Control Overhead Optimization in Wireless Resource Allocation Problems

A Ph.D. Dissertation

Submitted to the Faculty

of

Drexel University

by

Jie Ren

Advisor: John MacLaren Walsh

June 2016
© Copyright 2016
Jie Ren. All Rights Reserved.
Acknowledgments

First of all, My most sincere thanks go to my advisor and mentor Prof. John MacLaren Walsh, who introduced me to the area of information theory, for all of his wise guidance in research, and his advice about self-discipline. I would like to thank the members of my committee, Prof. Garcia-Frias, Prof. Kandasamy, Prof. Dandekar and Prof. Weber, for their support and insightful comments. I would like to thank my fellow doctoral students, especially Bradford Boyle, Gwanmo Ku, and Solmaz Torabi, who collaborated with me, and gave me great help during my Ph.D. study. I would like to thank my parents Xuhe Ren and Fengying Liu for raising me up and encouraging me to study in US. I would like to thank all my friends for their friendship. Finally, I would like to thank my wife Hanyan Li, for her understanding, patience and love. It is the most amazing thing to meet you in my life, and I wish you all the best in your Ph.D. study.
# Table of Contents

LIST OF FIGURES ........................................................................................................... v  
ABSTRACT ....................................................................................................................... vii  
1. Introduction .............................................................................................................. 1  
  1.1 Control Overhead in the Resource Allocation Problems .................. 1  
  1.2 Outline and Contributions .................................................................................. 2  
2. Background .............................................................................................................. 7  
  2.1 The Discrete Distributed Model for Function Computation ............... 7  
  2.1.1 Fundamental Limits in the Distributed Function Computation Problems ............................................................. 7  
  2.1.2 Practical Quantizer Designs in the Distributed Function Computation Problems ................................................... 8  
  2.1.3 Interactive Communication ......................................................................... 10  
  2.1.4 Distributed Markov Decision Process ......................................................... 11  
  2.2 The Lossless Function Computation Problem .............................................. 14  
  2.3 The Interactive Function Computation Problem ........................................... 17  
  2.3.1 The Two-terminal Interactive Function Computation Model ............ 17  
  2.3.2 The Collocated Multi-terminal Interactive Function Computation Model ........................................................................... 19  
  2.4 The Markov Decision Process and Dynamic Programming ..................... 20  
3. Control Overhead Optimization in the model of Lossless Function Computation ............................................................................................................... 23  
  3.1 Problem Model: Control Overhead Optimization for Downlink Resource Allocation ................................................................................................................. 23  
  3.1.1 Distortion Measures ....................................................................................... 24  
  3.2 Minimizing the Control Overhead by distributed function computation and graph coloring ........................................................... 26  
  3.3 Converse of Determining the Extremization Functions ......................... 37  
  3.4 Scaling in Number of Users .............................................................................. 42  
4. Control Overhead Optimization in the Model of Interactive Communication .............................................................................................................. 45  
  4.1 Problem Model – An Interactive Scheme to Maximize the System Throughput ............................................................ 45  
  4.2 Analysis .............................................................................................................. 48  
  4.3 Thresholds vs. Number of Users ...................................................................... 51  
  4.4 Results—Interaction in the arg max case ......................................................... 52  
  4.5 Results—Interaction in the max and (arg max, max) Case ....................... 54  
  4.6 Scaling Laws ...................................................................................................... 55  
  4.7 Comparison with Other Interactive Schemes ............................................... 58  
5. A Framework for Rate Efficient Control of Distributed Discrete Systems .............................................................................................................. 62  
  5.1 Omniscient Control of a Distributed Markov Decision Process ............. 64  
  5.1.1 Simulating the Omniscient MDP via Information Exchange ............... 69
5.2 Minimal Coordination Communication Required for Distributed Simulation of the MDP .......................................................... 71
  5.2.1 One Shot, Non-Interactive Distributed Simulation of the Omniscient MDP ........................................................ 72
  5.2.2 Interactive, Collocated Network, Simulation of the Omniscient MDP ................................................................. 75
5.3 Incorporating Communication Cost into the Reward Function ........ 79
5.4 Finding Candidate Quantizations through Alternating Optimization ... 83
6. Conclusion ................................................................. 93
BIBLIOGRAPHY .............................................................. 98
Bibliography ................................................................. 98
# List of Figures

1.1 Supported MCS Index & Associated CQI in the LTE Standard ............... 2

1.2 A resource allocation model in which a basestation wishes to maximize the system spectral efficiency by scheduling the user with the highest channel quality. ................................................................. 3

2.1 Two-terminal interactive function computation ......................... 18

2.2 An illustration of states, control actions, transition probabilities, and transition rewards ................................................................. 20

3.1 Coloring the characteristic graph under the optimal staggered function ... 37

4.1 MTIS Vs. non interactive for the arg max problem with $N_1 = 8$ .............. 48

4.2 MTIS Vs. non interactive for the arg max problem with $|X_1| = 64$ ........... 49

4.3 A comparison of sending thresholds and sending number of users with $|X_1| = 16$................................................................................. 53

4.4 MTIS Vs. non-interactive for the max problem with $N_1 = 8$ .............. 54

4.5 MTIS Vs. non-interactive for the max problem with $|X_1| = 16$ .............. 55

4.6 The 3 users case of the CEO extremization problem with RIS................. 59

4.7 The 3 users case of the CEO extremization problem with NBIS.............. 59

4.8 A comparison of the three interactive schemes with $N_1 = 8$................. 59

4.9 A comparison of the three interactive schemes with $|X_1| = 16$ .............. 61

5.1 The resource allocation with buffer limitation model: optimizing control decisions and encoder mappings jointly ........................................ 63

5.2 The collocated setup: any third party can learn the control decision after receiving the encoded messages from users and the BS.................. 67

5.3 An illustration of the package dropping process: packages for each user take turn moving into the buffer ...................................................... 68
5.4 Rate required to select a user attaining the max with 3 users each observing an independent RV with support \{1, 2, 3, 4\} when users take turns sending messages one at a time [1, 2]. The fundamental limit on the rate given by (5.22) is compared to that obtained by the best scalar quantizer to use in the collocated network interactive scheme (Optimal Scalar. Hete. Q). This is substantially less than the rate required if all three users must send each message in parallel, then hear collectively all of the sent messages, then send each message in parallel again, etc, which is the interactive model employed in [3].

5.5 The control overhead versus the expected throughput tradeoff with $\lambda \in [0, 0.36]$ in the augmented reward function in (5.26) ........................................................................ 82

5.6 The control overhead versus the expected packet dropping cost tradeoff with $\lambda \in [0, 0.36]$ in the augmented reward function in (5.26) ................ 83

5.7 A game setup of the optimal reward function and control overhead ........ 84

5.8 The expected system throughput with respect to the weighting coefficient $\lambda$ in (5.26) ............................................................... 87

5.9 The control overhead per time-slot with respect to the weighting coefficient $\lambda$ in (5.26) ............................................................... 88

5.10 The control overhead versus the expected throughput tradeoff computed by both the round-robin and the greedy alternating optimization algorithm with $\lambda \in [0, 4]$ in the augmented reward function in (5.26) .................... 88

5.11 The control overhead versus the expected packet dropping cost tradeoff computed by both the round-robin and the greedy alternating optimization algorithm with $\lambda \in [0, 4]$ in the augmented reward function in (5.26) .. 89
Abstract
Control Overhead Optimization in Wireless Resource Allocation Problems

Jie Ren
Advisor: John MacLaren Walsh, Ph.D.

Under many traditional resource allocation and link adaptation architectures of modern wireless communication systems, such as those utilized in the Long Term Evolution 3GPP standard, channel quality indicators (CQI) and other channel parameters are sent as uplink feedback, and these are utilized to determine which user should get which resource blocks as well as deciding the coding scheme. Then the base station must signal the resource decision on the downlink as overhead control information in addition to the data to be transmitted to the user itself. These resource decisions control information, along with the channel quality feedback utilized to make them, create control overhead in the multiuser OFDMA system that are surprisingly large, with the control information such as resource decisions and reference signals typically occupying roughly a quarter to a third of all downlink transmission in the LTE standard. Inspired by this observation, in this dissertation, we focus on determining how to efficiently encode control information salient for resource allocation from the view of both information theory and communication.

In the first section of the dissertation, we model the CQI feedback process as a lossless distributed function computation problem, and consider the special case that there is an infinite backlog of traffic at the basestation that is waiting and destined for each user. Via computation of related fundamental information theoretic limits, we show that, if the basestation is not allowed to interact with the users, a small amount of control information compression beyond that required to simply forward channel quality information is possible. However, the amount of information saved
by even the best possible lossless non-interactive scheme is small and does not scale with the number of users.

Next, switching to a model in which the resource decisions can be made interactively with the users, we show that substantial control rate compression is possible for this CQI feedback process. In particular, a feedback process that is structured in a similar manner to an auction yields rate savings which are proven to scale proportionally with the number of users. This shows that, provided interaction is included into the model, substantial rate savings relative to simple channel quality information forwarding can be achieved with simple schemes.

Next, we improve the model of the resource allocation problem by incorporating downlink traffic arrival statistics into the scheduling, and the signals the basestation sends into the optimization. Including the consideration of the control signals sent into the model enables the optimization of control signals flowing on the downlink, and thus the potential to reduce the footprint of control information in the downlink, which was the problem which originally motivated the research. The incorporation of arrival statistics and the buffer at the basestation involves adding memory into the model, and the problem is no longer one involving simple multiterminal source compression, but one of a distributed Markov decision process (MDP). In particular, the basestation directly observes how much information is currently in the buffer destined for each user, and the users know their individual channel qualities. A MDP framework can be applied to this problem to yield optimal resource controllers, which form the resource allocation decision as a function of both the users’ channel qualities and the buffer information. While an omniscient controller having simultaneous access to all of these channel qualities and the buffer information could implement these resource decisions, the fact that the observations are spread throughout the network require coordination between the users in the form of control messages.
The final section of the dissertation considers the fundamental limits for such MDP problems where the global state is composed of a series of local states, each observable at a different node in a network. First, using recent results from multi-terminal information theory, bounds on the minimum amount of control information required for the nodes to perfectly simulate the omniscient controller are presented, both for one-shot and for interactive control messaging schemes. Next, this control overhead cost is included as a negative term into the reward function for the MDP, and a tradeoff between control overhead and controller performance is defined. The resulting optimization problem is complex, and hence an alternating optimization algorithm is presented to yield candidate messaging and controls schemes. It is shown that the presented algorithm yields a sequence of combined rewards which always converges, and when the associated control map and messaging converge, they always yield a Nash equilibrium for the associated optimization problem. Throughout this final section, each stage of the development is accompanied with a detailed example showing how to apply the framework to the wireless resource allocation problem. The dissertation then concludes with a number of fruitful directions for future research.
1. Introduction

1.1 Control Overhead in the Resource Allocation Problems

Orthogonal frequency division multiple access (OFDMA) is widely used in modern downlink cellular systems. In a multiuser OFDMA system, resource allocation and link adaptation are crucial ingredients. Under resource allocation, each sub-carrier needs to be assigned to a user by the base station, while under link adaptation a modulation and coding scheme must be selected for use for communicating to a user that enables it to reliably receive information.

Under many traditional resource allocation and link adaptation architectures, such as those utilized in the Long Term Evolution 3GPP standard, channel quality indicators are sent as uplink feedback, and these are utilized to determine which user should get which subcarriers as well as deciding the coding scheme. These channel quality indicators are often defined as the index of the highest rate modulation and coding scheme, among those that are possible to be selected by the basestation, that the user’s associated downlink channel can currently support with a target block error probability.

For instance, the modulation and coding schemes and associated channel quality indicator levels for the LTE standard are displayed in Table 1 as specified via Section 7.1 & 8.6 in [4] and discussed in [5][6][7][8][9][10][11].

Once the base-station has decided which user to schedule on a particular collection of subcarriers, and which modulation and coding scheme to employ in its communication with them, it must signal this resource decision, both the subcarriers assigned and the modulation and coding scheme, on the downlink as overhead control information in addition to the data to be transmitted to the user itself. These
Fig. 1.1: Supported MCS Index & Associated CQI in the LTE Standard

resource decisions control information, along with the channel quality indicator feedback utilized to make them, create control overhead in the multiuser OFDMA system that are surprisingly large, with the control information such as resource decisions and reference signals typically occupying roughly a quarter to a third of all downlink transmission in the LTE standard [12]. This overhead is growing significantly as cellular standards incorporate additional otherwise capacity improving features such as coordinated multipoint and carrier aggregation [5, 13, 14, 15, 16, 17, 18]. Since time frequency resources utilized for control information must be taken from resources that could be used for data transmission, the problem of determining how to efficiently encode channel quality feedback from the view of both information theory and communication is important [19].

1.2 Outline and Contributions

The central goal of this dissertation is to understand how to efficiently encode the control overhead information in the design of the downlink wireless resource allocation.
controllers under different problem models and design constraints. In this dissertation, Chapter 2 serves as a background chapter that introduce some important results in the subjects of distributed function computation, distributed quantization, graph coloring, interactive communication, communication complexity, dynamic programming and Markov decision process which inspire us in the design of efficient coding schemes in wireless resource allocation problems.

In Chapter 3, we model the encoding process of the control uplink channel quality feedback as a distributed discrete one-shot function computation problem in the sense that the basestation, after receiving all the messages regarding the user channel qualities, must compute a control decision on how to allocate the resource blocks. We assume that the channel qualities are independent and identically distributed from a known distribution on a discrete support set, and the basestation wishes to maximize the system throughput. Under this model, we investigate the minimum sum-rate that the users must send to help the basestation compute the control decision. We give low complexity encoding schemes based on graph coloring method and show that this scheme achieves the fundamental limit in sum-rate. A closed form expression of
the minimum sum-rate of this coding scheme is derived. The amount of information
saved by the proposed one-way graph coloring based encoding scheme is small and
does not scale with the number of users, i.e. we show in Section 3.4 of Chapter 3 that
at most 2 bits can be saved by the one-shot distributed function computation relative
to the scheme in which the users simply send their un-coded channel qualities to the
basestation. However, when switching to a model in which the resource decisions
can be made interactively with the users, we are able to show as in Chapter 4 that
substantial control rate compression is possible.

In Section 4.1 of Chapter 4, we provide the Multi-Threshold Interactive Scheme
for computing the resource allocation functions interactively between the basestation
and users. The Multi-Threshold Interactive Scheme operates as follows: During each
round of this scheme, the basestation updates its estimation on the user channel
quality, and broadcasts a message to declare a threshold based on the estimation to
the online users, where we say a user remains online if for instance, it is still possible
to be scheduled for downlink data transmission by the basestation, based on the
information it has received up until this round. The online users will reply a binary
message to let the basestation know whether or not its channel quality is above
the threshold. The basestation stops the interactive communication scheme with
enough information to make decisions for resource allocation. Aiming at determining
the optimal choice of the threshold to minimize the expected value of a given cost
metric, i.e. the aggregate communication rate, we formalize the process of the Multi-
Threshold Interactive Scheme as a dynamic programming problem. Simulation results
of solving the dynamic programming problem are shown in this chapter.

Finally in Chapter 5 of the dissertation, we study a distributed discrete control
system modeled a Markov decision process for whom the global system state com-
posed of a series of local states. Exchanging local state information by communication
is required to make control decisions since no node is given access to the global state. For a given control mapping that maps each of the global state to a control action, we compute the minimum amount of information that must be exchanged in the network to guarantee any node with only observing the control messages will be able to recover the control action of the system. Unlike the CEO model discussed in Chapter 3 and Chapter 4, here we focus on the collocated communication network in which we assume the exchanged control messages about users’ local states are sent via a broadcasting channel and every nodes can successfully decode the messages. We then incorporate the communication cost into the reward function. By doing so, we propose a joint optimization problem in which we want to find the optimal control mapping with respect to the communication scheme and find the minimum sum-rate achieving communication scheme to learn the control decision. Clearly, finding the best control mapping can be modeled as a Markov decision process while finding the best communication scheme can be modeled as both one-shot and interactive function computation problem. We show a tiny example of solving the joint optimization problem by searching over all possible control mappings for a given quantization and minimizing over all quantizations. In general, such a joint optimization problem is intractable due to the exponentially growing of the size of the searching space. Therefore we propose an alternating optimization algorithm where we alternatively optimize the control mapping with respect to the quantization and optimize the quantization with respect to the control mapping until a Nash equilibrium is achieved. Examples of the wireless resource allocation problem have been discussed through this chapter. In these examples, we assume the basestations has a shared buffer containing the data packets that must be sent to the users, and the users observe their individual channel qualities for the current time-slot. The system state is composed of both the buffer status and the local channel qualities which is not fully known by neither the bases-
tation nor any of the users. A control action of scheduling which user for downlink data transmission is made by purely observing every messages.
2. Background

2.1 The Discrete Distributed Model for Function Computation

To model the control overhead optimization in the wireless resource allocation problems, we consider a distributed system where all users communicate to a central node. These models are referred as the chief executive officer (CEO) problem [20][21]. We begin by reviewing fundamental limits and achievable schemes for non-interactive variants of the CEO type problem. We then review results for interactive CEO type problems that demonstrate significant rate savings may be possible. The non-interactive CEO problems have received considerably more attention than the interactive, and for the cases where fundamental limits are known, quantization followed by entropy coding closely approximates these limits.

2.1.1 Fundamental Limits in the Distributed Function Computation Problems

In information theory, the interest is usually on characterizing or providing inner/outer bounds for rate region. Berger et al. introduced the generic CEO problem, wherein the CEO wants to reproduce the source from the received signals[20]. The rate region for the source reproduction with constrained distortion problem remains unknown, except for the cases of Gaussian distribution with quadratic distortion[22][23], or i.i.d. discrete source distributions [24].

In this dissertation, we typically focused on having the CEO compute a function of all sources, this is referred as the distributed function computation problem, and was considered in [25] [26] [27]. This general formulation contains the specialized problem of function computation with side information; in this problem, the CEO knows all
but one of the sources[28] [29]. When the problem requires error free computation, it was shown that the minimum worst case rate is related to the chromatic number of the characteristic graph of the source[28]. In the case of lossless computation (in the Shannon sense), it was shown that the minimum average rate is the conditional graph entropy of the characteristic graph of the source[29], where the definition of graph entropy and its information theoretical meaning was first given in [30]. Building upon this line of research, the rate region for the lossless distributed function computation problem was characterized for certain problem instances[26] [27]. Sefidgaran et al. derived inner and outer bounds to the rate region for a class of tree structured networks (which includes the CEO problem) and showed that the inner and outer bound coincide with each other if the source random variables obey certain Markov properties[26]. Doshi et al. gave the rate region for the distributed function computation problem but under a different constraint that they referred to as the “zig-zag” condition [27]. They showed that any achievable rate point can be realized by graph coloring at each user and Slepian-Wolf [31] encoding the colors. Aside from that, Han and Kobayashi partitioned all distributed functions based on whether their achievable rate region always coincides with the Slepian-Wolf region[25].

The aforementioned literatures provide insightful outer bounds, but the achievable schemes used in the proofs usually needs block coding with infinite block length, which is not practical. In a real system, simpler achievable schemes with low computational complexity and performance close to the limits are needed.

2.1.2 Practical Quantizer Designs in the Distributed Function Computation Problems

A concern of signal processing is to provide optimized practical quantization algorithms for the distributed function computation system with performance close to
the rate-distortion limits[32][33][21][34][35][36][37]. There are asymptotical results for sufficiently high-rate-low-distortion problem that are usually derived by applying the high rate quantization theory[38], while there are also non-asymptotical results that are derived based on generalizing Lloyd’s algorithm.

For the high-rate-low-distortion scenario, Misra et al. considered a quantization scheme for the analysis of distributed scalar quantization[34]. It was shown that, with certain constraints on the objective function and source distributions, the high-resolution approach can asymptotically achieve the rate-distortion limits and the optimized quantization is regular. Sun et al. utilized a similar high-resolution approach, but with a simpler decoder design and looser source distribution requirements [35].

For the general rate distortion problem, an algorithm for building optimized distributed quantizers was given when the CEO will use the quantized observations to perform hypothesis testing[21]. When the users each have a noisy observation on the same source and the problem is to let the CEO reproduce the source with distortion, a two-stage distributed scheme was proposed in [33] [36]. The first stage does local quantization while the second stage does Slepian-Wolf coding on the quantized signals where syndrome codes was used in [36] and index reuse technique was used in [33].

Most of the provided distributed quantization schemes are non-interactive, which means the users each can only communicate to the CEO once and no feedback is allowed from the central node. For the problems in which the non-interactive fundamental limits are known, distributed quantization shows satisfactory performance comparing with the limit. However, very little work has been done in the interactive distributed cases.
2.1.3 Interactive Communication

*Interactive communication* is a scheme that allows message passing over multiple rounds, where at each round, the communicating parties are allowed to send messages based on what they have received in previous rounds as well as their local source observation [39]. For the lossy source coding problem with side information (Wyner-Ziv [40]), Ma and Ishwar showed (via an example) that the minimum rate for a given distortion constraint can be arbitrarily smaller than the non-interactive minimum rate obtaining the same distortion[41]. In follow up work, Ma and Ishwar showed (by an example again) that for the CEO problem, the minimum sum-rate for losslessly computing a function can be smaller than the non-interactive rate[42]. These results motivate us to consider interaction in the distributed quantization model.

