TIME-CRITICAL DECISIONS WITH REAL-TIME INFORMATION EXTRACTION

Xingran Chen

A DISSERTATION

in

Electrical and Systems Engineering

Presented to the Faculties of the University of Pennsylvania

in

Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

2023

Supervisor of Dissertation

Shirin Saeedi-Bidokhti, Assistant Professor of Electrical and Systems Engineering and Computer and Information Systems, University of Pennsylvania

Graduate Group Chairperson

Troy Olsson, Associate Professor of Electrical and Systems Engineering, University of Pennsylvania

Dissertation Committee

Saswati Sarkar, Professor of Electrical and Systems Engineering, University of Pennsylvania
Hamed Hassani, Assistant Professor of Electrical and Systems Engineering, Computer and Information Systems, and Statistics and Data Science at the Wharton Business School, University of Pennsylvania
Eytan Modiano, Richard C. Maclauring Professor in Aeronautics and Astronautics, Massachusetts Institute of Technology
Aylin Yener, Roy and Lois Chope Chair Professor in Engineering, Professor of Electrical and Computer Engineering, Computer Science and Engineering, and Integrated Systems Engineering, The Ohio State University

To my family.

ACKNOWLEDGEMENT

First and foremost, I would like to thank my advisor, Prof. Shirin Saeedi-Bidokhti, for her support and guidance during my Ph.D. study. Under her training and supervision, I have grown from a newbie to an independent scholar. Now I know how to solve a problem systematically, how to write an academic paper and grants, and how to give a wonderful talk. Needless to say, I will benefit from my experience in Upenn for my entire life.

I would like to thank Prof. Saswati Sarkar, Prof. Hamed Hassani, and Dr. Navid NaderiAlizadeh. You supervised me to do academic projects and give me the chance to touch areas I haven't touched, including Game Theory, Coding Theory, Graph Neural Networks, and Reinforcement Learning. Talking with you has always been inspiring. Through the discussions with you, I have learned how to think about the big picture and find interesting problems, and the ideas you share have improved my academic taste.

I would also thank all my collaborators, Hesam Nikpey, Jungyeol Kim, Mohammad Fereydounian, Konstantinos Gatsis, Renpu Liu, Shaochong Wang, Mohammad Hassan Lotf, and Xinyu Liao. These projects wouldn't be possible without the support from them.

I would like to thank the thesis committee members, your suggestions make this thesis more brilliant. Fortunately, I have explored a lot of meaningful and interesting research directions from your comments for my future career. I want to thank all my friends at Penn for the wonderful time spent together in my Master's and Ph.D. from 2016 - 2023. Last but not least, I want to thank my family for their unconditional love and support throughout my life.

ABSTRACT

TIME-CRITICAL DECISIONS WITH REAL-TIME INFORMATION EXTRACTION

Xingran Chen

Shirin Saeedi-Bidokhti

The Internet of Things and the next-generation networks have led to the generation, dissemination, and transformation of a massive amount of real-time information. The information is often governed by processes that evolve over time and/or space (e.g., on an underlying network). This thesis will develop theoretical foundations and algorithmic designs for time-critical decisions with real-time information extraction in networked systems and consider applications such as estimation and network coding in IoT, and testing and isolation for COVID-19.

In the first part, we study the timeliness of information transfer in communication networks. Timeliness was first captured and quantified in point-to-point channels through the metric of age of information (AoI) and has since become a new design criterion in communications. In this thesis, we go beyond point-to-point channels and consider multiple access networks with multiple senders, broadcast networks with multiple receivers, as well as general ad-hoc networks.

In Chapter 2, we study the problem of age minimization in random access channels. This problem is essential in the remote estimation and control of processes that are observed from decentralized sources in wireless networks. Scheduling policies in multiple access channels is no longer realistic due to a huge amount of communication and coordination among sources. We propose decentralized policies to minimize the time-average AoI and provide performance guarantees for them. In particular, we show that in the regime of low packet arrival rate ($< \frac{1}{eM}$, where M is the number of sources), the standard slotted ALOHA policy is asymptotically age-optimal as the number of sources gets large. However, when the packet arrival rate gets larger, the age performance for slotted ALOHA deteriorates and diverges when the time horizon is large. To overcome the challenge, we propose the notion of age-gain of a packet to quantify how much the packet will reduce the instantaneous age

of information at the receiver side upon successful delivery. We then utilize this notion to propose a transmission policy in which sources act in a decentralized manner based on the age-gain of their available packets. Each source sends its latest packet only if its corresponding age-gain is beyond a certain threshold which could be computed adaptively using the collision feedback or found as a fixed value analytically in advance. In the regime where the packet arrival rate is relatively large ($= \frac{1}{o(M)}$), we achieve the age-performance of $\frac{e}{2}$, which provides a multiplicative gain of at least two compared to the minimum age under slotted ALOHA (minimum over all arrival rates). We conclude that it is beneficial to increase the sampling rate (and hence the arrival rate) and transmit packets selectively based on their age-gain. This is surprising and contrary to common practice where the arrival rate is optimized to attain the minimum AoI.

In Chapter 3, we study the problem of estimation error minimization in random access channels. Real-time sampling and estimation for autoregressive Markov processes is considered. Two general classes of policies are investigated: oblivious policies and non-oblivious policies. In the former class, decision-making is independent of the processes that are monitored, and we prove that minimizing the expected time-average estimation error is equivalent to minimizing the expected time-average AoI. In the latter class, decision-making depends on the physical processes. We first provide a closed-form expression for the estimation error that is a function of the peak age, the transmission delay, a term which we call the silence delay, as well as the process realization. We then approximately propose an optimal thresholding policy with the threshold $\sigma\sqrt{eM}$ (where M is the number of sources and σ is the standard deviation in physical processes) and the resulting normalized estimation error is $\frac{e}{6}\sigma^2$. The proposed non-oblivious thresholding policy provides a 3-fold improvement compared to age-based oblivious policies, and the age-based oblivious policy provides a 2-fold improvement compared to the state-of-the-art. Simulation results verify that the proposed decentralized thresholding policy outperforms all state-of-the-art policies, and the performance is very close to that of centralized greedy policies.

The formulation and approaches outlined in Chapters 2 and 3 are specific to the simple network topology of random multi-access channels. We go beyond random access channels in Chapter 4

and study the problems of age minimization and estimation error minimization in ad-hoc networks. There are M statistically identical sources in an ad-hoc network, where every source transmits packets to connected sources. Collisions happen if two or more sources in proximity simultaneously transmit packets. Here, we seek to design decentralized policies for each source to decide when to sample, whom to communicate with, and what to transmit to minimize the time-average estimation error and/or age. To tackle the multi-dimensionality of the space of decision-making and to capture the complex topologies of wireless networks, we propose a graphical reinforcement learning framework since theoretical methods are no longer tractable. The proposed framework is proven to be permutation equivalent and enjoys desirable transferability property. In particular, the transmission policies trained on small- or moderate-size networks can be executed on large-scale topologies. We also demonstrate, via numerical experiments, that (i) the transmission policies obtained by the proposed framework outperform state-of-the-art baselines, (ii) the trained transmission policies are transferable to larger networks, and their performance gains increase with the number of agents, and (iii) the training procedure can withstand non-stationarity even if we utilize independent learning techniques.

In Chapter 5, we investigate the tradeoffs between timeliness and communication rate in broadcast networks. To shed light on the tradeoffs, we first propose a novel framework of network AoI on the broadcast channels under a transmission mechanism with coding. We then propose two classes of coding policies: coding policies with uncoded caching and coding policies with coded caching. Two general lower bounds and an upper bound are derived on the time-average AoI for any transmission policy. The bounds are functions of generation rates, erasure probabilities, and target rate constraints. Simulation results reveal that (i) coding is beneficial, and the benefits increase with the number of users; (ii) a good approximation of proposed policies is obtained based on the maximum clique size of the information graph; (iii) the tradeoff between rate and age exists, which implies that the system has to sacrifice age to achieve a higher rate.

