received chunks was not clearly analyzed. The overlapping fashion of OCC in the work of [27] is contiguous and in rounded-end fashion. The decodability is analyzed with chunk size, the number of contiguously overlapped blocks and the number of received coded blocks. However, achieving high decodability, i.e., the probability that a chunk is decodable, requires a large chunk size, which can make the computational complexity more significant. A small sized chunk was analyzed then in their later work [28]. However, the decoding process will start when the receiver has collected a sufficient number of coded blocks of all chunks in the worst case, and there might be no decodable chunks directly, i.e., without aid from the other decoded chunks. Then, higher storing ability at the receiver is required, and the decoding complexity is still significant. The design of OCC with the other overlapping fashion was proposed in the work of [29] where the overlapped blocks, i.e., the blocks taken from the other chunks, are randomly selected. Although the performance in decodability is better than OCC with the contiguously overlapping fashion [27], the decoding process still might start when a sufficient number coded blocks are received.

Batched sparse (BATs) codes proposed in the work of [30] inherit the feature of rateless code by employing fountain code as the outer code (chunk size obeys a degree distribution) and random linear network code as the inner code (RLNC is employed within each chunk). The degree distribution is determined using the empirical rank distribution to obtain the optimal performance in achievable rate. The decoding process starts when there is at least a decodable chunk, and back-substitution is done then. The inactivation decoding method might be applied when there are no more decodable chunks. The other design, which also employs the empirical rank distribution, is in the work of [31], where chunk size is fixed. Two degree distributions are defined, and a degree distribution is determined when another degree distribution is fixed to obtain the optimal achievable rate.

This work provides the design of OCC/CF with a condition of decodability, which might be applied with the designs of codes for single flow transmission, which are described above to apply in multi-source multi-relay networks. This work employs an OCC in a contiguously overlapping fashion to design OCC/CF because it is simple to determine which allocation for each source to obtain the desired performance since there are only two variables to be determined for each source. Although its performance in rate (channel efficiency, for this work) is not higher than the other designs of codes for single flow transmission, it has a potential to reduce the storage overhead and the computational complexity to suit its application with a low specification wireless node in multi-source multi-relay networks. The contributions of this work in the design of OCC/CF are as follows:

- analyzing the decodability for chunks received at the destination to design OCC for each source and providing a decodability condition to design OCC/CF;
- based on the condition of decodability, designing OCC/CF by employing an OCC with a contiguously overlapping, but non-rounded-end fashion at each source. The design is done by using the empirical rank distribution and the probability distribu-

tion of the participation factor of each source to the received codeword combinations per chunk transmission.;

- providing a decoding scheme based on the feature of the employed OCC. The decoding scheme considers the other opportunity of starting decoding besides back-substitution, the combination of chunks. The decoding complexity is bounded by the maximum number of combined chunks, and the storing overhead can be reduced;
- estimating the performance of the designed OCC/CF by following the decoding scheme and using table lookup for all allocations, i.e., the number of innovative blocks per chunk and the number of contiguously overlapped blocks for each source. The estimation is to determine which allocation can provide the desired performance such as high decodability, highest channel efficiency and acceptable decoding complexity.

3.4 Scenario and System Model

This chapter takes a scenario of a K sources L relays single destination network, i.e., $M = 1$. The direct links from the sources to the destination are not considered, and only the transmissions from the sources to the relays are considered. All sources apply the NLC with the same message rate $R = \log_2 q$, i.e., symmetric rate.

This chapter assumes that the transmissions from the relays to the destination are lossless, hence, the employed empirical probability distributions at the destination are the same as those provided by the relays. In this chapter, each relay generates only one codeword combination, i.e., $r_{ld_t}^{(i)} \leq 1$. This codeword combination is forwarded to the destination if it satisfies Condition (2.13). In addition, the allocation of channel use for each source is full, i.e., $\omega_{kd_t}^{(i)} = 1, \forall k, i$, and d_t. Since there is only one destination, the notation of the rank probability distributions $\rho_m(r_m)$ and $\lambda_{km}(\theta_{km})$ are simplified to $\rho(r)$ and $\lambda_k(\theta_k)$, respectively, for convenience. Hence, $r^{(i)}$ is the rank of matrix $\mathbf{C}^{(i)} \in \mathbb{F}_q^{D^{(i)} \times r^{(i)}}$, which is a set of $r^{(i)}$ linearly independent vectors taken from $L \cdot D_t$ vectors $\left[\mathbf{c}_{11}^{(i)}, \cdots, \mathbf{c}_{L1}^{(i)}, \mathbf{c}_{12}^{(i)}, \cdots, \mathbf{c}_{L2}^{(i)}, \cdots, \mathbf{c}_{L \cdot D_t}^{(i)}\right]$, where $\mathbf{c}_{ld_t}^{(i)}$ is the simplified notation of the combined coding coefficient vector of the codeword combination received at d_t slot at relay l for chunk i since $\iota \in \{0, 1\}$. In this chapter, $\rho(r)$ and $\lambda_k(\theta_k)$ are employed without considering the correlation between them.

3.5 Design of Overlapped Chunked Code over Compute-and-Forward

3.5.1 Decodability

The original blocks of all sources for chunk i are recoverable if there are $D^{(i)}$ linearly independent received codeword combinations for chunk i, i.e., $r^{(i)} = D^{(i)}$. If the channel state is stable, i.e., $r^{(i)}$ is constant for all i, all chunks can be decoded with a suitable value of D_t without the need for feedback from the destination. However, with the unstable channel state, $r^{(i)}$ varies with different chunks. Hence, without the aid of feedback, there are some chunks that are undecodable. In this chapter, $r_m^{(i)}$ is simplified to $r^{(i)}$, and r for any chunk. Then, the empirical probability distribution of r is simplified to $\rho(r)$ where for $r \in \{0, 1, \cdots, D_{\max}\}$, and $D_{\max} = \max\{D^{(i)}, \forall i\}$.

In order to analyze the decodability, in this chapter, p_d and p_k denote the probabilities that chunk i is decodable, i.e., $r^{(i)} = D^{(i)}$ and $\theta_k^{(i)} = d_k^{(i)}$, respectively, when employing an OCC which is designed by using $\rho(r)$ and $\lambda_k(\theta_k)$, respectively, in single-transmission flow, i.e., transmission from a source to a relay via an orthogonal channel. The overlapping fashions of OCCs corresponding to $\rho(r)$ and $\lambda_k(\theta_k)$ are the same.

This chapter considers the case that $K \geq L$, $D_t = d_{\max}$, and source k generates D_t coded blocks by the RLNC encoder with $d_k^{(i)}$ linearly independent coded blocks for chunk i and all k, i.e., $\left[\boldsymbol{\chi}_{k1}^{(i)}, \boldsymbol{\chi}_{k2}^{(i)}, \cdots, \boldsymbol{\chi}_{kD_t}^{(i)}\right] \in \mathbb{F}_q^{d_k^{(i)} \times D_t}$ should be pseudorandom to ensure the linear independence between coded blocks since q is not large. When employing OCC/CF, the codeword combinations of chunk i are recoverable at the destination if there are $D^{(i)}$ received linearly independent codeword combinations, i.e., $r^{(i)} = D^{(i)}$. To determine the probability that a chunk is decodable, this work studies two special cases as below:

- Case I: the combination coefficient vector computed at relay l at the d_t-th slot for chunk i, $\boldsymbol{\beta}_{ld_t}^{(i)} = \left[\beta_{1ld_t}^{(i)}, \beta_{2ld_t}^{(i)}, \cdots, \beta_{Kld_t}^{(i)}\right]$, is a unit vector, i.e., only an element of $\boldsymbol{\beta}_{ld_t}^{(i)}$ is equal to one, and the others are zero;

- Case II: $\boldsymbol{\beta}_{ld_t}^{(i)}$ does not have zero elements; there are only D_t linearly independent codeword combinations, and they are only forwarded by a relay.

For Case I, $r^{(i)}$ can be written as $r^{(i)} = \sum_{k=1}^{K} \theta_k^{(i)}$. Hence, the decodability of each chunk only depends on the OCC design using $\lambda_k(\theta_k)$ for all k. In this case, the original blocks of each source can be recovered independently since every received codeword combination corresponds to the coded blocks from only one source. By assuming that chunk i for all sources is decodable, i.e., $r^{(i)} = D^{(i)}$, if the chunks of all sources are decodable, thus the probability that a chunk for all sources is decodable, p_{dec}, can be written as $p_{\text{dec}} = \prod_{k=1}^{K} p_k$, which is independent of p_d or $\rho(r)$. In other words, $p_d = 1$ for this case.

For Case II, since $D_t \geq \max\{d_k^{(i)}, 1 \leq k \leq K, \forall i\}$ and there are $d_k^{(i)}$ linearly independent coded blocks from source k for chunk i, hence $\theta_k^{(i)} = d_k^{(i)}$ for all i and all k. Thus, $p_k = 1$ for all k. This case assumes that chunk i is not decodable and there are $\nu_k^{(i)} \leq d_k^{(i)}$ blocks inside chunk i for source k with $k \in \{1, 2, \cdots, K\}$, which also belong to the other chunks. If these $\nu_k^{(i)}$ blocks have been already recovered with the decoded chunks, then there are still $\sum_{k=1}^{K}\left(d_k^{(i)} - \nu_k^{(i)}\right) = D^{(i)} - \sum_{k=1}^{K} \nu_k^{(i)}$ to recover for chunk i. From another point of view, it is equivalent to the case that matrix $\mathbf{C}^{(i)}$ has $\nu_k^{(i)}$ eliminated rows, which are between row $1 + \sum_{k'=1}^{k-1} d_{k'}^{(i)}$ and row $\sum_{k'=1}^{k} d_{k'}^{(i)}$, and becomes a $\left(D^{(i)} - \sum_{k=1}^{K} \nu_k^{(i)}\right) \times D_t$ matrix, $\mathbf{C}'^{(i)}$. Since $\boldsymbol{\chi}_{kd_t}^{(i)}$ is randomly drawn from $\mathbb{F}_q^{d_k^{(i)}}$ for $d_t \in \{1, 2, \cdots, D_t\}$, hence $\mathbf{C}'^{(i)}$ can be approximately also drawn. In addition, $\mathbf{C}'^{(i)}$ can be approximately obtained by eliminating $\sum_{k=1}^{K} \nu_k^{(i)}$ rows from a $D^{(i)} \times D_t$ matrix, which is randomly drawn from $\mathbb{F}_q^{D^{(i)} \times D_t}$. It looks like that $\sum_{k=1}^{K} \nu_k^{(i)}$ blocks are back-substituted into a chunk i when employing OCC in single flow transmission. Then, the decodability of each chunk when employing OCC/CF is the same as when employing OCC designed using $\rho(r)$ in single-flow transmission. Hence, in this case, $p_{\text{dec}} = p_d$. With Case II, the feature is that already recovered $\nu_k^{(i)} \leq d_k^{(i)}$ blocks can be perfectly back-substituted into chunk i, i.e., without waste.

In contrast, for the other case, by taking $\theta_k^{(i)} < d_k^{(i)}$ and $\nu_k^{(i)} = d_k^{(i)} - \theta_k^{(i)}$ for example, these recovered blocks can successfully increase the number of linearly independent received coded blocks in chunk i if they are linearly independent of the existing received coded blocks in chunk i. In addition, the value of $\nu_k^{(i)}$ should be appropriately selected by using $\lambda_k(\theta_k)$ or $\theta_k^{(i)}$ for all i. From the point of view of $\mathbf{C}^{(i)}$, an example taking $K = 2$, $d_1^{(i)} = 2$, $d_1^{(i)} = 2$, $D_t = 3$, $\nu_1^{(i)} + \nu_2^{(i)} = 2$ (according to the OCC design using $\rho(r)$)

and $q = 7$ is shown in Figure 3.1. In Figure 3.1a,b, $r^{(i)} = 3$, $\theta_1^{(i)} = 2$ and $\theta_2^{(i)} = 2$ are given. By taking $\nu_1^{(i)} = 1$ and $\nu_2^{(i)} = 1$, then there are five linearly independent blocks in chunk i after back-substitution, i.e., chunk i is decodable, with four out of six chances. On the other hand, if taking $\nu_1^{(i)} = 0$, $\nu_2^{(i)} = 2$ as in Figure 3.1b, then chunk i is decodable with all three possibilities. Therefore, a suitable selection of $\nu_1^{(i)}$ and $\nu_2^{(i)}$ can provide better performance for OCC/CF. Hence, the OCC design using $\lambda_k(\theta_k)$ for all k is needed. In Figure 3.1c, $r^{(i)} = 3$, $\theta_1^{(i)} = 2$ and $\theta_2^{(i)} = 3$ are given. By taking any two different recovered blocks, chunk i is decodable with nine of ten chances. The undecodable outcome should be caused by the selection of $\nu_1^{(i)} + \nu_2^{(i)} = 2$, i.e., the OCC design using $\rho(r)$. Figure 3.1c represents Case II where $\theta_1^{(i)} = d_1^{(i)} = 2$, $\theta_2^{(i)} = d_2^{(i)} = 3$.

For the general case, by combining the two cases above, the effective probability that each chunk is decodable when OCC/CF is applied, denoted by p_{deff}, can be approximately obtained by:

$$p_{\text{deff}} = p_d \cdot \prod_{k=1}^{K} p_k. \tag{3.1}$$

From Condition (3.1), it seems that the design of OCC for multi-source multi-relay networks, i.e., OCC/CF, becomes multiple design of OCC in single-flow transmission while satisfying the content constraint of each chunk for all sources, e.g., $D^{(i)} = \sum_{k=1}^{K} d_k^{(i)}$. Hence, the overlapping fashion of OCC in each design of OCC in single-flow transmission must be the same for a convenient design.

On the other hand, for the case that $K > L > 1$, the values of D_t and $d_k^{(i)}$ for $k \in \{1, 2, \cdots, K\}$ should be selected appropriately such that any chunk i can be decoded by itself, i.e., directly with $r^{(i)} = D^{(i)}$. For example, if taking $d_1^{(i)} = d_2^{(i)} = \cdots = d_K^{(i)} = d_{\max}$ for all i, then $d_{\max}$ should be chosen as a multiple of L, and $D_t \geq \frac{K \cdot d_{\max}}{L}$ to ensure that there is at most $L \cdot D_t$ codeword combinations to recover $K \cdot d_{\max}$ original blocks. For the case that $1 < K \leq L$, D_t can be taken by $D_t \geq d_{\max}$ for any value of $d_k^{(i)} > 0$, $\forall k$.

3.5.2 Channel Efficiency

In this chapter, channel efficiency is defined as the ratio of the total number of decoded blocks from all sources to the total transmission time (the total number of time slots for OCC/CF or for the transmission schemes without the need for feedback from the relays)

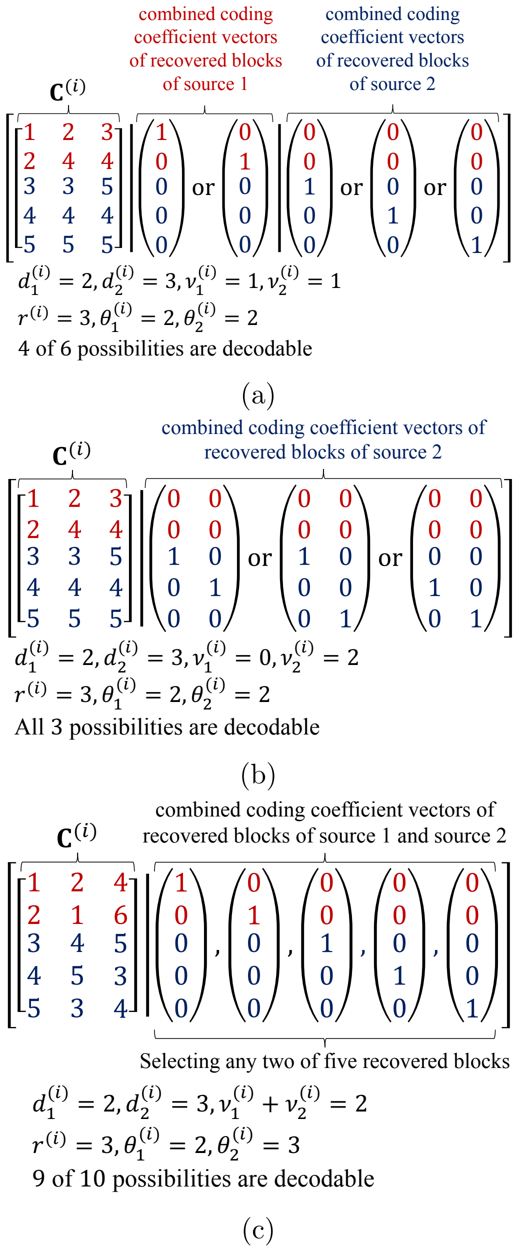


Figure 3.1: Example from point of view of the combined coding coefficient matrix. (**a**) $\nu_1^{(i)} = 1, \nu_2^{(i)} = 1, \theta_1^{(i)} = 2, \theta_2^{(i)} = 2$; (**b**) $\nu_1^{(i)} = 0, \nu_2^{(i)} = 2, \theta_1^{(i)} = 2, \theta_2^{(i)} = 2$; (**c**) $\nu_1^{(i)} + \nu_2^{(i)} = 2, \theta_1^{(i)} = 2, \theta_2^{(i)} = 3$. The decodability of chunk i is given with different possibilities of recovered blocks.

taken from the sources to the relays. η and η_{eff} denote the channel efficiency corresponding to p_d and p_{deff}, respectively.

For a K-source L-relay network, the ideal value of channel efficiency, which is obtained with lossless transmission and without linear dependence between codeword combinations for this scenario, is $\min\{K, L\}$. Thus, for $L = 1$ and $r_{ld_t}^{(i)} \leq 1$, the channel efficiency is like in the case of single flow transmission via an orthogonal channel. Hence, applying an orthogonal channel might be a better option, i.e., the channel use allocation should be modified to obtain better performance. This work only considers the case that $L > 1$ while $r_{ld_t}^{(i)} \leq 1$.

On the other hand, $\bar{r}$ denotes $\sum_{r=1}^{D_{\max}} r \cdot \rho(r)$, and $\bar{\eta}$ denotes $\dfrac{\bar{r}}{D_t}$. $\bar{\eta}$ is called channel capacity in this work, i.e., the upper bound of η_{eff}. Many OCC designs in single-flow transmission try to obtain η_{eff} close to $\bar{\eta}$. In this work, the design overhead is defined as the gap between η_{eff} and $\bar{\eta}$.

3.6 Design with Contiguously Overlapped Chunked Code

3.6.1 Encoding

This work applies an OCC in a contiguously overlapping fashion, which is similar to the works of [27, 28], but not in a rounded-end fashion to the design of OCC/CF in a multi-source multi-relay network where CF based on NLC is employed. The applied overlapping fashion is shown in Figure 3.2. In this fashion, for source k and each chunk, there are $\mu_k > 0$ innovative blocks, i.e., linearly independent blocks if comparing to the blocks of the other chunks, and there are γ_k overlapped blocks between two contiguous chunks. Hence, there are $d_k^{(i)} = \mu_k + \gamma_k^{(i)}$ blocks inside chunk i for source k, where $\gamma_k^{(1)} = 0$ and $\gamma_k^{(i)} \neq 0$ is constant for chunk $i > 1$, since it is not the rounded-end fashion. d_k denotes $d_k^{(i)}$ for $i > 1$, for convenience.

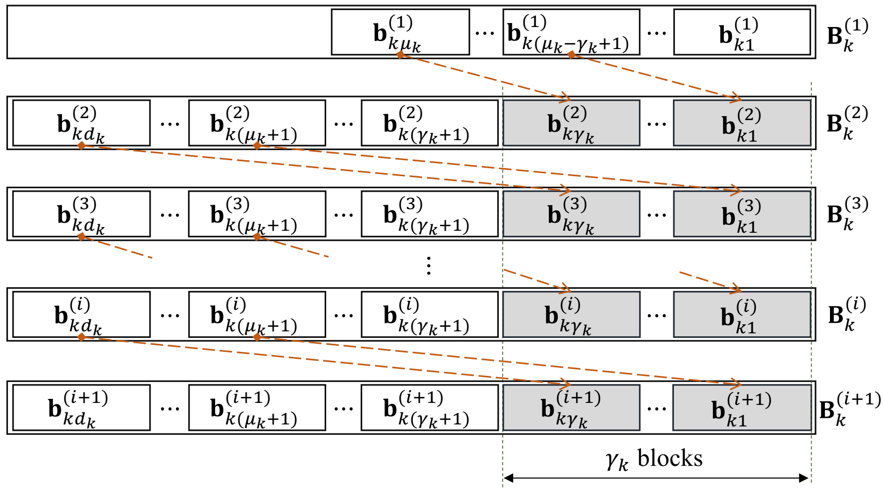


Figure 3.2: Applied overlapping fashion of overlapped chunked codes (OCC) for source k. The blocks in grey for chunk i are the blocks taken from the previous chunk, i.e., chunk $i-1$.

In addition, μ and γ are defined as $\sum_{k=1}^{K} \mu_k$ and $\sum_{k=1}^{K} \gamma_k$, respectively. There are $\min\{D_t, d_k^{(i)}\}$ linearly independent coded blocks among D_t coded blocks for chunk i and source k, where $\left[\boldsymbol{\chi}_{k1}^{(i)}, \boldsymbol{\chi}_{k2}^{(i)}, \cdots, \boldsymbol{\chi}_{kD_t}^{(i)}\right] \in \mathbb{F}_q^{d_k^{(i)} \times D_t}$ should be pseudorandomly generated to achieve this goal.

3.6.2 Decoding

The feature of OCC is that a decoded chunk can help the other undecodable chunks in decoding by using back-substitution (b.s). The recovered blocks of the decoded chunk are substituted into the undecodable chunks that consist of the same blocks, i.e., the overlapped blocks. Thus, the number of linearly independent codeword combinations of the back-substituted chunks might be increased, and it depends on the value of q and the pairs (μ_k, γ_k) for all k [61]. With the OCC employed in this work, for chunk i, left back substitution (l.b.s) and right back-substitution (r.b.s) denote b.s by the decoded neighboring chunk on the left, i.e., chunk $i-1$, and on the right, i.e., chunk $i+1$, respectively.

In addition to b.s, this work considers the other decoding opportunity, called combination of chunks (co.cs) for the applied OCC. co.cs combines the contiguous undecodable chunks into the form of chain of chunks (ch.cs) with length $\phi \geq 1$, where ϕ is the number of the combined chunks. The decoding process can start without the need for at least an already decoded neighboring chunk as with b.s. The form of combined coding coefficient matrix of ch.cs, $\mathbf{C}_c$, is shown in Figure 3.3.

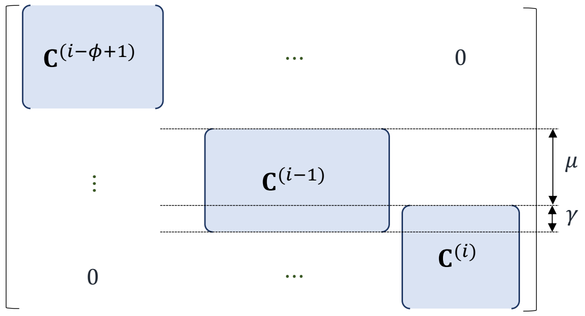


Figure 3.3: The form of combined coding coefficient matrix of the chain of chunks with length ϕ.