The related works about interactive communication are mainly under two setups, the communication complexity setup and the interactive information theory setup. Yao introduced the two terminal communication complexity model in [43]. In this model two terminals each know an input and they both wish to compute a joint function of their inputs with or without an error. This communication complexity literature is concerned with finding communication protocols that minimize the sum rate subject to different sets of constraints. Introductory overviews of communication complexity can be found in [44][45]. Communication complexity is defined as the sum-rate cost minimized over all protocols and maximized over all possible input pairs (worst case cost). Average cost has also been studied for randomized coding protocols. Much of the communication complexity literature mainly addresses 2 users. Some models of $N$-terminals setup were considered in [46][47]. The authors were focused on providing communication complexity bounds with the restrictions that the function must be boolean and the message sent at each round must be binary. However, in our work, we are interested in providing achievable schemes for the problems with no
1-bit restriction on the communication of each round and the function output.

Kaspi was the first to study interactive communication from the perspective of information theory [39]. The key difference between the information theory setup and the communication complexity setup is that the communication rate for each round does not have the requirement of being a single bit. Kaspi gave the rate region of any $N$-round interactive communication for source reproduction with distortion [39]. This line of research was continued by Ma and Ishwar. They worked on both two and more than two terminals cases for computing any function of the sources in both lossy and lossless manner. They showed that interactive communication strictly improves the Wyner-Ziv rate-distortion function[41]. They also showed that in some distributed function computation problems, interactive communication can provide substantial benefits over non-interactive codes; even infinite-many rounds of interaction may still improve the rate-region[42].

### 2.1.4 Distributed Markov Decision Process

The framework of Markov decision processes (MDP) [48, 49] provides a principled method for the optimal design of controllers for discrete systems. By solving a Bellman’s equation, for example through either a value or policy iteration, one derives a control mapping assigning to each possible state of the system a control action to take, in a manner that maximizes a long run discounted expected reward. Of increasing interest, however, are those discrete systems which are decentralized or distributed, in the sense that no single participant in the system has full access to the global system state, but rather this global state is the concatenation of a series of local states, each of which are directly observed at different locations.

For instance, a series of agents may each observe their own local state, and have a set of local actions to choose from [50, 51, 52], and the desire may be to design indi-
vidual local controllers. The fact that the global system state is not available at any location either requires sufficient information to be exchanged, either through ordinary communications or through the system’s state [53, 54], to remedy the situation, or a modification of the control framework. One way to address in part this distributed knowledge of network state is to use the framework of partially observable Markov decision processes (POMDP) to synthesize controllers. In general optimal control of a POMDP requires the controller to maintain probabilistic belief-states about the current system state based on all previous control actions and all previous observations, then assigning a control action based on these belief states. Thus a key issue in the selection/design of POMDPs are problem structures which enable simple forms for this control action [55]. As they are even more complex, solving general decentralized multiagent (PO)MDPs are NEXP-complete [56], and a rich literature, e.g. [57, 58, 59, 60, 61, 62] and many others, have addressed finding approximate solutions.

A key issue in the literature about decentralized and/or distributed MDPs is the role and amount of communication and coordination. Relationships between communication and control have been established in several contexts in the literature. Information theoretic limits have been incorporated into the classical case of a single observer remotely controlling a linear system through a noisy channel [63, 64], with a focus on determining the minimum rate required to achieve control objectives [64] [65], [66]. In this vein, [67] proposes the notion of anytime capacity and gives a necessary and sufficient condition on communication reliability needed over a noisy channel to stabilize an unstable scalar linear process when the observer has access to noiseless feedback of the channel output. Building on these ideas, [68, 69, 70] provide explicit construction of anytime reliable codes with efficient encoding and decoding over noisy channels. Shifting to a decentralized control context, deep connections
between communications between controllers through the system state and network coding have recently been investigated [71, 53, 54].

Despite this long standing interest between relationships between communication and control, the role of information theoretic limits when state observations from a MDP are distributed has been less well developed in the literature. Here, the literature studying communication and coordination has not made use of related ideas from multiterminal information theory to compute communication cost. [72], for instance, considers a multi-agent coordination with agent decisions made in a self-interested environment, while [73] discusses the computational complexity of finding optimal decisions in a communicative multi-agent team decision problem (COM-MTDP) along the dimension of communication cost. Similarly [74, 75, 76], recognize that communication incurs a cost in the global reward function, and show that whether or not to communicate is also a decision to make. However, none of these models in [72, 73, 74, 75, 76] make use of relevant information theoretic limits when computing a communication cost. Part of the reason that information theoretic limits have not been fully brought to bear in the distributed MDP is that the limits for the relevant models, for instance for distributed [27, 26] and/or interactive function computation [77, 78], have been only somewhat recently derived. Bearing this in mind, this paper aims to harness information theoretic limits and coding designs that approach them, to help synthesis efficient control schemes for a distributed MDP.

In the remaining of this chapter, we will highlight the key ingredients in the references that we use to optimizing the control overhead in the wireless resource allocation problems.
2.2 The Lossless Function Computation Problem

In [29], a related problem is considered in which the node observing \( X_1 \) sends a message to the node observing \( X_2 \) in such a manner that the function \( f^S(X_1^S, X_2^S) = [f(X_1, s, X_2, s)]_{s \in [S]} \), taking values from the set \( Z^S \), can be computed losslessly. In this problem, a rate \( R \) is said to be achievable if for every \( \epsilon > 0 \) there exists a sufficiently large \( S \) and \( K \) with \( R \geq \frac{K}{S} \), and an encoder \( \varphi : \mathcal{X}^S \rightarrow \{0,1\}^K \) and a decoder \( \psi : \{0,1\}^K \times \mathcal{X}^S \rightarrow Z^S \) such that \( \mathbb{P}(\psi(\varphi(X_1^S), X_2^S) \neq f^S(X_1^S, X_2^S)) < \epsilon \). Orlitsky and Roche proved that for given \( X_1, X_2 \) and \( f \), the infimum of the set of achievable rates is

\[
L_f(X_1|X_2) = H_G(X_1|X_2)
\]  

(2.1)

where \( H_G(X_1|X_2) \) is the conditional graph entropy of the characteristic graph of this problem in [29]. The characteristic graph \( G \) of \( X_1, X_2 \), and \( f \) is a generalization of the definition given by Witsenhausen[28]. Its vertex set is the support set \( \mathcal{X} \) of \( X_1 \), and distinct vertices \( x_1, x'_1 \) are adjacent if there is a \( x_2 \) such that \( p(x_1, x_2), p(x'_1, x_2) > 0 \) and \( f(x_1, x_2) \neq f(x'_1, x_2) \). The conditional graph entropy is

\[
H_G(X_1|X_2) = \min_{W-X_1-X_2, X_1 \in W \in \tau(G)} I(W; X_1|X_2)
\]  

(2.2)

where \( \tau(G) \) is the set of all maximal independent sets in \( G \), \( W \) is a random variable that has \( \tau(G) \) as its support set, and the minimization is over all conditional probabilities \( p(w|x_1) \) which is supported on those maximal independent sets \( w \) containing the vertex \( x_1 \), with the constraint that \( W, X_1 \) and \( X_2 \) form a Markov chain.

Additionally, conditional graph entropy can be related to coloring a certain product graph. In particular, the OR-product graph \( G^S_1(V_S, E_S) \), based on the characteristic graph \( G_1 \) of \( X_1, X_2 \) and \( f \), has a vertex set \( V_S = \mathcal{X}^S \), and distinct vertices \((x_{1,1}, \ldots, x_{1,S}),(x'_{1,1}, \ldots, x'_{1,S})\) are connected if there exists an edge between \( x_{1,a} \) and
$x'_{1,s}$ in $G_1$ for any $s$. In [79], Doshi et al. showed that minimum-entropy coloring the OR-product graph, followed by lossless compression of the colors with Slepian-Wolf (SW) coding, yields a rate proportional to the conditional chromatic entropy, and can asymptotically reach the lower limit set out by the conditional graph entropy

$$\lim_{S \to \infty} \min_{c \in \mathcal{C}_\epsilon(G_1^S(f))} \frac{1}{S} H(c(X_1)) = H_G(X_1|X_2) \quad (2.3)$$

where $\mathcal{C}_\epsilon(G_1^S(f))$ is the set of all $\epsilon$-colorings of the product graph.

For the decentralized model where two users communicate with a CEO attempting to losslessly compute a function, Doshi et al. gave the rate region when the problem satisfies a given zig-zag condition, which requires that for any $(x_1, x_2)$ and $(x'_1, x'_2)$ in $\mathcal{X}_1 \times \mathcal{X}_2$, $p(x_1, x_2) > 0$ and $p(x'_1, x'_2) > 0$ imply either $p(x_1, x'_2) > 0$ or $p(x'_1, x_2) > 0$ [27]. The key idea is to let each user do an $\epsilon$-coloring [27] of the OR-product graph of its own source and transmits the color by a SW code.

[27] showed in Theorem 16 that the rate-region for the aforementioned distributed function computation problem under the zig-zag condition is the set closure of $\kappa$, where $\kappa$ is the intersection of $\kappa^\epsilon$ for all $\epsilon > 0$, and $\kappa^\epsilon$ is

$$\kappa^\epsilon = \bigcup_{n=1}^{\infty} \bigcup_{(c_{x_1}^n, c_{x_2}^n)} \mathcal{R}^n(c_{x_1}^n, c_{x_2}^n) \quad (2.4)$$

where the regions $\mathcal{R}^n(c_{x_1}^n, c_{x_2}^n)$ are given by

$$R_{x_1} \geq \frac{1}{n} H(c_{x_1}^n(X_1)|c_{x_2}^n(X_2))$$

$$R_{x_2} \geq \frac{1}{n} H(c_{x_2}^n(X_2)|c_{x_1}^n(X_1))$$

$$R_{x_1} + R_{x_2} \geq \frac{1}{n} H(c_{x_1}^n(X_1), c_{x_2}^n(X_2)) \quad (2.5)$$

In Theorem 18, [27] showed that the difference of the minimum sum-rate and
\( H_G(X_1|X_2) + H_G(X_2|X_1) \) is bounded by

\[
H_G(X_1|X_2) + H_G(X_2|X_1) - (R_{x_1} + R_{x_2}) \leq \min\{I_{G_1}(X_1;X_2), I_{G_2}(X_1;X_2)\} \quad (2.6)
\]

where \( I_{G_1}(X_1;X_2) \) is the graph information of \( X_1 \), and the right hand side is zero when \( X_1 \) and \( X_2 \) are independent. Note that \( H_G(X_1|X_2) = H_G(X_1) \) when the sources are independent, where the graph entropy \( H_G(X_1) \) is

\[
H_G(X_1) \triangleq \min_{W-X_1-X_2,W_1W_2 \in \tau(G)} I(W;X_1). \quad (2.7)
\]

Hence when the sources are independent, the rate-region is

\[
\begin{align*}
R_{x_1} &\geq H_G(X_1) \\
R_{x_2} &\geq H_G(X_2) \\
R_{x_1} + R_{x_2} &\geq H_G(X_1) + H_G(X_2).
\end{align*} \quad (2.8)
\]

Doshi et al. consider a very general class of problems, for which in general it is necessary to express the rate region in terms of the \( \epsilon \)-coloring, which essentially is an valid coloring on a high probability subset of the characteristic graph. We will now show how to apply these ideas and related ones to the extremization problems under investigation. In particular, we will show in Section 3.3 that we can achieve the fundamental limits of the sum-rate in the extremization problems by normally coloring the original characteristic graph as described in Section 3.2, thereby removing the need for both OR-product graph and \( \epsilon \)-coloring. The notion of an \( \epsilon - coloring \) is slightly different from the traditional notion of a coloring, hence we repeat the definition here.

**Definition 1.** Vertex coloring : A vertex coloring of a graph is any function \( c :
V → N of a graph G = (V, E) such that no two vertices map to the same number if there is an edge between them.

**Definition 2.** $\epsilon$-coloring : Let $\mathcal{A} \subset \mathcal{X}_1 \times \mathcal{X}_2$ be an $\epsilon$-high-probability set if

$$\sum_{(x_1, x_2) \in \mathcal{A}} p_{X_1, X_2}(x_1, x_2) \geq 1 - \epsilon$$  \hspace{1cm} (2.9)

Define $\hat{p}(x_1, x_2) = p(x_1, x_2) / \sum_{(x_1, x_2) \in \mathcal{A}} p_{X_1, X_2}(x_1, x_2)$ for any $(x_1, x_2) \in \mathcal{A}$, and $\hat{p}(x_1, x_2) = 0$ otherwise. Let the characteristic graph of $X_1$ with respect to $X_2$, $\hat{G}$ and $f$ be denoted by $\hat{G}_{x_1} = (\hat{V}_{x_1}, \hat{E}_{x_1})$ and the characteristic graph of $X_2$ with respect to $X_1$, $\hat{G}$ and $f$ be denoted by $\hat{G}_{x_2} = (\hat{V}_{x_2}, \hat{E}_{x_2})$. $c_{x_1}$ and $c_{x_2}$ are $\epsilon$-coloring of $G_1$ and $G_2$ if they are valid coloring of $\hat{G}_{x_1}$ and $\hat{G}_{x_2}$ for any $\epsilon$ high probability set $\mathcal{A}$.

where $c_{x_1}^n$ and $c_{x_2}^n$ are $\epsilon$-colorings. We will use this result together with the conditional graph entropy shortly to set out the fundamental limits for our problems, but first we wish to specify some achievable schemes that we will later show to approach these limits.

### 2.3 The Interactive Function Computation Problem

In this section, we introduce the key fundamental results of interactive communication for function computation problems in [42, 41, 78] and [80]. We start from introducing the two-terminal interactive communication problem as illustrated in Figure 2.1.

#### 2.3.1 The Two-terminal Interactive Function Computation Model

Consider n samples $X = X^n = (X_1, \cdots, X_n) \in \mathcal{X}^n$ of an information source are available at location A. A different location B has n samples $Y \in \mathcal{Y}^n$ of a sec-
Figure 2.1: Two-terminal interactive function computation

ond information source. Location A desires to produce a sequence $\hat{Z} \in Z_n^A$ such that $d_A^{(n)}(X, Y, \hat{Z}_A) \leq D_A$ where $d_A^{(n)}$ is a nonnegative distortion measure of $3n$ variables. Similarly, location B desires to produce a sequence $\hat{Z}_B \in Z_n^B$ such that $d_B^{(n)}(X, Y, \hat{Z}_B) \leq D_B$. All alphabets are assumed to be finite. To achieve the desired objective, $t$ coded messages, $M_1, \ldots, M_t$, of respective bit rates $R_1, \ldots, R_t$, are sent alternately from the two locations starting with location A or B. The message sent from a location can depend on the source samples at that location and on all the previous messages.

Note that under this problem model $Z_A$ and $Z_B$ can be a function of $X, Y$ as well as the source itself. Also note that, $Z_A$ can be 0 which represent only location B need to reproduce the source or a function of the sources.

Theorem (Two Terminal Interactive Function Computation, Ma and Ishwar[42]).
Let both locations to reliably compute some functions of the two sources, \( f_A(X, Y) \) and \( f_B(X, Y) \), then the rate-region for any t-way interactive function computation scheme starting from location A is characterized as,

\[
\mathcal{R}_t^A = \{ R | \exists U^t, s.t. \forall i = 1, \cdots, t \\
R_i \geq I(X; U_i|Y, U^{i-1}), \quad U_i - (X, U^{i-1}) - Y, \quad i \text{ odd} \\
R_i \geq I(Y; U_i|Y, U^{i-1}), \quad U_i - (Y, U^{i-1}) - X, \quad i \text{ even} \\
H(f_A(X,Y)|X,U^t) = 0, \quad H(f_B(X,Y)|Y,U^t) = 0 \}
\]  \hspace{1cm} (2.10)

2.3.2 The Collocated Multi-terminal Interactive Function Computation Model

It’s nature to extend the two-terminal interactive function computation idea to the case of multi-terminals. For this extension, the authors in [80] have studied the problem of interactive function computation in collocated networks. In the collocated network model, it is assumed that \( m \) source nodes each observe an i.i.d. source \( X_i \), whereas in node \( d \) who does not observe any source would like to compute a joint function \( Z = f(X_1, \ldots, X_m) \) with respect to all the local observations. The communication takes place over \( r \) rounds. In each round, source nodes broadcast messages according to the schedule \( 1, \ldots, m \). Each message depends on the source that the node observes and all previous messages that have been broadcasted. Nodes are collocated, meaning that every broadcasted message is recovered without error at every node. After \( mr \) message broadcasts over \( r \) rounds, the sink node computes the sample wise function based on all the messages. For this problem, the rate region is given as follows:

**Theorem** (Collocated, Ma and Ishwar[80]). Let \( t = rm \), the rate-region for the \( r- \)
round interactive collocated problem is

\[ R_r = \{ R | \exists U^t, s.t. \forall j \in [t], k = (j \mod m), \]
\[ R_j \geq I(X_k; U_j U_{j-1}), U_j = -(U_{j-1}, X_k) = -(X^{k-1}, X^m_{k+1}), \]
\[ H(f(X^m)|U^t) = 0 \}, \tag{2.11} \]

and the sum-rate cost in this problem is,

\[ R_{\text{sum}, r} = \min_{U^t} I(X^m; U^t) \tag{2.12} \]

2.4 The Markov Decision Process and Dynamic Programming

Figure 2.2: An illustration of states, control actions, transition probabilities, and transition rewards
A Markov Decision Process (MDP) is a discrete time stochastic control process [49, 48, 81]. As illustrated in Figure 2.2, the system is at a given state $s, s \in \mathcal{S}$ at each time instance, the controller will pick an action $a$ in the candidate action set $\mathcal{A}$. Then at the next round, the system will randomly moves to a new state $s'$ with the transition probability $P_a(s, s')$ of moving from state $s$ to state $s'$ that depends on the decision of the actions. Hence for given decision mapping of actions $c : \mathcal{S} \rightarrow \mathcal{A}$, the state transitions behaves the Markov property. In addition than that, for any action $a$, and any state $s'$ the system is moving to, there is a corresponding transition reward $R_a(s, s')$. The problem is to maximize the expected discounted rewards (or similarly minimize the transition costs) by appropriately choosing the actions for all states.

The argument to the solution to this optimization is a mapping $c^* : \mathcal{S} \rightarrow \mathcal{A}$ assigning to each state the optimum action to take, so that the optimal $A^*_t = c^*(s_t)$ for all $t \in \{0, 1, \ldots, \}$. Bellman’s equation [82, 48] states that the solution to this optimization must solve the following system of equations

\begin{align}
V^*(i) &= \sum_{j \in \mathcal{S}} p_{c^*(i)}(i, j) \left[ R^*(i, j) + \gamma V^*(j) \right] \quad \forall i \in \mathcal{S} \tag{2.13} \\
c^*(i) &= \arg \max_{a \in \mathcal{A}} \sum_{j \in \mathcal{S}} p_a(i, j) \left[ R_a(i, j) + \gamma V^*(j) \right] \quad \forall i \in \mathcal{S} \tag{2.14}
\end{align}

The solution to this simultaneous system of equations, and the associated optimal control mapping, can be found by first determining the limit $V^* = \lim_{k \rightarrow \infty} V_k$ of the following value iteration

\begin{align}
V_{k+1}(i) &= \max_{a \in \mathcal{A}} \sum_{j \in \mathcal{S}} p_a(i, j) \left[ R_a(i, j) + \gamma V_k(j) \right] \quad \forall i \in \mathcal{S} \tag{2.15}
\end{align}
then solving for the control policy via

\[
c^\ast(i) = \arg\max_{a \in A} \sum_{j \in S} p_a(i, j) [R_a(i, j) + \gamma V^\ast(j)] \quad \forall i \in S. \tag{2.16}
\]

Alternatively, one can utilize a policy iteration, which performs a recursion in which first

\[
c_k(i) = \arg\max_{a \in A} \sum_{j \in S} p_a(i, j) [R_a(i, j) + \gamma V_k(j)] \quad \forall i \in S \tag{2.17}
\]

is solved, followed by a solution of the linear system

\[
V_{k+1}(i) = \sum_{j \in S} p_{c_k(i)}(i, j) [R_{c_k(i)}(i, j) + \gamma V_{k+1}(j)] \quad \forall i \in S \tag{2.18}
\]

for \(V_{k+1}(\cdot)\). until the control mapping can be selected to remain the same under the update, at which point the iteration ceases, see, e.g. [49, 81, 55]
3. Control Overhead Optimization in the model of Lossless Function Computation

In this chapter we start from modeling the control overhead optimization as a distributed loss-less function computation problem. We present an achievable coding scheme on compressing the control overhead for the case that the base station wants to know which user has the best channel quality (the arg max problem), we then show the coding scheme we gave achieves the fundamental limits of the sum-rates. We then show that for the case that the base station wants to know the exact value of the best channel quality (the max problem) and the case that the base station want to know both the max and the arg max (the pair problem), there is also no need for sophisticated coding schemes, as simple Huffman coding achieves the fundamental limits of the sum-rates for these functions. We call the arg max, max and the (max, arg max) pair the extremization functions. These functions directly tie in with the control overhead minimization problem we pointed out in Chapter 1. In the next section, we will present the problem model first.