Going beyond age minimization in wireless networks, in the second part of this thesis, we study the problem of testing and control of spread processes. This problem is another instance of time-

critical decision-making with real-time information extraction. In Chapter 6, the spread of an undesirable contact process, such as an infectious disease (e.g., COVID-19), is contained through testing and isolation of infected nodes. The temporal and spatial evolution of the process (along with containment through isolation) render such detection as fundamentally different from active search detection strategies. Through an active learning approach, we design testing and isolation strategies to contain the spread and minimize the cumulative infections under a given test budget. We prove that the objective can be optimized, with performance guarantees, by greedily selecting the nodes to test. We further design reward-based methodologies that effectively minimize an upper bound on the cumulative infections and are computationally more tractable in large networks. These policies, however, need knowledge about the nodes' infection probabilities which are dynamically changing and have to be learned by sequential testing. We develop a message-passing framework for this purpose and, building on that, show novel tradeoffs between the exploitation of knowledge through reward-based heuristics and the exploration of the unknown through carefully designed probabilistic testing. The tradeoffs are fundamentally distinct from the classical counterparts under active search or multi-armed bandit problems (MABs). We provably show the necessity of exploration in a stylized network and show through simulations that exploration can outperform exploitation in various synthetic and real-data networks depending on the parameters of the network and the spread.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

CHAPTER 1

Introduction

1.1. Backgrounds and Motivations

The next-generation (NextG) wireless communications and the Internet-of-Things (IoT) will immerse wireless communication in systems that interact with the physical world. Examples range from vehicle-to-vehicle communication for autonomous driving, to smart cities, and public health. These applications will be enabled by separate developments in low-latency high-reliability communication, massive connections between low-power devices, and increased data rates. Unprecedented platforms have been provided for the generation, dissemination, and collection of huge real-time information. The information is often governed by processes that evolve over time and/or space (e.g. on an underlying network).

To achieve optimal (long-term) design criteria in time-critical applications, it is vital to make informed decisions that extract real-time information about the system (e.g. its current state). Such real-time information is subsequently used for decision-making. So decisions have a dual role: on one hand, decisions change the state of a system; on the other hand, decisions bring about timely observations of a system, which is useful for future decisions. For example, real-time decisions for minimizing time-average estimation error introduce instantaneous estimation errors, which is helpful for decision-making in the next time slot. So, it is important to propose efficient strategies carrying real-time information extraction and then achieve desired objectives. A few examples of such settings are as follows. In applications of remote sensing, estimation, and control in wireless networks, consider an IoT network with plenty of small sensors, which can collect information about some underlying physical processes to communicate with a control center. How should we devise sampling and communication strategies to estimate the underlying processes in a timely manner? In an epidemic network, an infectious disease (e.g. COVID-19) spread among individuals, and this disease process is formed by individuals' everyday interactions. How to design efficient testing and control strategies to track or contain the disease?

The universal challenge in all these problems is how to sequentially make decisions using real-time (partial) observations of the process as well as the network. Inherent in these problems, the *timeliness* plays a significantly important role, which leads to a contrast between optimal and timely information extraction. Information theory traditionally seeks fundamental tradeoffs and optimal solutions to maximally extract information. Considering asymptotic performances, these solutions have been less successful in challenging settings where data is collected in real-time from information sources and the timeliness of communication/estimation/detection matters. From the perspective of learning, these settings are often studied in reward-based formulations such as Markov decision processes (MDPs) or partially observable MDPs where there is a notion of the instantaneous value of an action referred to as reward or regret (e.g. instantaneous estimation error). Such reward-based approaches have been relatively successful in centralized settings by providing approximate solutions. But in general, especially in settings where decision-making is decentralized or based on partial or noisy observations, solutions are available only for special cases. This thesis has developed new frameworks for (decentralized) sequential decision-making in multi-agent networks.

1.2. Outline of the Thesis

Information freshness is a new system design criterion motivated by this observation that information usually has the highest value when it is fresh. For example, think about system status updates or samples that are taken from a Markov process: Once a new update packet or sample packet is given, all older packets and the information they carry become insignificant. Information freshness is quantified by the metric of the *age of information (AoI)* as introduced in [1, 2] and it captures both how often information is transmitted and how much delay information experiences in a network. To keep the information freshest is an everlasting goal in time-sensitive applications, such as cyber-physical systems, the Internet of Things, smart cities, as well as healthcare systems.

In Chapter 2, we investigate the *information freshness* in random access channels, where M identical source nodes transmit information updates over a shared wireless medium. In every time slot, each node generates a packet by a Bernoulli process, and decides, *only* depending on its local information, whether or not to transmit its packets to the common receiver. If more than 2 nodes transmit packets

in the same time slot, then collisions happen, and no packets can be delivered successfully. The aim is to *minimize the time-average AoI* in the network by proposing *decentralized* transmission policies. The main challenge arises as follows: if nodes transmit packets at a low rate, the information would not be fresh at the receiver, while if nodes transmit packets frequently with a high rate, many fresh packets are transmitted but may collide leading to the staleness of information at the receiver. Thus, we need to select smartly which nodes can transmit packets and design effective decentralized mechanisms. To solve this challenge, we propose age-based policies, with performance guarantees, to minimize AoI and show that contrary to common practice it is beneficial to increase the sampling rate (and hence the arrival rate) and transmit packets selectively based on their age-gain.

Having designed decentralized policies that minimize AoI in Chapter 2, a natural question arises: can we incorporate the AoI into other objective functions and in particular the *estimation error*? We ask if AoI is a good proxy for the estimation error and if age-based policies perform well concerning the estimation error. Chapter 3 establishes this bridge and goes beyond AoI minimization. We consider the problem of real-time sampling and estimation over reliable and unreliable random access channels with M Markov (autoregressive) physical processes (sources). These processes are to be observed, sampled, and communicated wirelessly with a fusion center for *timely estimation*. Considering applications in IoT and CPS, it is not realistic to assume a central scheduler that monitors all the sensors for decision-making; we, therefore, seek to design near-optimal *decentralized* sampling and communication strategies. In other words, nodes decide, in a decentralized manner, when to sample and transmit to best utilize the network, and the controller has a good estimate of the process at each time. We propose two classes of policies: *oblivious policies* in which decision-making is independent of the source realizations, and *non-oblivious policies* in which sources are observed causally for decision-making and establish an analytic framework that proves a 3 fold improvement in performance compared to state of the art.

Chapter 2 and Chapter 3 discuss the simple case of random access channels. In Chapter 4, we go beyond random access channels and seek policies that minimize age and/or estimation error in *ad-hoc networks*. This brings about a new challenge and that is the inherent network topology.

In particular, we extend the minimization of the AoI and/or estimation error problems to general ad-hoc networks, where every source can be either a receiver or sender, and collisions happen when sources in proximity transmit together simultaneously. Each network source has to make decisions in a decentralized manner: *when* to transmit, *who* to communicate with, and *what* to transmit. Here, the analytical solutions are no longer tractable. We hence propose to use a *multi-agent reinforcement learning* framework in which each source is modeled by a *graph neural network*, trained in a centralized or decentralized manner and executed in a decentralized manner to minimize the AoI and/or estimation error. We make this approach practical for large-scale networks by proving the *transferability* property for our proposed solution.