A ch.cs is decodable if the rank of $\mathbf{C}_c$, rank $(\mathbf{C}_c)$, is equal to the total number of original blocks inside that ch.cs, which is denoted by r_{ch}. A ch.cs is considered as a directly undecodable ch.cs without waiting to receive a new chunk if rank $(\mathbf{C}_c)$ is lower than a threshold value denoted by r_{th}. r_{ch} and r_{th} are determined as described in Algorithm 1 by using the feature of the applied OCC. Chunk i can participate in the co.cs process if $r_p^{(i)} \leq r^{(i)} \leq D^{(i)} - 1$, where $r_p^{(i)}$ is determined as described in Algorithm 2. In Algorithms 1 and 2, l.b.s.s(i) and r.b.s.s(i) refer to the state of l.b.s and r.b.s, respectively, for considered chunk i. l.b.s.s(i) and r.b.s.s(i) declare whether undecodable chunk i has not been back-substituted by its left neighboring decoded chunk, i.e., chunk $i-1$, and by its right neighboring decoded chunk, i.e., chunk $i+1$, respectively. $r_p^{(t)} = 0$ means that chunk t cannot participate in co.cs, and its decodability depends on $r^{(t)}$. The process of co.cs is described in Algorithm 3, where d.s(ch.cs) refers to the decodability state of the currently obtained ch.cs.

The decoding process can be done as described in Algorithm 4, where d.s(t) and d.s($t - \phi + 1 : t$) declare whether chunk t and ch.cs, combining from chunk t back to $t - \phi + 1$, respectively, are decoded or not. The decoding process for a ch.cs with length $\phi \geq 2$ can be done by using the inactivation decoding method [49] in order to reduce the decoding complexity. However, Gaussian elimination is applied for the decoding process in this work, and applying the inactivation decoding method is considered as a future work.

The chunks that are considered as directly undecodable chunks without waiting for the next received chunks can become decodable by the aid of feedback from the destination back to the sources. Otherwise, they can be discarded in order to reduce the storage overhead if they do not affect the recovery of all blocks, i.e., the original message. The latter option can be achievable by applying precoding before OCC at each source. With precoding, the original blocks can be recovered when a fraction of all coded blocks is decoded [23].

Algorithm 1 Determining r_{th} and r_{ch} of a co.cs starting from chunk t with length ϕ.

if $t - \phi + 1 = 1$ **then**
    $r_{\text{ch}} = \phi \cdot \mu$
    if r.b.s.s(t) = **true then**
        $r_{\text{th}} = \phi \cdot \mu$
    else
        $r_{\text{th}} = \phi \cdot \mu - \gamma$
    end if
else
    $r_{\text{ch}} = \phi \cdot \mu + \gamma$
    if l.b.s.s($t - \phi + 1$) = **true and** r.b.s.s(t) = **true then**
        $r_{\text{th}} = \phi \cdot \mu + \gamma$
    else if l.b.s.s($t - \phi + 1$) = **true and** r.b.s.s(t) = **false then**
        $r_{\text{th}} = \phi \cdot \mu$
    else if l.b.s.s($t - \phi + 1$) = **false and** r.b.s.s(t) = **true then**
        $r_{\text{th}} = \phi \cdot \mu$
    else if l.b.s.s($t - \phi + 1$) = **false and** r.b.s.s(t) = **false then**
        $r_{\text{th}} = \phi \cdot \mu - \gamma$
    end if
end if

Algorithm 2 Determining $r_p^{(t)}$ for chunk t.

if $t = 1$ **then**
  if r.b.s.s(t) = **true then**
    $r_p^{(t)} = 0$
  else
    $r_p^{(t)} = \mu - \gamma + 1$
  end if
else
  if l.b.s.s(t) = **true and** r.b.s.s(t) = **true then**
    $r_p^{(t)} = 0$
  else if l.b.s.s(t) = **true and** r.b.s.s(t) = **false then**
    $r_p^{(t)} = \mu + 1$
  else if l.b.s.s(t) = **false and** r.b.s.s(t) = **true then**
    $r_p^{(t)} = \mu + 1$
  else if l.b.s.s(t) = **false and** r.b.s.s(t) = **false then**
    $r_p^{(t)} = \mu - \gamma + 2$
  end if
end if

Algorithm 3 Combination of chunks.

Starting from chunk i
Taking $t = i$, d.s(ch.cs) = **false**, $\phi = 1$, $\mathbf{C}_c = \mathbf{C}^{(t)}$
if $\text{rank}\left(\mathbf{C}_c\right) = r_{\text{ch}}$ **then**
  Update d.s(ch.cs) = **true**
  return d.s(ch.cs),ϕ
else
  if $t = 1$ **or** $\text{rank}\left(\mathbf{C}_c\right) < r_p^{(t)}$ **or** l.b.s.s$\left(t\right)$ = **true then**
    return d.s(ch.cs),ϕ
  end if
end if
while $t - 1 > 0$ **and** $r_p^{(t-1)} \leq r^{(t-1)} \leq D - 1$ **and** r.b.s.s$\left(t-1\right)$ = **false do**
  Update $\phi = \phi + 1$, $\mathbf{C}_c = \left[\mathbf{C}_c, \mathbf{C}^{(t-1)}\right]$
  if $\text{rank}\left(\mathbf{C}_c\right) = r_{\text{ch}}$ **then**
    Update d.s(ch.cs) = **true**
    return d.s(ch.cs),ϕ
  else
    Update $t = t - 1$
  end if
end while

Algorithm 4 Decoding process.

```
1:  Obtaining the codeword combinations of chunk i
2:  Take t = i
3:  if r^(t) = D and d.s(t) = false then
4:      Conduct decoding and update d.s(t) = true, t = t − 1
5:      Go to 3
6:  else
7:      if t − 1 > 0 and d.s(t − 1) = true and l.b.s.s(t) = false then
8:          Conduct l.b.s and update l.b.s.s(t) = true
9:      end if
10:     if d.s(t + 1) = true and r.b.s.s(t) = false then
11:         Conduct r.b.s and update r.b.s.s(t) = true
12:     end if
13:     if r^(t) = D and d.s(t) = false then
14:         Conduct decoding and
            update d.s(t) = true, t = t − 1
15:         Go to 3
16:     else
17:         if l.b.s.s(t) = false and t − 1 > 0
            and d.s(t − 1) = false then
18:             Conduct co.cs to obtain ch.cs with length φ
19:             if d.s(ch.cs) = true then
20:                 Conduct decoding and
                    update d.s(t − φ + 1 : t) = true, t = t − φ
21:                 Go to 3
22:             end if
23:         end if
24:     end if
25: end if
26: Wait to receive the codeword combinations of chunk i + 1
```

3.6.3 Design with Applied Overlapped Chunked Code

The design of the applied OCC for multi-source multi-relay, i.e., the design of OCC/CF, is to determine an allocation $[(\mu_1, \gamma_1), (\mu_2, \gamma_2), \cdots, (\mu_K, \gamma_K)]$ with the desired p_{deff} or with the desired effective channel efficiency η_{eff}. For convenience, this work takes D_t as the maximum chunk size, i.e., $D_t \geq \max\{d_k, 1 \leq k \leq K\}$ and D_t can provide $\min\{\theta_k, 1 \leq k \leq K\} > 0$. If N_k is the total number of blocks for source k, the number of chunks that contain the blocks of all sources is equal to $\min\left\{\left\lfloor \frac{N_k}{\mu_k} \right\rfloor, 1 \leq k \leq K\right\}$.

The finite number of chunks for the applied OCC might cause high overhead in single-flow transmission if compared with the other codes such as in the works of [29–31, 62], which have a rateless feature. However, the design of the applied OCC in a multi-source multi-relay network might be simpler if compared with the other designs that determine the probability distribution of chunk size for all sources, for example. By taking the design of BATs codes [30] as an example, d_k is selected according to a determined degree distribution $\Psi_k = \{\psi_0, \psi_1, \cdots, \psi_{D_t}\}$ for each chunk. Determining Ψ_k for all k must consider the outputs of p_k, p_d and p_{deff}, while Ψ_k for all k needs to satisfy $\Psi_1 * \Psi_2 * \cdots * \Psi_K = \Psi$ [63], where Ψ is the degree distribution of $D = \sum_{k=1}^{K} d_k$ and sign $*$ refers to the discrete-time convolution operation. It becomes more complicated to determine Ψ_k for all k when K is large.

For the design of OCC/CF, d_k is fixed for all chunks except for the first chunk. Thus, there are $1 + 2 + \cdots + D_t = \dfrac{D_t\,(D_t + 1)}{2}$ candidates of (μ_k, γ_k) for source k and $\left[\dfrac{D_t\,(D_t + 1)}{2}\right]^K$ candidates of $[(\mu_1, \gamma_1)\,, (\mu_2, \gamma_2)\,, \cdots, (\mu_K, \gamma_K)]$ for all sources. It would be less if fixing $d_k = D_t$ and varying only γ_k for all k, as in the work [64], where there are only D_t candidates for each source and D_t^K candidates for all sources. However, the decoding complexity and the storage overhead at the destination might be high. For this work, the chunk size for each source is not large and bounded by D_t; however, it can be large, as in the work of [27]. A larger chunk size with a large number of overlapped blocks can improve the decodability of all chunks, but it can cause high computational complexity, especially decoding complexity and storage overhead at the destination. An undecodable chunk needs to wait for several new received chunks to start decoding, i.e., ϕ is large. For example, if taking $\mu = 14$, $\gamma = 18$, $\bar{r} = 15$ and $D_t < D$, where $D = \mu + \gamma = 32$, then there are no chunks that can be decoded by themselves, i.e., $r = D$ without using b.s or co.cs. The decoding process can only start by co.cs, where ϕ at least satisfies:

$$\phi \cdot \bar{r} \geq \phi \cdot \mu + \gamma, \tag{3.2}$$

hence, $\phi \geq 18$.

With a large chunk size, the decoding process rarely starts with back-substitution, i.e., the ch.cs with length $\phi = 1$. The decoding process only start with co.cs with large ϕ. In this work, the length ϕ is bounded for the purpose of low decoding complexity by providing more opportunities to conduct b.s and to reduce latency, as an undecodable chunk needs to wait to become decodable.

3.7 Estimation of Decodability

3.7.1 Overview

In order to select an appropriate allocation $[(\mu_1, \gamma_1), (\mu_2, \gamma_2), \cdots, (\mu_K, \gamma_K)]$ for the desired purpose, the estimation of p_{deff} (also η_{eff}) is done for each possible allocation. The estimations of p_k and p_d are conducted separately for each possible allocation. Then, p_{deff} is determined by (3.1), and η_{eff} is determined by:

$$\eta_{\text{eff}} = \mu \cdot p_{\text{deff}}. \tag{3.3}$$

In this work, the estimation is done by conducting table lookup and the accumulative sum of the probabilities that ch.cs with the maximum length $\phi_{\max}$ are decodable for all possible combinations of $\left[r^{(i-\phi+1)}, r^{(i-\phi+2)}, \cdots, r^{(i)}\right]$ and $\left[\theta_k^{(i-\phi+1)}, \theta_k^{(i-\phi+2)}, \cdots, \theta_k^{(i)}\right]$ to determine p_d and p_k, respectively, using $\rho(r)$ and $\lambda(\theta_k)$, respectively, for $\phi \in \{1, 2, \cdots, \phi_{\max}\}$ and $i > 1$. For convenience, only the estimation of p_d is described, and the estimation of p_k for all k can be done similarly.

At the start, it is assumed that an OCC with the fashion as in Figure 3.2 is applied from a sender to a receiver in single-flow transmission, and $\rho(r)$ with $r \in \{1, 2, \cdots, D_{\max}\}$ is the obtained empirical rank probability distribution. The chunk size D is selected from the range value of r, then the maximum number of linearly independent codeword combinations (coded blocks) received per chunk becomes D. $\rho(r)$ is updated to $\rho'(r')$ where $r' \in \{1, 2, \cdots, D\}$. This work estimates $\rho'(r')$ from $\rho(r)$ by:

$$\rho'(r') = \begin{cases} \rho(r), & \text{if } r' \in \{1, 2, \cdots, D-1\}. \\ \sum_{r=D}^{D_{\max}} \rho(r), & \text{if } r' = D. \end{cases} \tag{3.4}$$

For the estimation and for convenience, another four back-substitution states for a chunk are defined as below:

- not back-substituted state (n.b.s.s): equivalent to the event that l.b.s.s and r.b.s.s for a chunk are all false;

- half back-substituted state (h.b.s.s): equivalent to the event that one of l.b.s.s and r.b.s.s is true;
- full back-substituted state (f.b.s.s): equivalent to the event that l.b.s.s and r.b.s.s are all true;
- quasi-half back-substituted state (q.b.s.s): equivalent to n.b.s.s in the b.s process and equivalent to h.b.s.s in the co.cs process.

n.b.s.s(r'), for example, refers to a chunk that has r' linearly independent coded blocks, and its state is n.b.s.s. $\varrho_n(r')$, $\varrho_h(r')$, $\varrho_q(r')$ and $\varrho_f(r')$ denote the probabilities that a chunk has r' linearly independent coded blocks and has n.b.s.s, h.b.s.s, f.b.s.s and q.b.s.s, respectively. They satisfy the condition below.

$$\sum_{r'=1}^{D} \left[\varrho_n(r') + \varrho_h(r') + \varrho_q(r') + \varrho_f(r')\right] = 1. \tag{3.5}$$

Initially, $\varrho_n(r') = \rho'(r')$ and $\varrho_h(r') = \varrho_q(r') = \varrho_n(r') = 0$ for $r' \in \{1, 2, \cdots, D\}$ are given. The estimation here is to update $\varrho_n(r')$, $\varrho_h(r')$, $\varrho_q(r')$ and $\varrho_f(r')$ according to the decoding process for all values of r'. If $\varrho_d(r')$ denotes the probability that a chunk has r' linearly independent coded blocks after conducting the updating process, then:

$$\varrho_d(r') = \varrho_n(r') + \varrho_h(r') + \varrho_q(r') + \varrho_f(r'), \text{for } r' \in \{1, 2, \cdots, D\}. \tag{3.6}$$

In the updating process, the chunk with n.b.s.s, h.b.s.s and q.b.s.s is active, i.e., $\varrho_n(r')$, $\varrho_h(r')$ and $\varrho_q(r')$ are used to conduct the updating process, and the chunk with f.b.s.s is inactive, i.e., $\varrho_f(r')$ cannot be used to conduct the updating process and is only used in determining $\varrho_d(r')$. The updating process is to try transforming n.b.s.s(r') for all r' to the chunks with other states, i.e., to make $\varrho_n(r')$ tend to zero for all r'. At the end of the updating process, p_d is obtained by taking $p_d = \varrho_d(D)$.

The updating process is divided into two parts: b.s and co.cs, which are for $\phi = 1$ and for $2 \leq \phi \leq \phi_{\max}$, respectively. This work assumes that the estimation of decodability is done at the destination. The destination informs the desired allocation to the sources via feedback.

3.7.2 Combination of Chunks

A ch.cs with $\mathbf{C}_c$ as in Figure 3.3 with length ϕ, where $2 \leq \phi \leq \phi_{\max}$, is considered. The rank array of $\mathbf{C}_c$ is $\left[r'^{(i-\phi+1)}, r'^{(i-\phi+2)}, \cdots, r'^{(i)}\right]$, where $r'^{(t)}$ is the rank of $\mathbf{C}^{(t)}$ for $t \in \{i-\phi+1, i-\phi+2, \cdots, i\}$. A combination of $\left[r'^{(i-\phi+1)}, r'^{(i-\phi+2)}, \cdots, r'^{(i)}\right]$ for a ch.cs (simply combination for convenience) with length ϕ is considered as a possible combination to be taken into account in the estimation if it satisfies:

$$\sum_{t=i-\phi+1}^{i} r'^{(t)} \geq \phi\mu + \gamma, \tag{3.7}$$

and it does not contain any possible combination with length ϕ' inside, where $\phi' \in \{1, 2, \cdots, \phi - 1\}$. The ch.cs corresponding to a possible combination is decodable if $\text{rank}\left(\mathbf{C}_c\right) = \phi \cdot \mu + \gamma$. p_s denotes the probability that a combination can make the correspondent ch.cs decodable and $q_s = 1 - p_s$. All possible combinations and their p_s are obtained by conducting a computation in MATLAB in this work and known by the destination where table lookup is done while doing the estimation.

Based on three different locations of a chunk in a ch.cs, the other three probabilities are defined as below.

- $\varrho_b\left(r'\right)$: the probability that a chunk that has r' linearly independent coded blocks can play the role of the beginning chunk, i.e., $t = i$. The beginning chunk can be with n.b.s.s or h.b.s.s or q.b.s.s.

- $\varrho_i\left(r'\right)$: the probability that a chunk that has r' linearly independent coded blocks can play the role of the intermediate chunk, i.e., $t \in \{i-\phi+2, i-\phi+3, \cdots, i-1\}$. The intermediate chunk only can be with n.b.s.s. Hence, $\varrho_i\left(r'\right) = \varrho_n\left(r'\right)$.

- $\varrho_e\left(r'\right)$: the probability that a chunk that has r' linearly independent coded blocks can play the role of the ending chunk, i.e., $t = i - \phi + 1$. The ending chunk can be with n.b.s.s, or h.b.s.s, or q.b.s.s. The beginning chunk and the ending chunk have a similar property because the combinations are reversible. Thus, only the beginning chunk is studied, and $\varrho_e\left(r'\right) = \varrho_n\left(r'\right) + \varrho_h\left(r'\right) + \varrho_q\left(r'\right)$ is taken.

Chunk t is decodable, i.e., ch.cs that contains chunk t and $t \in \{i-\phi+1, \cdots, i\}$ is decodable, by probability p_c, which is:

$$p_c = p_s \varrho_b(r'^{(i)}) \varrho_e(r'^{(i-\phi+1)}) \prod_{t=i-\phi+2}^{i-1} \varrho_i(r'^{(t)}). \tag{3.8}$$

When the beginning chunk $t = i$ is focused on, it must have $r'^{(t)} \geq \mu+1$, as in the case that one of l.b.s.s (t) and r.b.s.s (t) is true in Algorithm 2. For example, by taking $D = 18$ and $\gamma = 4$, a combination with rank array $[16, 14, 14, 17]$ is a possible combination where $r'^{(t)} = 16$ and $p_s \approx 0.9663$. However, the combination with rank array $[16, 15, 14, 17]$ is not a possible combination because it contains a possible combination with rank array $[15, 14, 17]$.

When the intermediate chunk $t \in \{i - \phi + 2, \cdots, i - 1\}$ is focused on, it must have $r'^{(t)} \geq \mu - \gamma + 2$, as in the case that l.b.s.s (t) and r.b.s.s (t) are all false in Algorithm 2. However, for $r'^{(t)} > \mu$, the possible combinations that have the same rank arrays as those of the possible combination when focusing on the beginning chunk with the same value of $r'^{(t)}$ are not included. In this estimation, the possible combinations that are both available while focusing on the beginning chunk and the intermediate chunk and have the same elements of the rank array, and the same focused chunk with n.b.s.s is applied once in the updating process with co.cs. This is because there is no constraint on the order of chunks in a ch.cs for Relation (3.8). For example, by taking $D = 18$ and $\gamma = 4$, two combinations with rank array $[16, 13, 15, 16]$ focusing on the chunk with $r'^{(t)} = 13$ and with rank array $[16, 15, 16]$ focusing on the chunk with $r'^{(t)} = 15$ are used because combinations with rank arrays $[13, 16, 15, 16]$ focusing on the chunk with $r'^{(t)} = 13$ and with rank array $[15, 16, 16]$ focusing on the chunk with $r'^{(t)} = 15$ are not possible combinations. However, the combination with rank array $[16, 13, 15, 16]$ focusing on the chunk with $r'^{(t)} = 15$ is not a possible combination because the combination with rank array $[15, 16, 13, 16]$ focusing on the chunk with $r'^{(t)} = 15$ is a possible combination.

The purpose of introducing q.b.s.s is described by the following example. By taking $D = 18$ and $\gamma = 3$, a combination with rank array $[17, 16, 14, 17]$ has $p_s \approx 0.9473$. However, it is not a possible combination since the combination with rank array $[17, 16]$ is a possible combination with $p_s \approx 0.8369$. Then, no matter how the rest of the decoding process (with b.s) is done, the combination with rank array $[17, 16, 14, 17]$ is decodable with

p_s less than 0.8369, which is lower than the real value. Hence, in order to fix this problem, q.b.s.s is introduced by assuming that the beginning chunk with n.b.s.s of an undecodable possible combination (with a probability of $q_c = q_s \varrho_b(r'^{(i)}) \varrho_e(r'^{(i-\phi+1)}) \prod_{t=i-\phi+2}^{i-1} \varrho_i(r'^{(t)})$) becomes a chunk with $r'^{(i)} = D-1$, but no longer with n.b.s.s, and with a state like h.b.s.s during the updating process with co.cs and with a state like n.b.s.s during the updating process with b.s. After introducing q.b.s.s, the ch.cs with the rank array $[17, 16, 14, 17]$ is decodable with a probability around 0.9498, which is close to the real value.

The relationship (transition) between four states of a chunk for the updating process with co.cs is shown in Table 3.1, where p_t refers to the transfer probability.

Table 3.1: Transition table between four chunk states for the updating process with the combination of chunks (co.cs). h.b.s.s, half back-substituted state; f.b.s.s, full back-substituted state; n.b.s.s, not back-substituted state; q.b.s.s, quasi-half back-substituted state.