3.1 Problem Model: Control Overhead Optimization for Downlink Resource Allocation

As stated previously, we are considering the $N$ user chief estimating officer (CEO) problem for estimating either max, arg max, or the pair (max, arg max). The $n^{th}$ user observes the sequence $\bar{x}_n \triangleq (X_{n,t} : t \in [T])$ of non-negative random variables.\footnote{For any integer $n$, let $[n] \triangleq \{1, \ldots , n\}$}. Let $\bar{x} \triangleq (\bar{x}_n : n \in [N])$. We assume that the sources are independent and identically
distributed (i.i.d.) across both users \((n)\) and the sequence \((t)\); that is

\[
f_{\bar{X}}(\bar{X}) = \prod_{n=1}^{N} \prod_{t=1}^{T} f_{X_n}(x_{n,t}). \tag{3.1}
\]

The quantities we are interested in for our problem are

\[
Z_A(t) \triangleq \{j \mid X_{j,t} = \max\{X_{n,t} : n \in [N]\}\}
\]

as the users with the maximum \(n^{th}\) source output and

\[
Z_M(t) \triangleq \max\{X_{n,t} : n \in [N]\}
\]

as the maximum \(n^{th}\) source output. Specifically for the arg max case, we consider a class of problems where we need not estimate the set \(Z_A(t)\), but rather any representative user from this set.

### 3.1.1 Distortion Measures

To the best fit of the resource allocation problem in wireless OFDMA systems. We propose the following distortion measures for each of the three extremization functions. For estimating the max, the distortion measure is

\[
d_M((X_{1,t}, \ldots, X_{N,t}), \hat{Z}_M(t)) = \\
\begin{cases} 
Z_M(t) - \hat{Z}_M(t) & \text{if } \hat{Z}_M(t) \leq Z_M(t) \\
Z_M(t) & \text{otherwise}
\end{cases}
\tag{3.4}
\]

For estimating the arg max, the distortion measure is
\[ d_A((X_{1,t}, \ldots, X_{N,t}), \hat{Z}_A(t)) = \begin{cases} 0 & \text{if } \hat{Z}_A \in Z_A \\ Z_M(t) - X_{\hat{Z}_A(t),t} & \text{otherwise} \end{cases} \]  

This distortion measures the loss between source value of the user with the actual max and the source value for the user estimated to have the max. This distortion has been selected because in the resource allocation in OFDMA-like systems, this distortion is reflecting the loss in system capacity associated with the imperfect knowledge of channel state that the controller is utilizing. Finally, for estimating the pair of values \((\max, \arg\max)\) we propose a distortion measure

\[ d_{M,A}((X_{1,t}, \ldots, X_{N,t}), (\hat{Z}_M(t), \hat{Z}_A(t))) = \begin{cases} Z_M - \hat{Z}_M(t) & \text{if } \hat{Z}_M(t) \leq X_{\hat{Z}_A(t),t} \\ Z_M & \text{otherwise} \end{cases} \]  

This distortion is a combination of under-estimating the max value, provided the estimate does not exceed the value of the user estimated as having the max value. It also captures the loss due to over-estimation, both exceeding the estimated arg max user’s value or exceeding the actual max value. If the rate selected is lower than the selected user’s, than the allocation will be successful, and the loss is the difference between that rate/price offered and the original rate the user observed. However, if the rate offered is higher than that of the selected user, then the allocation will be unsuccessful, because the communication/decoding will be unsuccessful, and the full potential achieved by an omniscient controller (the actual maximum) rate is lost.

Depending on the extrema problem being considered, let \(d(t)\) be

1. \(d_{M,A}((X_{1,t}, \ldots, X_{N,t}), (\hat{Z}_M(t), \hat{Z}_A(t)))\);
2. \( d_A((X_{1,t}, \ldots, X_{N,t}), \hat{Z}_A(t)) \), or;

3. \( d_M((X_{1,t}, \ldots, X_{N,t}), \hat{Z}_M(t)) \)

and define

\[
\begin{align*}
\bar{d}((\bar{x}_1, \ldots, \bar{x}_N), \bar{z}) &= \frac{1}{T} \sum_{t=1}^{T} d(t) \tag{3.7}
\end{align*}
\]

as the distortion between sequences. Finally, denote

\[
D = \mathbb{E} \left[ d((\bar{x}_1, \ldots, \bar{x}_N), \bar{z}) \right] \tag{3.8}
\]

where the expectation is with respect to joint distribution on the sources. In the next section, we consider the problem of finding the minimum sum rate necessary for computing the different extremization functions with arbitrarily small probability, which we will show, for the problems under consideration, to be equivalent to constraining the distortion to \( D = 0 \).

3.2 Minimizing the Control Overhead by distributed function computation and graph coloring

We start with the arg max problem. We consider \( N \) users, each observing \( \bar{X}_n = (X_{n,s}|s \in \{1, \ldots, S\}, X_{n,s} \in \mathcal{X}) \) and assume that \( \mathcal{X} = \{\alpha_1, \alpha_2, \ldots, \alpha_L\} \) s.t. \( \alpha_1 < \alpha_2 < \ldots < \alpha_L \) w.l.o.g.. For each element \( X_{1,s}, \ldots, X_{N,s} \) of these sequences, we are interested in the aggregate rate required to enable the CEO to learn a \( \hat{Z}_A(s) \) in the arg max such that

\[
d_A((X_{1,s}, \ldots, X_{N,s}), \hat{Z}_A(s)) = 0. \tag{3.9}
\]

**Definition 3.** A rate \( R \) will be said to be achievable if for every \( \epsilon \) there exists \( S, R_1, \ldots, R_N \) with \( R = \sum_{n=1}^{N} R_n \), \( N \) encoder maps \( \phi_n : \mathcal{X}^S \to \{0,1\}^{S-R_n} \), \( n \in \{1, \ldots, N\} \), and a decoder map \( \psi : \{0,1\}^{S-R_1} \times \{0,1\}^{S-R_2} \ldots \times \{0,1\}^{S-R_N} \to \{1, \ldots, N\}^S \).
such that $d_A((X_{1,s}, \ldots, X_{N,s}), \psi(\phi_1(X_1^S), \phi_2(X_2^S), \ldots, \phi_N(X_N^S))) < \epsilon$.

We say a tie happens in the arg max of the $s^{th}$ sources if two or more users attain the maximum value. Note that the arg max is not unique in such a case, because when a tie happens, the CEO can choose any user that achieves the maximum and will attain zero distortion. In other words, the extremization function is not uniquely determined everywhere. This will be useful when minimizing the amount of information necessary to determine this function.

**Definition 4.** A function $f_N : \mathcal{X}^N \to \{1, \ldots, N\}$ is a candidate arg max function if and only if

$$d_A((X_{1,s}, \ldots, X_{N,s}), f_N(X_{1,s}, \ldots, X_{N,s})) = 0. \quad (3.10)$$

Let $\mathcal{F}_{A,N}$ be the set of all such candidate arg max functions with $N$ inputs. For any $f_N \in \mathcal{F}_{A,N}$, it indicates the index of a user attaining the max.

**Theorem 1.** An achievable sum-rate for losslessly determining the argmax among a set of $N$ users is

$$R_A = \min_{f_N \in \mathcal{F}_{A,N}} \sum_{n=1}^{N} \min_{c_n \in \mathcal{C}(G_n(f_N))} H(c_n(X_n)) \quad (3.11)$$

where the first minimization is over all candidate arg max functions, and $\mathcal{C}(G_n(f))$ is the set of all colorings of the characteristic graph of user $n$ w.r.t. the function $f_N$.

**Proof.** This achievable scheme directly follows the result from [27] with a block size $S = 1$ and by observing that an ordinary coloring is also an $\epsilon$-coloring. Following [27], we color the characteristic graph for each arg max function and transmit the colors by a SW code. (3.11) is the minimum sum-rate over all such schemes w.r.t. all candidate arg max functions and all possible coloring schemes on the OR-product graph of size $S = 1$. \qed
In order to solve the optimizations in (3.11), the following two lemmas will be useful. Throughout the discussion below, we will use \( \{\alpha_i, \alpha_j\} \in G \) to denote the existence of an edge between node \( \alpha_i \) and \( \alpha_j \) in the characteristic graph \( G \), and use \( \{\alpha_i, \alpha_j\} \notin G \) to denote that there is no such edge.

**Lemma 1.** For any function \( f_N \in \mathcal{F}_{A,N} \) that determines the arg max, no 3 vertices can form an independent set in its characteristic graph \( G_i(f_N) \) for any user \( i \).

**Proof.** For any 3 vertices, there must exist two of them that their indices are not adjacent in number, say vertex \( \alpha \) and vertex \( \beta \), hence \( \exists \) vertex \( \gamma \), \( \alpha < \gamma < \beta \) such that \( f_N(x_1 = \gamma, \ldots, x_i-1 = \gamma, x_i = \alpha, x_{i+1} = \gamma, \ldots, x_N = \gamma) \neq f_N(x_1 = \gamma, \ldots, x_i-1 = \gamma, x_i = \beta, x_{i+1} = \gamma, \ldots, x_N = \gamma) \). Therefore, an edge must exist between \( \alpha \) and \( \beta \) in \( G_i(f_N) \), and they cannot be in the same independent set. \( \square \)

**Lemma 2.** For any function \( f_N \in \mathcal{F}_{A,N} \) that determines the arg max, if \( \{\alpha, \beta\} \notin G_i(f_N) \), then \( \{\alpha, \beta\} \in G_n(f_N) \forall n \in [N] \setminus \{i\} \).

**Proof.** Without loss of generality, we suppose \( \alpha < \beta \). From the condition that \( \{\alpha, \beta\} \notin G_i(f_N) \), we know that \( \forall \mathbf{x}_{\setminus \{i\}} \in \mathcal{X}^{N-1}, f_N(x_i = \alpha, \mathbf{x}_{\setminus \{i\}}) = f_N(x_i = \beta, \mathbf{x}_{\setminus \{i\}}) \). In particular, we consider the following input sequences

\[
\begin{align*}
\mathbf{x}^1 &= (x_1^1, \ldots, x_N^1) \text{ s.t. } x_n^1 = \alpha \forall n \in [N], \\
\mathbf{x}^2 &= (x_1^2, \ldots, x_N^2) \text{ s.t. } x_i^2 = \beta, x_n^2 = \alpha \forall n \in [N] \setminus \{i\}, \text{ and} \\
\mathbf{x}^3 &= (x_1^3, \ldots, x_N^3) \text{ s.t. } x_j^3 = \beta, x_n^3 = \alpha \forall n \in [N] \setminus \{j\}.
\end{align*}
\]  

(3.12)

Begin by observing that \( f_N(\mathbf{x}^2) = i \) and \( f_N(\mathbf{x}^3) = j \) since \( \beta > \alpha \) and the positions associated with other users are all \( \alpha \). Next, we observe that \( f_N(\mathbf{x}^1) = f_N(\mathbf{x}^2) \) because \( \{\alpha, \beta\} \notin G_i(f_N) \) by assumption. This then implies \( f_N(\mathbf{x}^1) \neq f_N(\mathbf{x}^3) \), and hence there exists \( \mathbf{x}_{\setminus \{j\}} = (\alpha, \ldots, \alpha) \) such that the function result differs for \( x_j = \alpha \) and \( x_j = \beta \), and there is an edge between \( x_j = \alpha \) and \( x_j = \beta \). \( \square \)
As we mentioned above, the minimum achievable sum-rate $R_A$ depends on how we break the ties (i.e. how we choose the candidate arg max function). Denote $\mathcal{F}_{A,N}^*$ as the set of all candidate arg max functions that achieve $R_A$, the following theorem specifies the solution to the optimization problem introduced in Theorem 1.

**Theorem 2.** There exists a series of functions $\{f_n^* | n \in [N]\}$ where $f_n^*$ is a candidate arg max function for $n$ users satisfying the properties that

1. $f_1^*(x) = 1$ for any $x \in \{\alpha_1, \ldots, \alpha_L\}$,

2. $\forall x \in S_n^-(\alpha_i)$, where $\alpha_i \in \mathcal{X}$ and $S_n^-(\alpha_i) = \{(x_1, \ldots, x_n) | x_1 = \alpha_i, \max\{x_{\{i\}}\} < \alpha_i\}$,

   $f_n^*(x) = 1$,

   \[ f_n^*(x) = \begin{cases} 1 & \text{mod } (n, 2) = \text{mod } (i, 2) \\ f_{n-1}^*(x_{\{i\}}) + 1 & \text{otherwise,} \end{cases} \tag{3.13} \]

3. $\forall x \in S_n^+(\alpha_i)$, where $\alpha_i \in \mathcal{X}$ and $S_n^+(\alpha_i) = \{(x_1, \ldots, x_n) | x_1 = \alpha_i, \max\{x_{\{i\}}\} = \alpha_i\}$,

   $f_n^*(x) = f_{n-1}^*(x_{\{i\}}) + 1$,

   \[ f_n^*(x) = f_{n-1}^*(x_{\{i\}}) + 1, \tag{3.14} \]

4. $\forall x \in S_n^+(\alpha_i)$, where $\alpha_i \in \mathcal{X}$ and $S_n^+(\alpha_i) = \{(x_1, \ldots, x_n) | x_1 = \alpha_i, \max\{x_{\{i\}}\} > \alpha_i\}$,

   $f_n^*(x) = f_{n-1}^*(x_{\{i\}}) + 1$,

   \[ f_n^*(x) = f_{n-1}^*(x_{\{i\}}) + 1, \tag{3.15} \]

such that
1. The minimum sum-rate achieved by graph coloring w.r.t. $f_n^*$ for any $n \geq 2$ is

$$R(f_n^*) = (n - 2)H(X) + \min_{c_1 \in \mathcal{C}(G_1(f_n^*)))} H(c_1(X_1)) + \min_{c_2 \in \mathcal{C}(G_2(f_n^*)))} H(c_2(X_2))$$

$$= -(n - 2) \sum_{i=1}^{L} p_i \log_2 p_i - \sum_{i=1}^{L-1} p_{i+1} \log_2 p_{i+1} - p_1 \log_2 p_1 - p_L \log_2 p_L$$

(3.16)

where $p_i = \mathbb{P}(X = \alpha_i)$ and $p_{i+1} = p_i + p_{i+1}$.

2. $f_n^* \in F_{A,n}$, i.e. $R_A$ can be achieved by $f_n^*$ for all $n \in [N]$. 

Proof. First we prove 1. In user 1’s characteristic graph $G_1(V_1, E_1)$ where $V_1 = \mathcal{X} = \{\alpha_1, \ldots, \alpha_L\}$, by Lemma 1, we must have $\{\alpha_i, \alpha_j\} \in G_1$ if $|i - j| \geq 2$. Now we consider the pair of vertices $\{\alpha_i, \alpha_{i+1}\}$ for any $i \in \{1, \ldots, L - 1\}$. When $\text{mod} \ (n, 2) \neq \text{mod} \ (i, 2)$, there exists a sequence $x_\{1\} = (\alpha_i, \ldots, \alpha_i)$ satisfying $f_n^*(\alpha_i, x_\{1\}) = f_{n-1}^*(x_\{1\}) + 1$ by (3.14), and $f_n^*(\alpha_i+1, x_\{1\}) = 1$ by (3.13). Note that $f_n^*(x) > 0$ for all $n$. This implies $f_n^*(\alpha_i, x_\{1\}) > f_n^*(\alpha_i+1, x_\{1\})$, hence $\{\alpha_i, \alpha_{i+1}\} \in G_1$. Next we will prove $\{\alpha_i, \alpha_{i+1}\} \notin G_1(f_n^*)$ if $\text{mod} \ (n, 2) = \text{mod} \ (i, 2)$. Since

$$\mathcal{X}^{n-1} = \{x_\{1\}| \max\{x_\{1\}\} < \alpha_{i+1}\} \cup \{x_\{1\}| \max\{x_\{1\}\} = \alpha_{i+1}\}$$

$$\cup \{x_\{1\}| \max\{x_\{1\}\} > \alpha_{i+1}\}$$

(3.17)

it suffices to show that for any given $x_\{1\}$ in these three sets, the function will not differ when $\text{mod} \ (n, 2) = \text{mod} \ (i, 2)$. For $x_\{1\} \in \{x_\{1\}| \max\{x_\{1\}\} < \alpha_{i+1}\}$, we observe that $f_n^*(\alpha_{i+1}, x_\{1\}) = f_n^*(\alpha_i, x_\{1\}) = 1$ by (3.13) and (3.14).

For $x_\{1\} \in \{x_\{1\}| \max\{x_\{1\}\} = \alpha_{i+1}\}$, we observe that,

$f_n^*(\alpha_{i+1}, x_\{1\}) = f_n^*(\alpha_i, x_\{1\}) = f_{n-1}^*(x_\{1\}) + 1$ by (3.14) and (3.15). Finally for $x_\{1\} \in \{x_\{1\}| \max\{x_\{1\}\} > \alpha_{i+1}\}$, we also observe that $f_n^*(\alpha_i, x_\{1\}) = f_n^*(\alpha_{i+1}, x_\{1\}) = \ldots$
where (a.1) and (a.3) hold by (3.15), (a.2) hold by (3.13) if max \( \{x \}\) holds by (3.14), and another sequence \( (\alpha_i, \alpha_{i+1}, \alpha_i, \ldots, \alpha_i) \) satisfying \( f^*_n(\alpha_i, \alpha_{i+1}, \alpha_i, \ldots, \alpha_i) = 2 \) by (3.15). This implies \( \{\alpha_i, \alpha_{i+1}\} \in G_2 \) if \( \text{mod}(n, 2) = \text{mod}(i, 2) \). Next we will prove \( \{\alpha_i, \alpha_{i+1}\} \not\in G_2(f^*_n) \) if \( \text{mod}(n, 2) \neq \text{mod}(i, 2) \). Since

\[
\mathcal{X}^n = \{x_{\langle 1,2 \rangle} | \max\{x_{\langle 1,2 \rangle}\} < \alpha_{i+1}\} \bigcup \{x_{\langle 1,2 \rangle} | \max\{x_{\langle 1,2 \rangle}\} = \alpha_{i+1}\} \bigcup \{x_{\langle 1,2 \rangle} | \max\{x_{\langle 1,2 \rangle}\} > \alpha_{i+1}\}
\]

(3.19)

it suffices to show that for any given \( x_1 \in \mathcal{X} \) and \( x_{\langle 1,2 \rangle} \) in these three sets, the function will not differ when \( \text{mod}(n, 2) \neq \text{mod}(i, 2) \). For \( x_1 < \max\{x_{\langle 1,2 \rangle}\} \) and \( x_{\langle 1,2 \rangle} \in \{x_{\langle 1,2 \rangle} | \max\{x_{\langle 1,2 \rangle}\} < \alpha_{i+1}\} \), we observe that

\[
f^*_n(x_1, \alpha_i, x_{\langle 1,2 \rangle}) \stackrel{(a.1)}{=} f^*_n(\alpha_i, x_{\langle 1,2 \rangle}) + 1 \stackrel{(a.2)}{=} 2
\]

and

\[
f^*_n(x_1, \alpha_{i+1}, x_{\langle 1,2 \rangle}) \stackrel{(a.3)}{=} f^*_n(\alpha_{i+1}, x_{\langle 1,2 \rangle}) + 1 \stackrel{(a.4)}{=} 2
\]

(3.20)