While freshness of information is important in many applications oftentimes, it is not the only design criteria of interest. In particular, it is important to understand the fundamental *tradeoffs* between the AoI and other performance metrics such as communication rate. We investigate this problem in an *erasure broadcast network* with M senders and M receivers. We ask (i) if *coding* is beneficial to the reduction of the AoI; (ii) if there exist tradeoffs between the AoI and communication rate. We propose a novel framework of network AoI for the broadcast channels under transmission protocols with coding. This allows us to utilize the optimality of coding and the efficiency of scheduling.

Finally, we investigate timely inference and detection for *processes* that not only evolve over time but also on an underlying network, such as the spread of an infectious disease in a contact network, the spread of a computer virus on the world wide web, or the spread of misinformation in social networks. How to contain the spread as soon as possible is another instance where timeliness is crucial because infected nodes who are not isolated may infect others. Our goal is to develop sequential and adaptive learning frameworks for deciding which nodes to "test" depending on observations. In Chapter 6, we consider one of the processes that evolve both *temporally and spatially*, COVID-19 infection. It evolves in time (e.g. through different stages of the disease for an infected individual) and over a contact network and its spread can be contained by testing and isolation. It is vital to judiciously decide who should be tested and isolated in the presence of limitations on the number of individuals who can be tested and isolated on a given day, and contain the spread as soon as possible. We

propose a *testing and isolation strategy* that capture *exploitation-exploration tradeoffs* on networks to contain and control the spread of disease, i.e., minimize cumulative infections. And we further show that exploration is necessary.

1.3. Summary of Contributions

Contributions in Chapter 2 In presenting our results below, we assume large symmetric networks in which we have M transmitters and each transmitter has an arrival rate θ. The key ideas are summarized in Table 1.1.

Algorithms or Bounds	Key ideas	Normalized age performance ($M \to \infty$)
Proposition 1	There is always a fresh packet to be transmitted	Lower bound $\frac{1}{2C_{RA}}$; C_{RA} is the capacity of the RA channel
Proposition 2	All packets are delivered upon arrival	Lower bound $\frac{1}{\theta M}$; tight when $\theta < \frac{1}{eM}$
Slotted ALOHA	See details in [152, Chapter 4.2.3]	Normalized age $\frac{1}{\theta M}$ when $\theta < \frac{1}{eM}$
Algorithm 1	Adaptive age-based thinning (ALOHA)	Decreases age and increases throughput simultaneously
Algorithm 2	Stationary age-based thinning (ALOHA)	Normalized age $\frac{e}{2}$ for $\theta = \frac{1}{o(M)}$
Algorithm 3	Stationary age-based thinning (RA with maximum throughput C)	Normalized age $\frac{1}{2C}$ for $\theta = \frac{1}{o(M)}$

Table 1.1: Summary of the proposed algorithms and bounds in Chapter 2.

We first derive two general lower bounds on AoI for any transmission policy by considering two ideal cases: (i) there is always a fresh packet to be transmitted and hence delivered packets are assumed to experience minimum delay; (ii) all packets are delivered instantaneously upon their arrivals with minimum delay, but without experiencing collisions. The former lower bound turns out to be active as the arrival rate (θ) approaches 1, and the latter lower bound becomes active when θ is small, i.e., when the inter-arrival time is the dominant term of the inter-delivery time.

We analyze the well-known slotted ALOHA algorithm. It is known that slotted ALOHA is stable when the sum arrival rate is below the infamous critical point $\frac{1}{e}$. But it becomes unstable when the sum arrival rate is larger than $\frac{1}{e}$. We prove that when the sum arrival rate is below $\frac{1}{e}$, the normalized age performance of a (stabilized) slotted ALOHA algorithm, properly defined later, is approximate $\frac{1}{M\theta}$ in the limit of large M and is optimal. We further show numerically that the normalized age performance is close to that of centralized max-weight policies that schedule based on *age-gain* (which is formally defined in Section 2.4) when the sum arrival rate is less than $\frac{1}{e}$. Simulation results show that as the sum arrival rate increases beyond this critical point, the normalized age of slotted ALOHA explodes.

We then ask if we can reduce age as the sum arrival rate increases beyond the critical point $\frac{1}{e}$. This is an important question that sheds light on whether increasing the sampling rate is useful when communication is over a random access channel. We find an affirmative answer. We propose two age-based thinning algorithms, i.e., Algorithm 1 and Algorithm 2. The core idea for both algorithms is that transmitters selectively disregard packets to mimic an effective (sum) arrival rate equal to $\frac{1}{e}$. In particular, we develop a threshold policy that can be implemented in a decentralized manner at the transmitters and in which packets that offer large age-gains are transmitted, and those that offer small age-gains are disregarded. In Algorithm 1 we propose an adaptive threshold in which the threshold is updated and improved based on the channel feedback. Algorithm 2 proposes a stationary threshold, in which the threshold is predetermined and thus saves computation costs. Using Algorithm 2, i.e., the stationary thinning method, we prove asymptotically ($M \to \infty$) that for any θ that is not too small $\left(\theta = \frac{1}{o(M)}\right)$, the normalized age is approximate $\frac{e}{2}$ and twice better than that the minimum age that (stabilized) slotted ALOHA can attain. Furthermore, numerical results show that as θ approaches 1, the normalized age approaches 1 using Algorithm 1 (the adaptive thinning method) that adaptively optimizes the threshold in each time slot. Interestingly, we observe that the adaptive thinning algorithm attains a smaller age while increasing the throughput beyond what slotted ALOHA can achieve.

Finally, we generalize our stationary thinning mechanism (Algorithm 2) and demonstrate that the idea behind Algorithm 2 is useful for other random access technologies (e.g. CSMA), see Algorithm 3. In particular, we prove that given a technology that can achieve the throughput C (without coding), Algorithm 3 can attain the normalized age of $\frac{1}{2C}$. Numerical results show that it approaches order-optimality in the limit of large M.

Contributions in Chapter 3 In Chapter 3, we study sampling and remote estimation of M independent first-order autoregressive processes over reliable and unreliable *wireless collision channels*. As opposed to all prior works, we seek decentralized solutions in which the decision at each node is based solely on its local observations and channel collision feedback. Our goal is to minimize the estimation error, specifically a normalized metric that we call the normalized average estimation

error (NAEE). This metric looks at the expected time-average estimation error, normalized by the number of source nodes M. We are interested in the asymptotic regime where $M \to \infty$.

In reliable random access channels, three general cases of autoregressive processes are considered, i.e., stationary autoregressive processes (with autoregressive coefficient $0 < \gamma < 1$), random walk processes ($\gamma = 1$), and explosive autoregressive processes ($\gamma > 1$). In different regimes of γ, two general classes of policies are considered, namely oblivious policies and non-oblivious policies. In the former class, decision-making is independent of the processes that are monitored, while in the latter class, decision-making depends on the processes. In the class of oblivious policies, when $\gamma = 1$, we prove that minimizing the expected time-average estimation error, is equivalent to minimizing the age of information. This leads to lower and upper bounds on the minimum achievable estimation error in this class along with efficient oblivious policies that are age-based. In particular, the NAEE under age-based policies is lower bounded by $.88\sigma^2$ and upper bounded by $\frac{e}{2}\sigma^2$. When $\gamma \neq 1$, the oblivious optimal policies essentially minimize the age of information, so the results can be straightforwardly to extend the cases when $\gamma \neq 1$.