Focused Chunk t with $r'^{(t)}$	**From**	**To**	$\varrho_b(r'^{(i)})$	p_t
$t = i$; $r'^{(t)} \geq \mu + 1$	h.b.s.s($r'^{(t)}$)	f.b.s.s(D)	$\varrho_h(r'^{(t)})$	p_c
$t = i$; $r'^{(i)} \geq \mu + 1$	h.b.s.s($r'^{(t)}$)	h.b.s.s($r'^{(t)}$)	$\varrho_h(r'^{(t)})$	q_c
$i - \phi + 2 \leq t \leq i - 1$; $r'^{(t)} \geq \mu - \gamma + 2$	n.b.s.s($r'^{(t)}$)	f.b.s.s(D)	$\varrho_e(r'^{(i)})$	p_c
$i - \phi + 2 \leq t \leq i - 1$; $r'^{(t)} \geq \mu - \gamma + 2$	n.b.s.s($r'^{(t)}$)	n.b.s.s($r'^{(t)}$)	$\varrho_e(r'^{(i)})$	q_c
$t = i$; $r'^{(t)} \geq \mu + 1$	n.b.s.s($r'^{(t)}$)	h.b.s.s(D)	$\varrho_n(r'^{(t)})$	p_c
$t = i$; $r'^{(t)} \geq \mu + 1$	n.b.s.s($r'^{(t)}$)	q.b.s.s($D - 1$)	$\varrho_n(r'^{(t)})$	q_c
$t = i$; $r'^{(t)} = D - 1$	q.b.s.s($r'^{(t)}$)	h.b.s.s(D)	$\varrho_q(r'^{(t)})$	p_c
$t = i$; $r'^{(t)} \geq \mu + 1$	q.b.s.s($r'^{(t)}$)	q.b.s.s($r'^{(t)}$)	$\varrho_q(r'^{(t)})$	q_c

The updating process with co.cs is done by applying the relationship in Table 3.1 and by following the concept of same increment same decrement, i.e., the decrement of the probability of a state (n.b.s.s or h.b.s.s or q.b.s.s) at a value of r' is the increment of probability of another state at the same value of r' or at the others. For example, by taking $D = 18$, $\gamma = 3$, $q = 7$, $\varrho_n(r') = \{0.1, 0.2, 0.4, 0.3\}$ and $\varrho_h(r') = \varrho_q(r') = \varrho_f(r') = \{0, 0, 0, 0\}$ for $r' \in \{15, 16, 17, 18\}$ and by considering a chunk with n.b.s.s and $r' = 15$ in a possible combination with rank array $[16, 15, 17]$, then $p_s \approx 0.8166$ and $p_c = 0.0065$ are obtained. Then, $\varrho_n(15)$ decreases by p_c, and $\varrho_f(18)$ increases by p_c. The updating process with co.cs can be done repeatedly for a number of iterations. $\varrho_n(r')$, $\varrho_h(r')$, $\varrho_q(r')$ and $\varrho_f(r')$ for all values of r' are updated after an iteration of the updating process with co.cs and are used for the next iteration.

3.7.3 Back-Substitution

The work in [61] provides the probability that ν recovered (overlapped) blocks successfully help undecodable chunk t that has already had $r'^{(t)}$ linearly independent coded blocks turn into a decodable chunk, where $r'^{(t)} + \nu \geq D$ and $\nu \in \{\gamma, 2 \cdot \gamma\}$. In this estimation, the increment of the number of linearly independent coded blocks in an undecodable chunk after conducting b.s by using ν recovered blocks is also considered. In addition, the probability distribution of the increment is used.

In order to obtain these data, a computation in MATLAB is conducted to obtain the rank probability distribution of matrix $\mathbf{C}^{(t)} \in \mathbb{F}_q^{D \times r'^{(t)}}$ with ν rows eliminated, i.e., a $(D-\nu) \times r'^{(t)}$ matrix; hence, the obtained rank $r_{\text{bs}}^{(t)}$ has range of values $\left\{0, 1, \cdots, r'^{(t)}_{\text{max}}\right\}$, where $r'^{(t)}_{\text{max}} = \min\{D-\nu, r'^{(t)}\}$. The probability distribution of $r_{\text{bs}}^{(t)}$ is denoted by $\rho_{\text{bs}}\left(r_{\text{bs}}^{(t)}\right)$. Since the initial rank of $\mathbf{C}^{(t)}$ is $r'^{(t)}$, then the probability distribution of the rank of the $(D-\gamma') \times r'^{(t)}$ matrix, denoted by $\rho'_{\text{bs}}\left(r'^{(t)}_{m}\right)$ where $r'^{(t)}_{\text{bs}} \in \{r'^{(t)}_{\text{min}}, \cdots, r'^{(t)}_{\text{max}}\}$ and $r'^{(t)}_{\text{min}} = \max\{0, r'^{(t)} - \nu\}$, is approximately obtained by:

$$\rho'_{\text{bs}}\left(r'^{(t)}_{\text{bs}}\right) = \left\{\rho_{\text{bs}}\left(r'^{(t)}_{\text{min}}\right), \cdots, \rho_{\text{bs}}\left(r'^{(t)}_{\text{max}}\right)\right\} \cdot \frac{\sum_{r_{\text{bs}}^{(t)}=0}^{r'^{(t)}_{\text{max}}} \rho_{\text{bs}}\left(r_{\text{bs}}^{(t)}\right)}{\sum_{r_{\text{bs}}^{(t)}=r'^{(t)}_{\text{min}}}^{r'^{(t)}_{\text{max}}} \rho_{\text{bs}}\left(r_{\text{bs}}^{(t)}\right)}. \tag{3.9}$$

The rank increment from the aid of ν overlapped blocks to undecodable chunk t is denoted by $r_{\text{in}}^{(t)}$; hence, $r_{\text{in}}^{(t)} = r'^{(t)}_{\text{bs}} + \nu - r'^{(t)}$, where $r_{\text{in}}^{(t)} \in \{0, 1, \cdots, \nu\}$. If $\rho_{\text{in}}\left(r_{\text{in}}^{(t)}\right)$ is the probability distribution of $r_{\text{in}}^{(t)}$, then $\rho_{\text{in}}\left(r_{\text{in}}^{(t)}\right) = \rho'_{\text{bs}}\left(r'^{(t)}_{\text{bs}}\right)$. $p_{\text{bs}} = \varrho_n(D) + \varrho_h(D)$ is taken as the fraction (probability) of the decoded chunks that can be used for b.s, and $q_{\text{bs}} = 1 - p_{\text{bs}}$. For the case that only half b.s (l.b.s or r.b.s) is executable, then $\nu = \gamma$. If full b.s is executable, then $\nu = 2 \cdot \gamma$. The transition table between four chunk states for the updating process with b.s is shown in Table 3.2. The focused chunk t has $\mathbf{C}^{(t)}$ with rank $r'^{(t)}$.

Table 3.2: Transition table between four chunk states for the updating process with b.s.

Half or Full b.s or N/A	From	To	p_t
Half	n.b.s.s$(r'^{(t)})$	h.b.s.s$(r'^{(t)} + r_{\text{in}}^{(t)})$	$2 \cdot p_{\text{bs}} \cdot q_{\text{bs}} \cdot \rho_{\text{in}}\left(r_{\text{in}}^{(t)}\right)$
Full	n.b.s.s$(r'^{(t)})$	f.b.s.s$(r'^{(t)} + r_{\text{in}}^{(t)})$	$p_{\text{bs}}^2 \cdot \rho_{\text{in}}\left(r_{\text{in}}^{(t)}\right)$
N/A	n.b.s.s$(r'^{(t)})$	n.b.s.s$(r'^{(t)})$	q_{bs}^2
Half	h.b.s.s$(r'^{(t)})$	f.b.s.s$(r'^{(t)} + r_{\text{in}}^{(t)})$	$p_{\text{bs}} \cdot \rho_{\text{in}}\left(r_{\text{in}}^{(t)}\right)$
N/A	h.b.s.s$(r'^{(t)})$	h.b.s.s$(r'^{(t)})$	q_{bs}
Half	q.b.s.s$(r'^{(t)})$	h.b.s.s$(r'^{(t)} + r_{\text{in}}^{(t)})$	$2 \cdot p_{\text{bs}} \cdot q_{\text{bs}} \cdot \rho_{\text{in}}\left(r_{\text{in}}^{(t)}\right)$
Full	q.b.s.s$(r'^{(t)})$	f.b.s.s$(r'^{(t)} + r_{\text{in}}^{(t)})$	$p_{\text{bs}}^2 \cdot \rho_{\text{in}}\left(r_{\text{in}}^{(t)}\right)$
N/A	q.b.s.s$(r'^{(t)})$	q.b.s.s$(r'^{(t)})$	q_{bs}^2

The updating process with b.s is done by applying the relationship of four chunk states shown in Table 3.2 and by following the concept of the same increment same decrement as in the updating process with co.cs. The updating process with b.s can be done repeatedly as with co.cs. After the end of each iteration of the updating process with b.s, p_{bs} is added to $\varrho_f(D)$, and it is updated according to the updated value of $\varrho_h(D)$ because $\varrho_n(D)$ becomes zero after the first iteration of the updating process with b.s.

3.7.4 Decoding Complexity

From the work in [30], to encode a block by applying RLNC from d_k input blocks required $\mathcal{O}(n \cdot d_k)$ finite field operation, where n is the number of symbols per block. By applying Gaussian elimination for the decoding process, decoding a chunk with size d_k requires $\mathcal{O}(d_k^2 + n \cdot d_k)$ finite field operations per block on average. Since decoding complexity is more significant than encoding complexity, thus only decoding complexity is discussed in this work.

With the applied decoding scheme, the main key to look at the decoding complexity is the mean length of ch.cs, $\bar{\phi}$. With OCC/CF, it is hard to estimate $\bar{\phi}$ since a successful b.s or co.cs does not depend on the selection of $[\mu, \gamma]$ alone, but actually on the selection of $[(\mu_1, \gamma_1), (\mu_2, \gamma_2), \cdots, (\mu_K, \gamma_K)]$. However, this work employs the OCC designed using $\rho(r)$, i.e., the selection of $[\mu, \gamma]$, to estimate $\bar{\phi}$ since $p_d \geq p_{\text{deff}}$, and the obtained value of $\bar{\phi}$ might rely on the way of conducting the estimation.

$\varrho_\phi(\phi)$ denotes the probability (faction) that a ch.cs with length ϕ is successfully decoded. $\varrho_\phi(\phi)$ for $2 \leq \phi \geq \phi_{\max}$ can be estimated or collected along with the process of updating probabilities with co.cs, i.e., the increase in $\varrho_h(D)$ and in $\varrho_f(D)$ for a considered ϕ is the increase in $\varrho_\phi(\phi)$. In addition, $\varrho_\phi(1)$ can be obtained by $p_d - \sum_{\phi=2}^{\phi_{\max}} \varrho_\phi(\phi)$. Hence, $\bar{\phi}$ can be obtained by:

$$\bar{\phi} = \frac{\sum_{\phi=1}^{\phi_{\max}} \phi \cdot \varrho_\phi(\phi)}{p_d}. \tag{3.10}$$

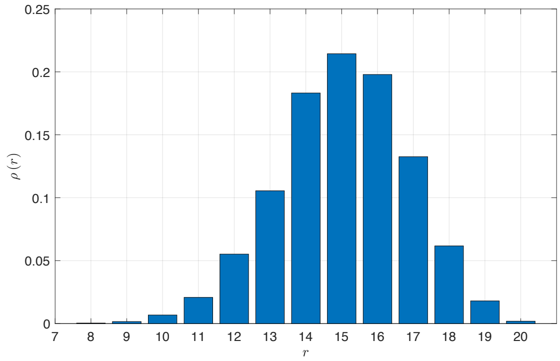
Therefore, while employing OCC/CF, for each successfully decoding, it would need $\mathcal{O}\left[\left(\bar{\phi}\cdot\mu+\gamma\right)^2 + n\cdot\left(\bar{\phi}\cdot\mu+\gamma\right)\right]$ finite field operations per block. It would be lower since $\mathbf{C}_c$ is an approximately sparse matrix, as shown in Figure 3.3.

3.7.5 Obtaining the Estimated Decodability

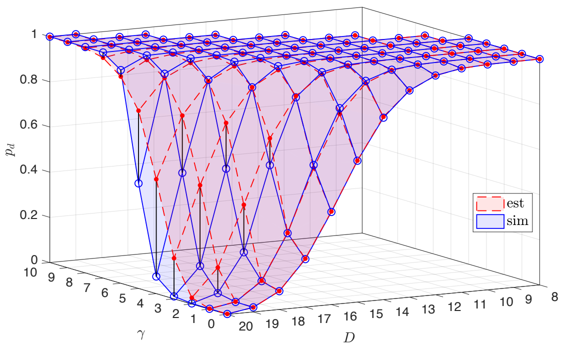
The process of updating probabilities with co.cs and b.s above can be done repeatedly and also alternately. Since there are μ innovative blocks per chunk, then chunk t with $r'^{(t)} > \mu$ can help the undecodable chunks to decode. This work considers that applying the updating process with b.s first might make the value of $\varrho_n'(r'^{(t)})$ where $r'^{(t)}$ is close to D tend to zero early and then might make the other $\varrho_n'(r'^{(t)})$ hardly tend to zero via the updating process. Therefore, this work applies the updating process with co.cs first.

By taking the overlapping fashion of the applied OCC as the pseudo-reference, the updating process with co.cs is applied for one iteration, then the updating process with b.s is applied for $\phi_{\max}$ iterations. If N is the total number of original blocks, then there are approximately $N_{\text{ch}} = \left\lceil \frac{N}{\phi_{\max}} \right\rceil$ chains of chunks with length $\phi_{\max}$. It is assumed that one time of the updating process is the updating process with co.cs for one iteration and then with b.s for $\phi_{\max}$ iterations. Then, N_{ch} times of the updating process are needed such that the effect of the first ch.cs reaches the last ch.cs, and thus, $2\cdot N_{\text{ch}}-1$ times of the updating process are needed such that this effect returns back to the first chunk. In this work, one round of the updating process is $2\cdot N_{\text{ch}}-1$ times the updating process. In order to obtain the ultimate $\varrho_d(r')$ for all r', the updating process is done repeatedly until the increment of obtained p_d is lower than an assigned ϵ_{thr}, then the updating process terminates, and the ultimate p_d is obtained. The channel efficient η is obtained by $\eta = \mu \cdot p_d$. The computational complexity for the estimation depends on ϵ_{thr}. It is higher when ϵ_{thr} is smaller, but the accuracy might be higher.

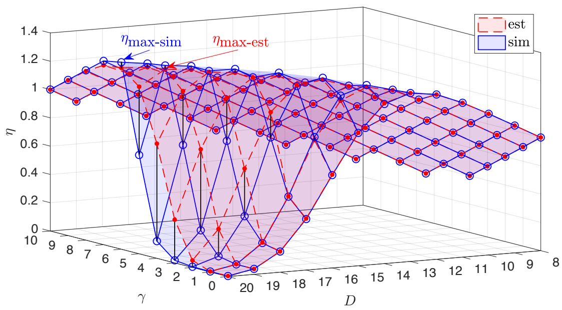
For example, by taking $K = L = 2$, $D_t = 10$, $\phi_{\max} = 5$, $\mathrm{SNR}_{\mathrm{SR}}^{(1,1)} = \mathrm{SNR}_{\mathrm{SR}}^{(2,2)} = \mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 35$ dB, $\mathrm{SNR}_{\mathrm{SR}}^{(2,1)} = 15$ dB, $E_8/7E_8$ as NLC, $\epsilon_{\mathrm{thr}} = 10^{-4}$, and by taking only the case that $\mu - \gamma > 0$, the empirical rank distribution $\rho(r)$, the estimated p_d and the correspondent η are shown in Figure 3.4. The simulation result obtained with the same condition and by using the empirical rank distribution in Figure 3.4a is also shown in Figure 3.4b,c for comparison. The process of updating probabilities with b.s is done first for the estimation of decodability (the former case). A continuous line connects the values obtained from the estimation and from the simulation for the same pair (μ, γ) in Figure 3.4b,c,d.



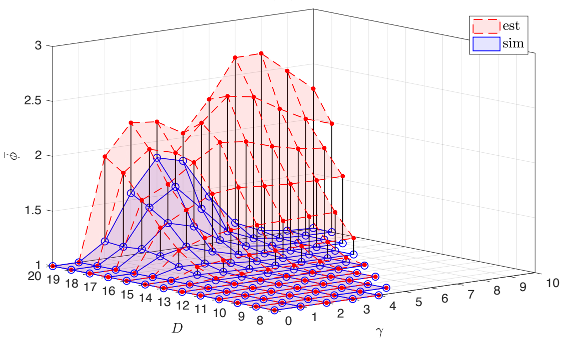
(a)



(b)



(c)



(d)

Figure 3.4: Comparison between the estimated p_d and η (est) and the result from the simulation (sim) that were obtained from an empirical rank distribution $\rho\,(r)$ while the process of updating probabilities with co.cs is done first. (**a**) Empirical rank distribution $\rho\,(r)$; (**b**) decodability; (**c**) channel efficiency; (**d**) mean length of ch.cs.

From Figure 3.4b, the estimation causes high deviation from the simulation result for the allocations (μ, γ) that provide low p_d, e.g., lower than 0.9, and it causes low deviation for the allocations (μ, γ) that provide high p_d. The error might be caused by the inaccuracy of the updated rank distribution $\rho'(r')$ or by the imperfectness of the process of updating probability with limited length of ch.cs. However, because the performance with high p_d is preferred for this work, thus the estimation applied in this work is acceptable.

On the other hand, from Figure 3.4c, the maximum η from the estimation, $\eta_{\text{max-est}}$, is obtained by taking allocation $(14, 5)$. However, the maximum η from the simulation, $\eta_{\text{max-sim}}$, is obtained by taking allocation $(14, 6)$. Both allocations have the same μ, but different output p_d. From the work of [27, 28], without limiting the length of ch.cs, larger γ with the same μ can provide higher p_d. Due to the estimation deviation, the allocation to provide $\eta_{\max}$ is not correctly given. However, during the application, the allocation providing $\eta_{\text{max-est}}$ can be switched to the other allocation with larger γ, but with the same μ to find out which allocation is more appropriate.

From Figure 3.4d, the estimated $\bar{\phi}$ is much larger than $\bar{\phi}$ from the simulation, since the process of updating probabilities with co.cs is executed first, which is different from the real fact that b.s should be conducted as soon as possible according to the decoding scheme described in Algorithm 4. Thus, $\varrho_\phi(\phi)$ is abnormally high for $\phi \geq 2$, especially, when γ is large. However, the obtained results of $\bar{\phi}$ with the same $\mu = D - \gamma$, but with different set values (μ, γ) from both estimation and simulation show that larger γ results in higher $\bar{\phi}$, hence higher decoding complexity. In addition, $\bar{\phi}$ from the estimation somehow can serve as $\bar{\phi}$ obtained in the worst case.

For more details of observation, Figure 3.5 provides the numerical result obtained with the same parameters but in the case that the process of updating probabilities with b.s is done first for the estimation of decodability (the latter case) comparing to the former case. The shown result is the differences of p_d and $\bar{\phi}$ between two cases, which are shown in Figure 3.5a and Figure 3.5b, respectively. The difference obtained by subtracting those of the latter case from those of the former case.

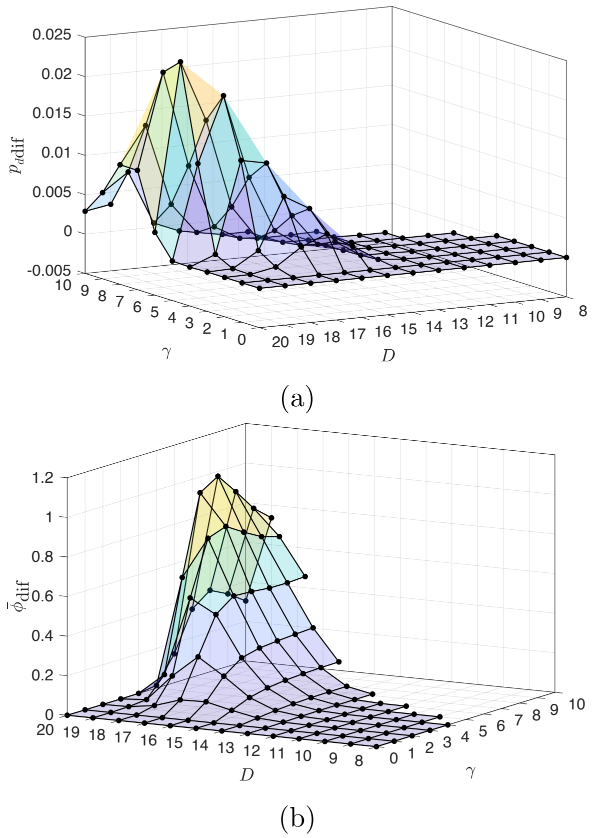


Figure 3.5: The differences of the estimated p_d and of the estimated $\bar{\phi}$ between the former case and the latter case. (**a**) difference of p_d, $p_{d\text{dif}}$; (**b**) difference of $\bar{\phi}$, $\bar{\phi}_{\text{dif}}$.

From Figure 3.5a, p_d obtained in the former case is higher than p_d obtained in the latter case, especially around the allocations which provide high channel efficiency η. However, $\bar{\phi}$ in the latter is much lower than $\bar{\phi}$ in the former in Figure 3.5b, especially for the allocations with higher γ. If comparing to the result from the simulation, it seems that p_d in the former case is more accurate than p_d in the latter case for the allocations providing high η. Hence, the estimations of p_d and η should be done in the former case. On the other hand, $\bar{\phi}$ should be done in the latter case since it follows the decoding scheme (b.s has high priority than co.cs).

3.8 Performance Observation

3.8.1 Examples of Allocations

This work assumes that the fairness between sources is achieved if $\eta_1 = \eta_2 = \cdots = \eta_K$ where η_k is the channel efficiency for source k. η_k is defined as the ratio of the number of decoded blocks from source k to the number of time slots taken from the sources to the relays. If individually decoding is not considered, η_k for all k only depends on p_{deff}. In this case, the fairness is achieved by taking $\mu_1 = \mu_2 = \cdots = \mu_K$.

By taking the data in Figure 3.4, Table 3.3 and Table 3.4 lists some allocations and their performances in p_{deff} and η_{eff} from estimation and simulation, respectively. In addition, $\bar{\phi}$ is the average value of ϕ, and it is counted when a ch.cs with length $\phi \geq 1$ is decoded and for the case that the process of updating probabilities with co.cs is done first. $\bar{\phi}$ represents the computational complexity of the decoding process.

Table 3.3: Performance in p_{deff} and η_{eff} of some example allocations from estimation.

Allocations					**Estimation**						
No.	μ_1	μ_2	γ_1	γ_2	p_1	p_2	p_{d}	p_{deff}	η_{eff}	$\bar{\phi}$
1	7	7	2	3	0.9982	0.9722	0.9206	0.8935	1.2509	1.5949
2	8	6	2	3	0.9868	0.9935	0.9206	0.9026	1.2636	1.5949
3	8	6	2	4	0.9868	0.9951	0.9152	0.8986	1.2580	2.1681
4	6	8	3	2	0.9988	0.8309	0.9206	0.7641	1.0687	1.5949
5	7	6	3	4	0.9975	0.9951	0.9756	0.9684	1.2589	2.3152
6	7	6	2	4	0.9982	0.9951	0.9779	0.9714	1.2628	1.4033
7	7	6	1	4	0.9989	0.9951	0.9710	0.9710	1.2623	1.2389
8	7	6	0	4	0.9981	0.9951	0.9674	0.9710	1.2577	1.1833
9	7	6	1	2	0.9989	0.9947	0.9697	0.9635	1.2525	1.1202

Table 3.4: Performance in p_{deff} and η_{eff} of some example allocations from simulation.