(3.21)

where (a.1) and (a.3) hold by (3.15), (a.2) hold by (3.13) if \( \max\{x_{\langle 1,2 \rangle}\} < \alpha_i \), and by (3.14) and the fact that \( \text{mod}(n, 2) \neq \text{mod}(i, 2) \) implies \( \text{mod}(n - 1, 2) = \text{mod}(i, 2) \) if \( \max\{x_{\langle 1,2 \rangle}\} = \alpha_i \), and (a.4) hold by (3.13). For \( x_1 < \max\{x_{\langle 1,2 \rangle}\} \) and
$\mathbf{x}_{\{1,2\}} \in \{ \mathbf{x}_{\{1,2\}} | \max \{ \mathbf{x}_{\{1,2\}} \} = \alpha_{i+1} \}$, we observe that

$$f_n^*(x_1, \alpha_i, \mathbf{x}_{\{1,2\}}) \overset{(b.1)}{=} f_{n-1}^*(\alpha_i, \mathbf{x}_{\{1,2\}}) + 1 \overset{(b.2)}{=} f_{n-2}^*(\mathbf{x}_{\{1,2\}}) + 2 \quad (3.22)$$

and

$$f_n^*(x_1, \alpha_{i+1}, \mathbf{x}_{\{1,2\}}) \overset{(b.3)}{=} f_{n-1}^*(\alpha_{i+1}, \mathbf{x}_{\{1,2\}}) + 1 \overset{(b.4)}{=} f_{n-2}^*(\mathbf{x}_{\{1,2\}}) + 2 \quad (3.23)$$

where (b.1) and (b.3) hold by by (3.15), (b.2) hold by (3.15), and (b.4) hold by (3.14) and the fact that $\mod (n, 2) \neq \mod (i, 2)$ implies $\mod (n - 1, 2) \neq \mod (i + 1, 2)$. For $x_1 < \max \{ \mathbf{x}_{\{1,2\}} \}$ and $\mathbf{x}_{\{1,2\}} \in \{ \mathbf{x}_{\{1,2\}} | \max \{ \mathbf{x}_{\{1,2\}} \} > \alpha_{i+1} \}$, we observe that

$$f_n^*(x_1, \alpha_i, \mathbf{x}_{\{1,2\}}) \overset{(c.1)}{=} f_{n-1}^*(\alpha_i, \mathbf{x}_{\{1,2\}}) + 1 \overset{(c.2)}{=} f_{n-2}^*(\mathbf{x}_{\{1,2\}}) + 2 \quad (3.24)$$

and

$$f_n^*(x_1, \alpha_{i+1}, \mathbf{x}_{\{1,2\}}) \overset{(c.3)}{=} f_{n-1}^*(\alpha_{i+1}, \mathbf{x}_{\{1,2\}}) + 1 \overset{(c.4)}{=} f_{n-2}^*(\mathbf{x}_{\{1,2\}}) + 2 \quad (3.25)$$

where (c.1) (c.2) (c.3) (c.4) all hold by (3.15). For $x_1 = \max \{ \mathbf{x}_{\{1,2\}} \}$ and $\mathbf{x}_{\{1,2\}} \in \{ \mathbf{x}_{\{1,2\}} | \max \{ \mathbf{x}_{\{1,2\}} \} < \alpha_{i+1} \}$, we observe that

$$f_n^*(x_1, \alpha_i, \mathbf{x}_{\{1,2\}}) \overset{(d.1)}{=} f_{n-1}^*(\alpha_i, \mathbf{x}_{\{1,2\}}) + 1 \overset{(d.2)}{=} 2 \quad (3.26)$$

and

$$f_n^*(x_1, \alpha_{i+1}, \mathbf{x}_{\{1,2\}}) \overset{(d.3)}{=} f_{n-1}^*(\alpha_{i+1}, \mathbf{x}_{\{1,2\}}) + 1 \overset{(d.4)}{=} 2 \quad (3.27)$$

where (d.1) holds by (3.14) if $x_1 = \alpha_i$ and by (3.15) if $x_1 < \alpha_i$, (d.2) holds by (3.14) if $\max \{ \mathbf{x}_{\{1,2\}} \} = \alpha_i$ and by (3.13) if $\max \{ \mathbf{x}_{\{1,2\}} \} < \alpha_i$, (d.3) holds by (3.15), and (d.4)
holds by (3.13). For \( x_1 = \max\{x_{\{1,2\}} \} \) and \( x_{\{1,2\}} \in \{x_{\{1,2\}} \mid \max\{x_{\{1,2\}} \} = \alpha_{i+1}\} \), we observe that

\[
f_n^*(x_1, \alpha_i, x_{\{1,2\}}) \overset{(e.1)}{=} 1
\]

and

\[
f_n^*(x_1, \alpha_{i+1}, x_{\{1,2\}}) \overset{(e.2)}{=} 1
\]

where (e.1) (e.2) both hold by (3.14) and the fact that \( \text{mod} \ (n, 2) \neq \text{mod} \ (i, 2) \) implies \( \text{mod} \ (n, 2) = \text{mod} \ (i + 1, 2) \). For \( x_1 = \max\{x_{\{1,2\}} \} \) and \( x_{\{1,2\}} \in \{x_{\{1,2\}} \mid \max\{x_{\{1,2\}} \} > \alpha_{i+1}\} \), which means \( x_1 = \max\{x\} \), we observe that if \( x_1 = \alpha_j \) where \( \text{mod} \ (j, 2) = \text{mod} \ (n, 2) \), then

\[
f_n^*(x_1, \alpha_i, x_{\{1,2\}}) \overset{(f.1)}{=} 1
\]

and

\[
f_n^*(x_1, \alpha_{i+1}, x_{\{1,2\}}) \overset{(f.2)}{=} 1
\]

where (f.1) (f.2) both hold by (3.14). If \( x_1 = \alpha_j \) where \( \text{mod} \ (j, 2) \neq \text{mod} \ (n, 2) \), then

\[
f_n^*(x_1, \alpha_i, x_{\{1,2\}}) \overset{(f.3)}{=} f_{n-1}^*(\alpha_i, x_{\{1,2\}}) + 1 \overset{(f.4)}{=} f_{n-2}^*(x_{\{1,2\}}) + 2
\]

and

\[
f_n^*(x_1, \alpha_{i+1}, x_{\{1,2\}}) \overset{(f.5)}{=} f_{n-1}^*(\alpha_{i+1}, x_{\{1,2\}}) + 1 \overset{(f.6)}{=} f_{n-2}^*(x_{\{1,2\}}) + 2
\]

where (f.3) (f.4) both hold by (3.14), and (f.5) (f.6) both hold by (3.15). For \( x_1 > \max\{x_{\{1,2\}} \} \) and \( x_{\{1,2\}} \in \{x_{\{1,2\}} \mid \max\{x_{\{1,2\}} \} < \alpha_{i+1}\} \), we observe that if \( x_1 \leq \alpha_i \),
then
\[ f_n^*(x_1, \alpha_i, x_{\{1,2\}}) \overset{(g.1)}{=} 2 \] (3.34)

and
\[ f_n^*(x_1, \alpha_{i+1}, x_{\{1,2\}}) \overset{(g.2)}{=} 2 \] (3.35)

where (g.1) holds by (3.14) if \( x_1 = \alpha_i \), and by (3.15) if \( x_1 < \alpha_i \), and (g.2) holds by (3.15). If \( x_1 > \alpha_i \), then
\[ f_n^*(x_1, \alpha_i, x_{\{1,2\}}) \overset{(g.3)}{=} 1 \] (3.36)

and
\[ f_n^*(x_1, \alpha_{i+1}, x_{\{1,2\}}) \overset{(g.4)}{=} 1 \] (3.37)

where (g.3) holds by (3.13), and (g.4) holds by (3.14) and the fact that \( \text{mod } (n, 2) \neq \text{mod } (i, 2) \) implies \( \text{mod } (n, 2) = \text{mod } (i + 1, 2) \) if \( x_1 = \alpha_{i+1} \), and by (3.13) if \( x_1 > \alpha_{i+1} \). For \( x_1 > \max\{x_{\{1,2\}}\} \) and \( x_{\{1,2\}} \in \{x_{\{1,2\}} \mid \max\{x_{\{1,2\}}\} \geq \alpha_{i+1}\} \), we observe that
\[ f_n^*(x_1, \alpha_i, x_{\{1,2\}}) \overset{(h.1)}{=} 1 \] (3.38)

and
\[ f_n^*(x_1, \alpha_{i+1}, x_{\{1,2\}}) \overset{(h.2)}{=} 1 \] (3.39)

where (h.1) (h.2) both hold by (3.13). Therefore in user 2’s characteristic graph \( G_1 \), we have
\[ \{\alpha_i, \alpha_j\} \not\in G_2 \iff \text{mod } (n, 2) \neq \text{mod } (i, 2) \& j = i + 1. \] (3.40)

By Lemma 2 and user 1 and 2’s characteristic graphs, we know user \( i \)'s characteristic graph will be complete for all \( i \in [N] \setminus \{1, 2\} \). Therefore if \( N \) is odd and \( L \) is odd,
the set of all maximal independent sets for each of the users will be

\[
\Gamma(1) = \{\{\alpha_1, \alpha_2\}, \{\alpha_3, \alpha_4\}, \{\alpha_5, \alpha_6\}, \ldots, \{\alpha_{L-2}, \alpha_{L-1}\}, \{\alpha_L\}\}
\]

\[
\Gamma(2) = \{\{\alpha_1\}, \{\alpha_2, \alpha_3\}, \{\alpha_4, \alpha_5\}, \ldots, \{L - 1, L\}\}
\]

\[
\Gamma(i) = \{\{\alpha_1\}, \{\alpha_2\}, \ldots, \{\alpha_L\}\} \forall i \in [N], i \not\in \{1, 2\}.
\] (3.41)

If \(N\) is odd and \(L\) is even, the set of all maximal independent sets for each of the users will be

\[
\Gamma(1) = \{\{\alpha_1, \alpha_2\}, \{\alpha_3, \alpha_4\}, \{\alpha_5, \alpha_6\}, \ldots, \{\alpha_{L-1}, \alpha_L\}\}
\]

\[
\Gamma(2) = \{\{\alpha_1\}, \{\alpha_2, \alpha_3\}, \{\alpha_4, \alpha_5\}, \ldots, \{L - 2, L - 1\}, \{L\}\}
\]

\[
\Gamma(i) = \{\{\alpha_1\}, \{\alpha_2\}, \ldots, \{\alpha_L\}\} \forall i \in [N], i \not\in \{1, 2\}.
\] (3.42)

If \(N\) is even and \(L\) is odd, the set of all maximal independent sets for each of the users will be

\[
\Gamma(1) = \{\{\alpha_1\}, \{\alpha_2, \alpha_3\}, \{\alpha_4, \alpha_5\}, \ldots, \{\alpha_{L-1}, \alpha_L\}\}
\]

\[
\Gamma(2) = \{\{\alpha_1, \alpha_2\}, \{\alpha_3, \alpha_4\}, \ldots, \{L - 2, L - 1\}, \{L\}\}
\]

\[
\Gamma(i) = \{\{\alpha_1\}, \{\alpha_2\}, \ldots, \{\alpha_L\}\} \forall i \in [N], i \not\in \{1, 2\}.
\] (3.43)

If \(N\) is even and \(L\) is even, the set of all maximal independent sets for each of the users will be

\[
\Gamma(1) = \{\{\alpha_1\}, \{\alpha_2, \alpha_3\}, \{\alpha_4, \alpha_5\}, \ldots, \{\alpha_{L-2}, \alpha_{L-1}\}, \{L\}\}
\]

\[
\Gamma(2) = \{\{\alpha_1, \alpha_2\}, \{\alpha_3, \alpha_4\}, \ldots, \{L - 1, L\}\}
\]

\[
\Gamma(i) = \{\{\alpha_1\}, \{\alpha_2\}, \ldots, \{\alpha_L\}\} \forall i \in [N], i \not\in \{1, 2\}.
\] (3.44)

Note that in each \(\Gamma(n), n \in [N]\), no vertex belongs to two maximal independent sets. Also note that \(\{\alpha_i, \alpha_{i+1}\}\) appears exactly once in \(\{\Gamma(n)\}|n \in [N]\) for all \(\alpha_i \in \mathcal{X}\). To
achieve the minimum sum-rate, the optimal coloring method would be assigning a color for each of the independent sets (see Figure 3.1). For the case that both $L$ and $N$ are odd, we have

$$R_A(f^*_n) = \sum_{i=1}^{n} \min_{c_i \in C_i(f^*_n)} H(c_i(X_i))$$

$$= (n - 2) H(X) + \min_{c_1 \in C(G_1(f^*_n))} H(c_1(X_1)) + \min_{c_2 \in C(G_2(f^*_n))} H(c_2(X_2))$$

$$= -(n - 2) \left( \sum_{i=1}^{L} p_1 \log_2 p_i \right) - \left( \sum_{i=1}^{L-1} p_{2i-1,2i} \log_2 p_{2i-1,2i} \right) - p_L \log_2 p_L$$

$$- p_1 \log_2 p_1 - \left( \sum_{i=1}^{L-1} p_{2i,2i+1} \log_2 p_{2i,2i+1} \right)$$

$$= -(n - 2) \left( \sum_{i=1}^{L} p_1 \log_2 p_i \right) - \left( \sum_{i=1}^{L-1} p_{i,i+1} \log_2 p_{i,i+1} \right) - p_1 \log_2 p_1 - p_L \log_2 p_L$$

(3.45)

For the other cases, we will get the exact same expression of the sum-rate although there exists a minor variation on the argument.

Now we will show that for any function $f_n \in \mathcal{F}_{A,n}, n \in [N]$ the sum-rate under $f_n$ will be no lower than (3.16) and $f^*_n \in \mathcal{F}^*_A$. By Lemma 1 we know that no three vertices can be assigned the same color and hence only the neighbor pair can share the color. By applying Lemma 2 $N-1$ times, we know that if a neighbor pair $\{\alpha_i, \alpha_{i+1}\}$ are given the same color in user 1’s characteristic graph then they have to have different colors in all other users’ graphs. Therefore for all candidate arg max functions, we can have at most $L-1$ different consecutive pairs $\{\alpha_1, \alpha_2, \cdots, \alpha_{L-1}, \alpha_L\}$ that share the color, all other vertices have to have their own colors, and by assigning distinct colors to each of the $L-1$ node pairs and all other single nodes and encoding the colors by SW coding, (3.16) is achieved.

For the case that $N = 2$, since there will be $L$ different ties that need to be
Figure 3.1: Coloring the characteristic graph under the optimal staggered function distinguished, and $2^L$ different candidate functions that need to be considered, we have:

**Corollary 1.** Among all $2^L \arg \max$ functions, the one that achieves the lowest sum-rate under minimum entropy graph coloring satisfies the property that for all $\alpha_i \in \mathcal{X}$,

$$f_2^*(\alpha_i, \alpha_i) = \begin{cases} 1, & i \text{ odd} \\ 2, & i \text{ even.} \end{cases} \quad (3.46)$$

Having introduced this scheme, we will show in the next section that no scheme can have a higher rate-savings than this one.

### 3.3 Converse of Determining the Extremization Functions

The following lemma is necessary to aid in drawing the conclusion that joint graph coloring achieves the fundamental limit and there is no benefit from the OR-product graph for $\arg \max$.

**Lemma 3.** For any given candidate $\arg \max$ function $f_N \in \mathcal{F}_{A,N}$, the conditional probability $p(w|x)$ which is supported on the maximal independent sets $w$ containing the vertices $x$ to achieve the minimum mutual information in the graph entropy expression (2.7) must be either 1 or 0 for all $n \in [N]$. 
Proof. We prove this by showing that no vertex exists in two different maximal independent sets. Without loss of generality, we consider vertex $\alpha_i$ in $X_1$’s characteristic graph. By Lemma 1, the two maximal independent sets that $\alpha_i$ may belong to are $w_1 = \{\alpha_{i-1}, \alpha_i\}$ and $w_2 = \{\alpha_i, \alpha_{i+1}\}$. If vertex $\alpha_i \in w_1$ under the arg max function $f_N$ (which means there is no edge between $\alpha_{i-1}$ and $\alpha_i$), then $\forall x_{\{1\}} \in X^{N-1}$, we have

$$f_N(\alpha_i, x_{\{1\}}) = f_N(\alpha_{i-1}, x_{\{1\}}). \quad (3.47)$$

In particular, there exists $x_{\{1\}} = (\alpha_i, \alpha_{i-1}, \ldots, \alpha_{i-1})$ such that

$$f_N(\alpha_i, x_{\{1\}}) = f_N(\alpha_{i-1}, x_{\{1\}}) = 2, \quad (3.48)$$

and obviously

$$f_N(\alpha_{i+1}, x_{\{1\}}) = 1. \quad (3.49)$$

Therefore, $x_1 = \alpha_i$ is connected to $x_1 = \alpha_{i+1}$, and the set $w_2 = \{\alpha_i, \alpha_{i+1}\}$ is not an independent set in $X_1$’s characteristic graph, and we have

$$p(w|x_1 = \alpha_i) = \begin{cases} 1, & w = \{\alpha_{i-1}, \alpha_i\} \\ 0, & \text{otherwise}. \end{cases} \quad (3.50)$$

\[\square\]

**Theorem 3.** To losslessly determine the arg max, the fundamental limit of the sum-rate can be achieved by coloring the characteristic graph of each user, hence the OR-product graph is not necessary.

**Proof.** As reviewed at (2.8), the fundamental limit of the sum-rate with independent sources problems is the sum of the graph entropy, i.e. $R^*_A = \sum_{n=1}^N R_n$ with $R_n =$
$H_G(X_n)$. By Lemma 3, for any given candidate arg max function,

$$H_G(X_n) = \min_{p(w_n|x_n) \in \{0,1\}, w_n \in \tau(G_n)} I(W_n; X_n)$$

$$= \min_{p(w_n|x_n) \in \{0,1\}, w_n \in \tau(G_n)} H(W_n) - H(W_n|X_n) \quad (3.51)$$

$$= \min_{p(w_n|x_n) \in \{0,1\}, w_n \in \tau(G_n)} H(W_n).$$

Note that the proof of Lemma 3 implies that the maximal independent sets are disjoint, and the fact that one can always Huffman encode the maximal independent sets with any given distribution. Consider using colors to represent the maximal independent sets, then Huffman encode the sets is the same as Huffman encode these colors. This color representation is a normal coloring method w.r.t. the characteristic graph since any two vertices connected by an edge will be in two different independent sets and no two maximal independent sets share the same color. Also note that when the maximal independents are disjoint, distinguish the vertices in the same independent set will result in a higher mutual information in the graph entropy optimization, since for any probabilities $p_i, p_{i+1}$ of the the nodes in a pairwise maximal independent set $\{\alpha_i, \alpha_{i+1}\}$, the difference of the mutual information will be

$$-p_i \log_2 p_i - p_{i+1} \log_2 p_{i+1} + (p_i + p_{i+1}) \log_2 (p_i + p_{i+1})$$

$$= -p_i \log_2 \left( \frac{p_i}{p_i + p_{i+1}} \right) - p_{i+1} \log_2 \left( \frac{p_{i+1}}{p_i + p_{i+1}} \right) \quad (3.52)$$

$$= (p_i + p_{i+1}) h_2 \left( \frac{p_i}{p_i + p_{i+1}} \right) \geq 0$$

where $h_2()$ is the binary entropy function. Therefore (3.51) can be achieved by graph coloring. Since the scheme we present in Theorem 2 is the optimal coloring method w.r.t. the non-product characteristic graph, it must achieve the minimum in (3.51). Therefore we have the following relationship for all $n \in [N]$ and the fundamental limit
of the sum-rate can be achieved by Theorem 2.

\[
H_G(X_n) \overset{(a)}{=} \lim_{S \to \infty} \frac{1}{S} \min\limits_{S \in G_n(f)} H(c_n(X_n)) \overset{(b)}{=} \min\limits_{c_n \in \mathcal{C}(G_n(f))} H(c_n(X_n)) \overset{(c)}{=} H_G(X_n)
\]  

where (a) holds by [79]; (b) holds by achievability: an ordinary coloring is also an \(\varepsilon\)-coloring, and a valid ordinary coloring on the characteristic graph can be used in replication to achieve a valid coloring on the OR-product graph; and we have proved (c) above. \hfill \Box

We now give the fundamental limit of the sum-rate in the problem that the CEO needs to determine \(Z_M\) and \(Z_{A,M}\) respectively.

**Definition 5.** A function \(f \in \mathcal{F}_M\) is a candidate max function if \(f\) satisfies

\[
d_M((X_{1,s}, \ldots, X_{N,s}), f(X_{1,s}, \ldots, X_{N,s})) = 0
\]  

where \(\mathcal{F}_M\) is the set of all candidate max functions.

**Theorem 4.** In the problem that the CEO needs to decide \(\hat{Z}_M\), if \(\min \mathcal{X} > 0\), then the minimum sum-rate will be

\[
R^*_M = \sum_{n=1}^{N} H(X_n)
\]  

**Proof.** The distortion measure \(d_M\) is

\[
d_M((X_{1,s}, \ldots, X_{N,s}), \hat{Z}_M(s)) = \begin{cases} 
Z_M(s) - \hat{Z}_M(s) & \text{if } \hat{Z}_M(s) \leq Z_M(s) \\
Z_M(s) & \text{otherwise.}
\end{cases}
\]  

Given \(\min \mathcal{X} > 0\), \(Z_M(s)\) can never be 0, the only way to make (3.54) happen is to
let \( \hat{Z}_M(s) \) exactly estimate \( Z_M(s) \), in other words, (3.54) is satisfied if and only if

\[
f(X_{1,s}, \ldots, X_{N,s}) = \max(X_{1,s}, \ldots, X_{N,s}).
\]  

(3.57)

For any node pair \((\alpha_i, \alpha_j)\) in user \( n \)'s characteristic graph, assume \( i < j \) w.l.o.g., we will have \((\alpha_i, \alpha_j) \in G_n\) since there exists \( x_{\setminus \{n\}} = (\alpha_i, \ldots, \alpha_i) \) such that

\[
f(\alpha_i, x_{\setminus \{n\}}) \neq f(\alpha_j, x_{\setminus \{n\}}).
\]  

(3.58)

Therefore, the characteristic graph of user \( n \) w.r.t. \( f \) is complete, and \( \tau(G_n) = \{\{\alpha_i\} : \alpha_i \in \mathcal{X}\} \), and the graph entropy is the same as the entropy of each source.

\[\square\]

Remark. To achieve this limit, we simply need each user to Huffman encode its source.