We next ask if non-oblivious policies can provide a significant gain by observing the processes as they progress. Since all source nodes are provided with channel collision feedback, they can compute their age functions and reproduce their respective estimated processes (at the destination) in each time slot. Furthermore, using the collision feedback, the nodes can implicitly coordinate for communication. We define the notion of *error process* at each node which is a function of the sample values and age. We then propose a threshold policy, called error-based thinning, in which source nodes become active only when their corresponding error process is beyond a given threshold. Once a node becomes active, it transmits stochastically following a slotted ALOHA policy. In the regime of $\gamma = 1$, to find an optimal threshold and find a closed-form solution for the resulting NAEE, we first provide a closed-form expression for the NAEE that is a function of the peak age, the transmission delay, a term which we call the silence delay, as well as the process realization. We approximately find the NAEE under an optimal threshold policy by considering the underlying autoregressive Markov process as a discretized Wiener process. An optimal threshold is

then shown to be approximately $\sigma\sqrt{eM}$ and the resulting NAEE to be $\frac{e}{6}\sigma^2$. Compared to oblivious policies, the multiplicative gain provided by the thresholding non-oblivious policy equals 3. The approximation error increases linearly as a function of the variance of the innovation process and decreases as M gets large. In the regime of $\gamma \neq 1$, there is no closed-form expression for the NAEE. We propose numerical solutions for an optimal threshold by approximating the expected time period in which nodes are keeping silent. We observe numerically that the multiplicative gain offered by the thresholding non-oblivious policy increases with γ.

Next, we extend our framework to unreliable random access channels. Similar to reliable ones, when $\gamma = 1$, the closed-form of NAEE under oblivious and non-oblivious policies are provided. The multiplicative gain is the same as that in reliable random access channels, which equals 3 and is independent of the channel erasure probability ϵ. Under non-oblivious policies, an optimal threshold is approximately $\sigma\sqrt{eM/(1-\epsilon)}$ and the corresponding NAEE is $\frac{e}{6(1-\epsilon)}\sigma^2$. When $\gamma \neq 1$, the numerical approximations of the optimal threshold for reliable random access channels can be extended to unreliable random access channels straightforwardly.

Simulation results show that the proposed decentralized threshold policy outperforms oblivious policies. Moreover, oblivious policies are shown to outperform state-of-the-art policies (both oblivious and non-oblivious) that impose a fixed rate (without using the collision feedback) on reliable and unreliable random access channels in all regions of γ. When $\gamma = 1$, it is numerically shown that the performance of the optimal threshold policy is very close to that of centralized greedy policies that schedule transmissions according to the instantaneous error reduction or age reduction. When $0 < \gamma < 1$, the performance of the optimal threshold policy is better than the centralized greedy policies that schedule transmissions according to the instantaneous age reduction, which implies that age is not a good representative in this case.

Contributions in Chapter 4 In this chapter, we study sampling and remote estimation of M independent random walk processes over wireless collision channels in *ad-hoc networks*. Every source can decide, based on its own information: (i) *when to sample*, i.e., if the source transmits a packet in a time slot; (ii) *who to communicate with*, i.e., which neighbor is the receiver of a

transmitted packet; and (iii) *what to transmit*, i.e., which packet is chosen and transmitted to the receiver. Our goal is to minimize the time-average estimation error and/or time-average age of information (AoI) by proposing decentralized sampling and transmission policies. Two classes of policies are considered: oblivious policies and non-oblivious policies [34]. We first show that in oblivious policies, the minimization of estimation error is equivalent to the minimization of the AoI. Then, we unify the error minimization and age minimization problems and propose a general solution framework.

As compared to decision making in random access channels [34, 37], the dimension of the decision space in our considered problem is significantly higher. In addition, complex network topologies need to be incorporated into the decision-making process. Thus, theoretical frameworks are no longer tractable, and we use techniques from MARL to address this problem. We propose a graphical MARL framework, which is an extension and modification of the classical actor-critic framework and has three main changes: (i) the critic is a GNN rather than a fully-connected neural network, (ii) the actor is a GRNN rather than a fully-connected neural network, (iii) an action distribution operator is added, which receives output node embeddings of the actor GRNN as input, and generates action distributions for all aforementioned decisions at its output. The numbers of parameters in the actor and critic models are independent of the number of agents in the network. Moreover, our framework is agnostic to the permutations of agent indices, given the fact that GNNs, GRNNs, and our proposed action distribution operator are permutation-equivariant. We further show that the actor (the GRNN) built on graph filters and the actor distribution operator exhibits the transferability property, implying that the near-optimal solutions trained on small or moderate networks can be applied to large-scale networks, while maintaining near-optimality.

In our implementations, we utilize two widely-used RL training methods: (i) independent learning (IL) (e.g., IPPO, IA2C), where every agent has a distinct actor and distinct critic to train in a decentralized manner, and (ii) centralized training and decentralized execution (CTDE) (e.g., MAPPO, MAA2C), where a centralized critic learns a joint state value function. Then, we have graphical RL frameworks with IL and CTDE. Numerical experiments reveal interesting observations: (i) Our

proposed policies outperform the state-of-the-art. The graphical RL framework outperforms the classical RL framework. And the graphical RL with CTDE outperforms the graphical RL with IL because the former can make the most of global, network-wide information during training. (ii) As we expected, classical RL with IL suffers from non-stationarity, and the performance deteriorates as training proceeds. However, opposite to the classical RL with IL, the performance of the graphical RL with IL is convergent. This convergence implies that the graphical RL with IL can help ameliorate non-stationarity. (iii) Transferability is verified by enhanced performance gains in large-scale networks during execution, i.e., the benefits of the proposed graphical RL framework compared to the baselines increase with the number of agents (or, equivalently, the network size).

Contributions in Chapter 5 In this chapter, we consider erasure networks and devise broadcast strategies that are efficient both in AoI and rate. The inherent tradeoff can be explained as follows. On the one hand, a higher rate can correspond to a smaller delay/AoI (both in the sense that queues get emptied faster and that fewer uses of the network are needed in total to transmit a fixed number of information bits). On the other hand, to achieve high rates, we need to wait for the arrival of other packets and change transmission priorities to facilitate coding, and this can lead to a larger AoI. To shed light on the above tradeoff, we consider an erasure wireless network with M users. Motivated by the success of age-based scheduling in wireless networks, we propose a scheduling framework where we schedule various useful coding actions as opposed to the users. Within this framework, we can capture both rate efficiency and age efficiency. Coding is known to provide significant benefits compared to time sharing especially as the number of users M increases [153, 154, 155, 156]. Our work shows, for the first time, that coding also provides benefits in terms of age and the gain increases by M sharply, especially when the generation rate is small and/or the channel erasure probability is large. More generally, we design deterministic coding policies that minimize the average AoI under given rate constraints.

Specifically, we first propose a novel framework of network AoI on the broadcast channels under transmission mechanism with coding (Section 5.2). Then, (near-)optimal coding policies with uncoded and coded caching are proposed (Section 5.3 and Section 5.4). Two general lower bounds

and an upper bound are derived on EAoI for any transmission policy (Section 5.5, Theorem 17 and Theorem 19). The bounds are functions of generation rates, erasure probabilities, and target rate constraints. Finally, simulation results reveal that (a) coding is beneficial, and the benefits increase with the number of users; (b) a good approximation of proposed policies is obtained based on the maximum clique size of the information graph; (c) the tradeoff between rate and AoI exists, which implies that the system has to sacrifice AoI to achieve a higher rate.

Contributions in Chapter 6 In this chapter, we study a spreading process such as Covid-19 and design sequential testing and isolation policies to contain the spread. Our contributions are as follows.

Formulating the spread process through a compartmental model and a given contact network, we show that the problem of minimizing the total cumulative infections under a given test budget reduces to minimizing a supermodular function expressed in terms of nodes' probabilities of infection and it thus admits a *near-optimal greedy* policy. We further design reward-based algorithms that minimize an upper bound on the cumulative infections and are computationally more tractable in large networks.

The greedy policy and its reward-based derivatives are applicable if nodes' probabilities of infection were known. However, since the set of infected nodes is unknown, these probabilities are unknown and can only be learned through *sequential testing*. We provide a message-passing framework for sequential estimation of nodes' *posterior* probabilities of infection given the history of test observations.