Allocations					**Simulation**		
No.	μ_1	μ_2	γ_1	γ_2	p_{deff}	η_{eff}	$\bar{\phi}$
1	7	7	2	3	0.9107	1.2750	1.4702
2	8	6	2	3	0.9295	1.3013	1.4264
3	8	6	2	4	0.9341	1.3077	1.4820
4	6	8	3	2	0.6674	0.9344	1.6135
5	7	6	3	4	0.9991	1.2988	1.1812
6	7	6	2	4	0.9986	1.2982	1.1767
7	7	6	1	4	0.9938	1.2919	1.1679
8	7	6	0	4	0.9807	1.2749	1.1549
9	7	6	1	2	0.9833	1.2783	1.0815

From Table 3.3 and Table 3.4, Allocation 1 can provide the fairness between sources, but it cannot provide the highest channel efficiency, while Allocations 2 and 3 can provide the highest channel efficiency by estimation and simulation, respectively. The interchanged Allocation 3, i.e., Allocation 4, shows the affect of an unsuitable selection of

(μ_2, γ_2), which has low p_{deff}. Allocations 1–4 do not provide high p_{deff}. On the other hand, Allocation 5 can provide high p_{deff}, but not the highest channel efficiency. However, the decoding complexity is lower since $\bar{\phi}$ is smaller. If the difference between the provided channel efficiency and the highest channel efficiency is small, then this allocation can be applied instead if lower decoding complexity is required. Allocations 5–9 have the same μ, but different γ, which varies from 7 to 3. They show the outcome of different values of γ to p_{deff} and $\bar{\phi}$. From the result in Table 3.3, larger γ provides higher p_{deff}, but higher decoding complexity.

On the other hand, the highest channel efficiency can be obtained by the precoding process at each source before employing OCC. However, this might cause additional decoding complexity and latency caused from re-ordering blocks after the decoding process. This work assumes that there is no precoding overhead, i.e., the number of required received coded blocks is equal to the number of original blocks when the maximum channel efficiency is considered.

3.8.2 Impact of the Participation Factor of Each Source

From now on, this work uses the term OCC as the applied OCC, which uses the allocation providing the highest channel efficiency. In addition, OCC′ refers to the applied OCC that uses the allocation providing the highest channel efficiency with condition $p_d \geq p_{\text{thr}}$ or $p_k \geq p_{\text{thr}}$. OCC/CF and OCC′/CF refer to the transmission schemes employing OCC and OCC′, respectively, before NLC in a multi-source multi-relay network. The decodability condition for OCC′/CF is $p_{\text{deff}} \geq p_{\text{thr}}$.

$\eta_{\max}$ denotes the maximum channel efficiency that can be provided by the applied OCC designed using $\rho(r)$. If individually decoding is not considered, from (3.1), the upper bound of η_{eff} is $\eta_{\max}$. As mentioned above, $\bar{\eta} = \dfrac{\bar{r}}{D_t}$ is the channel capacity or the upper bound of the channel efficiency for the transmission scheme employing OCC/CF from the sources to the destination. By taking $K = L = 2$, $D_t = 10$, $\phi_{\max} = 5$, $E_8/7E_8$ as NLC, $\epsilon_{\text{thr}} = 10^{-4}$, $p_{\text{thr}} = 0.97$, $\text{SNR}_{\text{SR}}^{(1,1)} = \text{SNR}_{\text{SR}}^{(2,2)} = 35$ dB, $\text{SNR}_{\text{SR}}^{(1,2)} \in \{5, 20, 35\}$ dB, $\text{SNR}_{\text{SR}}^{(2,1)} \in \{0, 5, 10, 15, 20, 25, 30, 35\}$ dB. The performance of OCC/CF and OCC′/CF in decodability and channel efficiency from estimation (with postfix “-est”) and simulation (with postfix “-sim”) is shown in Figure 3.6.

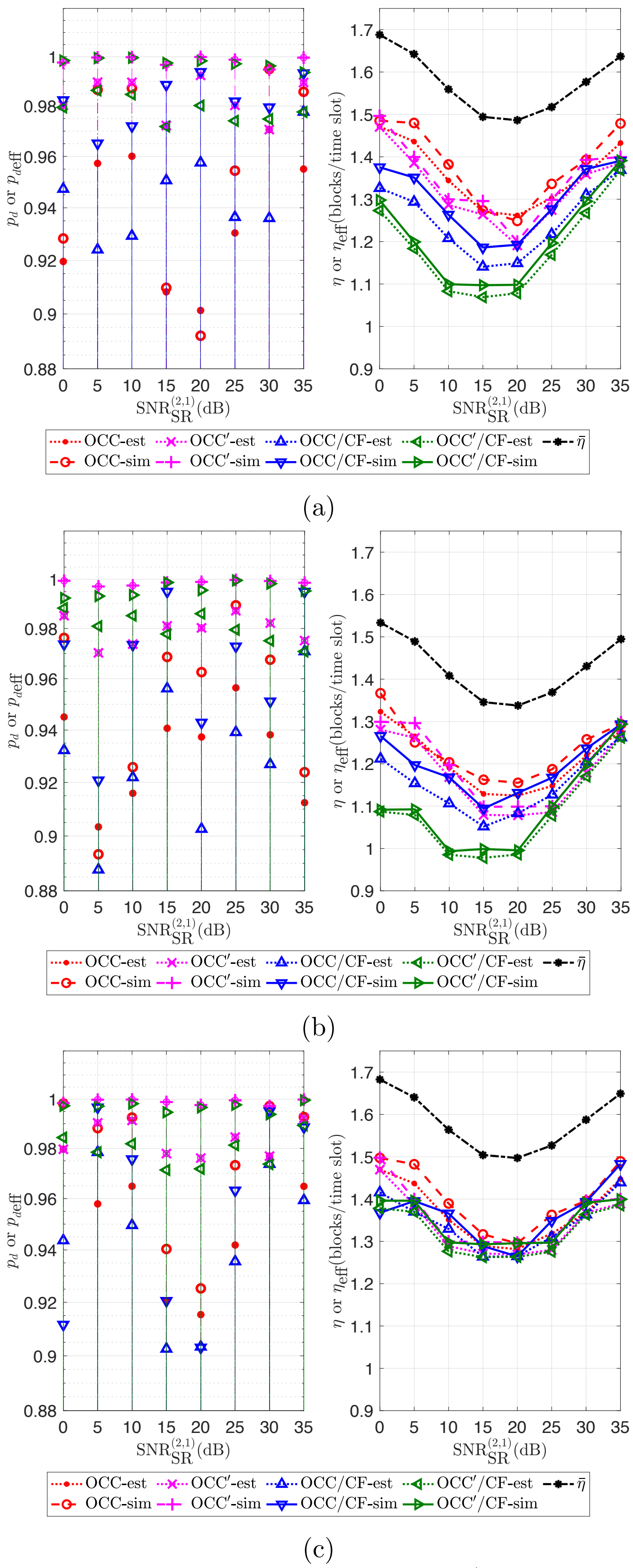


Figure 3.6: Decodability and channel efficiency of OCC/compute-and-forward (CF) and OCC′/CF from estimation and simulation. (**a**) $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 5$ dB; (**b**) $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 20$ dB; (**c**) $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 35$ dB.

From Figure 3.6, the estimated channel efficiency of OCC/CF and OCC′/CF are around 97.95% and 98.34%, respectively, of those of the channel efficiency obtained from simulation on average. In addition, the gap between the channel efficiency of OCC and $\bar{\eta}$ represents the design overhead of applied OCC. From the simulation result, $\eta_{\max}$ is around 87.71% of $\bar{\eta}$ on average. The design overhead of OCC/CF should be the aggregate of the design overhead using $\rho(r)$ and $\lambda_k(\theta_k)$ for all k. The channel efficiency of OCC/CF and OCC′/CF is close to $\eta_{\max}$ when $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)}$ or $\mathrm{SNR}_{\mathrm{SR}}^{(2,1)}$ is close to (as high as) $\mathrm{SNR}_{\mathrm{SR}}^{(1,1)}$ or $\mathrm{SNR}_{\mathrm{SR}}^{(2,2)}$. This is because, in this case, the participation factor of each source is dense around $\theta_k = d_k$. When $\lambda_k(d_k)$ is very dense, most of the allocations (μ_k, γ_k) can provide p_k close to one. Hence, the design of OCC/CF or OCC′/CF can only depend on $\rho(r)$.

In the case that $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)}$ and $\mathrm{SNR}_{\mathrm{SR}}^{(2,1)}$ are low, the received combined coded blocks are almost plain coded blocks, i.e., $\boldsymbol{\beta}_{lm}^{(i)}$ are almost in the form of unit vectors for all l and m. The design overheads of OCC/CF and OCC′/CF in this case should be the aggregate of the design overhead of OCCs using $\lambda_k(\theta_k)$ for all k. Since the participation factor of each source is not dense around $\theta_k = d_k$, the design overheads of OCC and OCC′ might be high and higher than those of the prior case.

In order to make $\lambda_k(d_k)$ denser, in addition to forcing to obtain $\boldsymbol{\beta}_{lm}^{(i)}$ without zero elements at each relay, improving the diversity of the received codeword combinations by increasing the number of participating relays or equipping more antennas at relays as in work of [41, 45] might be a solution.

3.9 Performance Evaluation

3.9.1 Reference Schemes

Because the original work in [11] did not consider the retransmission, a feedback-based transmission scheme is used as the reference scheme instead to evaluate the performance of OCC/CF and OCC′/CF, and it is called CF with protocol overhead (CF/PO) in this work. For each round of CF/PO, each source applies NLC without OCC and needs feedback from the destination after sending a block to know which blocks have been decoded and which blocks need to be retransmitted. Feedback is forwarded by relays via an orthogonal channel. There is no decoding delay constraint, i.e., a source can transmit a new original block although the previous blocks of the other sources have not been decoded [48].

The protocol overhead (the transmission time of feedback and the loss of feedback) are taken into account. The feedback reception success rate is denoted by p_f, and the ratio of the transmission time of feedback to a slot time is denoted by τ_f. p_f is obtained by conducting a simulation where NLC is applied on a link from a source to a relay with the highest SNR via an orthogonal channel. Because it is hard to track the performance of CF/PO with varying p_f, only the performance with different values of τ_f is considered.

In addition to the channel efficiency, the transmission efficiency ε_t is also considered to evaluate the performances of OCC/CF and OCC′/CF with a scheme called RLNC via orthogonal channel (RLNC/OC) where RLNC is applied before NLC at each source for the transmissions from the sources to the relays via an orthogonal channel. For each source, original blocks are grouped into disjoint chunks with D_t blocks per chunk. RLNC is applied within each chunk, and a feedback (ACK) is needed when a transmitted chunk is decodable. The protocol overhead is also considered, and it is assumed that feedback cannot be received instantaneously by the source to stop transmitting [32]. In this work, ε_t is defined as the ratio of the total number of decoded blocks to the total number of transmissions taken between the sources and the relays, while the transmission of feedback is also taken into account. In this work, τ_f is assumed as the ratio of the length of feedback data to the length of payload data per block. The performance in transmission efficiency reflects the energy consumption of each scheme.

On the other hand, a transmission scheme employing LT code [58, 62] at each source before NLC, called fountain code over CF (FC/CF), is also used as a reference scheme. The considered parameters of LT code are $c_{\rm fc}$ and $\delta_{\rm fc}$. In the case of single flow transmission with N_k original blocks, the receiver can recover all blocks with probability $1 - \delta_{\rm fc}$ if receiving $N_k + 2 \cdot \log_e(S_{\rm fc}/\delta_{\rm fc}) \cdot S_{\rm fc}$ coded blocks, where $S_{\rm fc} \equiv c_{\rm fc} \cdot \log_e(N_k/\delta_{\rm fc}) \cdot \sqrt{N_k}$.

In the CF/PO and FC/CF schemes, the decoding process is done for each source block transmission. The destination tries to decode if there are K linearly independent codeword combinations. If undecodable, the destination stores the undecodable blocks after decoding using (2.4) and waits for the next received codeword combinations.

Since feedback is not needed in OCC/CF, OCC′/CF and FC/CF, their transmission efficiency is $1/K$ times their channel efficiency. If $N_{\rm dec}$, $N_{\rm fb}$ and $N_{\rm ts}$ denote the total number of decoded blocks, the total number of feedback and the total number of time slots taken excluding the transmission time of feedback, respectively, then the channel efficiencies and the transmission efficiencies of CF/PO and RLNC/OC can be written as below.

$$\eta_{\text{eff\_CF/PO}} = \frac{N_{\text{dec}}}{N_{\text{ts}} + \tau_f \cdot N_{\text{fb}}}. \tag{3.11}$$

$$\varepsilon_{t\text{\_CF/PO}} = \frac{N_{\text{dec}}}{K \cdot N_{\text{ts}} + \tau_f \cdot N_{\text{fb}}}. \tag{3.12}$$

$$\eta_{\text{eff\_RLNC/OC}} = \frac{N_{\text{dec}}}{N_{\text{ts}}}. \tag{3.13}$$

$$\varepsilon_{t\text{\_RLNC/OC}} = \frac{N_{\text{dec}}}{N_{\text{ts}} + \tau_f \cdot N_{\text{fb}}}. \tag{3.14}$$

3.9.2 Numerical Results and Discussion

This work considers two scenarios to observe the performance of the transmission schemes employing OCC/CF and OCC′/CF by comparing with the reference schemes. The first scenario investigates the performance in a two-source two-relay network with an asymmetric channel state at relays, i.e., average SNRs of all links from all sources to a relay might be different, as used in Section 3.8.2. The second scenario considers a varying number of relays with a symmetric channel state, i.e., average SNRs of all links from all sources to a relay are the same, and the number of sources is fixed to two. The numerical results are obtained by conducting simulations taking $E_8/7E_8$ as NLC, $\epsilon_{\text{thr}} = 10^{-4}$, $D_t = 10$, $N_k = 1000$ for all k, $c_{\text{fc}} = 0.01$, $\delta_{\text{fc}} = 0.01$, $\tau_f = 0.05$ for RLNC/OC and $\tau_f \in \{0, 0.05\}$ for CF/OF. The simulation frequency is 100 times. Each simulation terminates when there is at least source k having the rest of innovative blocks less than μ_k for OCC/CF, OCC′/CF and RLNC/OC and when all original blocks of at least one source are recovered for CF/PO and FC/CF.

The performances of OCC/CF and OCC′/CF in the first scenario are the same as in Figure 3.6. The second scenario takes $\text{SNR}_{\text{SR}}^{(1,1)} = \text{SNR}_{\text{SR}}^{(2,2)} = \text{SNR}_{\text{SR}}^{(1,2)} = \text{SNR}_{\text{SR}}^{(2,1)} \in \{30, 35, 40\}$ dB with correspondent $p_f \in \{0.8266, 0.9021, 0.9449\}$. The performances in channel efficiency and transmission efficiency of all schemes in Scenario 1 and 2 are shown in Figures 3.7 and 3.8, respectively.

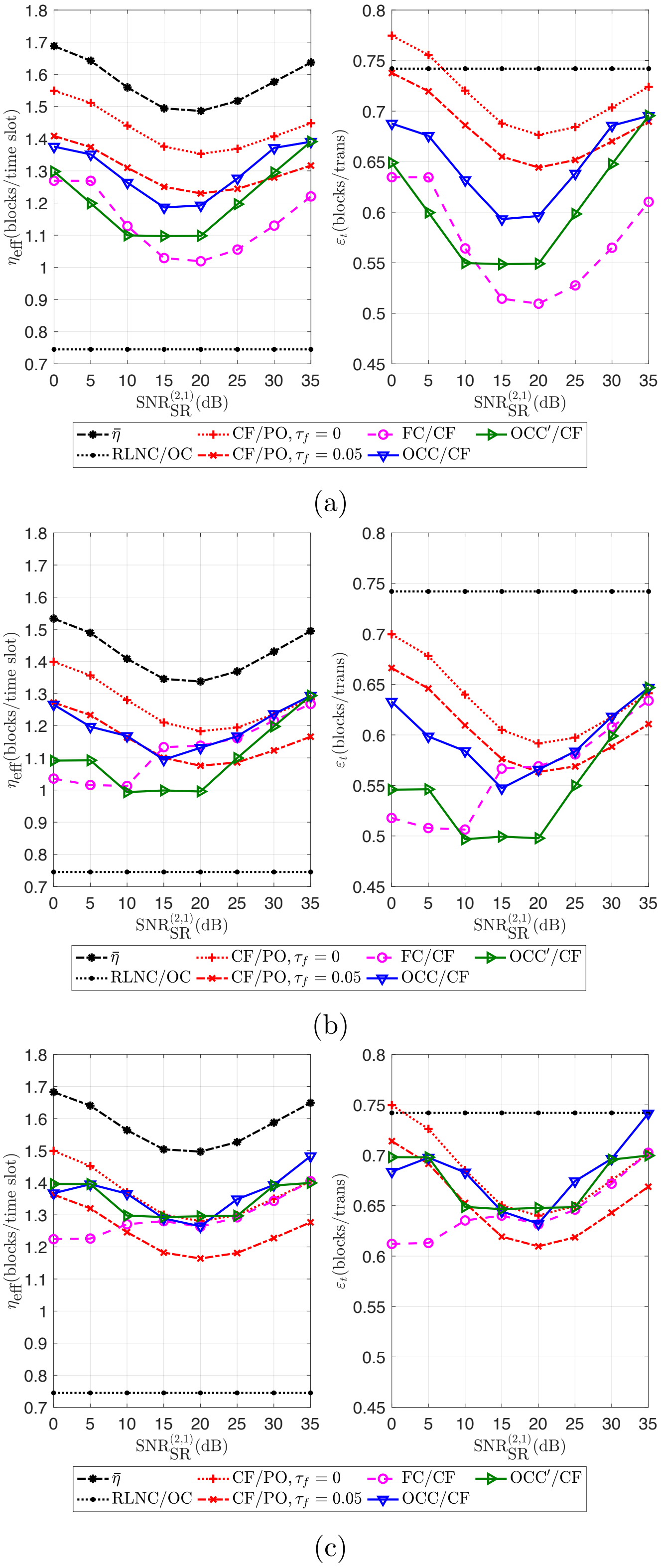
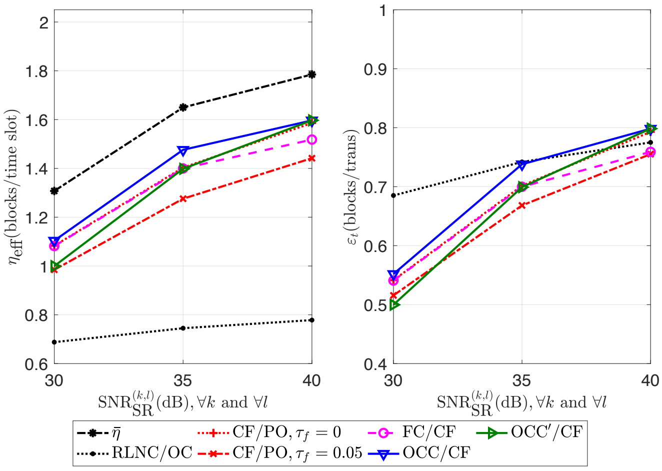
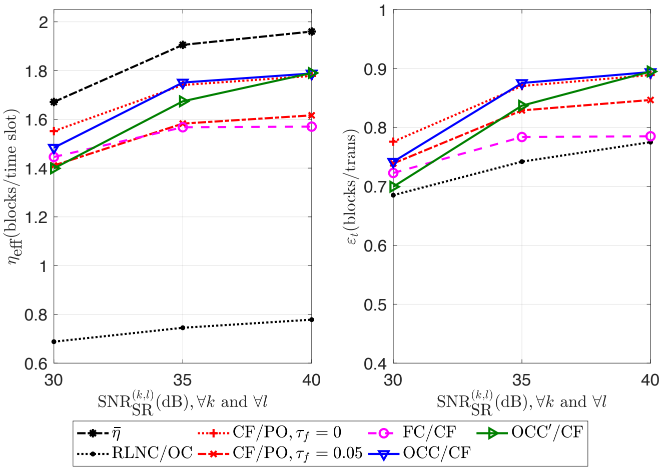


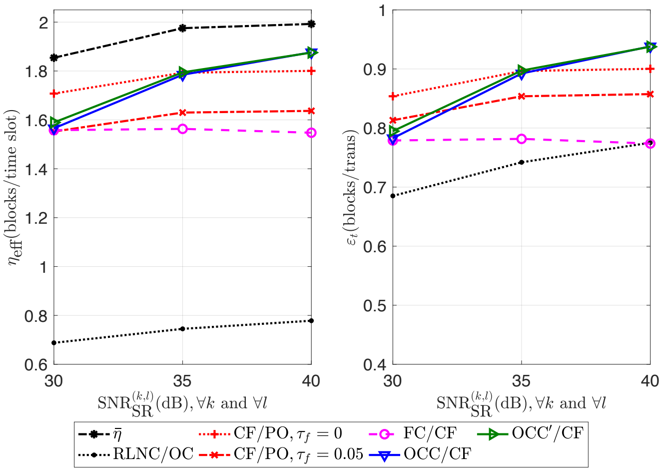
Figure 3.7: Channel efficiency and transmission efficiency for Scenario 1. (**a**) $\text{SNR}_{\text{SR}}^{(1,2)} = 5$ dB; (**b**) $\text{SNR}_{\text{SR}}^{(1,2)} = 20$ dB; (**c**) $\text{SNR}_{\text{SR}}^{(1,2)} = 35$ dB.



(a)



(b)



(c)

Figure 3.8: Channel efficiency and transmission efficiency for Scenario 2. (**a**) Two-source two-relay network; (**b**) two-source three-relay network; (**c**) two-source four-relay network.

From Figure 3.7, OCC/CF and OCC′/CF increased channel efficiency by 72.98% and

63.28%, respectively, on average if comparing with RLNC/OC. For $\tau_f = 0.05$, OCC/CF had a 4.71% increment of channel efficiency on average if comparing with CF/PO. However, OCC/CF and OCC′/CF provided higher channel efficiency than CF/PO only when $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)}$ or $\mathrm{SNR}_{\mathrm{SR}}^{(2,1)}$ was similarly high as $\mathrm{SNR}_{\mathrm{SR}}^{(1,1)}$ or $\mathrm{SNR}_{\mathrm{SR}}^{(2,2)}$, i.e., when the participation factor of source k was dense around $\theta_k = d_k$ for all k. OCC/CF and OCC′/CF provided the increment of channel efficiency up to 16.41% and 13.41% if comparing with CF/PO for $\tau_f = 0.05$. On the other hand, by comparing with FC/CF, OCC/CF had higher channel efficiency than FC/CF in almost all cases. The increment was around 9.36% on average. For OCC′/CF, because of the higher design overhead, it sometimes could not provide higher channel efficiency than FC/CF. There was only a 2.91% increment of channel efficiency on average if comparing with FC/CF. The issue of employing fountain code in this scenario was that the number of coded blocks to ensure the desired decodability was larger than the expected number, because some coded blocks from a source could not be extracted from the codeword combinations forwarded from the relays during each source block transmission.

On the other hand, the transmission efficiency of RLNC/OC was higher than the other schemes in all cases except CF/PO for $\tau_f = 0$. Thus, OCC/CF and OCC′/CF had less of a chance to perform better than CF/PO when τ_f was small. The transmission efficiency of OCC/CF and OCC′/CF was 86.85% and 81.98% on average, respectively, of the transmission efficiency of RLNC/OC. It seems the performance of the proposed schemes was a trade-off between the increment of channel efficiency and the decrement of transmission efficiency if comparing with an orthogonal channel transmission scheme, e.g., RLNC/OC for this scenario. In order to improve the transmission efficiency of the proposed schemes, increasing the diversity of codeword combinations at relays, i.e., increasing the number of participating relays, was considered and discussed in Scenario 2.