**Corollary 2.** In the problem that the CEO needs to decide \((\hat{Z}_A, \hat{Z}_M)\), if \( \min \mathcal{X} > 0 \), then the minimum sum-rate will be

\[
R^*_{A,M} = \sum_{n=1}^{N} H(X_n)
\]  

(3.59)

Proof. This directly follows the proof of Theorem 4, the characteristic graph is also complete if \( \min \mathcal{X} > 0 \).

\[\square\]

**Corollary 3.** In the problem that the CEO needs to decide \( \hat{Z}_M \), if \( \min \mathcal{X} = 0 \), then the minimum sum-rate satisfies

\[
R^*_M = NH(X) + N \left( p_1 \log_2 p_1 + p_2 \log_2 p_2 - (p_1 + p_2) \log_2 (p_1 + p_2) \right)
= NH(X) - N \left( p_1 + p_2 \right) h_2 \left( \frac{p_1}{p_1 + p_2} \right)
\]  

(3.60)

where \( \alpha_1 = \min \mathcal{X} = 0, p_1 = \mathbb{P}(X = \alpha_1), p_2 = \mathbb{P}(X = \alpha_2) \).
Proof. Let $f \in \mathcal{F}_M$ satisfies that $f(\alpha_1, \ldots, \alpha_L) = \alpha_2$, then $(\alpha_1, \alpha_2) \not\in G$ for the characteristic graph of each source $X_i$ w.r.t. $f$. The graph is not complete and the set of independent sets is $\tau(G_n) = \{\{\alpha_1, \alpha_2\}, \{\alpha_3\}, \ldots, \{\alpha_L\}\}$ for all $n \in [N]$. Hence by a similar proof as in Theorem 2 and Theorem 3, we achieve

$$R^*_M = \sum_{n=1}^{N} H_G(X_n) \tag{3.61}$$

where

$$H_G(X_n) = -(p_1 + p_2) \log_2 (p_1 + p_2) - \sum_{i=3}^{L} p_i \log_2 p_i \tag{3.62}$$

for all $n \in [N]$. \qed

### 3.4 Scaling in Number of Users

In this subsection, we consider the rate-saving performance of graph coloring in a large scale of $N$. Define the rate savings as the difference between the scheme that each user Huffman encode its source and the scheme by Theorem 2, i.e.

$$\Delta_A \triangleq \sum_{n=1}^{N} H(X_n) - R^*_A. \tag{3.63}$$

**Theorem 5.** To losslessly determine the arg max, the savings $\Delta$ is bounded by

$$\max_{p_1, \ldots, p_L} \Delta_A = \max_{p_1, \ldots, p_L} \sum_{i=1}^{L-1} (p_i + p_{i+1}) \frac{p_i}{p_i + p_{i+1}} \leq 2 \tag{3.64}$$

where $p_i = \mathbb{P}(X = \alpha_i)$, and hence the per user saving satisfies that

$$\lim_{N \to \infty} \frac{\Delta_A}{N} = 0 \tag{3.65}$$
Proof. By Theorem 2 we have

\[
\Delta_A = NH(X) - (N - 2)H(X) + \sum_{i=1}^{L-1} (p_{i,i+1} \log_2 p_{i,i+1}) + p_1 \log_2 p_1 + p_L \log_2 p_L
\]

\[
= -2 \sum_{i=1}^{L} (p_1 \log_2 p_i) + \sum_{i=1}^{L-1} (p_{i} \log_2 p_{i,i+1}) + \sum_{i=1}^{L-1} (p_{i+1} \log_2 p_{i,i+1}) + p_1 \log_2 p_1 + p_L \log_2 p_L
\]

\[
= - \sum_{i=1}^{L-1} (p_i \log_2 p_i) - \sum_{i=2}^{L} (p_i \log_2 p_i) + \sum_{i=1}^{L-1} (p_{i} \log_2 p_{i,i+1}) + \sum_{i=1}^{L-1} (p_{i+1} \log_2 p_{i,i+1})
\]

\[
= \sum_{i=1}^{L-1} (p_i + p_{i+1}) h_2 \left( \frac{p_i}{p_i + p_{i+1}} \right)
\]

\[
\leq \sum_{i=1}^{L-1} (p_i + p_{i+1}) < 2 \sum_{i=1}^{L} p_i = 2. \tag{3.66}
\]

Hence,

\[
\lim_{N \to \infty} \frac{\Delta_A}{N} = 0. \tag{3.67}
\]

\[
\square
\]

Corollary 4. In the problem that the CEO needs to decide \( \hat{Z}_M \), the per user saving \( \Delta_M/N \) satisfies

\[
\lim_{N \to \infty} \frac{\Delta_M}{N} = \begin{cases} 
0 & \text{if } \min \mathcal{X} = 0 \\
-(p_0 + p_\gamma) h_2 \left( \frac{p_0}{p_0 + p_\gamma} \right) & \text{if } \min \mathcal{X} > 0
\end{cases} \tag{3.68}
\]

where \( p_0 = \mathbb{P}(x = 0) \), \( \gamma = \min \mathcal{X} \setminus \{0\} \), and \( p_\gamma = \mathbb{P}(x = \gamma) \).
Proof. By Theorem 4, we get no savings if \( \min \mathcal{X} > 0 \). If \( \min \mathcal{X} = 0 \), we have

\[
\frac{\Delta_M}{N} = NH(X) - \sum_{n=1}^{N} H_G(X_n)
\]

\[
= N (-p_1 \log_2 p_1 - p_2 \log_2 p_2 + (p_1 + p_2) \log_2 (p_1 + p_2))
\]

\[
= N (p_1 + p_2) h_2 \left( \frac{p_1}{p_1 + p_2} \right)
\]

and

\[
\lim_{N \to \infty} \frac{\Delta_M}{N} = (p_1 + p_2) h_2 \left( \frac{p_1}{p_1 + p_2} \right).
\]

(3.69)

(3.70)

Corollary 5. In the problem that the CEO needs to decide \((\hat{Z}_A, \hat{Z}_M)\), the per user saving \(\Delta_{A,M}/N\) goes to 0 as \(N\) goes to infinity.

Proof. Observe that

\[
R^*_A \leq R^*_{A,M} \leq \sum_{n=1}^{N} H(X_n).
\]

(3.71)

As we shall see in Chapter 4, this lack of savings in this lossless non-interactive problem structure stands in stark contrast to an interactive setup in which it can be showed that, by allowing the CEO to communicate with the users over multiple rounds, a substantial saving in sum rate relative to the the non-interactive scheme can be achieved [83] while still obtaining the answer losslessly.
4. Control Overhead Optimization in the Model of Interactive Communication

4.1 Problem Model – An Interactive Scheme to Maximize the System Throughput

Comparing with the straightforward scheme in which each user uses a Slepian-Wolf (SW) code to forward its observations to the CEO to enable it to learn the arg max, we showed in Chapter 3 that it is possible to save some rate by applying graph coloring. However we showed that the maximum possible such savings is small: one can save at most 2 bits for independent and identically distributed sources and the per user saving as the number of users goes to infinity will be 0. This motivated us to investigate other coding strategies capable of delivering a larger reduction in rate. While one may consider strategies that enabled this rate reduction by relaxing the requirement that the CEO compute the extremizations perfectly to computing them in a lossy manner, here we will still consider the requirement that the extremizations are computed with zero distortion or losslessly and but focus on rate savings obtainable through interactive communication.

*Interactive communication* is defined to be a method of communication that allows message passing forward and backward multiple times between two or more terminals [39]. It has been shown that interactive communication can provide substantial rate savings over non-interactive communication in some distributed function computation problems [42]. Here, we will apply interactive communication to the extremization problems, and show that a large reduction in rate is possible relative to the non-interactive lossless limits presented in Chapter 3. We will demonstrate that through interaction we can obtain substantial rate savings.
In particular, inspired by the selective multiuser diversity (SMUD) [84] scheme as well as the multi-predefined thresholds [85] scheme which is an extension of SMUD, we propose here the Multi-Thresholds Interactive Scheme (MTIS) between the CEO and the users that efficiently encodes the feedback necessary for the lossless computation of the extremization problems. We show that the MTIS achieves a large reduction in the rate when interaction is utilized when compared with the rate results of Theorem 1 in Chapter 3 in which each user sends its own message to the CEO by graph coloring.

Here we will model the observations of the users’ channel qualities as identically distributed discrete random variables with support set \( \mathcal{X} = \{\alpha_1, \ldots, \alpha_L\} \) s.t. \( 0 < \alpha_1 < \alpha_2 < \ldots < \alpha_L \), and cumulative distribution function \( F_x(x) \). The users each initially occupy a fraction of a bandwidth to communicate to the CEO. The CEO knows the user index and the part of the bandwidth that it corresponds to at the beginning. The interactive communication will occur over multiple rounds indexed by \( \omega \). During each round, only a subset of the users called the online users will participate in the communication, and the CEO will know which users are offline by the information it exchanges with the online users. For instance, in the arg max case, a user remains online only while it is still possible to be the arg max based on the information it has received up until this round, and is offline otherwise. The part of communication bandwidth associated with offline users is freed up for use by other communications and is thus not wasted. During round \( \omega \), given the CDF \( F_\omega(x) \), the support set \( \mathcal{X}_\omega = \{\alpha_1^\omega, \ldots, \alpha_L^\omega(\omega)\} \) and the \( N_\omega \) conditioned on the information that the CEO obtained about the online users thus far, it will determine and send a common message \( V_\omega \) to declare a threshold to each of the online users, and each online user \( n \) responds with a message \( U_n^\omega \) to let the CEO know whether or not it is above this threshold for all \( n \in [N_\omega] \). The user will stay online for the next round if it feeds back a 1. Alternatively, if a user feeds back a 0, but the next threshold \( \lambda_{\omega+1} \)
is lower than $\lambda_\omega$ (which indicating that all users replied 0 at round $\omega$), it will also stay online, otherwise this user becomes offline. After receiving all of the feedback bits, the CEO can obtain the information $F_{\omega+1}(x), x_{\omega+1}$ and $N_{\omega+1}$ for next round’s communication. If there is only one user above the threshold $\lambda_\Omega$ at the round $\Omega$, this user is the arg max and the communication process stops. Similarly, if $|x_\Omega| = 1$, then all of the online users in the next round attain the max, and the communication process stops since the CEO can pick any one of these users to be the arg max. If more than one online user replies a 1, then conditioned on all the information received thus far, the new channel distribution parameters for the next round are

$$N_{\omega+1} = \sum_{n=1}^{N_\omega} \mathbb{1}_{x_n \geq \lambda_\omega}$$

$$\alpha_1^\omega = \lambda_\omega$$

$$\alpha_{L(\omega+1)}^\omega = \alpha_{L(\omega)}^\omega$$

$$F_{\omega+1}(x) = \frac{F_\omega(x) - F_\omega(\lambda_\omega)}{F_\omega(\alpha_{L(\omega)}^\omega) - F_\omega(\lambda_\omega)}$$

(4.1)

While if all users reply 0, then conditioned on all the information received thus far at the CEO, the new channel distribution parameters for the next round are

$$N_{\omega+1} = N_\omega$$

$$\alpha_1^{\omega+1} = \alpha_1^\omega$$

$$\alpha_{L(\omega+1)}^{\omega+1} = \lambda_\omega$$

$$F_{\omega+1}(x) = \frac{F_\omega(x) - F_\omega(\alpha_1^{\omega+1})}{F_\omega(\lambda_\omega) - F_\omega(\alpha_1^{\omega+1})}$$

(4.2)

The threshold for next round can be generated based on the new information. Hence the algorithm of MTIS operates as follows.
Algorithm 1: Muti-Thresholds Interactive Scheme

**Result:** Let the CEO decide the arg max

 initialization: number of online users $N_1 = N$, the support set and the CDF of 
 the discrete source random variables $\mathcal{X}_1 = \mathcal{X}, F_1(x) = F_x(x)$

**while** $N_\omega > 1 \Leftrightarrow |\mathcal{X}_\omega| > 1$ **do**

  step 1) CEO sends threshold $\lambda_\omega$ to all users
  step 2) online users generate the parameters $\mathcal{X}_\omega$ and $F_\omega(x)$ according to 
  Equation 4.1 and Equation 4.2, and decide to stay online or not
  step 3) online users send $U_{\omega}^n = 1_{x_n \geq \lambda}$ for all $n \in [N_\omega]$
  step 4) CEO generates the parameters $N_{\omega+1}, \mathcal{X}_{\omega+1} and F_{\omega+1}(x)$ according to 
  Equation 4.1 and 4.1

---

**Figure 4.1:** MTIS Vs. non interactive for the arg max problem with $N_1 = 8$

### 4.2 Analysis

Our aim in this subsection is to determine the optimal choice of the thresholds in 
the interactive scheme in the sense of minimizing the average total amount of rates 
must incur.

Define $R$ to be the total expected number of overhead bits exchanged when using
the series of threshold levels \( \lambda_1, \lambda_2, \cdots \), and define \( R^* \) to be

\[
R^* = \min_{\lambda_1, \lambda_2, \ldots} R(\lambda_1, \lambda_2, \ldots)
\]  \hspace{1cm} (4.3)

It is clear that \( R^* \) will be a function of the initial number of users \( N_1 \) (all of whom are initially online) and \( \mathcal{X}_1 \). We will need the following theorem to solve the optimization problem.

**Theorem 6.** Problem (4.3) is a dynamic programming problem.

**Proof.** We first show there will be a finite stop \( \Omega \) for (4.3). The threshold \( \lambda_\omega \) is picking from the support set of the sources \( \mathcal{X} = \{ \alpha^1_1, \ldots, \alpha^w_{L(\omega)} \} \). After each round of communication, the support set will be updated to either \( \{ \alpha^1_{\lambda_1}, \ldots, \lambda_\omega \} \) or \( \{ \lambda_\omega, \ldots, \alpha^w_{L(\omega)} \} \), hence the size of the support set is monotone decreasing. Therefore only a finite number of rounds is needed to decrease the support set to have cardinality 1, and then the communication stops.

Also, we observe that if policy \( \lambda^*_1, \ldots, \lambda^*_\Omega \) is the optimal choice of thresholds for initial condition \( N_1, \{ \alpha^1_1, \ldots, \alpha^1_{L(1)} \} \) and \( F_1(x) \) then the truncated policy \( \lambda^*_\omega, \ldots, \lambda^*_\Omega \)
will be the optimal choice of thresholds for initial condition $N_\omega$, $\{\alpha_1^\omega, \ldots, \alpha_L^\omega\}$ and $F_\omega(x)$, and thus the problem has the form of a dynamic programming problem.

In order to solve this problem, we begin with a one round analysis in which we assume to pick $\lambda_\omega$ as the threshold for round $\omega$ and that the thresholds after round $\omega$ have been optimized already. Define $R_\omega(\lambda_\omega)$ as the expected aggregate rate from round $\omega$ to the end, then $R_\omega(\lambda_\omega) =$

$$
H(\lambda_\omega|\lambda_1, \mathcal{X}_1, N_1, \ldots, \lambda_{\omega-1}, \mathcal{X}_{\omega-1}, N_{\omega-1}) + N_\omega + \mathbb{E}[R_{\omega+1}]
\tag{4.4}
$$

where the first term represents the minimum number of bits needed to let the users know the threshold in round $\omega$, the second term represents the total number of bits of feedback from the $N_\omega$ users, and the last term represents the expected rate cost for future rounds which can be further expressed as

$$
\mathbb{E}[R_{\omega+1}] = \sum_{n=0}^{N_\omega} p_n \mathbb{E}[R_{\omega+1}|n] = (F_\omega(\lambda_\omega))^{N_\omega} R^*(N_\omega, \alpha_1^\omega, \lambda_\omega)
$$

$$
+ \sum_{n=1}^{N_\omega} (1 - F_\omega(\lambda_\omega))^n F_\omega(\lambda_\omega)^{N_\omega-n} \binom{N_\omega}{n} R^*(n, \lambda_\omega, \alpha_L^\omega)
\tag{4.5}
$$

where $p_n$ represents the probability of $n$ users reply 1 at round $\omega$. The optimal choice of threshold at round $\omega$ then must satisfy

$$
\lambda_\omega^* = \arg \min_{\lambda_\omega} R_\omega(\lambda_\omega)
\tag{4.6}
$$

(4.4) (4.5) and (4.6) together form a policy iteration algorithm[81] for this dynamic programming problem.
4.3 Thresholds vs. Number of Users

Let us now consider several possible methods of encoding the threshold, and hence several possible values for the quantity $H(\lambda_\omega|\lambda_1, X_1, N_1, \cdots, \lambda_{\omega-1}, X_{\omega-1}, N_{\omega-1})$ in (4.4). Based on SW codes, the minimum information the CEO needs to broadcast should be the conditional entropy of the threshold given all previous knowledges that the online users have.

For the purposes of comparison, and ease of the associated algorithm encoder design, let us also consider two additional coding strategies which are easy to implement. We will see that these two strategies also require less communication than the non-interaction scheme. The first strategy is to encode the threshold with no conditioning

$$U_\omega = H(\lambda_\omega) = \log_2 |\mathcal{X}_\omega|$$  \hspace{1cm} (4.7)

Motivated by the idea that the users may calculate the optimal choice of threshold themselves rather than receiving it, we provide the second strategy that the BS broadcasts the number of currently online users. Observe that the optimal policy $\lambda_\omega^*$ at each round is determined by the information the CEO has, including $N_\omega$, $f_\omega(x)$ and $\mathcal{X}_\omega = \{\alpha_1^\omega, \ldots, \alpha_{L(\omega)}^\omega\}$. We show that it is enough to let the users calculate the threshold by broadcasting $N_\omega$ by induction.

**Theorem 7.** The number of online users $N_\omega$ is a sufficient statistic of the optimal threshold $\lambda_\omega^*$.

**Proof.** (4.4) (4.5) (4.6) show that the CEO determines the $\lambda_\omega^*$ by the information of $\{(F_i(x), \mathcal{X}_i, N_i) : i \in [\omega]\}$, hence it suffices to show that the users can learn $F_\omega(x)$ and $\mathcal{X}_\omega$ by knowing $N_\omega$ at round $\omega$. We prove it by induction. At round 1, each user has the CDF $F_1(x)$, the support set $\mathcal{X}_1 = \{\alpha_1^1, \ldots, \alpha_{L(1)}^1\}$ and its own value $x_n$, hence the optimal threshold $\lambda_1^*$ can be calculated after receiving the initial number of the online
users $N_1$. Suppose that at round $\omega - 1$ the users successfully compute the threshold $\lambda_{\omega-1}^*$ by the information $N_{\omega-1}$, $F_{\omega-1}(x)$ and $X_{\omega-1} = \{\alpha_{1}^{\omega-1}, \ldots, \alpha_{L(\omega-1)}^{\omega-1}\}$. Now at round $\omega$ for any user $n \in [N_{\omega-1}]$, if it receives $N_\omega = N_{\omega-1}$ and its value is below the threshold $\lambda_{\omega-1}$ which means it replied a 0 at previous round, then it knows that every user must be below the previous threshold and $X_\omega = \{\alpha_{1}^{\omega-1}, \lambda_{\omega-1}^*\}$; similarly if it receives $N_\omega = N_{\omega-1}$ and its value is above the threshold $\lambda_{\omega-1}$, then it knows that every user must be above the previous threshold and $X_\omega = \{\lambda_{\omega-1}^*, \alpha_{L(\omega-1)}^{\omega-1}\}$. Therefore the $X_\omega$ can be renewed at each user by the following rules

$$X_{\omega} = \begin{cases} \{\lambda_{\omega-1}^*, \alpha_{L(\omega-1)}^{\omega-1}\} & \text{if } N_\omega < N_{\omega-1} \\ \{\alpha_{1}^{\omega-1}, \lambda_{\omega-1}^*\} & \text{if } N_\omega = N_{\omega-1} \text{ and } \lambda_{\omega-1}^* > x_n. \\ \{\lambda_{\omega-1}^*, \alpha_{L(\omega-1)}^{\omega-1}\} & \text{if } N_\omega = N_{\omega-1} \text{ and } \lambda_{\omega-1}^* \leq x_n \end{cases} \quad (4.8)$$

Note that the user will turn offline if $N_\omega < N_{\omega-1}$ and $\lambda_{\omega-1}^* > x_n$ and stay online otherwise. The updated CDF $F_\omega(x)$ can be get by (4.1) (4.2) once $X_\omega$ has been renewed. Therefore, the threshold $\lambda_\omega^*$ can be determined after each user receiving the $N_\omega$.

4.4 Results—Interaction in the $\text{arg max}$ case

Having identified the policy iteration form of the problem of minimizing the expected aggregate rate exchanged for the MTIS scheme for determining the user with the $\text{arg max}$, we now solve the policy iteration for the various methods of communicating the thresholds. We will measure the amount of communication required in each case and compare with the amount of information which must be transmitted without interaction. As we mentioned before, (4.4) (4.5) (4.6) can be solved by iteration with
the boundary condition

$$ R^*(N_\omega, \mathcal{X}_\omega) = 0 $$

(4.9)

if $N_\omega = 1$ or $|\mathcal{X}_\omega| = 1$. Fig. 4.1, 4.2, 4.8, 4.9, 4.3, 4.4 and 4.5 present the number of bits communicated under the various schemes when the sources are uniformly distributed. Figure 4.1 compares the bits communicated by MTIS, with SW coded thresholds achieving the conditional entropies (4.4), and the non-interactive scheme with $N_1 = 8$, while Figure 4.2 performs the same comparison with $\mathcal{X}_1 = 64$. From both figures we can see significant rate savings through interaction when calculating the arg max.

As mentioned in previous section, we suggested two simple encoding strategies for the base station to broadcast which include Huffman encoding the $\lambda^*_\omega$ with no conditioning on previous thresholds and Huffman encoding the $N_\omega$. Figure 4.3 shows the number of bits that must be exchanged when these methods are used. The strategy of sending the threshold outperforms the strategy of sending the number of users in the situation that the initial number $N_1$ is large; while when $N_1$ is small, the latter shows

Figure 4.3: A comparison of sending thresholds and sending number of users with $|\mathcal{X}_1| = 16$. 