We argue that testing has a dual role: (i) discovering and isolating the infected nodes to contain the spread, and (ii) providing more accurate estimates for nodes' infection probabilities which are used for decision-making. In this sense, *exploitation* policies in which decision-making only targets (i) can be suboptimal. We prove in a stylized network that when the belief about the probabilities is wrong, exploitation can be arbitrarily bad, while a policy that combines exploitation with random testing can contain the spread. This points to novel *exploitation-exploration tradeoffs* that stem

from the lack of knowledge about the location of infected nodes, rather than the network or spread process.

Following these findings, we propose *exploration* policies that test each node probabilistically according to its reward. The core idea is to balance the exploitation of knowledge (about the nodes' infection probabilities and the resulting rewards) and exploration of the unknown (to get more accurate estimates of the infection probabilities). Through simulations, we compare the performance of exploration and exploitation policies in several synthetic and real-data networks. In particular, we investigate the role of three parameters on when exploration outperforms exploitation: (i) the *unregulated delay*, i.e., the period when the disease spreads without intervention; (ii) the *global clustering coefficient* of the network, and (iii) the average *shortest path length* of the network. We show that when the above parameters increase, exploration becomes more beneficial as it provides better estimates of the nodes' probabilities of infection.

CHAPTER 2

Information Freshness in Random Access Channels

In this chapter, we investigate the information freshness in random access channels, where M identical source nodes transmit information updates over a shared wireless medium. The aim is to minimize the time-average AoI in the network by proposing *decentralized* transmission policies. The main challenge arises as follows: if nodes transmit packets at a low rate, the information would not be fresh at the receiver, while if nodes transmit packets frequently with a high rate, many fresh packets are transmitted but may collide leading to the staleness of information at the receiver. Thus, we need to select smartly which nodes can transmit packets and design effective decentralized mechanisms. The major part of this chapter deals with random access technologies such as slotted ALOHA that do not assume carrier sensing capabilities. The underlying reason is threefold: (i) Status packets are generally very short (as opposed to traditional settings such as streaming where packets are long) and so CSMA is not efficient; (ii) Transmitters have low power capabilities. As such, it is not very efficient (in terms of energy and cost) to perform carrier sensing when the arrival rate is large and CSMA is not particularly useful when the arrival rate is small. More importantly, since transmission power is low, the hidden node problem will be a major issue under CSMA-type protocols; (iii) Our analytical results are clearer without the additional complexity of CSMA. In Section 2.5.5, we describe how our findings generalize and apply to CSMA.

2.1. Literature Review

Age of information (AoI), introduced in [1, 2], measures the freshness of information at the receiver side. AoI is a function of both how often packets are transmitted and how much delay packets experience in the communication network. When the rate of communication is low, the receiver's AoI will increase (implying that the receiver's information is stale) because the transmitter is not sending packets frequently enough. But even when the transmitter is sending packets frequently, if the system design imposes a large delay for the packets, the information at the receiver will still be stale. The metric of AoI is of great importance in the Internet of Things applications where

timeliness of information is crucial (e.g. in monitoring the status of a system). Assuming a first come first serve (FCFS) policy, the works in [3, 4] show in queue theoretic setups that AoI is minimized at an optimal update rate. Relaxing the restriction of FCFS policies, [4, 5] propose packet management policies that discard old packets and improve AoI in wide regimes of operation. This already points to the fact that under the metric of AoI, rate, and reliability have little relevance in the design of timely communication schemes. This is because AoI implicitly assumes that the information content of the packets forms a Markov process and hence fresh packets render older packets obsolete. In the past few years, various extensions and new dimensions have also been studied in the paradigm of timely communication: source and channel coding were studied in [6, 7, 8, 9], multi-hop networks were studied in [10, 11, 12], and scheduling algorithms were studied in [13, 14, 15, 16, 17, 18, 19].

Prior work such as [13, 14, 18, 20] consider scheduling policies in multiple access channels that are controlled in a centralized manner. However, in decentralized (random access) applications, employing such policies would require a huge amount of communication and coordination rendering them inapplicable. Towards designing *decentralized* algorithms for minimizing the age of information, [21, 22] analyze stationary randomized policies under the assumption that sources generate packets in every time slot (i.e., all sources are active at all times). Considering the more realistic scenario where packets are generated at random times, [23] analyzes round-robin scheduling techniques with and without packet management and also presents partial results for stationary randomized policies. Round-robin policies are proved to be age-optimal in [24] when the number of transmitters is large and the arrival rate is constant. The follow-up work [25] additionally assumes that nodes are provided with carrier sensing capabilities and proposes distributed schemes that have good performance in simulations; Nevertheless, [25] does not address how the parameters of the proposed algorithms should be designed theoretically. The concurrent work [26, 27] (published after our work [28]) investigate variants of decentralized age-based schemes for CSMA under energy constraints. In an unslotted, uncoordinated, unreliable multiple access collision channel, [29] provides the exact system age and an accurate individual age approximation for a small number of sources. The work [30] which was done independently and concurrently with this work considers a threshold-based lazy version of Slotted ALOHA where each transmitter attempts to access the channel with a cer-

tain probability when its corresponding age exceeds a certain threshold. Optimizing the threshold and the transmission probabilities are non-trivial and the authors provide analysis only for $M = 2$ transmitters for the special case where the arrival rate is equal to 1.

2.2. System Model

We consider a wireless architecture where a controller monitors the status of M identical source nodes over a shared wireless medium. To provide analytical frameworks and closed-form solutions, we focus on symmetric systems (instead of asymmetric ones) and use the profile of all sources as an estimate of an individual source and look at the limit behaviour. Let time be slotted. At the beginning of every slot k, $k = 1, 2, \ldots$, the source node i, $i = 1, \ldots, M$, generates a new packet encoding information about its current status with probability θ and this packet becomes available at the transmitter immediately. We denote this generation/arrival process at the transmitter by $A_i(k)$, where $A_i(k) = 1$ indicates that a new packet is generated at time slot k and $A_i(k) = 0$ corresponds to the event where there is no new update. New packets are assumed to replace undelivered older packets at the source (i.e., older packets are discarded), relying on the fact that the underlying processes that are monitored in physical systems are oftentimes Markovian[1].

The communication media is modeled by a collision channel: If two or more source nodes transmit at the beginning of the same slot, then the packets interfere with each other (collide) and do not get delivered at the receiver. We use the binary variable $d_i(k)$ to indicate whether a packet is transmitted from source i and received at the destination in time slot k. Specifically, $d_i(k) = 0$ if source i does not transmit at the beginning of time slot k or if collisions occur; $d_i(k) = 1$ otherwise.

We assume a delay of one-time units in the delivery of packets, meaning that packets are transmitted at the beginning of time slots and, if there is no collision, they are delivered at the end of the same time slot. We assume that all transmitters are provided with channel collision feedback at the end of each time slot. Specifically, at the end of time slot k, $c(k) = 1$ if collisions happened and $c(k) = 0$ otherwise. In the event that collision occurs, the involved transmitters can keep the

[1]We show in Appendix A.1 that this assumption can be made without loss of generality when the performance measure is Age of Information.

undelivered packets and retransmit them according to their transmission policy (until the packets are successfully delivered or replaced by new packets).

Our objective is to design *decentralized* transmission mechanisms to minimize the time-average age of information per source node. A decentralized transmission policy is one in which the decision of transmitter i at time k is dependent only on its history of actions, the packets arrived so far, $\{A_i(j)\}_{j=1}^{k}$, as well as the collision feedback received so far, $\{c(j)\}_{j=1}^{k-1}$.