From Figure 3.8, the channel efficiency of OCC/CF and OCC′/CF increased when $\mathrm{SNR}_{\mathrm{SR}}^{(k,l)}$ increased, since the channel capacity $\bar{\rho}$ also increased with $\mathrm{SNR}_{\mathrm{SR}}^{(k,l)}$. OCC/CF and OCC′/CF performed similarly at high $\mathrm{SNR}_{\mathrm{SR}}^{(k,l)}$, and they had a chance to perform better than CF/PO for $\tau_f = 0$ in the case of four relays at $\mathrm{SNR}_{\mathrm{SR}}^{(k,l)} = 40$ dB since the loss of feedback also impacted the performance of CF/PO. If comparing with CF/PO, OCC/CF and OCC′/CF increased channel efficiency up to 3.93% and 4.17%, respectively, for $\tau_f = 0$, and up to 16.18% and 14.59%, respectively, for $\tau_f = 0.05$. If comparing with RLNC/OC, OCC/CF and OCC′/CF increased channel efficiency up to 140.32% and 140.88%, respectively. OCC′/CF sometimes performed slightly better than OCC/CF, because of the estimation deviation. In addition, the channel efficiency of OCC/CF and OCC′/CF increased faster than that of FC/CF when $\mathrm{SNR}_{\mathrm{SR}}^{(k,l)}$ increased or the number of relays increased. This is because the overhead of fountain code was the same if the parameters c_{fc} and δ_{fc} were fixed. In addition, it might have been because of the condition of stopping the simulation, i.e., all original blocks of a source were recovered, and those of the other source had not been all recovered, the channel efficiency of FC/CF did not increase when $\mathrm{SNR}_{\mathrm{SR}}^{(k,l)}$ increases or the number of relays increased, as shown in Figure 3.8c.

For the performance in transmission efficiency in Figure 3.8, OCC/CF and OCC′/CF had higher transmission efficiency than RLNC/OC when the number of relays was higher than the number of sources. However, employing more relays might have increased the complexity of the network such as how to select which relays to join, how to achieve time synchronization at all relays, etc.

On the other hand, the performance of OCC/CF and OCC′/CF could be improved, especially at low $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)}$ and low $\mathrm{SNR}_{\mathrm{SR}}^{(2,1)}$ in the first scenario by applying decoding individually, but this might have increased the complexity of the decoding process if K were large.

3.10 Summary

This chapter describes a design of OCC that is applied before NLC in multi-source multi-relay networks, called OCC/CF. A decodability condition was provided for the design. This work took an OCC with a contiguously overlapping fashion, but not a rounded-end fashion, to design OCC/CF. The decoding scheme and the estimation of designed OCC/CF are provided. The estimation is done for each allocation, i.e., the number of innovative blocks per chunk and the number of blocks taken from the previous chunk, to search for which allocation can provide the desired performance such as the highest channel efficiency, or the preferred decodability, or the acceptable decoding complexity. The estimation deviation is low when the decodability is sufficiently high. Since there are a limited number of chunks for the designed OCC/CF, the design overhead is high if comparing with channel capacity. From the numerical results, the advantage of OCC/CF over a feedback-based transmission scheme depends on the level of protocol overhead, i.e., the transmission time and the size of feedback, the feedback loss rate. The performance of OCC/CF, especially transmission efficiency when comparing with an orthogonal channel transmission, can be improved by increasing the number of relays. Future work is to consider decoding individually and the cooperation between feedback and OCC/CF for higher performance.

Chapter 4

A Retransmission Scheme using Overlapped Chunked Code over Compute-and-Forward

4.1 Introduction

This chapter describes a proposed retransmission scheme for multi-source multi-relay networks, called RLNC/CF, which applies random linear network coding within each chunk at each source before generating nested lattice code codewords (CWs) for simultaneous transmission from the sources to the relays. Compute-and-forward (CF) approach is employed at relays to generate linear combinations of CWs for forwarding to the destination via orthogonal channel. The number of input blocks are the same for each chunk and for each source. Feedback is needed for each chunk transmission for all sources from the destination to manage the retransmission. Lossless transmission of feedback is assumed. Block acknowledgement (BACK) scheme is employed. The content of BACK is the numbers of blocks to be taken from the previous chunk at all sources for an expected reception state (ERS). These numbers are determined using the empirical probability distributions. The different selections of ERS can provide different outcomes, quick recovery of undecodable blocks or high channel efficiency. From the numerical results, RLNC/CF scheme can provide some improvement in channel efficiency over a cooperative CF scheme, however, there is a trade-off between the increment of channel efficiency and the decrement of transmission efficiency if comparing to a transmission scheme employing orthogonal channel.

This chapter is organized as follows. The problems are stated in Section 4.2. Related works are described in Section 4.3. Section 4.4 describes the scenario of this chapter. The details of the proposed scheme is described in Section 4.5. After that, the performance evaluation including the description of the reference schemes, the performance with different expected channel state and the numerical results with discussion is given in Section 4.6. At the end, Section 4.7 gives the summary for this chapter.

4.2 Problem Statement

While CF based on NLC is applied in multi-source multi-relay networks, each relay computes the superimposed CWs transmitted from the sources to obtain the linear combinations of these CWs and forwards the obtained linear combinations to the destination. The destination can recover correctly the original information in source CWs if there are enough number of the qualified CW combinations (CWCs), thus, the destination could not recover them due to insufficient forwarded CWCs. By considering symmetric message rate and symmetric transmit power as in the previous chapter, the number of qualified CWCs can be increased by allowing more number of relays (by comparing to the number of sources) [41] or equipping more number of antennas at each relay [45] to conduct data forwarding. Although more number of participating relays can improve the diversity of the CWCs, however it might bring higher system complexity such as synchronization, more complicated cooperation, and the energy consumption by considering for the whole network (in case that the reception energy consumption is also considered). Furthermore, it might affect the other data transmission if there are some relays also participate in the data transmission of the other sources.

On the other hand, with fewer number of relays and small size of finite field, q, the linear dependence of CWCs might occur if there is no cooperation between relays. To solve this problem, the precoding process which is to make the combination coefficient vectors linearly independent to each other such as in the work of [65]. However, this solution relies on the correctness of the knowledge of the channel state information (CSI) obtained at the sources. If the channel coefficient varies quickly, then the CSI estimated at the relays for the next transmission, which will be informed to the sources via feedback, might be deviated. Hence, there would be error, and the issue of the linear dependence between the CWCs computed at relays could not be completely solved. The consideration of the different level of transmit power [51, 52], different source message rates [11, 52] and channel access allocation [46] might not ensure completely ensure the recovery of the source messages at the destination since the channel state is not stable, and there might be error in the estimation of channel state. Although the transmission considering the worst channel state can be an option, but it would be not efficient. Alternatively, retransmission would be considered. Thus, the original source information could not be recovered for every transmission, and the retransmission would be needed alternatively. However, the retransmission scheme for CF approach has not been well investigated, and there are few works related to the retransmission scheme for PNC up to the present. Should not-yet recovered source blocks of all sources be retransmitted or just only blocks of some sources be retransmitted in order to obtain more efficient data transmission, i.e., high channel efficiency or high transmission efficiency? In addition, how do a retransmission scheme should take use the received CWCs that could not be decoded at the last transmission to obtain efficient retransmission? Furthermore, if the feedback for each transmission of source block is needed, then the protocol overhead would be high. Hence, the performance would be degraded. This work aims to provide an efficient retransmission scheme which can improve the network throughput, energy efficiency while CF based on NLC is employed in multi-source multi-relay networks.

4.3 Related Works

While employing CF in multi-source multi-relay network, the interaction between the relays and the destination might be needed to determine which CWCs are to be forwarded and which relays are supposed to forward before forwarding. Hence, if the destination fails to have enough CWCs to recover the original blocks of all sources, the retransmission would be only made by the sources. If the destination stores the CWCs that could not recover the original blocks of all sources, then how to efficiently take use of these stored CWCs for the retransmission should be considered. In the work of [48], if the block of a source is recovered during a round of transmission, then that source can transmit a new block regardless of the fact the blocks of the other sources have not been recovered. However, if there is no any recovered source blocks from the existing CWCs, then more efficient retransmission would be needed to take advantage of CWCs. For example in case of two sources, if the destination stored a CWCs that is not a plain CW of any source, then there might be only one source that needs to retransmit the previous block, and the another source can transmit a new block while unrecovered blocks do not needed to become recovered after the retransmission.

If each source transmits a block for each round and the feedback is needed after the transmission of each round, then the protocol overhead would be significant. Successful reception and failed reception can be informed each source by using acknowledgement (ACK) and negative ACK (NACK), respectively. In order to reduce the protocol overhead, block ACK (BACK) scheme which is employed in IEEE 802.11n standard and IEEE 802.11e standard [66–68] can be used to reduce the amount of needed feedback by compressing the content of sending feedback. A source puts a certain number of blocks into the transmission window with successive ID, and after these blocks are all transmitted to a receiver, a request for BACK, BACKReq, is sent by the source. The receiver sends a BACK containing the reception states of the transmitted blocks inside the transmission window. The source only retransmits the unsuccessfully received blocks within the transmission window, and the process of BACK scheme repeats until all blocks in the transmission window are all successfully received. Then, successfully received blocks are released, and new blocks are added into the transmission window. In this case, sending feedback is triggered by the request from the source. However, up to the present, it is seems that there are rare works employing BACK with PNC.

The other approach is selective ACK (SACK) scheme which is employed in the work of [48] for the transmission with PNC in cross-atom topology. The transmission window begins with the block which has not not received, and it can also include the blocks which have already been received. Each source keeps transmitting the blocks have not been acknowledged within the transmission window in round robin fashion until the number of received blocks at the relay reaches a certain number, then a compressed ACK about the reception states of transmitted blocks is sent back to the sources. The transmission window is refreshed according to the information from the feedback. In this case, sending feedback is triggered by the reception state at the relay. For the case of multiple relays, i.e., multi-source multi-relay networks, all relays might need to cooperate each other after each block transmission while employing SACK scheme. Then, extra protocol overhead is caused. It might be impossible if the relays are out of transmission range of each other. In addition, the feedback probably could not arrive immediately to stop the transmission from the sources since the transmission time of feedback could not be ignored.

On the other hand, RLNC can reduce the protocol overhead because the receiver only needs to send back an ACK to the sender when it receives linearly independent correct coded blocks with the same number as the input blocks. If blocks are encoded with RLNC within chunk before encoded with NLC encoder at each source before transmitting, the destination needs to collect linearly independent qualified CWCs with at least the same number as the total number of blocks of all sources per chunk [46]. For the case of single relay, sending feedback can be triggered by the event that the number of linearly independent CWCs reaches the desired amount, i.e., the total number of blocks per chunk[46]. However, for the case of multiple relays, how to manage to send feedback should be put into consideration.

Alternatively, if source blocks are grouped into overlapped chunks [27, 57], then decoded chunks can help undecodable chunks which have common blocks in the decoding process by increasing the number of linearly independent coded blocks in the undecodable chunks via back-substitution. This work takes the blocks from the previous chunk (from the toe of the previous chunk as in previous chapter) to add into the new chunk for retransmission at each source in multi-source multi-relay network, and the other blocks of the previous chunk are expected to be recovered after the transmission of the next chunks, hence they can be released from the transmission window. However, the (maximum) number of coded blocks to be transmitted for each chunk is the same for all sources and is limited by a certain number, and BACK scheme is employed for each hop transmission. Each feedback transmission is assumed lossless in this chapter. With BACK scheme, the proposed retransmission scheme in this work might work well when the number of sources become large, and there is no overhead caused from late reception of feedback. A BACKReq is sent by each source to its assigned relay separately after a chunk transmission. After receiving the forwarded CWCs from the relays, the destination send feedback, i.e. BACKs, to the sources via the relays. The content of feedback is the numbers of overlapped blocks for each source to manage the retransmission, which are determined by using the empirical probability distributions related to the channel state of the links from the sources to the relays, the probability distribution of the number of qualified linearly independent CWCs received for a chunk, and the probability distributions of the participation factor of a source in these CWCs. This determination is done for an expected reception state of the sending chunk ,i.e., the retransmitting chunk. A set of the numbers of overlapped blocks for all sources determined in this work provides an expected decodability of the previous chunks and the retransmitting chunk, which results in a different performance, high decodability or high throughput, for example. The contribution in this chapter are as below:

- providing a retransmission scheme using BACK scheme and contiguously overlapped chunks for multi-source multi-relay network;
- providing an approach for determining the number of overlapped blocks for all sources for the retransmission by using the empirical probability distributions to obtain highest decodability of the blocks of the previous chunks and of the retransmitting chunk for an expected reception state;
- providing a discussion on the performance of the proposed scheme with different selections of the expected reception state.

The numerical results shows that selection of expected channel state does a trade-off between channel efficiency (i.e., network throughput) and reception delay (i.e., the recovery speed of the undecoded blocks) for the proposed scheme. In addition, the propose scheme can increase channel efficiency if comparing to a cooperative CF scheme. However, it does a trade-off between the improvement of channel efficiency and the degradation of transmission efficiency (i.e., energy efficiency) if comparing to a transmission scheme employing orthogonal channel.

4.4 Scenario

This chapter considers a scenario of multi-source multi-relay single-destination network. There are K sources, L relays and single destination. Each node is equipped with a single antenna. The consideration of multiple antennas could be done as in the work of [45]. All sources need to transmit their messages to the destination via the relays, and the direct link transmission from the sources to the destination is not considered in this chapter. An example of the scenario with two-source two-relay network is shown in Figure 2.1. The scenario can be extended to the case of multiple destinations and the case that the direct link transmission (or overhearing) is considered, for example, the scenario with cross topology [48] or two-way relay channel.

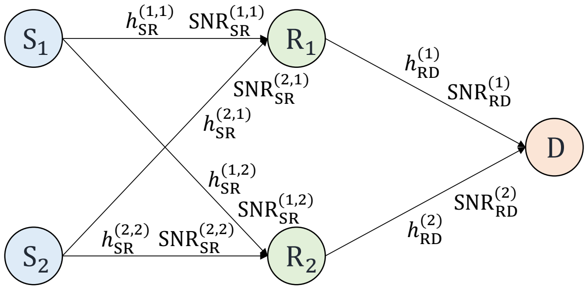


Figure 4.1: Scenario for the case of a two-source two-relay single-destination network.

In Figure 4.1, $h_{\text{RD}}^{(l)}$ denotes the channel coefficient corresponding to the instantaneous received signal-to-noise ratio (SNR) of the link from relay l to the destination. $\text{SNR}_{\text{RD}}^{(l)}$ denotes the average received SNR of is link. A source is assigned with a relay which is in charge of receiving BACKReq from that source and sending BACK back to that source. In addition, assigned relay will help its corresponding source to forward message to the destination if orthogonal channel is employed. This work assumes that the average received SNR of the link from the source to its assigned relay can be fixed, but the average SNR of the link from the source to the other relays might not be assured. In this chapter, lossless transmissions from the relays to the destination are not assumed.

The transmissions of any control messages including ACK, NACK, BACKReq, BACK and the control messages for the interaction between the relays and the destination, are lossless, but their transmission time is the same and not negligible. The transmission time are counted in time slot. τ_f denotes the transmission time of feedback, normalized by a slot time. The set up phase and tear down phase [67, 68] in employing BACK scheme are skipped in this work.

Similarly, each source employs the same NLC with identical message rate $R_1 = R_2 = \cdots = R_K = R = \log_2 q$. However, in this chapter, In this chapter, the chunk size is

D_t for each chunk and each source. In addition, relay l (for $l \in \{1, 2, \cdots, K\}$) generates $r_{ld_t}^{(i)} \le K$ from the d_t-th transmitted source block of chunk i. Similarly, the rank probability distributions are denoted by $\rho(r)$ and $\lambda_k(\theta_k)$ for convenience. Hence, $r^{(i)}$ is the rank of matrix $\mathbf{C}^{(i)} \in \mathbb{F}_q^{D_t \times r^{(i)}}$, which is a set of $r^{(i)}$ linearly independent vectors taken from $K \cdot L \cdot D_t$ vectors $\left\{\mathbf{c}_{ld_t\iota}^{(i)}, l \in \{1, 2, \cdots, K\}, d_t \in \{1, 2, \cdots, D_t\}, \iota \in \{0, 1, \cdots, K\}\right\}$, where $\mathbf{c}_{ld_t\iota}^{(i)}$ is the simplified notation of the combined coding coefficient vector of the ι-th CWC computed at the d_t-th slot at relay l for chunk i. $r_{\max}$ denotes $K \cdot D_t$ which is the total number of blocks of all sources per chunk. The allocation of channel use for each source is full, i.e., $\omega_{kd_t}^{(i)} = 1$, $\forall k, i$, and d_t. Each relay must forward all the CWCs which are demanded by the destination.

4.5 Retransmission Scheme using Contiguously Overlapped Chunked Code

4.5.1 Overview

This chapter assumes that the original blocks are successively added into the transmission window with size D_t, i.e., a chunk with size D_t alike at each source, hence, chunk is addressed instead of transmission window from now on, and chunk $i-1$ and chunk i refers to the previous chunk and the current chunk (retransmitting chunk), respectively. RLNC is employed within chunk to generate D_t coded blocks. For chunk i at source k, there are $\mu_k^{(i)}$ new blocks, i.e., innovation blocks comparing to the previous transmitted blocks, and there are $\gamma_k^{(i)}$ blocks taken from chunk $i-1$. These $\gamma_k^{(i)}$ blocks are selected from the end of chunk $i-1$ as shown in Figure 4.2.

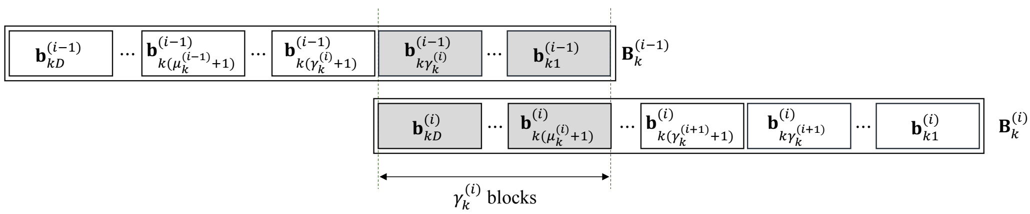


Figure 4.2: The blocks of chunk $i-1$ and chunk i, where the grey blocks refer to the overlapped blocks, i.e., the common blocks of two chunks.

After all sources have transmitted D_t coded blocks of chunk $i-1$, for example, to the relays, all sources send a BACKReq to their assigned relay via orthogonal channel in time domain to avoid the collision. BACKReq also serves to inform the relays that the transmissions of chunk $i-1$ from the sources have ended. It might take at least $K \cdot \tau_f$ time slots for sending BACKReq from all sources. As a relay receives BACKReq, it will send a feedback to inform the source of the success reception of BACKReq and also to inform the destination about its reception state information (RSI), i.e., the combined coding coefficients of the CWCs and the computation rates of the CWCs for the interaction with the destination. After the destination receives all this information from all relays, it tries to find the maximum number of the linearly independent CWCs with the highest sum of computation rates for chunk $i-1$. The CWCs are needed by the destination if they are useful for decoding, but they are not needed to be linearly independent of the CWCs of the previous chunks. The destination will determine which CWCs belong to which relay, and then, it informs all relays about their forwarding decision information (FDI), i.e., which CWCs to be forwarded and by which relay by broadcasting this information to all relays. Hence, it might take at least $(L+1) \cdot \tau_f$ time slots for the interaction between the relays and the destination.

After that, each relay alternately forwards the demanded CWCs to the destination by using BACK scheme (immediate BACK scheme as described in [67, 68]). The destination broadcasts the retransmission request information (RRI) to the relays after all demanded CWCs have been successfully received. RRI contains the information about the number of blocks taken from the previous chunk for all sources, i.e., $\gamma_k^{(i)}$ for all k. Each assigned relay captures the desired information from received RRI before forwarding it to the correspondent source for preparing the retransmission, i.e., the transmission of chunk i. Alternatively, RRI is sent to the assigned relays separately to inform $\gamma_k^{(i)}$. The format of RRI can be the same as BACK which using bitmap. The bits corresponding to the overlapped blocks are set to be zero, and the other blocks are set to be one. On the other hand, the content of FDI might also includes the values of $\gamma_k^{(i)}$ for all k. It might be useful if the transmission between the sources to the relays and the transmissions between the relays and the destination employ different channel, or if the relays work in full-duplex mode. However, in this work, only RRI contains the information of $\gamma_k^{(i)}$. The brief illustration of this transmission scheme applying for a chunk is shown in Figure 4.3

for the case two-source two-relay single-destination.

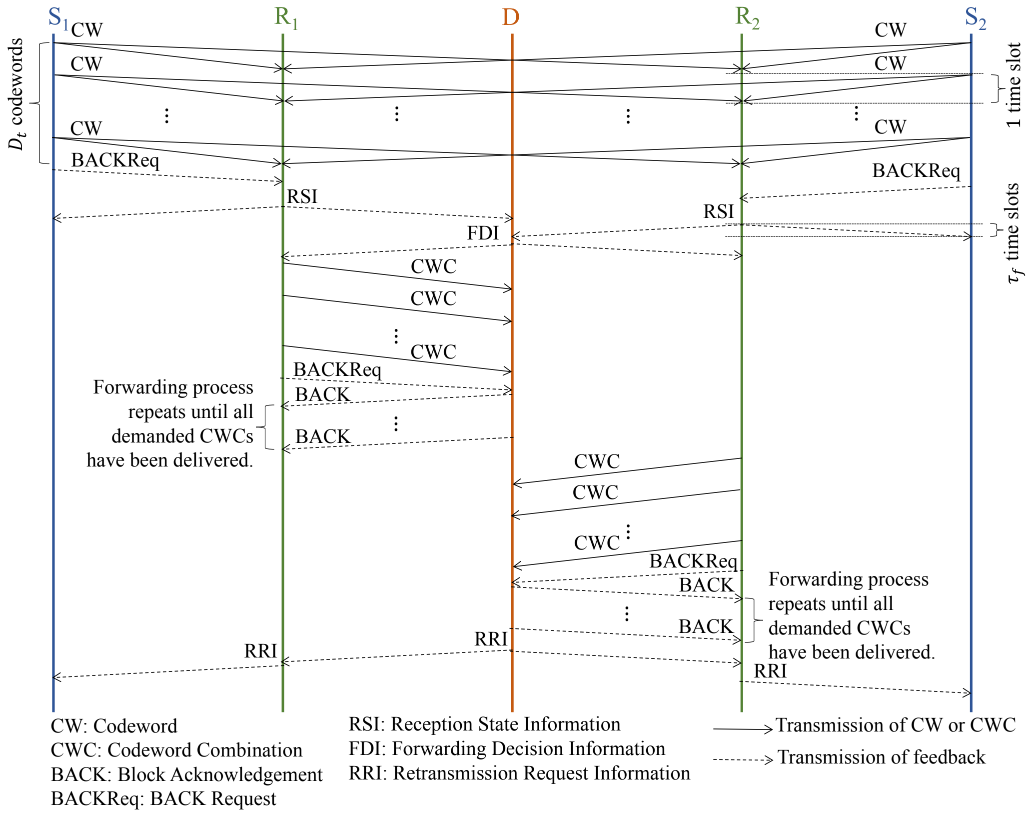


Figure 4.3: Brief illustration of the proposed transmission scheme applying for a chunk with the case two-source two-relay single-destination network.