![Graph showing comparison between sending thresholds and sending number of users.](image-url)
Figure 4.4: MTIS Vs. non-interactive for the max problem with $N_1 = 8$

better performance. The minimum between these two schemes requires an amount of communication close to the best scheme, which SW encodes the thresholds.

4.5 Results– Interaction in the max and (arg max, max) Case

We can also apply the achievable interaction scheme in the problem that the exact maximum value need to be decided as well as the problem that both the max and arg max need to be decided, following the same analysis as (4.1) to (4.6) with the only difference being the boundary conditions. Instead of (4.9), we will have the following condition for determining the max and the pair

$$R^*(N_\omega, \mathcal{X}_\omega) = 0 \iff |\mathcal{X}_\omega| = 1.$$  \hspace{1cm} (4.10)

For the problem that the CEO wants to learn the max or the pair (max, arg max), Figure 4.4 compares the bits communicated by MTIS, with SW coded thresholds achieving the conditional entropies (4.4), and the non-interactive scheme with $N_1 = 8$, while Figure 4.5 performs the same comparison with $\mathcal{X}_1 = 16$. Note that case 1 and
case 3 share the same boundary conditions and hence have the same rates because once the CEO knows the maximum value, it can pick any one of the online users that achieves the maximum. Also note that by Theorem 4, the one-way fundamental limit of determining the max is $NH(X)$ because we have selected $\min X > 0$.

4.6 Scaling Laws

We have shown for the lossless non-interactive communication, one can have at most 2 bits saving for the arg max case, and the per user saving goes to 0 as the number of users goes to infinity. Now we will see our proposed interactive scheme will exhibit a better scaling law.

**Theorem 8.** For the case that two users each observe uniformly distributed independent discrete sources, the aggregate expected rate required to losslessly determine the arg max by interactive communication satisfies

$$R^* < 6 - 6 \left( \frac{1}{2} \right)^{\log_2 L} < 6$$

(4.11)
\[
\lim_{L\to\infty} \frac{R^*}{L} = 0 \quad (4.12)
\]

**Proof.** We will derive an upper bound on the amount of information exchanged by MTIS by choosing non-optimal thresholds and transmitting \(N_\omega\) instead of the threshold. The users, instead of computing \(\lambda^*_\omega\) by dynamic programming, will always pick the median of \(X_\omega\) as the threshold and send a 1 bit message indicating whether its observation is in \(\{\alpha^\omega_1, \lambda^-\omega\}\) or \(\{\lambda_\omega, \ldots, \alpha^\omega_{L(\omega)}\}\), where \(\lambda^-\omega\) is the nearest level to \(\lambda_\omega\) that \(\lambda^-\omega < \lambda_\omega\). The CEO then also replies a 1 bit message indicating whether or not the two users are in the same region. The communication process stops if the two users are not in the same region, otherwise the problem degenerates to a 2-user arg-max problem with support set shrinking to a half of the original size. Define \(R(L)\) as the expected aggregate rate by this interactive scheme with support set \(\{\alpha_1, \ldots, \alpha_L\}\) in the 2-user arg-max problem.

\[
R(L) \stackrel{(a)}{=} 2 + 1 + (p^\omega_1 p^\omega_2) (R(\lceil L/2 \rceil) + R(\lfloor L/2 \rfloor))
\leq 3 + 2p^\omega_1 p^\omega_2 R(\lceil L/2 \rceil)
\leq 3 + 0.5 (R(\lfloor L/2 \rfloor)) \quad (4.13)
\]

where \(p^\omega_1 = \mathbb{P}(x \in \{X^\omega_1, \lambda^-\omega\})\), \(p^\omega_2 = \mathbb{P}(x \in \{\lambda_\omega, X^\omega_{L(\omega)}\})\). Where the 2 in (4.13) stands for the 2 bits communicated by the two users in this round, the 1 stands for the replied bit from the CEO, and the last term stands for the case that both users either reply 1 or 0. As (4.9) suggests, we have \(R(1) = 0\), hence for any \(X = \{\alpha_1, \ldots, \alpha_L\}\) we have

\[
R(L) - 6 \leq \frac{1}{2} ((R(\lceil L/2 \rceil)) - 6)
\leq \left(\frac{1}{2}\right)^m (R(1) - 6)
= -6 \left(\frac{1}{2}\right)^m \quad (4.14)
\]
where $2^{m-1} \leq L \leq 2^m$, and therefore

$$R^* \leq R(L) \leq 6 - 6\left(\frac{1}{2}\right)^m < 6. \tag{4.15}$$

\begin{proof}

\textbf{Theorem 9.} Let $\Delta_A = R_A^* - R^*$ be the rate saving of the proposed interactive scheme w.r.t. the lossless non-interactive limit $R_A^*$ in the arg max problem, the per-user saving $\Delta_A/N$ satisfies

$$\lim_{N \to \infty} \frac{\Delta_A}{N} \geq H(X) - 1 \tag{4.16}$$

\textbf{Proof.} We propose an interactive scheme which will derive an upper bound on the amount of information exchanged by MTIS by choosing $\lambda = \max \mathcal{X}$. Define $R_U(\mathcal{X}, N)$ as the expected aggregate rate of this scheme, we know $R_U \geq R^*$, and

$$R_U(\{\alpha_1, \ldots, \alpha_L\}, N) = (1 - p_L)^N R_U(\{\alpha_1, \ldots, \alpha_{L-1}\}, N)$$

$$+ (1 - (1 - p_L)^N) \cdot 0 + H(X) + N$$

$$\leq (1 - p_L)^N R_U(\{\alpha_1, \ldots, \alpha_L\}, N) + H(X) + N$$

$$\leq (1 - p_L)^N NH(X) + H(X) + N$$

where the first two terms in the last equation stand for the expected rate cost for future rounds, $p_L = \mathbb{P}(X = \alpha_L)$, $H(X)$ stands for the bits required to send the threshold $\lambda = \max \mathcal{X}$ and $N$ stands for the bits replied by the $N$ users. Hence by
(3.16), and the fact that \( \lim_{N \to \infty} (1 - p_L)^N = 0 \), we have

\[
\lim_{N \to \infty} \frac{\Delta A}{N} \geq \frac{1}{N} \sum_{i=1}^{L-1} \log_2 p_{i,i+1} - \log_2 p_1 - p_L \log_2 p_L \\
- (1 - p_L)^N \cdot \frac{1}{N} \sum_{i=1}^{L-1} \log_2 p_{i,i+1} - p_1 \log_2 p_1 - p_L \log_2 p_L \\
- (1 - p_L)^N \cdot NH(X) - H(X) - N \\
= H(X) - 1.
\]

(4.17) (4.18)

### 4.7 Comparison with Other Interactive Schemes

As an interesting point of comparison, we compare the MTIS with another two interactive schemes. Both of the two schemes are given in [42] as examples that show interaction can enable rate savings relative to non-interactive schemes in distributed function computation problems. In both schemes, it is assumed that when the user sends a message, the CEO knows without cost which user this message is from. Additionally, in the first scheme, referred to as Relay Interactive Scheme (RIS), the users transmit sequentially with one user transmitting at a time for reception by the next user. The second scheme, called Non-Broadcasting Interactive Scheme (NBIS), has an additional constraint that all communication must occur between the CEO and users and the CEO can only communicate to one user at a time. Here we illustrate the schemes for 3 users. Pseudocode for the two schemes is provided in Algorithms 2 and 3 respectively. Note that both of the two schemes require computing the max function, therefore by Theorem 4, no rate can be saved by graph coloring.

In Figure 4.8 and Figure 4.9, we see that for uniformly distributed sources, the NBIS has better performance than MTIS when there is only two users. For most of
Figure 4.6: The 3 users case of the CEO extremization problem with RIS

Figure 4.7: The 3 users case of the CEO extremization problem with NBIS

Figure 4.8: A comparison of the three interactive schemes with $N_1 = 8$
**Algorithm 2: Relay Interaction Scheme**

**Result:** Let the CEO decide the arg max
initialization: number of users $N$, the support set and the CDF of the source
random variables $\mathcal{X} = \{\alpha_1, \ldots, \alpha_L\}, F_x(x)$;
step 1) user 1 sends its value to user 2;
step 2) user 2 computes $\max\{x_1, x_2\}$ and sends it with its index (the arg max) to user 3;
\[\vdots\]
step L-1) user $N-1$ computes $\max\{x_1, \ldots, x_{N-1}\}$ and sends it with its index to user $N$;
step L) user $N$ computes $\max\{x_1, \ldots, x_N\}$ and sends its index to the CEO;

**Algorithm 3: Non-Broadcasting Interaction Scheme**

**Result:** Let the CEO decide the arg max
initialization: number of users $N$, the support set and the CDF of the source
random variables $\mathcal{X} = \{\alpha_1, \ldots, \alpha_L\}, F_x(x)$;
step 1) user 1 sends its value to the CEO;
step 2) CEO forwards user 1’s value to user 2;
step 3) user 2 computes $\max\{x_1, x_2\}$ and sends it to the CEO;
step 4) CEO learns both the arg max and the max of the first 2 users and forwards $\max\{x_1, x_2\}$ to user 3;
\[\vdots\]
step 2N-3) user $N-1$ computes $\max\{x_1, \ldots, x_{N-1}\}$ and sends it to the CEO;
step 2N-2) CEO learns both the arg max and the max of the first $N-1$ users and forwards $\max\{x_1, \ldots, x_{N-1}\}$ to user $N$;
step 2N-1) user $N$ computes $\max\{x_1, \ldots, x_N\}$ and sends it to the CEO;

The cases, the MTIS utilizes fewer overhead bits than the other two schemes.

In summary, we observe from Figure 4.1 – Figure 4.9 that the MTIS provides a substantial saving in sum rate relative to the non-interactive scheme as well as the RIS and the NBIS while still obtaining the answer losslessly. In fact, we observe from Theorem 9 that the per-user rate goes to 1 as the number of users goes to infinity, which is a very large reduction relative to the minimum necessary communication if non-interaction is required.
Figure 4.9: A comparison of the three interactive schemes with $|\mathcal{X}_1| = 16$
5. A Framework for Rate Efficient Control of Distributed Discrete Systems

So far we considered resource allocation problems in which a resource controller needs to compute an extremization function over a series of \( N \) remote users. The designs were developed that minimized the amount of information exchange necessary for this remote function computation. We have shown that, in most of the cases where the extremization must be computed losslessly, at most two bits can be saved relative to the direct scheme in which the users simply forward their metrics to the controller which computes the function. In contrast to this lossless case, we observed that substantial rate savings can be achieved if the controller and the users are allowed to interactively communicate, and the rate savings by applying interaction improves with the number of users.

The results we got were all built on the assumption that the objective of the base station is to pick the user with highest channel quality to schedule the data transmission, in other words, we simply wish to maximize the potential system throughput. However, in a real communication system, the base station would like to distinguish not only good channel quality users with bad channel quality users, but also heavy load (i.e. video downloading) users with light load (i.e. texting) users. Motivated by this, we introduce a buffer setup in the wireless resource allocation problem where we assume the base station has a shared buffer containing the data packets that must be sent to the users (illustrated as in Figure 5.1). The resource allocation decision now must be made by considering both the users’ channel qualities and the buffer information, i.e. a user with the best channel quality but no data packet left for downloading will no more be scheduled. We consider the control overhead minimization in the buffer related wireless resource allocation problem by modeling the problem as a
Figure 5.1: The resource allocation with buffer limitation model: optimizing control decisions and encoder mappings jointly

distributed stochastic control problem where communication cost is involved in the transition reward function of the distributed Markov decision process of this control problem. Under this model setup, the resource allocation decision is not given explicitly as any extremization functions. Hence we have to jointly consider two problems, one is to find the minimum sum-rate achieving coding scheme as in Chapter 3 and Chapter 4, the other is to find the optimal resource allocation decision with respect to the encoded control messages.

As we can see in the following sections, we study the problem of determining distributed discrete control decision with information theoretic optimized communication cost. We assume that no single node has full knowledge of the global system state which is composed of local states hence communication must be implemented. We consider the collocated communication network where every one can overhear the other users’ message about their local states. We also assume that overhearing other users’ messages is enough for a node to learn the control action even if it has no
access to any local state. For this problem, we first study the fundamental limit of
the minimum communication cost to learn the control action. We then incorporate
into the control decision making to form a control-communication joint optimization
problem. Finally, we propose an alternating optimization algorithm to solve the joint
optimization problem.

The rest of this chapter is organized as follows. In Section 5.1 we present the model
of the discrete stochastic control problem. Minimum communication cost required for
solving the Markov decision process under both non-interactive and interaction com-
munication setup is discussed in Section 5.2. Finally in Section 5.3, communication
is involved in making the control decisions by considering communication cost in the
transition reward function.

5.1 Omniscient Control of a Distributed Markov Decision Process

Consider a distributed discrete stochastic control system modeled a Markov De-
cision Process (MDP) for whom the global system state at time $t$, $S_t \in \mathcal{S}$, is itself
a vector composed of a series of local states $S_{n,t} \in \mathcal{S}_n$ each observed at one network
node $n \in \mathcal{N} = \{1, \ldots, N\}$, so that

$$S_t = [S_{n,t}|n \in \mathcal{N}] \quad \text{and} \quad \mathcal{S} = \mathcal{S}_1 \times \cdots \times \mathcal{S}_N \quad (5.1)$$

The global network state $S_t$ evolves according to a Markov Chain whose transition
matrix at time $t$ is selected by the control action $A_t$

$$\mathbb{P}[S_{t+1} = j | S_t = i, A_t = a] = p_{a}(i,j) \quad \forall i, j \in \mathcal{S}. \quad (5.2)$$

Additionally, there is a reward function $R_a(i,j)$ indicating the payment obtained when
the global system state transitions from $i$ to $j$ after action $a$ is taken.
An omniscient controller having access to the series of global states $S_t, t \in \mathbb{N}$ would select the actions $A_t$ maximizing the total discounted expected reward

\[
\min_{c: \mathcal{S} \to \mathcal{A}} \sum_{t=0}^{\infty} \gamma^t \mathbb{E}[R_{A_t}(S_t, S_{t+1})]. \tag{5.3}
\]

The argument to the solution to this optimization is a mapping $c: \mathcal{S} \to \mathcal{A}$ assigning to each state the optimum action to take, so that the optimal $A_t^* = c(S_t)$. Bellman’s equation states that the solution to this optimization must solve the following system of equations

\[
V^*(i) = \sum_{j \in \mathcal{S}} p_{c^*(j)(i,j)} [R_{c^*(j)(i,j)} + \gamma V^*(j)] \quad \forall i \in \mathcal{S} \tag{5.4}
\]

\[
c^*(i) = \arg \max_{a \in \mathcal{A}} \sum_{j \in \mathcal{S}} p_a(i,j) [R_a(i,j) + \gamma V^*(j)] \quad \forall i \in \mathcal{S} \tag{5.5}
\]

The solution to this simultaneous system of equations, and the associated optimal control mapping, can be found by first determining the limit $V^* = \lim_{k \to \infty} V_k$ of the following value iteration

\[
V_{k+1}(i) = \max_{a \in \mathcal{A}} \sum_{j \in \mathcal{S}} p_a(i,j) [R_a(i,j) + \gamma V_k(j)] \quad \forall i \in \mathcal{S} \tag{5.6}
\]

then solving for the control policy via

\[
c^*(i) = \arg \max_{a \in \mathcal{A}} \sum_{j \in \mathcal{S}} p_a(i,j) [R_a(i,j) + \gamma V^*(j)] \quad \forall i \in \mathcal{S}. \tag{5.7}
\]

Alternatively, one can utilize a policy iteration, which performs a recursion in which first

\[
c_k(i) = \arg \max_{a \in \mathcal{A}} \sum_{j \in \mathcal{S}} p_a(i,j) [R_a(i,j) + \gamma V_k(j)] \quad \forall i \in \mathcal{S} \tag{5.8}
\]
is solved, followed by a solution of the linear system

\[ V_{k+1}(i) = \sum_{j \in S} p_{ca(i)}(i,j) \left[ R_{ca(i)}(i,j) + \gamma V_{k+1}(j) \right] \quad \forall i \in \mathcal{S} \quad (5.9) \]

for \( V_{k+1}(\cdot) \) until the control mapping can be selected to remain the same under the update, at which point the iteration ceases, see, e.g. [81][55].

Note that in many problems, for a given \( i \), there is more than one choice for \( c^*(i) \) achieving the maximum in (5.5). In this instance, one can derive a set of candidate (omniscient) control functions

\[
\mathcal{C} := \left\{ c^* : \mathcal{S} \to \mathcal{A} \mid c^*(i) \in \arg\max_{a \in \mathcal{A}} \sum_{j \in \mathcal{S}} p_a(i,j) \left[ R_{c^*(i)}(i,j) + \gamma V^*(j) \right] \quad \forall i \in \mathcal{S} \right\}
\quad (5.10)
\]

each of which achieves the maximum long run expected reward.

**Example 1** (Downlink Wireless Resource Allocation). An example of a practical problem having the structure outlined by (5.4) and (5.5) is that of distributing resources on a wireless downlink. Time is slotted. As shown in Figure 5.2, there is a basestation which has a shared buffer containing individual information that must be sent to a series of users, with the amount waiting for a user \( n \in \{1, \ldots, N\} \) during time slot \( t \in \mathbb{N} \) being denoted by \( B_{n,t} \). Collectively, these buffer sizes form the local state at the basestation \( S_{n+1,t} = (B_{1,t}, \ldots, B_{N,t}) \). Each user has a channel state \( S_{n,t} \), indicating how much information can be reliably transmitted to this user during the present timeslot, which evolves independently of other users channel states according to a Markov chain with transition distribution \( p(i_n, j_n) = \mathbb{P}[S_{n+1} = j_n \mid S_{n,t} = i_n] \).

During each time slot \( t \), a random amount of additional traffic \( X_{n,t} \) arrives destined for each user \( n \in \{1, \ldots, N\} \) at the basestation’s buffer, independently of other time slots and previous arrivals. If the basestation’s buffer can not accommodate this
The Collocated Communication Setup

\[ \phi(q_1(c_1), q_2(c_2), q_0(b)) = c^*(c_1, c_2, b) \]

for all \((c_1, c_2, b) \in S\)

Figure 5.2: The collocated setup: any third party can learn the control decision after receiving the encoded messages from users and the BS

During each time slot, it must be decided which of the users to give the resource to, and thus this forms the action in the MDP, \(A = \{1, \ldots, N\}\). Additionally, both the users and the basestation must know the outcome of this decision. After the user to transmit to is selected, an amount of their traffic that is the minimum between their capacity during the slot and the amount of traffic waiting for them in the buffer will be successfully transmitted to them and removed from the buffer, yielding the Markov chain dynamics

\[
S_{N+1,t+1} = D(S_{N+1,t} - T(A_t, S_t) + X_{t+1}, X_{t+1}), \quad (5.11)
\]

where \(X_t = [X_{1,t}, \ldots, X_{N,t}]\) is the amount of new traffic arriving during the time slot
New Arrival Data
From the Internet

Figure 5.3: An illustration of the package dropping process: packages for each user take turn moving into the buffer

\[ T_n(A_t, S_{t-1}) = \begin{cases} 
\min\{S_{A_t,t}, B_{A_t,t}\} & n = A_t \\
0 & \text{otherwise}
\end{cases} \]  

while the function \( D \) performs the package dropping process when the arriving traffic can not be accommodated in the buffer, illustrated in Figure 5.3. Note that this assumption, that for this amount of information to be successfully transmitted and received, all the users and basestation need to know is who to schedule, is consistent with assumption that the physical layer below the scheduler uses a rateless code with feedback, which can be closely approximated with hybrid ARQ [86, 87, 88].

When making the decision of who to schedule, several important metrics can be considered, and thereby combined, into the reward. A very natural metric is the throughput, which measures how much information is transmitted, summed over all
the users. This gives the reward function

\[ R_a(S_t, S_{t+1}) = \min(S_{a,t}, B_{a,t}) \] (5.13)

Other metrics such as the average or maximum delay a user's traffic experiences are also reasonable metrics which can be incorporated into the reward function, for instance by adding them together with rates that can trade them for one another.

With the selected metrics included in the reward function, the MDP framework gives a formal way of deciding who to allocate the resources to during each time slot. A series of examples throughout the rest of the paper will find this optimal controller and investigate properties, such as how much information must be exchanged in order to perform it.

### 5.1.1 Simulating the Omniscient MDP via Information Exchange

However, as the system is distributed, no single node is given access to the global network state, and control must be performed by the nodes learning which action to take through some sort of communication enabled strategy. Additionally, we will introduce the constraint that an observer not given access to any of the local states, but rather accessing only all of the control messages the nodes share with one another, must be able to infer which action was taken. In order to enable the system to be easily monitored, we further require this user observing no state to be able to learn an optimal action \( A_t^* \) selected exclusively from the information shared during the time slot \( t \) during which the omniscient control action \( A_t \) must be taken.

For such a strategy, a key question is how much information must be shared in order to enable the optimal control action \( A_t^* = c(S_t) \) the omniscient controller would have taken to be selected based on the shared information. In other words, how much information must be shared in order to enable the system to simulate the omniscient
controller in the sense that every node, including the one having access to no local state observations and only observing the shared control messages during time slot $t$, can learn the action the controller will take, thereby enabling the distributed system to obtain the same expected (discounted) long run reward as the omniscient system.