The measure of performance in this work is Age of Information (AoI). Originally defined in [1, 2], AoI captures the timeliness of information at the receiver side. We extend the definition a bit further, formally defined below, to also account for the age of information at the source side. Aging at the source/transmitter is caused by the *stochastic nature of arrivals*.

Definition 1. *Consider a source-destination pair. Let* $\{k_\ell\}_{\ell\geq 1}$ *be the sequence of generation times of packets and* $\{k'_\ell\}_{\ell\geq 1}$ *be the sequence of times at which those packets are received at the destination. At any time* τ*, denote the index of the last generated packet by* $n_s(\tau) = \max\{\ell | k_\ell \leq \tau\}$ *and the index of the last received packet by* $n_d(\tau) = \max\{\ell | k'_\ell \leq \tau\}$*. The* source's age of information *is defined by* $w(k) = k - k_{n_s(k)}$ *and the* destination's age of information *is defined by* $h(k) = k - k_{n_d(k)}$.

It is clear from the above definition that once there is a new packet available at the transmitter, the older packet(s) cannot contribute to reducing the age of the system. We hence assume without loss of generality that buffers at transmitters are of size 1 and new packets replace old packets upon arrival. We formalize and prove this claim in Appendix A.1.

Following Definition 1, let $h_i(k)$ denote the destination's AoI at time slot k with respect to source i. The age $h_i(k)$ increases linearly as a function of k when there is no packet delivery from source i and it drops with every delivery to a value that represents how old the received packet is; within our framework, this would be the corresponding source's AoI (in previous time slot) plus 1. Without

loss of generality, we assume $w_i(1) = 0$ and $h_i(1) \geq 0$, and write the recursion of AoI as follows:

$$h_i(k) = \begin{cases} w_i(k-1) + 1 & d_i(k-1) = 1 \\ h_i(k-1) + 1 & d_i(k-1) = 0 \end{cases} \tag{2.1}$$

and

$$w_i(k) = \begin{cases} 0 & A_i(k) = 1 \\ w_i(k-1) + 1 & A_i(k) = 0. \end{cases} \tag{2.2}$$

Note that at the beginning of each time slot k, given the collision feedback $\{c(j)\}_{j \leq k-1}$ and local information about $\{A_i(j)\}_{j \leq k}$, transmitter i can compute its corresponding source's AoI $\{w_i(j)\}_{j \leq k}$ and destination's AoI $\{h_i(j)\}_{j \leq k}$.

We define the Normalized Average AoI (NAAoI) as our performance metric of choice[2]:

$$J^{\pi}(M) = \lim_{K \to \infty} \mathbb{E}[J_K^{\pi}], \; J_K^{\pi} = \frac{1}{M^2 K} \sum_{i=1}^{M} \sum_{k=1}^{K} h_i^{\pi}(k) \tag{2.3}$$

where π refers to the underlying transmission policy.

We consider *centralized policies* and *decentralized age-based policies* in this work. Centralized policies serve as benchmarks. They need a central scheduler who receives information about all arrival processes and previous transmission actions, and coordinate all the transmitters. When the number of transmitters M gets large, facilitating such scales of coordination is not feasible and we are hence interested in decentralized mechanisms. Randomized policies are easy to implement in a decentralized manner. Previous works [21, 22] fall into this class but they have the weakness of not utilizing local collision feedback at the transmitters. Utilizing the collision feedback, we aim to make age-based decisions at the transmitters in a decentralized manner.

[2]For any distributed transmission scheme, it is clear that the average AoI increases with the number of source node M for any fixed arrival rate θ. Note that our problem setup allows M to become very large, so to offset the effect introduced by the number of source nodes, we consider the proposed NAAoI.

2.2.1. Notation

We use the notations $\mathbb{E}[\cdot]$ and $\Pr(\cdot)$ for expectation and probability, respectively. We denote scalars with lowercase letters, e.g. s; vectors with underlined lowercase letters, e.g. $\underline{s}$, and matrices with boldface capital letters, e.g. $\mathbf{S}$. Notation $[\underline{s}]_i$ represents the i^{th} element of $\underline{s}$ and $[\mathbf{S}]_{ij}$ denotes the element in the i^{th} row and j^{th} column. Random variables are denoted by capital letters, e.g. S. We use M to denote the number of transmitters, K to denote the time horizon, and C to denote the capacity of a channel (under a given technology). The operator $(s)^+$ returns 0 if $s < 0$ and it returns s if $s \geq 0$. $\lfloor s \rfloor$ represents the largest integer j such that $j \leq s$. $O(\cdot)$ and $o(\cdot)$ represent the Big O and little o notations according to Bachmann-Landau notation, respectively. We summarize the notations in Table 2.1.

M	The number of sources
K	The time horizon
θ	The generation/arrival rate of new packets
$A_i(k)$	The indicator of the generation/arrival process
$d_i(k)$	The indicator of delivery at source i
$\lambda_i(k)$	The indicator of transmission at source i
$c(k)$	The indicator of collision in the channel
$h_i(k)$	The destination's AoI at time k w.r.t source i
$w_i(k)$	The source's AoI at time k w.r.t source i
π	A specific transmission and sampling policy
$J^{\pi}(M)$	Normalized Average AoI with M sources
C_{RA}	The sum-capacity of the random access channel
$\delta_i(k)$	The age-gain in time slot k at source i
$\{\ell_m(k)\}_m$	The distribution of age-gain in time slot k
$\mathtt{T}(k)$	The threshold under the AAT policy at time k
$\mathtt{T}^*$	The threshold under the SAT policy

Table 2.1: Useful notations in Chapter 2.

2.3. Lower Bound

We start by deriving two lower bounds on the achievable age performance. The first lower bound is derived by assuming that there is always a fresh packet to be transmitted (and hence delivered packets are assumed to experience unit-time delays). The second lower bound is derived by assuming that all packets are delivered instantaneously upon their arrivals (with unit-time delays, but without

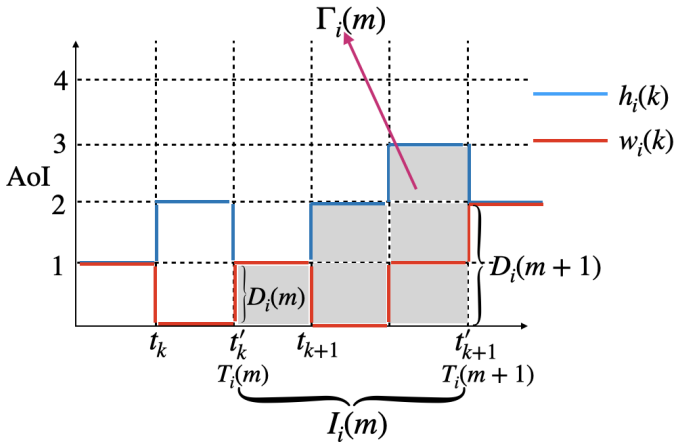


Figure 2.1: An example of $D_i(m)$, $I_i(m)$, and $\Gamma_i(m)$.

experiencing collisions). The former is active as θ approaches 1 and the latter is active when θ is small (when the inter-arrival time is the dominant term of the inter-delivery time).