Since this proposed scheme employs BACK scheme, it might works well when the number of sources is large, because the protocol overhead is only caused from the transmission of BACK and BACKReq, even if BACK or BACKReq might be loss. In some retransmission schemes, a sender needs to retransmit already received blocks if the feedback for these blocks was loss.

If $\gamma_k^{(i)}$ for all k are well selected such that the blocks of chunk $i-1$ and of chunk i are expected to be recovered after the transmission of chunk i or further latter chunks with high probability, then $D_t - \gamma_k^{(i)}$ blocks of chunk $i-1$ could be released from the transmission window. It might be useful if each source has many data to send. If chunk i is decodable, then $\gamma_k^{(i+1)}$ is set to be zero for all k.

Since the number of input blocks per chunk for each source is D_t, then there are totally $K \cdot D_t$ blocks of all sources for a chunk. Then, the blocks of all sources inside a chunk can be recovered by the destination if there are $K \cdot D_t$ linearly independent CWCs. This work assumes that the blocks of chunk $i-1$ with $\gamma_k^{(i-1)}=0$ for all k could not be recovered due to lacking CWCs, i.e., $r^{(i-1)} < K \cdot D_t$. In addition, $\hat{r}^{(i)}$ denotes the expected or estimated value of $r^{(i)}$. The purpose of this work is to determine $\gamma_k^{(i)}$ for $k \in \{1, 2, \cdots, K\}$ such that the blocks of chunk i and the blocks of chunk $i-1$ can be recovered with the highest probability for an expected value of $r^{(i)}$, i.e., $\hat{r}^{(i)}$, at the destination. The determination is done by using the empirical probability distributions $\rho(r)$ and $\lambda_k(\theta_k)$ for all k.

4.5.2 Decodability

If $\gamma^{(i)}$ denotes $\sum_{k=1}^{K} \gamma_k^{(i)}$ as in the previous chapter, then after the transmitting of chunk i, there are $2 \cdot K \cdot D_t - \gamma^{(i)}$ blocks to be recovered by up to $r^{(i-1)} + r^{(i)}$ forwarded CWCs. Hence, the blocks of two contiguous chunks can be recovered if the rank of the combined matrix of the combined coding coefficient vectors of both chunks, $\left[\mathbf{C}^{(i-1)}, \mathbf{C}^{(i)}\right]$, is equal to $2 \cdot K \cdot D_t - \gamma^{(i)}$. According to the result in the previous chapter, the decodability of all blocks while overlapped chunked code (OCC) is applied at each source in multi-source multi-relay networks not only depends on the decodabilty of OCC designed by using $\rho(r)$ in single-flow transmission, but also the decodability of OCCs designed by using $\lambda_k(\theta_k)$ for all k in single-flow transmission. In other words, the recovery of the blocks of both chunks does not only depend on the selection of $\gamma^{(i)}$, but also on the selection of $\gamma_k^{(i)}$ for all k to ensure $2 \cdot D_t - \gamma_k^{(i)}$ blocks could be recovered by $\theta_k^{(i-1)} + \theta_k^{(i)}$ coded blocks for all k, where the term coded block is used instead of CWC since single-flow transmission is considered. An example of the combined matrix $\left[\mathbf{C}^{(i-1)}, \mathbf{C}^{(i)}\right]$ with the other two combined matrices related to the participation factors of all sources is shown in Figure 4.4 while taking $K = L = 2, q = 7, D_t = 10, \gamma_1^{(i)} = 2$ and $\gamma_2^{(i)} = 5$.

In this chapter, $\vartheta_k\left(\gamma_k^{(i)}\right)$ denotes the probability that the blocks of source k in chunk $i-1$ and chunk i, i.e., $2 \cdot D_t - \gamma_k^{(i)}$ blocks, can be recovered after the transmission of chunk i for an assigned value of $\gamma_k^{(i)}$ while considering in single-flow transmission with channel state corresponding to $\lambda_k(\theta_k)$. Hence, for an assigned value of $\gamma^{(i)}$, and $\vartheta\left(\gamma^{(i)}\right)$ denotes the probability that the blocks of all sources in chunk $i-1$ and chunk i, i.e., $2 \cdot K \cdot D_t - \gamma^{(i)}$ blocks, can be recovered after the transmission of chunk i for an assigned value of $\gamma^{(i)}$ while considering in single-flow transmission with channel state corresponding to $\rho(r)$. Hence, for an assigned set of values $\left(\gamma_1^{(i)}, \gamma_2^{(i)}, \cdots, \gamma_K^{(i)}\right)$, the blocks of both chunks can be

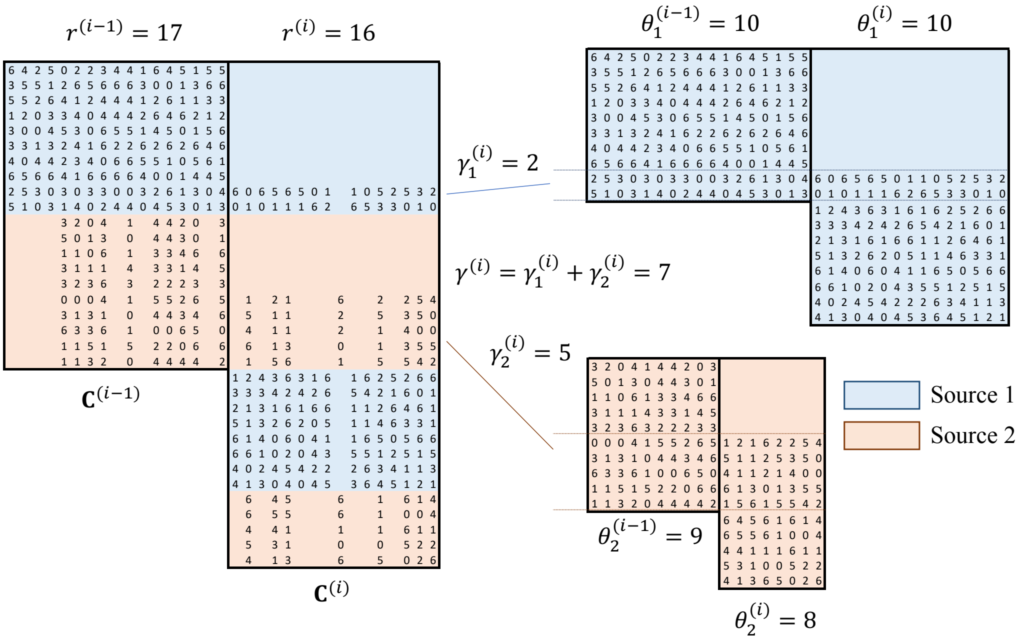


Figure 4.4: An example of $\left[\mathbf{C}^{(i-1)}, \mathbf{C}^{(i)}\right]$ while taking $K = L = 2, q = 7, D_t = 10, \gamma_1^{(i)} = 2$ and $\gamma_2^{(i)} = 5$.

recovered at the destination with probability $\vartheta_{\text{eff}}\left(\gamma^{(i)}, \gamma_1^{(i)}, \gamma_2^{(i)} \cdots, \gamma_K^{(i)}\right)$ (simply as ϑ_{eff}), where

$$\vartheta_{\text{eff}}\left(\gamma^{(i)}, \gamma_1^{(i)}, \gamma_2^{(i)} \cdots, \gamma_K^{(i)}\right) = \vartheta_m\left(\gamma^{(i)}\right) \cdot \prod_{k=1}^{K} \vartheta_k\left(\gamma_k^{(i)}\right), \tag{4.1}$$

while considering in multi-source multi-relay networks, i.e., simultaneous multiple-flow transmission.

For convenience, only $\vartheta\left(\gamma^{(i)}\right)$ corresponding to $\rho(r)$ is taken for study. Since the single-flow transmission is considered, each chunk size is $K \cdot D_t$. After transmitting the coded blocks of chunk i, with an assigned value of $\gamma^{(i)}$, the destination would have up to $r^{(i-1)} + r^{(i)}$ coded blocks to recover $2 \cdot K \cdot D_t - \gamma^{(i)}$ blocks. Thus, if $r^{(i-1)} + r^{(i)} < 2 \cdot K \cdot D_t - \gamma^{(i)}$, then the coded blocks could not be decoded. If $r_{\gamma\text{min}}^{(i)}$ is defined by

$$r_{\gamma\text{min}}^{(i)} = 2 \cdot K \cdot D_t - \gamma^{(i)} - r^{(i-1)}, \tag{4.2}$$

then $\vartheta\left(\gamma^{(i)}\right)$ can be estimated by

$$\vartheta\left(\gamma^{(i)}\right) = \sum_{r=r_{\gamma\text{min}}^{(i)}}^{K \cdot D_t} \rho(r) \cdot p_s(r), \tag{4.3}$$

where $p_s(r)$ is the probability that the rank of the combined matrix $\left[\mathbf{C}^{(i-1)}, \mathbf{C}^{(i)}\right]$ with rank array $\left[r^{(i-1)}, r\right]$ for $r \in \left\{r_{\gamma\min}^{(i)}, r_{\gamma\min}^{(i)}+1, \cdots, K \cdot D_t\right\}$ is $2 \cdot K \cdot D_t - \gamma^{(i)}$. $p_s(r)$ depends on the finite field size q. It can be estimated by conducting a simulation in MATLAB and then by conducting table lookup as in the previous chapter. In order to simplify the estimation, $p_s(r)$ can be approximately obtained as the probability that the rank of a $\gamma^{(i)} \times \left[r^{(i-1)} + r - 2 \cdot \left(K \cdot D_t - \gamma^{(i)}\right)\right]$ matrix where its elements are randomly drawn from the finite field is equal to $\gamma^{(i)}$. It is feasible because the elements of the combined matrix $\left[\mathbf{C}^{(i-1)}, \mathbf{C}^{(i)}\right]$ are randomly drawn from the finite field such that to obtain the rank array $\left[r^{(i-1)}, r\right]$ satisfying $r^{(i-1)} + r \geq 2 \cdot K \cdot D_t - \gamma^{(i)}$. A brief illustration of the approximated matrix is shown in Figure 4.5. In case that $\gamma^{(i)} = 0$, then $p_s(r)$ would be the probability

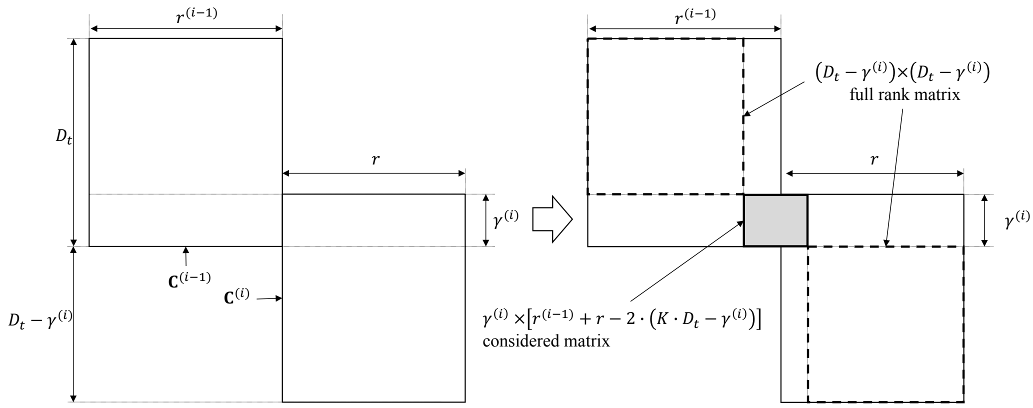


Figure 4.5: The approximated matrix, i.e., the $\gamma^{(i)} \times \left[r^{(i-1)} + r - 2 \cdot \left(K \cdot D_t - \gamma^{(i)}\right)\right]$ considered matrix.

that r linearly independent coded blocks can make an undecodable chunk with $r^{(i-1)}$ linearly independent coded blocks become decodable, and $p_s(r)$ can be obtained by using Lemma 1 in the work of [31]. Alternatively, with the approximation approach employed in this chapter, $p_s(r)$ would be the probability that the rank of $\left(2 \cdot K \cdot D_t - r^{(i-1)}\right) \times r$ matrix is $2 \cdot K \cdot D_t - r^{(i-1)}$. Similarly, $\vartheta_k\left(\gamma_k^{(i)}\right)$ can be obtained by using $\lambda_k(\theta_k)$.

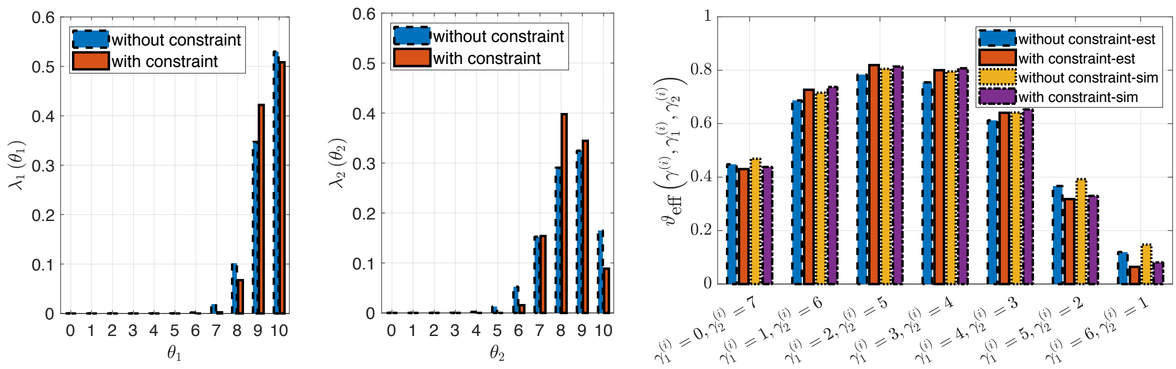
4.5.3 Selection of the Number of Overlapped Blocks

If $\hat{r}^{(i)}$ is the expected value of $r^{(i)}$, in this chapter, the corresponding value of $\gamma^{(i)}$ is determined by

$$\gamma^{(i)} = 2 \cdot K \cdot D_t - r^{(i-1)} - \hat{r}^{(i)}. \tag{4.4}$$

With a given $\gamma^{(i)}$, a set values of $\left(\gamma_1^{(i)}, \gamma_2^{(i)}, \cdots, \gamma_K^{(i)}\right)$ can be determined to obtain the maximum value of $\prod_{k=1}^{K} \vartheta_k\left(\gamma_k^{(i)}\right)$ selected from all values of $\prod_{k=1}^{K} \vartheta_k\left(\gamma_k^{(i)}\right)$ with all possible combinations of $\left(\gamma_1^{(i)}, \gamma_2^{(i)}, \cdots, \gamma_K^{(i)}\right)$ while satisfying $\sum_{k=1}^{K} \gamma_k^{(i)} = \gamma^{(i)}$.

The estimation of $\vartheta_k\left(\gamma_k^{(i)}\right)$ should be done by employing $\lambda_k(\theta_k)$ corresponding to the value of $\hat{r}^{(i)}$, i.e., with considering the correlation between $\lambda_k(\theta_k)$ and $\rho(r)$ or between $\lambda_k(\theta_k)$ and $\hat{r}^{(i)}$ (with constraint). However, employing $\lambda_k(\theta_k)$ for any value of $\hat{r}^{(i)}$, i.e., without considering the correlation (without constraint), would be acceptable since the value of r^i could not be well predicted. Figure 4.6 shows the difference effect of two considerations, which are obtained by estimation (shown in term of -est) and by simulation (shown in term of -sim). $\lambda_1(\theta_1)$ and $\lambda_2(\theta_2)$ are correspondent to matrix $\mathbf{C}^{(i)}$ in Figure 4.4 for $\hat{r}^{(i)} = r^{(i)} = 16$.



(a) (b) (c)

Figure 4.6: Example of determination of $\left(\gamma_1^{(i)}, \gamma_2^{(i)}, \cdots, \gamma_K^{(i)}\right)$ for two different consideration: with and without correlation between $\lambda_k(\theta_k)$ and $\hat{r}^{(i)}$. (a) $\lambda_1(\theta_1)$; (b) $\lambda_2(\theta_2)$; (c) ϑ_{eff}.

From Figure 4.6c, although there is deviation between $\lambda_k(\theta_k)$ of two considerations, but the desired set of $\left(\gamma_1^{(i)}, \gamma_2^{(i)}, \cdots, \gamma_K^{(i)}\right)$ is given the same even with $\hat{r}^{(i)} = r^{(i)}$. For the rest of chapter, $\lambda_k(\theta_k)$ for any value of $\hat{r}^{(i)}$ is employed.

The determination of $\gamma_k^{(i)}$ is done at the destination, and $\gamma_k^{(i)}$ can be informed by using BACK, for example, by acknowledging the last $\gamma_k^{(i)}$ blocks of chunk i are needed to be retransmitted and the rest are supposed to be successfully received, i.e., successfully received sooner or later by the transmission of next chunks. In this chapter, if the blocks of chunk $i-1$ and chunk i can be recovered, then $\gamma_k^{(i+1)}$ for all k are set to be zeros.

For the general case, undecodable chunk $i-1$ might be replaced by a combination of contiguous undecodable chunks with length ϕ, i.e., a chain of chunks from chunk $i-\phi$ to chunk $i-1$. The value of $\gamma^{(i)}$ corresponding to a given $\hat{r}^{(i)}$ is then determined by

$$\gamma^{(i)} = (\phi + 1) \cdot K \cdot D_t - \sum_{t=i-\phi+1}^{i-1} \gamma^{(t)} - \operatorname{rank}(\mathbf{C}_c) - \hat{r}^{(i)}, \tag{4.5}$$

where rank $(\mathbf{C}_c)$ is the number of linearly independent coded blocks inside the combination of the contiguous undecodable chunks. Then, $\gamma_k^{(i)}$ for all k can be determined as described above.

The decoding process can be done as in the previous chapter by using back-substitution and combination of contiguous undecodable chunk but by considering that $\gamma^{(i)}$ varies with i.

4.6 Performance Evaluation

4.6.1 Reference Schemes

This chapter takes two other transmission schemes as the reference schemes. The first transmission scheme, called RLNC/OC, employs RLNC via orthogonal channel both for the transmission from a source to its assigned relay and the transmission from that relay to the destination. The chunk size is set also to D_t. The coded blocks generated with RLNC is then encoded with NLC before transmission. For the transmissions from the source to the relay, when a relay receives D_t linear independent coded block, feedback (an ACK) is sent back to the source. However, the source could not receive the feedback immediately to stop transmitting the next coded block. On the other hand, BACK scheme is employed for the transmissions from the relay to the destination. In this work, BACK scheme is employed for the transmissions from the relays to the destination for all transmission scheme. A brief illustration of RLNC/OC scheme is shown in Figure 4.7.

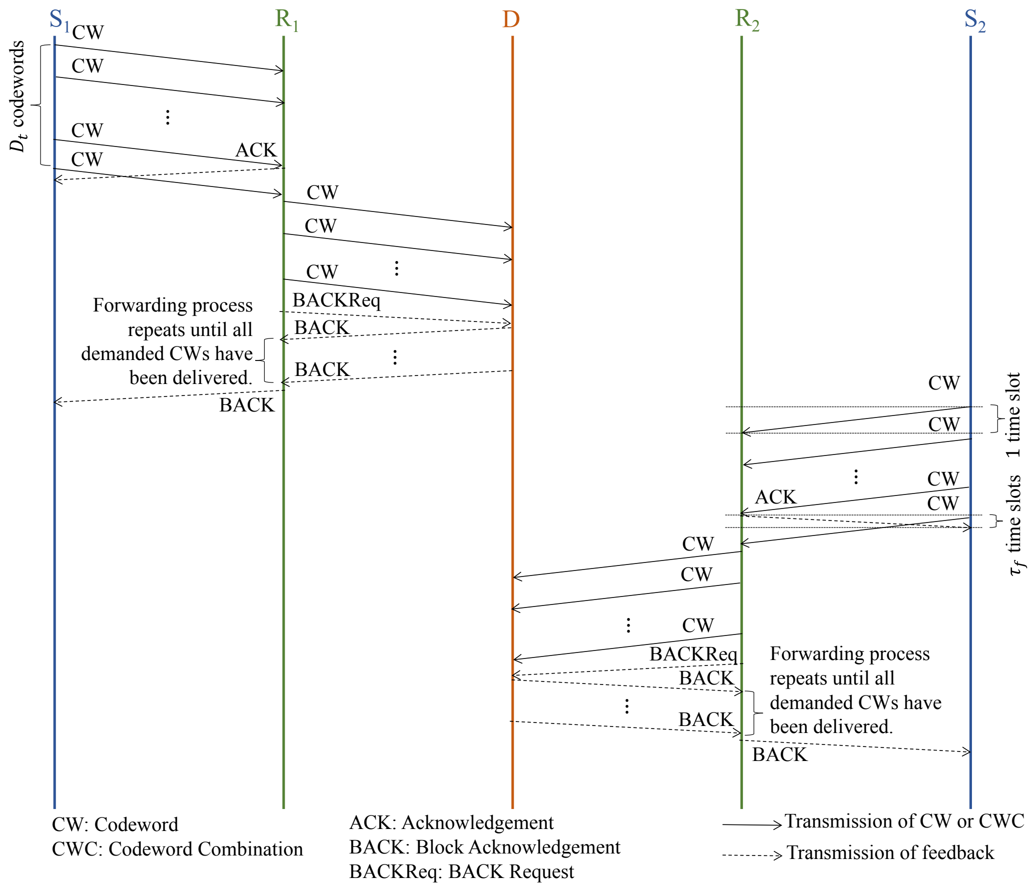


Figure 4.7: Brief illustration of RLNC/OC scheme with the case $K = L = 2$.

The other transmission scheme employs CF for each block transmission, and there is interaction between the relays and the destination before forwarding as in the proposed scheme. This scheme is called cooperative CF (CCF). For CCF, all sources transmits their CW simultaneously and wait for feedback from their assigned relay. Each relay generates up to K CWCs. After the interaction, each relay forwards the CWCs demanded from the destination by using BACK scheme. If the destination can recover the original blocks of certain sources, they send a feedback to those sources to inform the successful transmission, and those sources can transmit new blocks. If the destination could not recover the blocks of all sources, then it stores the forwarded CWCs and send feedback to inform all sources of need of retransmission. The stored CWCs are used for the recovery the sources blocks after retransmission. A brief illustration of CCF scheme is shown in Figure 4.8.