The answer to this question depends, of course, on the model for the way the information is exchanged, the control map $c \in C$, and particular characteristics of the transition kernels $p_n(i,j)$. Clearly the problem of designing this communication has been transformed into one of distributed function computation, as each node observing a local state $S_{n,t}$ must convey a message $M_{n,t}$ during the time slot $t$ such that $A^*_t = c(S_t)$ can be learned from the messages $M_{n,t}$, $n \in N$. Additionally, the capability to select any $c \in C$ and still achieve the maximum reward enables this amount of information exchanged to be further minimized over $c \in C$.

**Example 2** (Downlink Wireless Resource Allocation, Continued). The assumptions made above, and the associated problem of minimizing overhead, have special practical significance for the downlink wireless resource allocation setup of example 1. Only the basestation has direct access to the buffer and observes the amount of information that has arrived destined for the various users, and only the users observe their downlink channel qualities, yet sufficient information must be exchanged for the system to make an informed decision regarding who to schedule on the downlink and how much information to send to them. The omniscient controller having access to all of this state distributed throughout the network could make a series of decisions by solving the associated MDP. At present, for instance in the LTE and WiMax standards, these decisions are made by the basestation, which requests and receives channel quality statistics from the users, then schedules the users and how much information to send to them, transmitting its decisions on the downlink [5]. Additionally, it is desirable to minimize this amount of control information through an efficient design, as this type
of control measurement and decision information, together with reference signals, has reached roughly a quarter to a third of the time frequency footprint in LTE and LTE advanced [5]. Furthermore, it is essential that the messages exchanged during time slot \( t \) are all that must be overhead in order to learn the control decision action, as nodes come and go from the network, and it is essential that nodes that have just arrived in the current slot be able to determine what the control decisions were. Finally, we note that it is evident from the problem description for example 1, that the various local states, which are the channel state at each user and the buffer state at the basestation, evolve according to independent Markov chains given the control actions.

5.2 Minimal Coordination Communication Required for Distributed Simulation of the MDP

For general models with arbitrary dependence between observations, multiterminal information theory has yet to determine the minimum sum rate required for distributed function computation, however, these limits are known for a handful of special cases, including those where the local observations are independent. In this independent case, the transition kernel and initial state distribution admit a factorization

\[
p_a(i, j) = \prod_{n \in \mathcal{N}} p_a(i_n, j_n), \quad \text{and} \quad \mathbb{P}[S_0 = i] = \prod_{n \in \mathcal{N}} \mathbb{P}[S_{n,0} = i_n].
\]  

(5.14)

This factorization, in turn, implies that the local states evolve independently of one another, once an action has been specified, and as such, the quantities available to be encoded into messages at the nodes are independent of one another, so that

\[
\mathbb{P}[S_t = i] = \prod_{n \in \mathcal{N}} \mathbb{P}[S_{n,t} = i_n] \quad \forall i \in \mathcal{S}.
\]

The fundamental limits for this special case can be further subdivided based upon whether the messages \( M_{n,t} \) must all be sent in parallel and in a non-interactive manner, or if interaction between users over multiple
rounds of communication during one time slot is allowed.

5.2.1 One Shot, Non-Interactive Distributed Simulation of the Omni-
scient MDP

In the non-interactive case, under the assumptions made regarding the monitoring
node, the minimum sum-rate required for distributed function computation of a given
control map $c \in C$ with independent sources (in this case, the independent states)
is given by the sum of the graph entropies of the characteristic graphs for each user
[26][27].

The characteristic graph $G_n(c)$ for user $n$ has as its set of nodes $V_n = S_n$ the possible
local states of user $n$. An edge $\{i_n, j_n\} \in E_n$ exists in the characteristic graph if there
are values $i_n' \in S_n, n' \in N \setminus \{n\}$ such that $P[S_t = (i_1, \ldots, i_{n-1}, i_n, i_{n+1}, \ldots, i_N)] > 0$ and $P[S_t = (i_1, \ldots, i_{n-1}, j_n, i_{n+1}, \ldots, i_N)] > 0$ and $c(i_1, \ldots, i_{n-1}, j_n, i_{n+1}, \ldots, i_N) \neq c(i_1, \ldots, i_{n-1}, i_n, i_{n+1}, \ldots, i_N)$. Since the local states are independent, the probability distribution will be positive
on a product support $S_n^{>0}, n \in N$, and there is no edge, i.e. $\{i_n, j_n\} \notin E_n$; if for all pos-
sible values of other local states $i_n' \in S_n^{>0}, n' \in N \setminus \{n\}, c(i_1, \ldots, i_{n-1}, i_n', i_{n+1}, \ldots, S_N) = c(i_1, \ldots, i_{n-1}, j_n, i_{n+1}, \ldots, S_N)$. As such, there is a transitive property in the comple-
ment of the characteristic graph: namely if there is no edge $\{i_n, j_n\}$ and no edge
$\{j_n, k_n\}$ in the characteristic graph, then there is also no edge $\{i_n, k_n\}$, therefore the
maximal independent sets of the characteristic graph do not overlap, and form a par-
tition of the set of vertices of the graph. Owing to this transitivity property in the
complement of the characteristic graphs [26][89], each of these graph entropies is in
fact the chromatic entropy

$$H_{G_n(c)}(S_n, \tau) = \min_{r \in R(G_n(c))} H(r(S_n, \tau))$$  (5.15)
where $\mathcal{R}(\mathcal{G}_n(c))$ represent all colorings of the characteristic graph $\mathcal{G}_n(c)$. This minimum expected rate can be achieved within one bit by Huffman coding the coloring of the characteristic graph achieving the minimum entropy, which then can be achieved by assigning different colors to its different maximal independent sets. Selecting an omniscient control map $c \in \mathcal{C}$ requiring the minimum rate, then gives the minimum non-interactive rate of

$$R_{\text{ni}} = \min_c \sum_{n \in \mathcal{N}} \min_{r \in \mathcal{R}(\mathcal{G}_n(c))} H(r(S_{n,t}))$$

(5.16)

Note further than when searching minimum entropy colorings of the graph to calculate the chromatic entropy, it suffices to consider exclusively the greedy-colorings [90] obtained by iteratively removing maximal independent sets.

The following two examples describe the control rate required under this form of one-shot, non-interactive sharing of quantized local states, for two particular distributed MDPs.

**Example 3** (Minimum control information for arg max, Non-Interactive). Let’s assume in example 2 that the buffer size is infinite, and each user has infinitely many backlogged packets destined for it in the buffer. Then the control decision is made regarding only the users’ channel qualities, and the objective is to let the basestation learn the control decision of which user should occupy the resource block after observing all the messages sent from the users.

Let the local states $S_1, \ldots, S_N$ be independent and identically distributed downlink channel qualities from a known distribution on a discrete support set $\mathcal{S}$, if the basestation wishes to maximize the system throughput, the control decision becomes
finding one of the users with the best channel quality, i.e.

\[ c(S_1, \ldots, S_N) \in \arg \max_{n \in \mathcal{N}} S_n \tag{5.17} \]

For this problem, we have shown in Section 3.2 of Chapter 3 that the characteristic graphs \( G_1(c), \ldots, G_N(c) \) obey the properties that if \( \{i, j\} \notin \mathcal{E}_n \) then \( \{i, k\} \in \mathcal{E}_n, \{j, k\} \in \mathcal{E}_n, \forall k \in \mathcal{S} \), and that if \( \{i, j\} \notin \mathcal{E}_n \) then \( \{i, j\} \in \mathcal{E}_n', \forall n' \in \mathcal{N} \setminus \{n\} \). We have also shown in Section 3.4 of Chapter 3 that the minimum information required to determine the control action can be computed as in (5.16), and at most 2 bits can be saved relative to the scheme in which the users simply send their un-coded channel qualities to the basestation.

**Example 4** (Rate Required for Simulating an Omniscient Wireless Resource Controller with No Interaction). Return to the case of a finite buffer size without any backlogged packets in example 2, and assume that the channel qualities where \( S_{n,t} \) are independently uniformly distributed on the support \( \{0, 1, 2, 3\} \ \forall n \in \mathcal{N}, t \in \mathbb{N} \).

In addition, let the amount of additional traffic that arrives destined for each user be independent across users and time, and be distributed on the support \( \mathcal{X} = \{0, 1, 2\} \) with probabilities \( \{1/2, 1/3, 1/6\} \). Additionally, let the packet dropping function operate according to Algorithm 4, in a manner consistent with a total buffer size of \( BU_{\text{max}} = 3 \). Let the controller aim to maximize the throughput reward (5.13). If \( N = 2 \), which means there are 2 users and 1 basestation in the system, the optimal control decisions by solving the MDP problem of (5.4) and (5.10) with discounting factor \( \gamma = 0.9 \) will give a maximal total discounted reward of 9.249 if the system starts from the all 0 initial state \( S_0 = (S_{1,0}, S_{2,0}, S_{3,0}) = (0, 0, (0, 0)) \). Meanwhile, the expected amount of system throughput per time-slot will be 1.076 and the expected amount of data dropped per time-slot will be 0.257. Calculating the characteristic graphs and determining the Huffman codes associated with the minimum entropy
colorings, we find that the associated optimal control decision can be learned via a quantization of local states (the channel states at each of the two users and the buffer size at the basestation), with a minimum non-interactive rate of 3.5175 bits. The encoder mappings with respect to the minimum rate are given as:

\[
q_1(S_1) = \begin{cases} 
2 & \text{if } S_1 = 0 \\
1 & \text{if } S_1 \in \{1, 2, 3\} 
\end{cases}
\]

(5.18)

\[
q_2(S_2) = \begin{cases} 
1 & \text{if } S_2 \in \{0, 1\} \\
2 & \text{if } S_2 \in \{2, 3\} 
\end{cases}
\]

(5.19)

for the users, and

\[
q_3(S_3) = \begin{cases} 
1 & \text{if } S_3 \in \{(1, 0), (2, 0), (3, 0), (1, 1), (2, 1)\} \\
2 & \text{if } S_3 \in \{(0, 1), (0, 2), (0, 3), (0, 0)\} \\
3 & \text{if } S_3 \in \{1, 2\} 
\end{cases}
\]

(5.20)

for the basestation. A control mapping \(c' : q(S) \to A\) can be decided deterministically with the given optimal control \(c\) and the encoders \(q\), i.e. \(c'(q_1(S_1) = 1, q_2(S_2) = 2, q_3(S_3) = 3) = 2\) and \(c'(q_1(S_1) = 1, q_2(S_2) = 1, q_3(S_3) = 3) = 1\).

5.2.2 Interactive, Collocated Network, Simulation of the Omniscient MDP

In the interactive communication case, a natural lower bound on the rate can be obtained via a collocated network messaging model. In this model, communication happens over multiple rounds, in which users consecutively take turns sending a message which is overheard by all of the other users. In particular, in round \(r\), the node with index \(w(r) = (r - 1) \mod |N| + 1\) sends a message \(M_r^{(1)}\) based on its observation \(S_{w(r), t}\) and all of the messages \(M_t^{(1)}, \ldots, M_t^{(r-1)}\) sent in the previous rounds up until
Algorithm 4: The packet dropping process when the buffer is full

Result: Update next round buffer status $S_{N+1,t+1} = (B_{1,t+1}, \ldots, B_{N,t+1})$

Input: current buffer status $S_{N+1,t} = (B_{1,t}, \ldots, B_{N,t})$, buffer size $BU_{max}$, and new arriving packages $X_{t} = (X_{1,t}, \ldots, X_{N,t})$

- $S_{N+1,t+1} = S_{N+1,t}$
- Remaining $= BU_{max} - \sum_{n=1}^{N} B_{n,t}$
- New $= \sum_{n=1}^{N} X_{n,t+1}$

\[ n = 1 \]

while Remaining $> 0$ & New $> 0$

  if $X_{n,t+1} > 0$ then
    \[ B_{n,t+1} = B_{n,t+1} + 1 \]
    \[ X_{n,t+1} = X_{n,t+1} - 1 \]
    \[ Remaining = Remaining - 1 \]
    \[ New = New - 1 \]
    \[ n = (n + 1) \mod N \]

Output $S_{N+1,t+1}$

This time. After $R$ rounds the communication finishes and the optimum omniscient control action $A_{t}^* = c(S_{t})$ must be completely determined from $M_{t}^{(1)}, \ldots, M_{t}^{(R)}$. Ma and Ishwar [78] have shown that the minimum sum-rate, over all block codes, that can be obtained by such a strategy is lower bounded by the solution to following repeated convex geometric calculation. The solution is written with respect to the rate reduction functional, which maps the coordinates for the marginal probability distributions $p_{n} \in \mathcal{P}(S_{n})$ for each of the local observations $n \in \mathcal{N}$ to a conditional entropy,

$$\rho_{\tau}: \prod_{n \in \mathcal{N}} \mathcal{P}(S_{n}) \to \mathbb{R}, \quad \rho_{\tau}(p) := H(S|M_{(1)}^{(r)}, \ldots, M_{(r)}^{(r)})$$  \hfill (5.21)

if the marginal distribution of $S$ is given by $p$. The rate required if the function is to be computed after $R$ rounds of communication is then expressed as

$$R_{\tau}^{L,wi} = H(S_{t}) - \rho_{\tau}(p_{S_{t}}).$$  \hfill (5.22)
i.e. by evaluating the rate reduction functional at the marginal probability distribution \( p_S \). The rate reduction functional, in turn, is found via the following iterative convex program

\[
\rho_0(p) = \begin{cases} 
H(p) & \exists c \in C \text{ s.t. } c \text{ is constant on supp}(p) \\
-\infty & \text{otherwise}
\end{cases}
\]  

(5.23)

\[
\rho_T(p_1, \ldots, p_{w(r)-1}, p_{w(r)+1}, \ldots, p_N) = 
\text{concavify}(\rho_{T-1}(p_1, \ldots, p_{w(r)-1}, p_{w(r)+1}, \ldots, p_N))
\]  

(5.24)

Here for each fixed context \( p_n \in \mathcal{P}(S_n), \; n \in \mathcal{N} \setminus \{w(x)\} \), the operator concavify is computing the upper concave envelope of the function

\[
\rho_{T-1}(p_1, \ldots, p_{w(r)-1}, p_{w(r)+1}, \ldots, p_N),
\]  

(5.25)

i.e. viewing this restriction as a function from \( \mathcal{P}(S_n) \to \mathbb{R} \).

Note that for the problem at hand, this lower bound may not be achievable, as scalar quantization and coding strategies are required by the assumptions we have made in our distributed MDP setup, while this lower bound may in general only be achieved with a limit of vector quantization schemes. In particular, only scalar quantization is available in the problem under consideration because no repeated observations are available for use in a larger block-length as we have added the constraint that the user overhearing exclusively all of the messages during time slot \( t \) must be able to learn \( A^*_t = c(S_t) \). Nonetheless, as we demonstrate in the following example, often the best scalar quantization based interaction schemes still yield a rate which is very close to this fundamental limit.
Example 5 (Minimum control information for arg max, Interactive). Consider the infinite packet buffer backlog and throughput maximization variant of the wireless resource allocation as described in Example 3, with the added ability that the users and basestation can all interact with one another while sending their messages. In particular, the users can each take turns sending messages, one at a time, over $R$ rounds, such that all the other participants, including the basestation which sends no messages, overhear each message. The goal is to enable anyone who overhears all of these messages to learn the index of at least one user whose channel quality is the same as the maximum channel quality over all of the users. The curve labelled Fundamental limit in Figure 5.4 calculates the fundamental limit (5.24) lower bounding the total number of bits must be exchanged in order to perform this calculation for the case of $N = 3$ and for channel qualities uniformly distributed over the set $\{1, 2, 3, 4\}$. As explained above, this fundamental limit is in general only achievable with vector quantization schemes, while the problem setup at hand demands that scalar quantization schemes must be used. Additionally, a second curve in Figure 5.4, labelled “optimal scalar Hete. Q” gives the rate required by the best possible scalar quantization scheme, followed by Huffman coding, for this problem, and this is seen to be quite close to the vector quantization limit. This curve was found via exhaustive search over all scalar quantization schemes. In addition to presenting this problem in detail and describing these curves,[2] also considers reduced complexity restricted to smaller quantizer search spaces. Finally, the curve “Homo. scalar Q interactive” indicates the rate required if all of the users must send their messages in parallel, then, after all of these messages are received, can send another series in parallel and so on, which is the interactivity model considered in [3]. While this does count as a form of interaction, requiring the users to send their messages in parallel substantially increases the rate required.
Figure 5.4: Rate required to select a user attaining the max with 3 users each observing an independent RV with support \{1, 2, 3, 4\} when users take turns sending messages one at a time [1, 2]. The fundamental limit on the rate given by (5.22) is compared to that obtained by the best scalar quantizer to use in the collocated network interactive scheme (Optimal Scalar. Hete. Q). This is substantially less than the rate required if all three users must send each message in parallel, then hear collectively all of the sent messages, then send each message in parallel again, etc, which is the interactive model employed in [3].

5.3 Incorporating Communication Cost into the Reward Function

The previous discussion has assumed that the reward function is completely given, but in many problems, the cost of communicating over the network may subtract from the reward of the decisions. In this manner, it may be desirable to design the MDP to consider this cost explicitly by incorporating it as a weighted term into the reward function. In particular, suppose that the total number of bits communicated by the partial state sharing scheme in the messages $M_{n,t}$, $n \in N$ when the state vector is $S_t = i$ is denoted by $|q(i)|$, then we can form an augmented reward function

$$R'_a(i,j) = R_a(i,j) - \lambda|q(i)|$$

(5.26)
including the communication cost reflecting the number of bits transmitted in order to enable the system learn the action. The goal then shifts to solving the optimization problem

$$\max_{(c,q) \in F} \sum_{t \in \mathbb{N}} \gamma^t \left( \mathbb{E} \left[ R_{c(S_t)}(S_t, S_{t+1}) - \lambda |q(S_t)| \right] \right)$$

(5.27)

where the constraint set is defined as

$$F := \{ (c, q) | \forall i, i' \in S \text{ such that } q(i) = q(i'), \text{ we have also that } c(i) = c(i'). \}$$

(5.28)

or equivalently, the set of $c, q$ such that $c$ can be rewritten as a function of exclusively $q$.

$$F := \{ (c, q) | q : S \rightarrow B, \exists c' : q(S) \rightarrow A, c(i) = c'(q(i)) \forall i \in S \}. \quad (5.29)$$

Observe that while the observation of the encodings $q(i)$ form effectively an observation for a partial observed Markov decision process (POMDP) [55][91], the requirement we have made that we are able to determine the controller’s (i.e. running the full state knowledge MDP) action decisions from exclusively the observation during the same time step implies a different problem structure from a POMDP as the memory of past observations or action decisions for determining the state distribution must be neglected.

Select some map $\ell : S \rightarrow \{1, \ldots, |S|\}$ and, for any given controller $c : S \rightarrow A$, define the $|S| \times |S|$ transition matrix $P(c)$ whose $\ell(i), \ell(j)$th element is

$$[P(c)]_{\ell(i), \ell(j)} = p_{c(i)}(i, j), \quad \text{and define the row vector } [\pi]_{\ell(i)} = \mathbb{P}[S_0 = i]. \quad (5.30)$$
The objective function in the optimization can then be rewritten as

\[
\sum_{t \in \mathbb{N}} \gamma^t \left( \mathbb{E} \left[ R_c(S_t)(S_t, S_{t+1}) - \lambda |q(S_t)| \right] \right)
\]

\[
= \sum_{t \in \mathbb{N}} \gamma^t \sum_{i,j \in S} \mathbb{P}[S_t = i, S_{t+1} = j] \left( R_c(i, j) - \lambda |q(i)| \right)
\]

\[
= \sum_{t \in \mathbb{N}} \gamma^t \sum_{i,j \in S} \left[ \pi P(c)^t \right]_{\ell(i)} p_c(i, j) \left( R_c(i, j) - \lambda |q(i)| \right)
\]

\[
(5.31)
\]

\[
= \sum_{i,j \in S} \left[ \pi (I - \gamma P(c))^{-1} \right]_{\ell(i)} p_c(i, j) \left( R_c(i, j) - \lambda |q(i)| \right)
\]

The presence of the constraint that \((c, q) \in \mathcal{F}\) makes the joint optimization problem (5.27) a substantially more difficult combinatorial optimization problem than an ordinary MDP. For small problems, the set of control maps \(c : S \rightarrow A\) can be enumerated, and for each such control map, the component

\[
\max_{q | (c, q) \in \mathcal{F}} -\lambda \sum_{i \in S} \left[ \pi (I - \gamma P(c))^{-1} \right]_{\ell(i)} |q(i)|
\]

\[
(5.32)
\]

of the expected reward associated with the minimum control information overhead can be calculated using the results in section 5.2.1 for a non-interactive messaging scheme, while if interactive communications are enabled, then the results in section 5.2.2 can be used. In both cases, the probability distribution for \(S\) is selected as being multiplicatively proportional to \(\pi (I - \gamma P(c))^{-1} \ell(i)\) to ensure that the reward will be maximized by the encoding. After determining in this manner the expected discounted reward \(\max_{q | (c, q) \in \mathcal{F}} \sum_{t \in \mathbb{N}} \gamma^t \left( \mathbb{E} \left[ R_c(S_t)(S_t, S_{t+1}) - \lambda |q(S_t)| \right] \right)\), maximized over all encoding schemes, for each control map \(c\), the control map yielding the expected maximum reward for the particular \(\lambda\) can be selected. Furthermore, a tradeoff between the control overhead and the expected reward can be traced by
Figure 5.5: The control overhead versus the expected throughput tradeoff with $\lambda \in [0, 0.36]$ in the augmented reward function in (5.26)

Example 6 (Overhead Performance Tradeoff for Wireless Resource Allocation, Tiny Model). Consider again the setup of distributed wireless resource allocation described in example 1, with $N = 2$ mobile users each observing their local channel quality $S_{n,t}$ at each time instant $t$, where $S_{n,t}$ is uniformly distributed on the support $\{0, 1\}$ for all $n \in \mathcal{N}, t \in \{1, \ldots\}$. Additionally, let the amount of additional traffic that arrives destined for each user $X_{n,t}$ be uniformly distributed on the support $\mathcal{X} = \{0, 1\}$, and let the buffer size limit at the basestation be $BU_{max} = 2$, with the packet dropping process described as in Example 4. The system is started in the all 0 initial state, in which the buffer is empty and the channel qualities are all 0. Let (5.26) be the transition reward function where the term $R_a(i,j)$ represents the system throughput by choosing action $a$ at state $i$ as in (5.13), and consider the one-shot, non-interactive, control information sharing model in which each node (i.e. each user and the base station) sends a quantized representation of their state to everyone else, enabling them
Figure 5.6: The control overhead versus the expected packet dropping cost tradeoff with $\lambda \in [0, 0.36]$ in the augmented reward function in (5.26).

all to learn the control action directly from these messages. Finding the optimum solution to (5.27) via exhaustive search over all control mappings for different $\lambda$ enables the control overhead versus throughput tradeoff and the control overhead versus the packet dropping cost tradeoff plotted in Figure 5.5 and Figure 5.6 to be traced out. We observe that, at least 2 bits of control overhead are required to guarantee the expected system throughput achieves the limit.