Fix a large time horizon K and look at the packets of source i. Let $N_i(K)$ denote the number of delivered packets (from source i) up to and including time slot K. Now consider the m^{th} and $(m+1)^{th}$ *deliveries* at the receiver and denote the delivery time of them at the receiver by $T_i(m)$ and $T_i(m+1)$, respectively. The inter-delivery time

$$I_i(m) = T_i(m+1) - T_i(m)$$

is the time between these two consecutive deliveries. Upon the arrival of the m^{th} delivered packet at the receiver, the age of information at the receiver drops to the value $D_i(m)$ which represents how much delay the packet has experienced in the system. Fig. 2.1 illustrates the introduced notation. Now define $\Gamma_i(m)$ as the sum of age functions $h_i(k)$, where k is in the interval $[T_i(m), T_i(m+1))$:

$$\begin{aligned}\Gamma_i(m) = &\sum_{k=T_i(m)}^{T_i(m)+I_i(m)-1} h_i(k) = \frac{1}{2}\big(D_i(m) + I_i(m) + D_i(m)\big)\cdot I_i(m) - \frac{I_i(m)}{2} \\ =&\frac{1}{2}I_i^2(m) - \frac{1}{2}I_i(m) + D_i(m)I_i(m).\end{aligned} \tag{2.4}$$

It follows that in the limit of large K, we have

$$J^{\pi}(M) = \lim_{K\to\infty} \mathbb{E}[J_K^{\pi}] = \lim_{K\to\infty} \mathbb{E}\left[\frac{1}{M^2}\sum_{i=1}^{M}\frac{1}{K}\sum_{m=1}^{N_i(K)}\Gamma_i(m)\right].$$

Using this formulation, we next lower bound NAAoI. Let C_{RA} denote the sum capacity of the underlying random access channel. Note that in the limit of large K, $\frac{N_i(K)}{K}$ is the throughput of transmitter i and

$$\lim_{K\to\infty}\sum_{i=1}^{M}\frac{N_i(K)}{K} \le C_{RA}. \tag{2.5}$$

Then, we have the following propositions.

Proposition 1. *For any transmission policy π,*

$$J^{\pi}(M) \ge \frac{1}{2C_{RA}} + \frac{1}{2M}.$$

Proof. The proof is given in Appendix A.2. □

Proposition 2. *For any transmission policy π,*

$$J^{\pi}(M) \ge \frac{1}{M\theta}. \tag{2.6}$$

Proof. The proof is given in Appendix A.3. □

Let us give an example of how Proposition 1 can be utilized. Note that C_{RA} is not known in general. Nevertheless, any upper bound on C_{RA} gives a lower bound on the normalized age. Based on [157], the capacity of the random access channel with collision feedback, in the limit of large M, is upper bounded by $\lim_{M\to\infty} C_{RA} \le 0.568$ and hence

$$\lim_{M\to\infty} J^{\pi}(M) \ge .88. \tag{2.7}$$

Remark 1. *The lower bound in* (2.7) *does not assume CSMA capabilities. For CSMA, we have*

$C_{CSMA} \leq 1$ *and hence*

$$J^{\pi}(M) \geq \frac{1}{2} + \frac{1}{2M}. \tag{2.8}$$

We show the asymptotic optimality of this bound in Section 2.5.5 as $M \to \infty$.

2.4. Centralized Scheduling

The first class of schemes that we consider are centralized schemes that avoid collision by scheduling transmitters one by one. In particular, Max-Weight policies are shown to perform close to optimal in various works such as [13, 14, 20]. Although such schemes are not practical (due to the scale of required coordination), it turns out that they provide useful intuitions and they also serve as a benchmark for comparison in Section 2.5. We assume a central scheduler that can observe all arrival processes and coordinate/control all senders' actions in order to avoid collisions.

Denote by $\lambda_i(k) = 1$ the event that transmitter i sends a packet and recall that $d_i(k)$ indicates delivery of packets. Note that if $\lambda_j(k) = 1$ for another source $j \neq i$, then the packets collide and no packets will be delivered. One can thus write

$$d_i(k) = \lambda_i(k) \prod_{j \neq i} \big(1 - \lambda_j(k)\big). \tag{2.9}$$

The goal of a central scheduler is to select one source for transmission at each time. Denote $\underline{h}(k) = (h_1(k), h_2(k), \cdots, h_M(k))$. Following the works in [13, 14, 20], an age-based max-weight policy can be designed by considering the following Lyapunov function:

$$\mathcal{L}(\underline{h}(k)) = \sum_{i=1}^{M} h_i(k) \tag{2.10}$$

and minimizing its corresponding one-step Lyapunov Drift:

$$\Delta(\underline{h}(k)) = \mathcal{L}(\underline{h}(k+1)) - \mathcal{L}(\underline{h}(k)). \tag{2.11}$$

It turns out that the max-weight policy selects, in each time slot k, the transmitter that offers the highest *age-gain* $\delta_i(k)$, defined below:

$$\delta_i(k) := h_i(k) - w_i(k). \tag{2.12}$$

$\delta_i(k)$ quantifies how much the instantaneous receiver's age of information reduces upon successful delivery from transmitter i. Proposition 3 states the above max-weight policy more formally (see Appendix A.4 for the proof).

Proposition 3. *For every time slot k, define*

$$\ell(k) = \arg\max_i \delta_i(k). \tag{2.13}$$

An optimal policy to minimize the one-step drift in (2.11) *is to choose $\lambda_{\ell(k)}(k) = 1$ and $\lambda_j(k) = 0$ for all $j \neq \ell(k)$.*

Remark 2. *We will show in Section 2.5 how the notion of age-gain plays a central role also in the design of distributed age-based policies.*

2.5. Decentralized Age-Based Policies

In this section, we propose a new class of decentralized policies designed to *prioritize* transmissions for the purpose of minimizing the age of information. In each time slot k, transmitter i decides whether or not to send its packet depending on its local AoI, and in particular, based on $\delta_i(k)$ $\big($defined in (2.12)$\big)$.

To develop a deeper understanding of our proposed algorithm, let us focus on two regimes of operation assuming large M:

- The regime of infrequent arrivals, where $\theta \leq \frac{1}{eM}$,
- The regime of frequent arrivals, where $\theta > \frac{1}{eM}$.

The choice of these two regimes is made based on the well-established performance of slotted ALOHA

with respect to rate (throughput) [152, Chapter 4]. We will first develop our framework for the slotted-ALOHA random access technology and then generalize it to other random access technologies in Section 2.5.5.

The basic idea of slotted ALOHA is as follows: At every time slot k, transmitters send their packets immediately upon arrival unless they are "backlogged" after a collision in which case they transmit with a backoff probability. In this section, we focus on Rivest's stabilized slotted ALOHA [152, Chapter 4]. In this algorithm, all arrivals are regarded as backlogged nodes that transmit with the backoff probability $p_b(k)$. Let $c(k) = 1$ denote the event that collision occurred at time k and $c(k) = 0$ denote the complementary event. The backoff probability is calculated through a pseudo-Bayesian algorithm based on an *estimate* of the number of backlogged nodes $n(k)$ (see [152, Chapter 4.2.3]):

$$\begin{aligned} p_b(k) &= \min\left(1, \frac{1}{n(k)}\right) \\ n(k) &= \begin{cases} n(k-1) + M\theta + (e-2)^{-1} & \text{if } c(k) = 1 \\ \max\left(M\theta, n(k-1) + M\theta - 1\right) & \text{if } c(k) = 0. \end{cases} \end{aligned} \tag{2.14}$$

It is well known that this algorithm attains stability of queues for $\theta < \frac{1}{eM}$. In other words, transmitters can reliably send packets with a sum rate up to $\frac{1}{e}$ in a decentralized manner [152, Chapter 4.2.3]. Asymptotically, when $M \to \infty$, the probability of delivering a packet in each time slot is $1/e$, the probability of collisions is $1 - 2/e$, and the probability of having an idle channel is $1/e$ (see Appendix A.5). Note that when $M\theta < \frac{1}{e}$, the expected total number of delivered packets in every time slot is $M\theta$.

We find the asymptotic NAAoI (in the limit of large M) in Theorem 1 below.