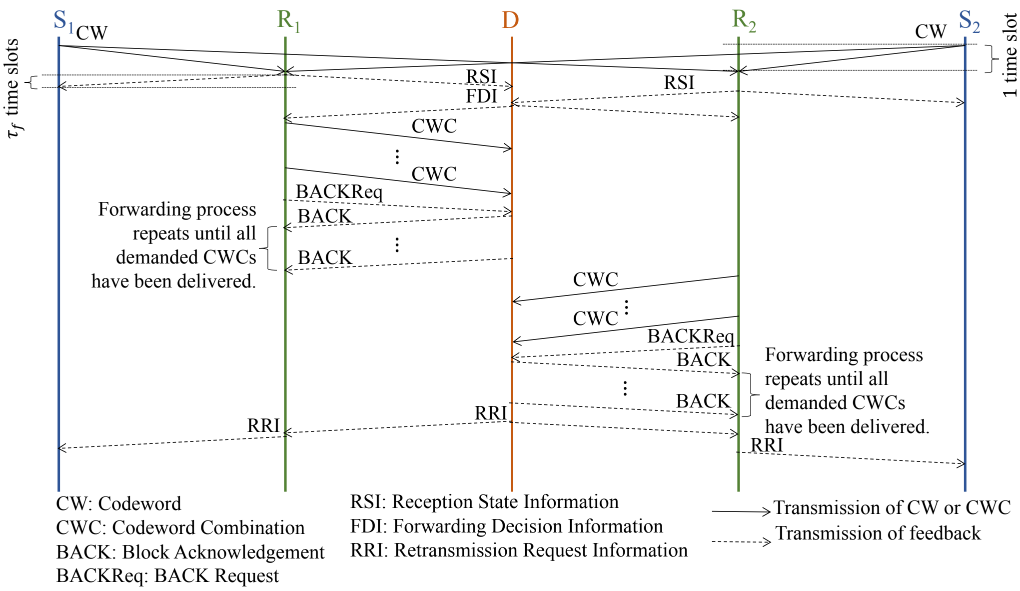


Figure 4.8: Brief illustration of CCF scheme with the case $K = L = 2$.

Channel efficiency and transmission efficiency are also taken as the term of performance to evaluate the proposed scheme in completing the transmission of a certain number of blocks in this chapter. Channel efficiency is defined by the ratio of the total number of decoded blocks of all sources to the total transmission time including the transmission time of feedback, and transmission efficiency is defined by the ratio of the total number of decoded blocks of all sources to the total number of transmissions (including transmission of feedback) taken by all nodes. In this chapter, channel efficiency, denoted by η_{eff}, corresponds to (end-to-end) network throughput, and transmission efficiency, denoted by ε_t, corresponds to (end-to-end) energy efficiency.

For the proposed transmission scheme, denoted by RLNC/CF, the ideal value of η_{eff} is $\frac{K}{1+K}$, and the ideal value of ε_t is $\frac{K}{K+K} = 0.5$. They are the same with CCF scheme. On the other hand, for RLNC/OC scheme, the ideal values of η_{eff} and ε_t are the same and equal to $\frac{K \cdot D_t}{2 \cdot K \cdot D_t} = 0.5$. The ideal values are obtained by assuming that there is no message loss and linear dependence between CWCs, and the protocol overhead is ignored. In this chapter, $\bar{\eta}$ and $\bar{\varepsilon}_t$ denote $\frac{\bar{r}}{D_t + \frac{\bar{r}}{p_{\text{RD}}}}$ and $\frac{\bar{r}}{K \cdot D_t + \frac{\bar{r}}{p_{\text{RD}}}}$, respectively, where p_{RD} is the forwarding success rate. $\bar{\eta}$ and $\bar{\varepsilon}_t$ can be counted as the upper bound of η_{eff} and ε_t, respectively, for RLNC/CF scheme. These upper bounds given here are obtained by ignoring the protocol overhead.

4.6.2 Expected Reception State

For RLNC/CF scheme, since the performance in term of channel efficiency and in term of transmission efficiency correlate to each other according to the definition, then only channel efficiency is taken for discussion. From Section 4.5.3, the values of $\gamma_k^{(i)}$ for all k depends on the expected reception state $\hat{r}^{(i)}$. Intuitively, $\bar{\eta}$ would be achieved by taking $\hat{r}^{(i)} = r_{\max}$, however the resulting η_{eff} probably could not attain $\bar{\eta}$ as expected because of the linear dependence between the CWCs that belong to undecodable contiguously overlapped chunks. The linear dependence is caused by the small size of finite field, i.e., the value of q, or by the small taken value of $\gamma_k^{(i)}$, which causes low $\vartheta_k\left(\gamma_k^{(i)}\right)$. Furthermore, high computational complexity of decoding would be caused because there might be combinations of undecodable chunks with long length according to the previous chapter. To describe the speed of recovery, τ_{rcv} denotes the average period of time from when a block is added into the transmission window to the time when block is recovered at the destination while only successfully recovered blocks are considered. τ_{rcv} is called reception delay in this chapter.

On the other hand, if considering the case that $\gamma_k^{(i-1)} = 0$ for all k and $r^{(i-1)} < K \cdot D_t$, i.e., $\phi = 1$, a value of $\hat{r}^{(i)}$ can be found for each retransmission to obtain ϑ_{eff} close to one. However, the overhead, i.e. the decrease in channel efficiency, to achieve this purpose would be high, although this can speed up the recovery of undecodable blocks. Alternatively, to ensure the recovery with high probability and better channel efficiency, a value of $\hat{r}^{(i)}$, denoted by r_d, can be found such that $\eta_d = \vartheta_{\text{eff}} \cdot \left(r^{(i-1)} + \hat{r}^{(i)}\right)$ is maximum. However, it might not work for the case $\phi > 1$ because the decodability of combination of contiguous undecodable chunks from chunk $i-\phi-2$ to chunk $i-2$ might still rely on the success rate of back-substitution. This success rate is correspondent to the probability that a matrix randomly drawn from $\mathbb{F}_q^{\left(D_t-\gamma_k^{(t+1)}\right)\times\theta_k^{(t)}}$ is full rank, where $t \in \{i-\phi-2, i-\phi-1, \cdots, i-2\}$.

The performance in term of η_{eff} and τ_{rcv} for a certain purpose for the selection of $\hat{r}^{(i)}$ can be estimated by using Markov chain, for example, however, the number of states might grow exponentially with the different values of r, θ_k and ϕ. Hence, it is seemly difficult to find the optimal performance of RLNC/CF scheme. However, the better performance can be obtained by conducting searching while employing RLNC/CF scheme. For example, searching process can be done by setting the threshold of ϑ_{eff} varying from high to low while selecting γ_k for all k. By taking the scenario as shown in Figure 4.1, $\text{SNR}_{\text{SR}}^{(1,1)} = \text{SNR}_{\text{SR}}^{(2,2)} = \text{SNR}_{\text{RD}}^{(1)} = \text{SNR}_{\text{RD}}^{(2)} = 35$ dB, $\text{SNR}_{\text{SR}}^{(1,2)} \in \{0, 20, 35\}$ dB, $\text{SNR}_{\text{SR}}^{(2,1)} = 0 \to 35$ dB, $D_t = 10$, $\tau_f = 0.05$ and $E_8/7E_8$ as NLC, the simulation results in term of channel efficiency and reception delay for different selections of $\hat{r}$ are shown in Figure 4.9. The simulation frequency is 100 times. Each simulation terminates when the 1024-th block or the latter of all sources is successfully received.

From Figure 4.9, the channel efficiency increases while the threshold of ϑ_{eff} decreases. However, there are some unexpected degradation of channel efficiency when the threshold is too small, for example, in Figure 4.9a and 4.9b for $\vartheta_{\text{eff}} \geq 0.1$, and in Figure 4.9c for $\vartheta_{\text{eff}} \geq 0.1$ and for $\vartheta_{\text{eff}} \geq 0.2$. It is due to small selected value of γ_k that causes unsatisfied decodability. On the other hand, the reception delay is longer when the threshold is lower. Special selection $\hat{r} = \bar{r}$ can provide acceptable and stable performance in channel efficiency but with long reception delay. For the other special selection $\hat{r} = r_d$, it can provide the shortest reception delay, but it does not provide the lowest channel efficiency, because it must require a high decodability to obtain the maximum of η_d, but not a perfect decodability. Therefore, there is trade-off between the channel efficiency and the reception delay for the proposed scheme. If there are constraint on the reception delay, to obtain an acceptable performance, the selection of $\hat{r}$ might be done by taking $\hat{r} = \bar{r}$ when $1 \leq \phi < \phi_{\max} - 1$, and by taking $\hat{r} = r_d$ when $\phi = \phi_{\max} - 1$, then the delay reception should be roughly bounded by $\tau_\phi = \phi_{\max} \cdot \left(D_t + \dfrac{\bar{r}}{p_{\text{RD}}} \right)$ time slots, excluding the transmission time of feedback. For the latter case, the determination of $\gamma_k^{(i)}$ is done by using similar way as in (4.5). This selection is called joint selection, and its corresponding selection of $\hat{r}$ is denoted by r_j.

In addition to reception delay, the failure rate, denoted by ϵ_{eff}, is defined by the ratio of total number of blocks that have not been recovered or their reception delay does exceed the bounded τ_ϕ to the considered total number of blocks. From Figure 4.9, selection $\hat{r} = r_j$ with $\phi_{\max} = 4$ can reduce reception delay and failure rate comparing to selection $\hat{r} = \bar{r}$, but there is some degradation of channel efficiency. In addition, the failure rate of select $\hat{r} = r_j$ are not much low as the failure rate of selection $\hat{r} = r_d$, it might reflect that selection $\hat{r} = r_j$ still inherits the lacking decodability of selection $\hat{r} = \bar{r}$. The other

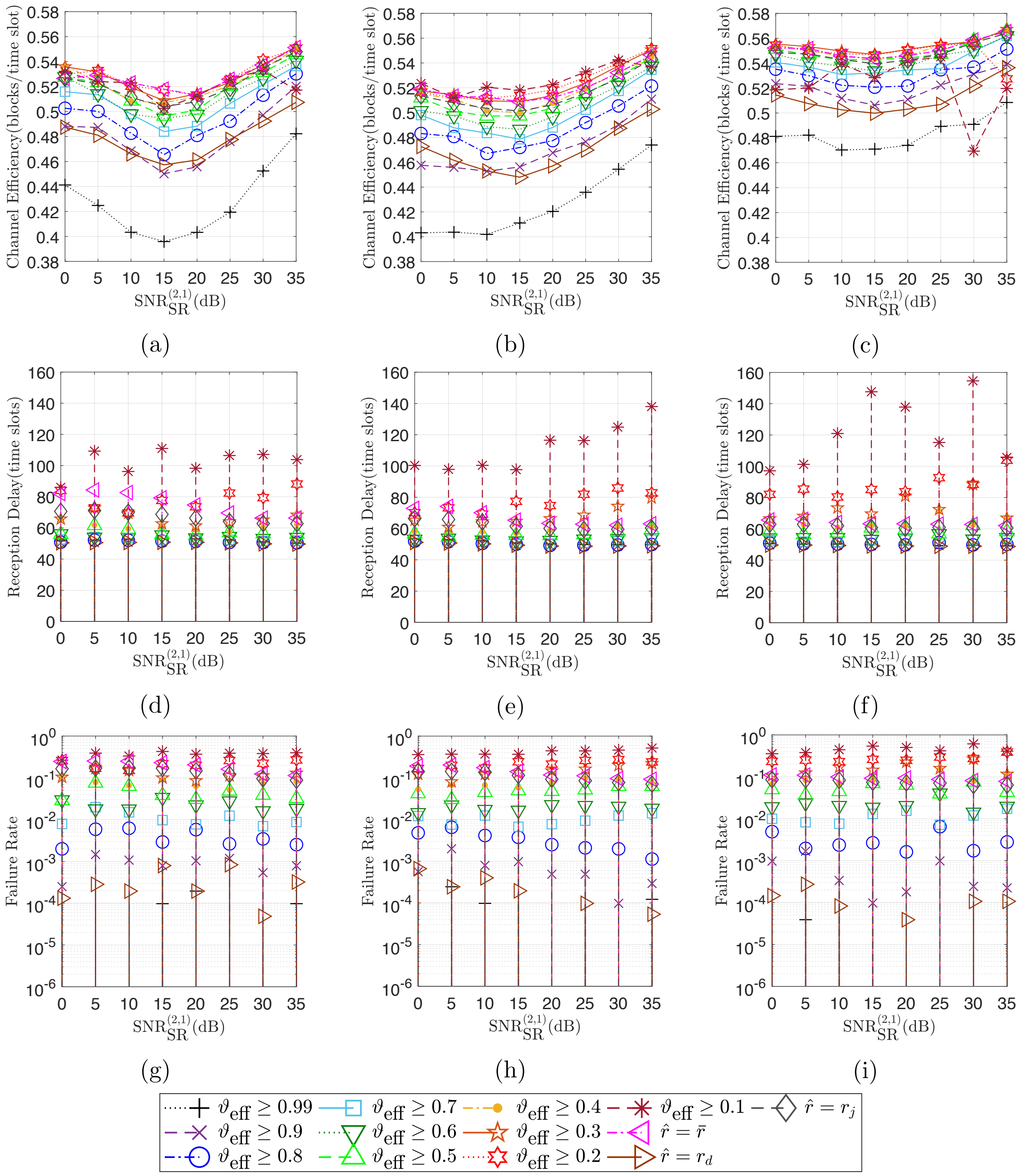


Figure 4.9: Channel efficiency and reception delay with different selections of $\hat{r}$. (a) η_{eff} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 5$ dB; (b) η_{eff} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 20$ dB; (c) η_{eff} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 35$ dB; (d) τ_{rcv} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 5$ dB; (e) τ_{rcv} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 20$ dB; (f) τ_{rcv} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 35$ dB; (g) ϵ_{eff} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 5$ dB; (h) ϵ_{eff} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 20$ dB;(i) ϵ_{eff} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 35$ dB.

selection that might ensure the decodability is to obtain $\prod_{k=1}^{K} \vartheta_k \left(\gamma_k^{(i)}\right)$ close to one while determining $\gamma_k^{(i)}$. However, this selection might still do the trade-off between channel efficiency and failure rate.

4.6.3 Numerical Results

In order to do comparison with the reference schemes, this work employed the same simulation parameters as in Section 4.6.2, but only took joint selection with $\phi_{\max} = 4$. The simulation results in term of channel efficiency and transmission efficiency of three schemes are shown in Figure 4.10. The performance in term of reception delay is not mentioned here because it is seemly not a fair comparison since the reception rely on the chunk size D_t.

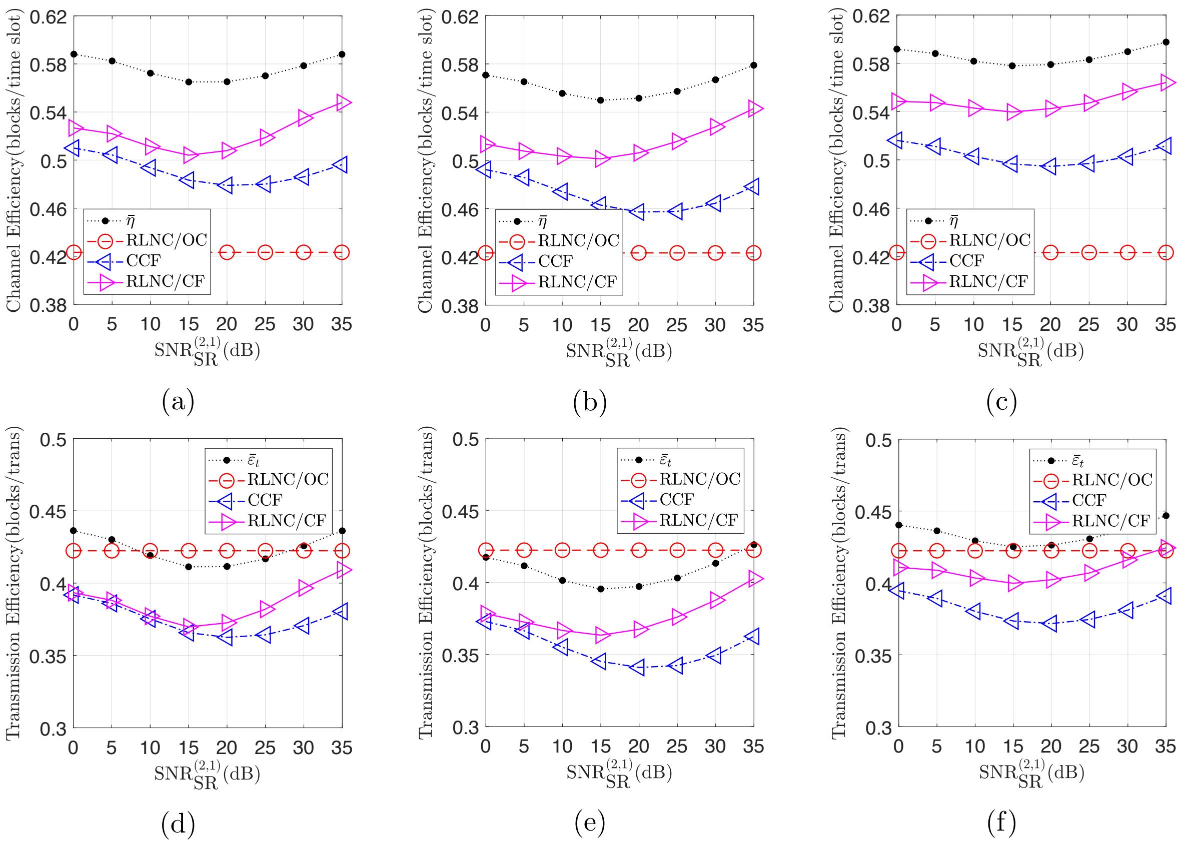


Figure 4.10: Channel efficiency and transmission efficiency for the case of two-source two-relay network. (a) η_{eff} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 5$ dB; (b) η_{eff} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 20$ dB; (c) η_{eff} for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 35$ dB; (d) ε_t for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 5$ dB; (e) ε_t for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 20$ dB; (f) ε_t for $\mathrm{SNR}_{\mathrm{SR}}^{(1,2)} = 35$ dB.

From Figure 4.10, RLNC/CF scheme increases channel efficiency by 8.08% on average if comparing to CCF scheme, where there should be no difference in the ideal case. This shows the impact from the protocol overhead and inefficient retransmission. It is because that all sources have to retransmit their previous block when these blocks could not be recovered during the last transmission round, although there is some stored CWCs. An efficient way can be done by allowing some sources to transmit a new block by assuming that the previously sent blocks will be recovered sooner or later. It is similar concept as of RLNC/CF scheme where the retransmission is to try to have the previous blocks recovered and some new blocks to be transmitted. Hence, high protocol overhead can be reduced, and the retransmission would become more efficient. There is similar improvement for the performance in term of transmission efficiency.

On the other hand, RLNC/CF scheme also increases channel efficiency by 24.83% on average if comparing to RLNC/OC scheme, which is supposed to be 33.33% in the ideal case. The deviation might be because that the block success rate is higher when orthogonal channel is employed comparing to when non-orthogonal channel is employed for the transmissions from the sources to the relays. It is because that the correctness of the data in the scheme employing CF not only depends on the AWGN built inside, but also depends on the error caused from forcing the channel coefficients to the integer values. With the same reason, the transmission efficiency of RLNC/CF is 92.49% of the transmission efficiency of RLNC/OC, while they should be the same in the ideal case, and although $\bar{\varepsilon}_t$ sometimes is higher than the transmission efficiency of RLNC/OC. This might reflects the overhead of the proposed scheme. Therefore, applying simultaneous transmission from the sources to the relays might take the trade-off between the improvement of channel efficiency and the reduction of transmission efficiency if comparing to a scheme employing orthogonal channel, RLNC/OC for example. In addition, whether letting many sources and just few sources to transmits their data simultaneously, i.e., allocation of channel use, would be considered, if the performance in term of transmission efficiency or energy efficiency is emphasized.

4.7 Summary

This chapter proposed a retransmission scheme, called RLNC/CF, where contiguously overlapped chunked code is applied before NLC at each source in multi-source multi-relay networks, and the number of blocks taken from the previous chunk at all sources can be determined to obtain the different purposes, high channel efficiency or quick recovery of the blocks of the previous chunks and the retransmitting chunk. Block ACK scheme is employed, and lossless transmission of feedback is assumed. From the numerical results, RLNC/CF scheme can provide some improvement in channel efficiency over a cooperative CF scheme, showing its advantage in reducing the protocol overhead and improving the efficiency of retransmission. However, in term of transmission efficiency, applying RLNC/CF scheme might need to do a trade-off between the improvement of channel efficiency and the degradation of transmission efficiency if comparing to a transmission scheme employing orthogonal channel. Alternatively, an allocation of channel use might be considered in order to obtain the desired performance while the energy consumption is focused.

Chapter 5

Conclusion and Future Works

5.1 Summary, Contribution and Discussion

The purpose of this dissertation is to investigate the performance of an IANC approach, OCC, employing with a physical-layer network coding approach, CF based on NLC, since these network coding techniques are expected to improve the network performance in term of network throughput, energy efficiency and latency when the network density is high. To achieve these purposes, a design of OCC and a feedback-based retransmission scheme employing contiguously OCC for multi-source multi-relay network are proposed. Both proposed works employ the empirical probability distributions in designing the proposed schemes, in estimating the network performance and in determining the parameters used in the proposed schemes. The performance in term of channel efficiency, transmission efficiency and decoding complexity (or reception delay) are considered instead of network throughput, energy efficiency and latency, respectively.

For the proposed design of OCC, a decodability condition which is essential for the design of OCC no matter for single-flow transmission or simultaneous multi-flow transmission (i.e., multi-source multi-relay networks) was provided. This condition splits the design of OCC for multi-source multi-relay networks into multiple designs of OCC in single-flow transmission. Hence, The OCCs that were proposed for single-flow transmission are applicable for the design of OCC in multi-source multi-relay networks. To investigate the performance of OCC in multi-source multi-relay network, an OCC with contiguously overlapping but non-round-end fashion was employed for the design. From the numerical results with the application of contiguously OCC, the performance of the designed code in term of channel efficiency and transmission efficiency depends on the level of the protocol overhead (the transmission time of feedback and the loss of feedback), and the design overhead is high due to not outstanding performance of contiguously OCC itself in single-flow transmission. However, the design with contiguously OCC is simple to obtain the desired performance, e.g., the highest channel efficiency or high decodability with the highest channel efficiency or low computational complexity (short reception delay), via the estimation of these terms of performance, which can be done with low complexity. In addition, the numerical results ensured that the design of OCC for multi-source multi-relay networks not only depends on the empirical probability distribution of the number of linearly independent qualified codeword combinations of a chunk, but also the empirical probability distributions of the participation factors of all sources into these codeword combinations. Moreover, the channel efficiency provided by the OCC designed for multi-source multi-relay networks can approach the channel efficiency provided by the OCC designed using the empirical probability distribution of the number of linearly independent qualified codeword combinations of a chunk (for single-flow transmission) if the the empirical probability distributions of the participation factors of all sources are dense around the chunk size. This aspect might motivate the consideration of relays selection for better performance. On the other hand, this work provided a decoding scheme for this application with contiguously OCC, which enables the destinations to begin decoding without waiting until all chunks have been received as for some rateless codes (fountain codes, BATs codes, etc.) while back-substitution is not applicable. The proposed design gives a starting point for the designs with the other OCCs, e.g. BATs codes, which outperform contiguously OCC in single-flow transmission. However, the way to design with low complexity or overhead should be considered since the design complexity increases with the number of sources. On the other hand, the proposed design was only for one-hop transmission, i.e., from the sources to the relays, then the advantage of OCC to overcome the protocol overhead does not completely show up. The application of the proposed designed with contiguously OCC might gain more advantages from the other kinds of topology, such as cross topology and line network with multiple relay stages, where the advantage can be obtained from high network density and overhearing. In addition, contiguously OCC can be counted as an erasure code, where the overlapped blocks can be served in checking the correctness of the blocks of two or more contiguously overlapped chunks. Moreover, with an assigned length of chain of chunks, this OCC can be designed to obtain low block error rate if belief propagation decoding approach is applied for example, while there might still be a trade-off between channel efficiency and decodability as shown in this work.