5.4 Finding Candidate Quantizations through Alternating Optimization

However, in many problems, the sort of exhaustive search approach to solving the combinatorial optimization (5.27) just described is nowhere near computationally feasible, as the number of possible control maps to search over are $|A|^{|\mathcal{S}|}$. In this case, an alternating optimization approach yields a lower complexity search method that can be well suited to finding candidate solutions to this optimization problem.

As shown in Figure 5.7, a reasonable goal for such an alternating optimization
method is to alternate between optimizing the control map, then optimizing the quantizer. Let the iteration index in this algorithm be $k$, and the control map and local state encoders at iteration $k$ be denoted by $c_k$ and $q_k$ respectively. At a given iteration in the algorithm, the control map minimizing the augmented value function among all control maps that can be determined from the present encoding $q_k$ could be selected by solving

$$c_{k+1} = \arg \max_{c \in \mathcal{C}(q_k)} \sum_{t \in \mathcal{N}} \gamma^t \left( \mathbb{E} \left[ R_{c(S_t)}(S_t, S_{t+1}) - \lambda |q_k(S_t)| \right] \right)$$

(5.33)

Next, the local encodings $q_{k+1}$ which achieve the minimum expected sum rate while enabling distributed computation of the new control map $c_{k+1}$ are selected

$$q_{k+1} = \arg \max_{q \in \mathcal{Q}(c_{k+1}, q)} \sum_{t \in \mathcal{N}} \gamma^t \left( \mathbb{E} \left[ R_{c_{k+1}(S_t)}(S_t, S_{t+1}) - \lambda |q(S_t)| \right] \right)$$

(5.34)

As this admits a form of an alternating maximization, the sequence of expected
values will be monotone increasing. As this sequence is bounded above via the global optimum, this sequence must converge to a limit. In general this limit may or may not be the global optimum, as all that can be guaranteed is that this limit is associated with a Nash equilibrium. In particular, the limit of this iteration has the property that in no unilateral change individually in the control map $c$ or the quantizer $q$ can yield a higher expected reward, although it may be possible to modify them both together and achieve a higher reward.

To solve (5.34), if the non-interactive communications structure is used, the results in section 5.2.1 and equation (5.16) can be used as a fairly tight and close bound with associated close achievability scheme (within one bit), while if interactive communications are enabled, then the results in section 5.2.2 and equations (5.22,5.24) can be used as a bound.

Solving (5.33) however, can be quite complicated, as the direct search solution to the combinatorial optimization has complexity $|\mathcal{A}|^{|\mathcal{S}|}$. To simplify matters, the control map update (5.33) can itself be attacked with an alternating optimization which has an overall iteration update complexity proportional to $|q(\mathcal{S})||\mathcal{A}|$. In its simplest form, this alternating minimization cycles through the different possible quantizations $m \in q(\mathcal{S})$, updating the associated $c'(m) = c(i)$, $i \in q^{-1}(m)$ in an order determined by a selected bijection $\mu : q(\mathcal{S}) \rightarrow \{0, 1, \ldots, |q(\mathcal{S})| - 1\}$ according to

$$m_{k,\ell} = \mu^{-1}(\ell \mod |q_k(\mathcal{S})|)$$  \hspace{1cm} (5.35)

$$c_{k,\ell} = \arg \max_{c \in \mathcal{C}(m_{k,\ell}, c_{k,\ell-1}, q_k)} \sum_{i \in q_k^{-1}\left(m_{k,\ell}\right)} \sum_{j \in \mathcal{S}} \left\{ [\pi (\mathbb{I} - \gamma P(c))^{-1}]_{\ell(i)} \cdot p_{c(i)}(i,j) \right\} \left( R_{c(i)}(i,j) - \lambda |q(i)| \right)$$  \hspace{1cm} (5.36)
\[ c_{k+1} = \lim_{\ell \to \infty} c_{k,\ell}, \quad c_{k,0} := c_k. \] (5.37)

wherein

\[
F(m_k, c_k, \ell, q_k) := \left\{ c : \mathcal{S} \to \mathcal{A} \middle| \exists c' : q(\mathcal{S}) \to \mathcal{A}, \ c = c' \circ q_k, \ c'(m_k, \ell) \in \mathcal{A}, \ c'(m) = c_{k,\ell-1}(m) \ \forall m \in q_k(\mathcal{S}) \setminus \{m_k, \ell\} \right\}. \] (5.38)

Alternatively, a more greedy form of the alternating optimization can be selected, which replaces (5.35) with

\[
m_k, \ell = \arg \max_{m \in \mathcal{S}} \max_{c \in F(m, c_k, \ell-1, q_k)} \sum_{i \in \mathcal{A}} \sum_{j \in \mathcal{S}} \left\{ \left[ \pi(\mathcal{I} - \gamma \mathcal{P}(c))^{-1} \right]_{\ell(i)} \cdot p(c(i), j) (R(c(i), j) - \lambda |q(i)|) \right\} . \] (5.39)

Putting these pieces together, the overall low complexity alternating optimization algorithm to find candidate solutions to (5.27) consists of (5.35) or (5.39) and (5.36),(5.37),and (5.34). As this algorithm is an alternating optimization, with the individual dimensions in the optimization being \(q\), and each of the \(c'(m), m \in q(\mathcal{S})\), the sequence of expected rewards achieved by each update is monotone increasing. As this sequence of expected rewards is bounded above by the global maximum (5.27), it must converge to a limit, and depending on the initialization \(q_0\), this limit may or may not be the global maximum. When the sequence of control maps and quantizations also converges it must at least be to a Nash equilibrium as summarized in the following theorem.

**Theorem 10.** The iterative method for solving the constrained MDP (5.27) that is described by (5.35) or (5.39) and (5.36),(5.37),and (5.34) yields a monotone increasing sequence of expected rewards which converges. Additionally, when the sequence of control maps and quantizations converges, \((c_*, q_*) = \lim_{k \to \infty} (c_k, q_k)\), the convergent
Figure 5.8: The expected system throughput with respect to the weighting coefficient \( \lambda \) in (5.26)

pair \((c_\ast, q_\ast)\) are a Nash equilibrium [92], in the sense that no unilateral deviation in any of the axes \( q \) or \( c'(m) \) for each \( m \in q(S) \), can yield an increase in the expected reward.

Example 7 (Overhead Performance Tradeoff for Wireless Resource Allocation, Larger Model). We apply the aforementioned alternating optimization algorithm to solve the following example. Consider again the wireless resource allocation setup of examples 1 and 6, but in which \( N(N = 2) \) mobile users observe their local channel quality \( S_{n,t} \) at time-slot \( t \), distributed on the support \( \{0,1,2,3,4\} \) with probabilities \( \{1/8, 2/8, 3/8, 1/8, 1/8\} \) \( \forall n \in \{1,2\}, t \in \{1,\ldots\} \). Additionally, the amount of additional traffic arrivals destined for each user, \( X_{n,t} \) be distributed on the support \( \mathcal{X} = \{0,1,2\} \) with probabilities \( \{1/2, 1/3, 1/6\} \). The basestation observes the buffer state \( S_3 \) which has a buffer limit \( BU_{\text{max}} = 4 \), and let the packet dropping process be the same as the one described in Example 4.

We refer to the algorithm consisting of (5.35),(5.36), (5.37), and (5.34) as the
Figure 5.9: The control overhead per time-slot with respect to the weighting coefficient $\lambda$ in (5.26)

Figure 5.10: The control overhead versus the expected throughput tradeoff computed by both the round-robin and the greedy alternating optimization algorithm with $\lambda \in [0, 4]$ in the augmented reward function in (5.26)
Figure 5.11: The control overhead versus the expected packet dropping cost tradeoff computed by both the round-robin and the greedy alternating optimization algorithm with $\lambda \in [0, 4]$ in the augmented reward function in (5.26)

round-robin alternating optimization algorithm, and the algorithm consisting of (5.39), (5.36), (5.37), and (5.34) as the greedy alternating optimization algorithm. For the quantizers and information sharing strategies to learn the action, we assume that a one-shot, non-interactive scheme must be used. Furthermore, both algorithms are initialized with the trivial quantizers which simply relay the full local state.

Note that, as observed by Thm. 10, while these alternating optimization methods will always yield a sequence of rewards which is monotone non-decreasing and converges, when the associated control map and quantizer converge, they will in general only be to a Nash equilibrium, and possibly not a global optimum. This local convergence was indeed observed in the experiments, as some multipliers $\lambda$ lead quantization and control mappings with negative expected discounted rewards, while when not sending any control information, the expected reward will be lower bounded by 0. Hence, while the quantization and control schemes presented in the remainder of this example are guaranteed to be Nash equilibria, there is a chance that they are
not globally optimal. Nonetheless, they are highly optimized, and, thus, the tradeoffs they yield, by varying $\lambda$ between the expected throughput and the control overhead are quite interesting.

By applying both the round-robin and the greedy algorithm, we find local optimal quantizations and control mappings for different choice of $\lambda$. The expected throughput, and the communication cost are computed as shown in Figure 5.8 and Figure 5.9. Based on the local optimal solutions we found, we also plot the throughput versus control overhead tradeoff and the packet dropping cost versus control overhead tradeoff in Figure 5.10 and Figure 5.11.

We observe from Figure 5.8 that the expected system throughput is maximized when $\lambda = 0$. In such a case, the communication cost is not involved in the transition reward function, the optimal control mapping $c$ becomes to pick the user that maximize the instant throughput, and the optimal encoders $q$ are designed to minimize the control overhead while guarantee that the optimal control mapping can be learned by any node with only observing all of the control messages. We also observe from Figure 5.9 that the control overhead become 0 when $\lambda \geq 3$, this is because, as $\lambda$ grows larger, the system realizes that the cost it pays to encode the local states weights more than the reward it could earn by sending data traffics to the destined user, hence the optimal decision is to not encode any local state and blindly schedule a user to occupy the resource block. Finally, we observe from Figure 5.8 that the expected system throughput per time-slot goes to 0 when the system decision is to blindly pick a user. This is because, although the basestation may pick users and send data traffics in the first few time-slots, however, in a long run, the global state must be absorbed to one of the recurrent classes in which, the instant throughput will always remain 0, as the buffer fills with traffic destined for the other users and
stays full. In fact, given a blind control mapping with no local observation as

$$c(S) = a, \text{ for all } S \in \mathcal{S},$$  \hfill (5.40)

those global states with the same buffer local state $S_{n+1} = (b_1, \ldots, b_N)$ satisfying

$$b_a = 0$$  \hfill (5.41)

and

$$\sum_{n \in \mathcal{N} \setminus a} b_n = B U_{max}$$  \hfill (5.42)

will form a recurrent class. The result that the expected system throughput becomes 0 when the system decision is to blindly pick a user indicates that a system with randomly picking users at all time-slots will perform better than the deterministic control mapping system, which matches the conclusion in [93][91]. It is important to note, however, that this would be precluded by the present model, which would require which user would be transmitted to be known deterministically to a participant just arriving in the network who in this case would not have observed anything since there is no control information being sent. Additional control rate savings and increased rewards enabled by randomization, which will require the assumption of synchronized common randomness at all participants in the scheme, are an important direction for future work.

In summary, this chapter analyzed a Markov decision process in which the state was composed of a series of local states, each observed at a different location in a network. Using recent results from multiterminal information theory regarding distributed and interactive function computation, the minimum amount of control information that would be necessary to exchange in order for the system to simulate
a centralized controller having access to the global state was determined. Next, the information theoretic cost of communication was incorporated into the reward function in the MDP, and the problem of simultaneously designing the controller and the messaging scheme to maximize the associated combined reward was formulated, creating a tradeoff between communication and performance. To provide candidate solutions for the associated optimization problem, an alternating optimization method was presented that produces a sequence of rewards that always converges, and when the associated messaging scheme and controller map converges, it converges to a Nash equilibrium for the problem. A series of running examples from downlink wireless resource allocation illustrated the ideas throughout.
In this dissertation, we have focused on the topic of control overhead minimization in wireless resource allocation problems.

In Chapter 3, we have shown how to efficiently encode the channel quality feedback for the purposes of resource allocation from the view of both information theory and communication by modeling the feedback encoding process as a distributed discrete function computation problem. We assume that the channel qualities are independent and identically distributed from a known distribution on a discrete support set, and the basestation wishes to maximize the system throughput.

Under this model, we have investigated the minimum sum-rate that the users must send. In Section 3.2, an achievable coding scheme has been presented on compressing the control overhead to help the basestation learn which user has the best channel quality (the arg max problem). This achievable scheme is designed by first building characteristic graphs for all candidate arg max functions and finding the chromatic entropy achieving coloring methods for the graphs, and then minimizing over all candidate arg max functions. The candidate arg max functions are these functions who take all users’ channel qualities as input and map to one of the indices with which the user has the maximum channel quality.

Then, we have shown in Section 3.3 that the coding scheme we provide in Section 3.2 achieves the fundamental limit of the sum-rates. This fundamental limit equals to the aggregate graph entropies across users.

Since the basestation can pick any of the users who have the maximum quality, there will be more than one candidate arg max function to minimize the control sum-rate with. However, we have been able to show in Section 3.4 that there exists a class of desired candidate arg max functions so that the minimum control sum-rate can be
computed with any given function in that class. Followed by that, we have derived a close form of the minimum sum-rate with which the users encode their own channel qualities and send to the basestation so that the basestation is able to pick a user maximizing the system throughput.

We have also visited the problems where the base station wants to know the best channel quality (the max problem), or both the best channel quality and the user who achieves the best quality (the max and arg max pair problem). For these two problems, we showed that to achieve the minimum control sum-rate, there is no need for sophisticated coding schemes than each user sending the maximum independent set that contains its channel quality.

The amount of information saved by even the best possible lossless non-interactive scheme is small and does not scale with the number of users, i.e. we have shown in Section 3.4 of Chapter 3 that at most 2 bits can be saved by the one-shot distributed function computation relative to the scheme in which the users simply send their uncoded channel qualities to the basestation. However, when switching to a model in which the resource decisions can be made interactively with the users, we have been able to show as in Chapter 4 that substantial control rate compression is possible.

In Section 4.1, we have provided the Multi-Threshold Interactive Scheme for computing the extremization functions interactively between the basestation and users. In this scheme, the users each initially occupy a fraction of a bandwidth to communicate to the basestation. The basestation knows the user index and the part of the bandwidth that it corresponds to at the beginning. During each round of this scheme, the basestation updates its estimation on the best user channel quality, and broadcasts a message to declare a threshold based on the estimation to the online users, where we say a user remains online if for instance in the arg max case, it is still possible to be the arg max based on the information it has received up until this round.
The online users will reply a binary message to let the basestation know whether or not its channel quality is above the threshold. The interactive communication stops if the basestation can compute the extremization function losslessly.

Aiming at determining the optimal choice of the threshold to minimize the expected value of a given cost metric, i.e. the aggregate communication rate, we in Section 4.2 have formalized the process of the Multi-Threshold Interactive Scheme as a dynamic programming problem.

Two encoder designs for broadcasting the message of the threshold have been discussed in Section 4.3. One is to encode the threshold only conditioning on the knowledge of the user channel quality’s support, the other is to encode the number of online users at every round. These two encoders, relative to the scheme that encode threshold with all historical knowledge, can give an expected minimum aggregate rate very close to the latter one with a lower encoding computation complexity.

We then have shown the simulations results of solving the dynamic programming problem with all the three mentioned encoding schemes in Section 4.4 and Section 4.5. The scaling law has been discussed in Section 4.6, and a comparison with other multi-agents interactive communication schemes has been given in Section 4.7.

Finally in Chapter 5 of the dissertation, motivated by an extension of the wireless resource allocation problem, we have studied a distributed discrete control system modeled a Markov decision process for whom the global system state is itself a vector composed of a series of local states known by a series of network nodes respectively. In this model, an control decision is made with a payment received for each of the global system states which will change the transition probability of the system moving to a new state in the future. Since no single node is given access to the global system state, therefore, to learn the control decision that maximizing expected reward received, it is required to exchange the local state information by communicating among the nodes.
We assume the communication network is collocated, that is, any message sent by any node in the network must be overheard by all the other nodes. We require the communication to guarantee that any node with no access to local state and only observing those exchanged information can still learn the control decision.

Examples of the wireless resource allocation problem have been discussed through this chapter. In these examples, we assume the basestation has a shared buffer containing the data packets that must be sent to the users, and the users observe their individual channel qualities for the current time-slot. The system state is composed of both the buffer status and the local channel qualities which is not fully known by neither the basestation nor any of the users. Each of the nodes broadcasts a message regarding its local state. A control action of scheduling which user for downlink data transmission must be made by purely observing every messages.

Under the model of distributed discrete control systems with collocated communication networks, we have focused on finding the optimal control mapping and the optimal communication scheme to learn this optimal control simultaneously.

With this objective, we in Section 5.2 have studied the minimum amount of information that must be exchanged in the network for a given fixed control mapping, and typically for the control mapping learned by an omniscient who is assumed to observe the global state. We have computed the fundamental limit of minimum amount communication cost under both the one-shot non-interactive communication setup and the interactive collocated communication setup in Section 5.2. By using the results from the theory of distributed function computation, it has been shown that the fundamental limit of the one-shot non-interactive communication cost to learn the control decision equals to the sum of the graph entropies of the characteristic graphs for each user. With the assumption that the local states are independent with each other, quantizers achieving the non-interactive fundamental limit within 1
bit by Huffman code has been given in Section 5.2.1. By using the results from the theory of interactive communication, we have computed the minimum amount of communication to learn control decisions in a collocated interactive manner in Section 5.2.2.

Further, we have incorporated the communication cost into the reward function in Section 5.3. Under the assumption we have made that any node with only observing the current shared information could learn the control action, we have modeled the problem of finding optimal control mapping with a given quantization as the reactive POMDP problem. We have shown a tiny example of solving the joint optimization problem by searching over all possible control mappings for a given quantization and minimizing over all quantizations. In general, such a joint optimization problem is intractable due to the exponentially growing of the size of the searching space.

Finally in Section 5.4, we have proposed an alternating optimization algorithm where we alternatively optimize the control mapping with respect to the quantization and optimize the quantization with respect to the control mapping until a Nash equilibrium is achieved. Resource allocation examples have been discussed where the communication cost versus system throughput tradeoffs have been established. Simulation results confirms that the proposed alternating optimization algorithm yields a lower computation complexity than the original joint optimization problem.

Important directions for future investigation involve allowing time varying messaging and control schemes, the use of historical observations of messages in a POMDP like framework, and the use of rate distortion theory to aid with the derivation of tradeoffs between communication and control reward in the present decentralized MDP context.
Bibliography


Vita

Jie Ren received the B.S. degree in Electrical Engineering from Tsinghua University, China, in 2009, and the M.S. degree in Telecommunication from Widener University, Chester, PA in 2011. Since September 2011 he has been pursuing a Ph.D. at Drexel University, Philadelphia, PA within the Adapative Signal Processing and Information Theory Research Group. He has interned with the Wireless Bigdata Research Team, Huawei(US) in 2015. His publications so far includes, “Overhead Performance Tradeoffs - A Resource Allocation Perspective”, “Interactive Communication for Resource Allocation”, “Interactive Scalar Quantization for Distributed Resource Allocation”, “Computing the Rate Distortion Region for the CEO Problem With Independent Sources”, “Computing the Rate Distortion Region for the CEO Problem With Independent Sources”, “Practical Interactive Scheme for Extremum Computation in Distributed Networks”, “Interactive Quantization for Extremum Computation in Collocated Networks”.