For the proposed feedback-based retransmission scheme, RLNC/CF, a BACK scheme is provided for end-to-end transmission in multi-source multi-relay networks to reduce the protocol overhead, and it is suitable for the case of multiple relays. Differently from traditional retransmission scheme, by employing contiguously OCC with feedback, some blocks ,i.e., overlapped blocks, can be randomly selected from the previous chunk for retransmission and the other blocks are expected to be recovered after the transmission of next chunks and can be released from the transmission window. The determination of the number of overlapped blocks for all sources using the empirical probability distributions for an expected reception state of the next chunk with the highest decodability is provided. The performance of the different selections of expected reception state, which are expected to provide different level of decodability of the unrecovered blocks of the previous chunks and of the next chunk are investigated. From the numerical results, the increment of channel efficiency and transmission efficiency can be provided by comparing to a cooperative CF scheme, showing its efficiently taking use of stored codeword combinations and its reducing the protocol overhead. The increment is 8.61% of the channel efficiency of the cooperative CF scheme when the joint selection of expected reception state is used. However, RLNC/CF scheme does a trade-off between the increase in channel efficiency and the decrease in transmission efficiency if comparing to a transmission scheme employing orthogonal channel, because the block success rate via orthogonal channel is higher than via non-orthogonal channel. Therefore, how to group the sources to use orthogonal channel or the allocation of channel use should be considered. On the other hand, the selection of expected reception state (or the compensation for the loss of the next chunk transmission) decides what will the performance of RLNC/CF scheme provide. This work gives the selection of expected reception state based on the threshold of the expected decodability that the expected reception state can provide. There is surely a trade-off between channel efficiency and reception delay (or failure rate). However, if deadline of the block reception is tolerant, then the better performance in term of channel efficiency can be obtained. The better selection for desired performance can be done by searching while employing RLNC/CF scheme. On the other hand, although the loss of feedback is not considered in the proposed scheme, but there will be only the overhead caused from the transmissions of feedback (e.g., BACK, BACKReq, etc.), not from wasted codeword transmissions which cost significantly. In addition, the proposed scheme might still work well when the number of sources becomes large since the main degradation of performance is caused by the protocol overhead. Nevertheless, the channel access allocation should be considered because the reception state while multiple sources simultaneously transmit their codeword via the same spectral channel becomes worse when the number of sources becomes larger. If the protocol overhead is high, then the transmission employing OCC/CF can be employed instead. Therefore, the estimation or the upper of the performance of RLNC/CF scheme should be considered to make decision which transmission scheme to be used. In addition, since the blocks that are not taken for the retransmission can be released from the transmission windows, then RLNC/CF scheme might have potential to reduce the congestion if the sources play a role of relay for the other senders and the deadline of block reception is tolerant.

5.2 Future Works

The work described in this dissertation might not completely to meet the vision. There are still some points for extension and for putting consideration into. The first extension is the application in the other topology, such as cross topology, line network, etc. For the application in cross topology, the main advantage comes from overhearing, i.e., receiving the unintended data from the other source, and the high network density that can make cross topology network founded. However, since each destination only requires the blocks from its corresponding source, therefore how to extract the desired data efficiency from the codeword combinations computed at the relay will be considered. In addition, individually decoding, i.e., decoding the desired blocks first, will be also considered. Furthermore, how to set up the cross topology network, i.e., hank-shaking mechanism, searching routing path and topology changing will be considered. The collected data related to channel state, the empirical probability distributions can serve to deal with routing path searching. Beaconing might be used by a new joining node to inform its existence. For line network, ensuring that the computation rate at further hop meets an threshold will be a challenge because the achieved computation rate at the next hop is up to the computation rate at the previous hop if the relays do not decode all source codeword before forwarding. Furthermore, in order to take advantage of overhearing, the linear combination of CWC computed at the front hop and the CWC computed at the back hop (separated by an intermediate node) should be acceptable by the intermediate node. Hence, an appropriate NLC will be considered.

As mentioned in Chapter 4, allocation of channel use might be a solution to obtain the desired performance in transmission efficiency. In addition, a maximum channel efficiency with this constraint might be obtained by using the empirical probability distributions collected at the destination with the aid of the relays. The allocation of channel use might be specific and random, while random access is seemly preferred. Hence, how to efficiently gathering or computing to obtain all possible allocations will be considered. Furthermore, how to select the appropriate allocation is also considered while the correlation between the two probability distributions (rank probability distribution and probability distribution of participation factor of each source) should be taken account into. In addition, since the work mainly relies on the empirical probability distributions, hence the scheme for managing the collected data, for example when there are error in collected data, will be considered too.

In order to decide whether to use OCC/CF or RLNC/CF, an upper bound of the performance of RLNC/CF will be sufficient for the decision, but to search for the optimal performance, i.e., which selection of expected channel state to be used for each retransmission, the performance analysis or the performance estimation of RLNC/CF will be considered. The estimation of RLNC/CF might be done with huge data needed somehow, for example, by using Markov chain. On the other hand, the simplified design of OCC employing the other OCC, e.g. BATs code, will be considered too to obtain better performance, especially, in line network.

At the end, the implementation of CF will be considered to prove the proposed work could also work not only in theory. GNU Radio might be employed to implement CF in a typical network topology for the start before the implementation of the proposed schemes. How to manage the synchronization, channel estimation at relays, etc., will be considered and studied.

5.3 Conclusion

With the growing network density, with existing limited channel bandwidth, new channel resource or new transmission technology have been considered and employed, such as beamforming, millimeter wave, light fidelity, etc. to avoid interference. However, when these new resources still could not satisfy the demanded network connections, employing non-orthogonal channel will be still a solution. Since the block error rate decreases when the number of senders employing non-orthogonal is larger, grouping the senders can be done by allocating orthogonal channel for each group. How to efficiently take advantage of non-orthogonal transmission among each group, network coding is a promised technique to improve the network performance by changing the way of transmission, and this dissertation tries to make a contribution into this consideration.

Bibliography

[1] W. P. Cisco, "Cisco visual networking index: Global mobile data traffic forecast update, 2016–2021," 2017.

[2] L. Jiang, D.-Y. Liu, and B. Yang, "Smart home research," in *Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on*, IEEE, vol. 2, 2004, pp. 659–663.

[3] L. G. Giordano and L. Reggiani, *Vehicular Technologies-Deployment and Applications*. 2013.

[4] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, "A survey on low latency towards 5g: Ran, core network and caching solutions," *IEEE Communications Surveys & Tutorials*, 2018.

[5] Y. Teng, M. Liu, F. R. Yu, V. C. Leung, M. Song, and Y. Zhang, "Resource allocation for ultra-dense networks: A survey, some research issues and challenges," *IEEE Communications Surveys & Tutorials*, 2018.

[6] M. Kobayashi, J. Boutros, and G. Caire, "Successive interference cancellation with siso decoding and em channel estimation," *IEEE Journal on selected areas in Communications*, vol. 19, no. 8, pp. 1450–1460, 2001.

[7] J. C. Liberti Jr, S. Moshavi, and P. G. Zablocky, *Successive interference cancellation*, US Patent 8,670,418, 2014.

[8] M. S. Rahman, Y. Li, and B. Vucetic, "An iterative zigzag decoding for combating collisions in wireless networks," *IEEE Communications Letters*, vol. 14, no. 3, 2010.

[9] TELCOMA, *Non-orthogonal multiple access (noma) for 5g systems*, https://telcomaglobal.com/blog/18015/noma-5g-non-orthogonal-multiple-access/, [Online; accessed 06-August-2018], 2018.

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

[11] B. Nazer and M. Gastpar, "Compute-and-forward: Harnessing interference through structured codes," *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6463–6486, 2011.

[12] B. Nazer, "Successive compute-and-forward," in *22th International Zurich Seminar on Communications (IZS)*, Eidgenössische Technische Hochschule Zürich, 2012.

[13] Y. Niu, Y. Li, D. Jin, L. Su, and A. V. Vasilakos, "A survey of millimeter wave communications (mmwave) for 5g: Opportunities and challenges," *Wireless Networks*, vol. 21, no. 8, pp. 2657–2676, 2015.

[14] Z. Ding, P. Fan, and H. V. Poor, "Random beamforming in millimeter-wave noma networks," *IEEE access*, vol. 5, pp. 7667–7681, 2017.

[15] Z. Liu, M. Li, L. Lu, C.-K. Chan, S.-C. Liew, and L.-K. Chen, "Optical physical-layer network coding," *IEEE Photonics Technology Letters*, vol. 24, no. 16, pp. 1424–1427, 2012.

[16] R. Ahlswede, N. Cai, S.-Y. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on information theory*, vol. 46, no. 4, pp. 1204–1216, 2000.

[17] C. Fragouli, D. Katabi, A. Markopoulou, M. Medard, and H. Rahul, "Wireless network coding: Opportunities & challenges," in *Military Communications Conference, 2007. MILCOM 2007. IEEE*, IEEE, 2007, pp. 1–8.

[18] M. Médard and A. Sprintson, *Network coding: Fundamentals and applications*. Academic Press, 2011.

[19] S.-Y. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE transactions on information theory*, vol. 49, no. 2, pp. 371–381, 2003.

[20] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Physical Communication*, vol. 1, no. 1, pp. 3–20, 2008.

[21] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4413–4430, 2006.

[22] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proceedings of the annual Allerton conference on communication control and computing*, The University; 1998, vol. 41, 2003, pp. 40–49.

[23] P. Maymounkov, N. J. Harvey, and D. S. Lun, "Methods for efficient network coding," in *Proc. 44th Annual Allerton Conference on Communication, Control, and Computing*, 2006, pp. 482–491.

[24] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, *Trading structure for randomness in wireless opportunistic routing*, 4. ACM, 2007, vol. 37.

[25] C. Qin, Y. Xian, C. Gray, N. Santhapuri, and S. Nelakuditi, "I^2mix: Integration of intra-flow and inter-flow wireless network coding," in *Sensor, Mesh and Ad Hoc Communications and Networks Workshops, 2008. SECON Workshops' 08. 5th IEEE Annual Communications Society Conference on*, IEEE, 2008, pp. 1–6.

[26] J. W. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "A digital fountain approach to reliable distribution of bulk data," *ACM SIGCOMM Computer Communication Review*, vol. 28, no. 4, pp. 56–67, 1998.

[27] A. Heidarzadeh and A. H. Banihashemi, "Overlapped chunked network coding," in *Information Theory (ITW 2010, Cairo), 2010 IEEE Information Theory Workshop on*, IEEE, 2010, pp. 1–5.

[28] ——, "Analysis of overlapped chunked codes with small chunks over line networks," in *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, IEEE, 2011, pp. 801–805.

[29] Y. Li, E. Soljanin, and P. Spasojevic, "Effects of the generation size and overlap on throughput and complexity in randomized linear network coding," *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 1111–1123, 2011.

[30] S. Yang and R. W. Yeung, "Batched sparse codes," *IEEE Transactions on Information Theory*, vol. 60, no. 9, pp. 5322–5346, 2014.

[31] B. Tang and S. Yang, "An ldpc approach for chunked network codes," *IEEE/ACM Transactions on Networking*, vol. 26, no. 1, pp. 605–617, 2018.

[32] A. Apavatjrut, C. Goursaud, K. Jaffrès-Runser, and J.-M. Gorce, "Fountain codes and network coding for wsns," *Network coding*, pp. 27–72, 2012.

[33] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "Xors in the air: Practical wireless network coding," *IEEE/ACM Transactions on Networking (ToN)*, vol. 16, no. 3, pp. 497–510, 2008.

[34] N. Lin, R. Ngeth, K. Sriviriyakul, Y. Lim, and Y. Tan, "Necomac: Network coding aware mac protocol for multirate wireless networks," in *Advanced Information Networking and Applications (AINA), 2016 IEEE 30th International Conference on*, IEEE, 2016, pp. 384–391.

[35] H. Seferoglu, A. Markopoulou, and K. Ramakrishnan, "I^2nc: Intra-and inter-session network coding for unicast flows in wireless networks," in *INFOCOM, 2011 Proceedings IEEE*, IEEE, 2011, pp. 1035–1043.

[36] L. Hai, H. Wang, Y. Liu, J. Wang, and Z. Tang, "R^2nc: Robust inter-session network coding in lossy wireless networks," *IET Communications*, vol. 9, no. 2, pp. 184–195, 2014.

[37] Q. Huang, K. Sun, X. Li, and D. O. Wu, "Just fun: A joint fountain coding and network coding approach to loss-tolerant information spreading," in *Proceedings of the 15th ACM international symposium on Mobile ad hoc networking and computing*, ACM, 2014, pp. 83–92.

[38] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: Analog network coding," *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 4, pp. 397–408, 2007.

[39] L. Dai, B. Wang, Z. Ding, Z. Wang, S. Chen, and L. Hanzo, "A survey of non-orthogonal multiple access for 5g," *IEEE Communications Surveys & Tutorials*, 2018.

[40] B. Nazer and M. Gastpar, "Reliable physical layer network coding," *Proceedings of the IEEE*, vol. 99, no. 3, pp. 438–460, 2011.

[41] L. Wei and W. Chen, “Compute-and-forward network coding design over multi-source multi-relay channels,” *IEEE Transactions on Wireless Communications*, vol. 11, no. 9, pp. 3348–3357, 2012.

[42] D. Kramarev, A. Sakzad, and E. Viterbo, “Implementation of a two-way relay network with compute-and-forward in gnu radio,” *Transactions on Emerging Telecommunications Technologies*, vol. 27, no. 4, pp. 484–493, 2016.

[43] K. N. Pappi, P. D. Diamantoulakis, H. Otrok, and G. K. Karagiannidis, “Cloud compute-and-forward with relay cooperation.,” *IEEE Trans. Wireless Communications*, vol. 14, no. 6, pp. 3415–3428, 2015.

[44] M. El Soussi, A. Zaidi, and L. Vandendorpe, “Compute-and-forward on a multi-user multi-relay channel,” *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 589–592, 2014.

[45] M. Hejazi, S. M. Azimi-Abarghouyi, B. Makki, M. Nasiri-Kenari, and T. Svensson, “Robust successive compute-and-forward over multiuser multirelay networks,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 10, pp. 8112–8129, 2016.

[46] J. Goseling, M. Gastpar, and J. H. Weber, “Random access with physical-layer network coding,” *IEEE Transactions on Information Theory*, vol. 61, no. 7, pp. 3670–3681, 2015.

[47] S. Ashrafi, C. Feng, and S. Roy, “Compute-and-forward for random-access: The case of multiple access points,” *IEEE Transactions on Communications*, 2018.

[48] J. He and S.-C. Liew, “Arq for physical-layer network coding,” *IEEE Transactions on Mobile Computing*, vol. 15, no. 7, pp. 1614–1631, 2016.

[49] M. A. Shokrollahi, S. Lassen, and R. Karp, *Systems and processes for decoding chain reaction codes through inactivation*, US Patent 6,856,263, 2005.

[50] F. Lázaro, G. Liva, and G. Bauch, “Inactivation decoding of lt and raptor codes: Analysis and code design,” *IEEE Transactions on Communications*, vol. 65, no. 10, pp. 4114–4127, 2017.

[51] Z. Chen, P. Fan, and K. B. Letaief, “Compute-and-forward: Optimization over multisource–multirelay networks,” *IEEE Transactions on Vehicular Technology*, vol. 64, no. 5, pp. 1806–1818, 2015.

[52] B. Nazer, V. R. Cadambe, V. Ntranos, and G. Caire, “Expanding the compute-and-forward framework: Unequal powers, signal levels, and multiple linear combinations,” *IEEE Transactions on Information Theory*, vol. 62, no. 9, pp. 4879–4909, 2016.

[53] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[54] U. Fincke and M. Pohst, “Improved methods for calculating vectors of short length in a lattice, including a complexity analysis,” *Mathematics of computation*, vol. 44, no. 170, pp. 463–471, 1985.

[55] M. N. Hasan and B. M. Kurkoski, “Practical compute-and-forward approaches for the multiple access relay channel,” in *Communications (ICC), 2017 IEEE International Conference on*, IEEE, 2017, pp. 1–6.

[56] M. Hejazi and M. Nasiri-Kenari, “Simplified compute-and-forward and its performance analysis,” *IET Communications*, vol. 7, no. 18, pp. 2054–2063, 2013.

[57] D. Silva, W. Zeng, and F. R. Kschischang, “Sparse network coding with overlapping classes,” in *Network Coding, Theory, and Applications, 2009. NetCod’09. Workshop on*, IEEE, 2009, pp. 74–79.

[58] M. Luby, “Lt codes,” in *Foundations of Computer Science, 2002. Proceedings. The 43rd Annual IEEE Symposium on*, IEEE, 2002, pp. 271–280.

[59] A. Shokrollahi, “Raptor codes,” *IEEE transactions on information theory*, vol. 52, no. 6, pp. 2551–2567, 2006.

[60] P. Maymounkov, “Online codes,” Technical report, New York University, Tech. Rep., 2002.

[61] B. Tang and S. Yang, “An improved design of overlapped chunked codes,” in *Communications (ICC), 2016 IEEE International Conference on*, IEEE, 2016, pp. 1–6.

[62] D. J. MacKay, “Fountain codes,” *IEE Proceedings-Communications*, vol. 152, no. 6, pp. 1062–1068, 2005.

[63] R. V. Hogg, J. McKean, and A. T. Craig, *Introduction to mathematical statistics*. Pearson Education, 2005.

[64] R. Ngeth, Y. Lim, B. M. Kurkoski, and Y. Tan, "A design of overlapped chunked code over compute-and-forward in multi-source multi-relay networks," in *2018 IEEE Global Communications Conference: Wireless Communications*, IEEE, (presented).

[65] A. Sakzad, E. Viterbo, J. Boutros, and Y. Hong, "Phase precoded compute-and-forward with partial feedback," in *Information Theory (ISIT), 2014 IEEE International Symposium on*, IEEE, 2014, pp. 2117–2121.

[66] "Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (mac)and physical layer (phy) specifications amendment 5: Enhancements for higher throughput," *IEEE Std 802.11n-2009 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, and IEEE Std 802.11w-2009)*, pp. 1–565, 2009.

[67] Y. Chen, S. Emeott, and R. R. Choudhury, "Wlc24-6: An analytical model of block acknowledgement and selective retransmission in an 802.11 e wlan network," in *IEEE Globecom 2006*, IEEE, 2006, pp. 1–5.

[68] S. Chieochan and E. Hossain, "Wireless fountain coding with ieee 802.11 e block ack for media streaming in wireline-cum-wifi networks: A performance study," *IEEE Transactions on Mobile Computing*, vol. 10, no. 10, pp. 1416–1433, 2011.

Publications

Journals

[1] R. Ngeth, Y. Lim, B. M. Kurkoski, and Y. Tan, "Retransmission Scheme for Compute-and-Forward in Multi-Source Multi-Relay Networks using Contiguously Overlapped Chunked," *IEEE Transactions on Mobile Computing*, (to be submitted).

[2] R. Ngeth, B. Kurkoski, Y. Lim, and Y. Tan, "A design of overlapped chunked code over compute-and-forward for multi-source multi-relay networks," *Sensors*, vol. 18, no. 10, p. 3225, 2018.

International Conferences

[3] R. Ngeth, Y. Lim, B. M. Kurkoski, and Y. Tan, "A Retransmission Scheme Using Contiguously Overlapped Chunked Code over Compute-and-Forward," in *The 2019 Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (IEEE PIMRC 2019)*, IEEE, Istanbul, Turkey, September 2019, (submitted).

[4] R. Ngeth, Y. Lim, B. M. Kurkoski, and Y. Tan, "A design of overlapped chunked code over compute-and-forward in multi-source multi-relay networks," in *2018 IEEE Global Communications Conference (GLOBECOM)*, IEEE, pp. 1-6, Abu Dhabi, UAE, December 2018.

[5] R. Ngeth, B. M. Kurkoski, Y. Lim, and Y. Tan, "Random linear network coding over compute-and-forward in multi-source multi-relay networks," in *Wireless Com-*

munications and Mobile Computing Conference (IWCMC), 2017 13th International, IEEE, pp. 805-810, Valencia, Spain, June 2017.

[6] Nyan Lin, R. Ngeth, K. Sriviriyakul, Y. Tan and Y. Lim, "necoMAC: Network Coding aware MAC Protocol for Multirate Wireless Networks," in *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, pp. 384-391, Switzerland, March 2016.

[7] Nyan Lin, R. Ngeth, Y. Tan, and Y. Lim, "Energy Efficient Data Gathering with Network Coding for Cluster-based Wireless Sensor Networks," in *Proceedings of the 14th International Conference on Computer Applications (ICCA)*, pp. 8-13, Yangon, Myanmar, February 2016.

Local Conferences

[8] R. Ngeth, Y. Lim, B. M. Kurkoski, and Y. Tan, "Study of retransmission protocol with compute-and-forward in multi-source multi-relay network," in *Proceedings of the IEICE Society Conference*, BS-7-24, S-70, 1, Kanazawa University, Ishikawa, September 2018.

[9] R. Ngeth, Nyan Lin, Y. Tan and Y. Lim, "Network Coding Scheme based on Selection of Number Input Packets in X-topology Network," in *Proceedings of the IEICE General Conference*, vol. 2, pp. 177, Kyushu University, Fukuoka, March 2016.

[10] Nyan Lin, R. Ngeth, Y. Tan, and Y. Lim, "Study of Balanced Cooperative Coding Scheme for Linear Data Exchange Problem," in *Proceedings of the IEICE General Conference*, vol. 2, pp. 176, Kyushu University, Fukuoka, March 2016.

[11] Nyan Lin, K. Sriviriyakul, R. Ngeth, Y. Tan, and Y. Lim, "Study of Network Coding-aware MAC protocol for Multihop Wireless Networks," in *Proceedings of the IEICE Society Conference*, vol. 2, pp. 67, Tohoku University, Sendai, August 2015.