Theorem 1. *Suppose* $\theta < \frac{1}{eM}$ *and define*

$$\eta = \lim_{M\to\infty} M\theta.$$

Any stabilized slotted ALOHA scheme achieves

$$\lim_{M\to\infty} J^{SA}(M) = \frac{1}{\eta}.$$

Moreover, (stabilized) slotted ALOHA is order optimal in terms of NAAoI.

Proof. The proof is presented in Appendix A.6. The idea is to divide the sources into two groups in every time slot k: sources with $\delta_i(k) = 0$ and sources with $\delta_i(k) > 0$. We show that (i) the contribution of the first group of sources to NAAoI is equal to $\frac{1}{M\theta}$, and (ii) the second group constitutes only a vanishing fraction of the nodes and therefore, even though the sources in this group have larger $\delta_i(k)$'s, their total contribution vanishes as $M \to \infty$. □

2.5.1. Age-Based Thinning

When the arrival rate θ of each transmitter approaches $\frac{1}{eM}$, the NAAoI of slotted ALOHA approaches e (see Theorem 1). As θ increases beyond $\frac{1}{eM}$, the arrival rate gets larger than the maximum channel throughput ($= e^{-1}$), $n(k)$ overestimates the number of active transmitters and $p_b(k)$ underestimates the optimal probability of transmission, causing the throughput to decrease and the NAAoI to sharply increase.

Noting that the maximum channel rate/throughput is $\frac{1}{e}$ when (stabilized) slotted ALOHA algorithms are applied, a natural question arises: What should the transmitters do in order to ensure a small age of information at the destination when $\theta \geq \frac{1}{eM}$? A naive solution to the above question would be to have each transmitter randomly drop packets and perform at the effective rate $\frac{1}{eM}$. But Theorem 1 shows that this only leads to NAAoI $\approx e$ which implies that we will not be able to benefit from the frequency of fresh packets to reduce age.

To benefit from the availability of fresh packets, we devise a decentralized age-based transmission policy in which transmitters prioritize packets that have larger age-gains. In particular, in each time slot k, transmitters find a common threshold $\mathtt{T}(k)$ in order to distinguish and keep packets that offer high age-gains. The core idea is to still fully use the channel (depending on the available

technology) but to carefully select, in a decentralized manner, what packets to send to minimize age. Recall that $\delta_i(k)$ denotes the age-gain of scheduling transmitter i. In our proposed algorithms, transmitters that have large age-gains become active and those with small age-gains stay inactive. More formally, Transmitter i is called *active* in time slot k if $\delta_i(k) \geq \mathtt{T}(k)$. Only active transmitters participate in the transmission policy. Alternatively viewed, at time k, we propose to discard a fresh packet at transmitter i if $0 \leq \delta_i(k) < \mathtt{T}(k)$ and to keep it otherwise. We refer to this process as thinning and this is done locally at the transmitters based on the AoI at the source/destination.

Note that no matter how the transmission policy is designed, since it is decentralized, multiple transmitters may try to access the channel at the same time, leading to collisions. For simplicity and clarity of ideas, we will restrict attention to slotted ALOHA techniques to resolve such collisions, and in particular, the Rivest's stabilized slotted ALOHA[3] described in (2.14).

The main underlying challenge is in the design of $\mathtt{T}(k)$. We propose two algorithms: an adaptive method of calculating $\mathtt{T}(k)$ for each time slot based on the local collision feedback and a fixed threshold value $\mathtt{T}^*$ that is found in advance and remains fixed for all time slots k.

In the remainder of this section, we assume that M is large, and $\theta > \frac{1}{eM}$. The following definition comes in handy in presenting our results.

Definition 2. *Consider transmitter i at time slot k. If $\delta_i(k) = m$, we say that transmitter i is an m-order node. Now let $\ell_m(k)$ be the expected fraction of m-order nodes in time slot k, i.e.,*

$$\ell_m(k) = \mathbb{E}\left[\frac{1}{M}\sum_{i=1}^{M} 1_{\{\delta_i(k)=m\}}\right]. \tag{2.15}$$

We define $\{\ell_m(k)\}_{m=0}^{\infty}$ as the average node distribution (of the age-gain) at time k.

[3]In classical slotted ALOHA, "backlogged nodes" represent the nodes who have experienced collisions and transmit with the backoff probability $p_b(k)$. In our version of Rivest's algorithm, since we have unit buffer sizes, we don't use the term "backlogged". We instead work with active nodes. In each time slot k, nodes decide based on their local age-gains whether they should be active. Active nodes transmit with probability $p_b(k)$, see (2.14) where $n(k-1)$ is the number of active nodes in time $k-1$.

2.5.2. Adaptive Threshold

Let $\mathtt{T}(k)$ denote the threshold for decision-making in slot k. We propose to choose $\mathtt{T}(k)$ such that it imposes an effective arrival rate equal to $\frac{1}{eM}$ per transmitter. If the effective arrival rate per transmitter is less than $\frac{1}{eM}$, we are not utilizing the channel efficiently. If it is larger, then we are not prioritizing efficiently. This is because we would get a larger pool of packets than the slotted ALOHA can support, leading to reduced throughput and a larger age. More specifically, we design $\mathtt{T}(k)$ in three steps:

(i) Compute an estimate of the node distribution of the age-gain;

(ii) Find $\mathtt{T}(k)$ based on the estimated distribution;

(iii) Update the estimate of the node distribution based on the chosen $\mathtt{T}(k)$ and the collision feedback.

Note that $\{\ell_m(k)\}_{m=0}^{\infty}$ is unknown in decentralized systems. We hence find an estimate of it $\{\hat{\ell}_m(k)\}_{m=0}^{\infty}$ in every time slot. We summarize the process as follows

$$\{\hat{\ell}_m(k)\}_{m=0}^{\infty} = F(c(k), \{\hat{\ell}_m(k-1)\}_{m=0}^{\infty}) \tag{2.16}$$

where $F(\cdot)$ is a function that will be determined later.

Suppose the estimated node distribution $\{\hat{\ell}_m(k-1)\}_m$ is known at (the end of) time slot $k-1$. We now describe how threshold $\mathtt{T}(k)$ is designed and how $\{\hat{\ell}_m(k)\}_m$ is updated. For clarity of ideas, let us view the time slot k in three stages: The first stage corresponds to the beginning of the time slot when new packets may arrive and replace the old packets. We denote the time just before the arrival of new packets by k^- and the time just after the arrival of packets by k^+. After the arrival of new packets, at time k^+, the source's AoI changes from $w_i(k^-)$ to $w_i(k^+)$ and the destination's AoI $h_i(k^+)$ remains the same as $h_i(k^-)$. So the age-gain values and their node distributions change. We denote the resulting node distribution in this stage by $\{\hat{\ell}_m(k^+)\}_m$. In the second stage, transmitters determine the threshold $\mathtt{T}(k)$ based on $\{\hat{\ell}_m(k^+)\}_m$. Transmissions

happen according to the designed threshold $\mathtt{T}(k)$. In the third phase, at the end of time slot k when collision feedback is also available, the node distribution is once again estimated. We slightly abuse notation and denote the final estimate of the node distribution at the end of time slot k with $\{\hat{\ell}_m(k)\}_m$. The aforementioned three stages of calculating $\mathtt{T}(k)$ are described next.

Stage 1: Suppose the estimated node distribution $\{\hat{\ell}_m(k-1)\}_m$ is known at the beginning of slot k before the arrival of new packets. The expected fraction of m-order nodes that receive new packets is $\theta\hat{\ell}_m(k-1)$. The order of these nodes increases and this changes the expected node distribution to $\{\hat{\ell}_m(k^+)\}_m$ as a function of $\{\hat{\ell}_m(k-1)\}_m$. Let $a_m(k)$ denote the expected fraction of nodes that have just become m-order nodes at time k^+ for $m \geq 1$.

Lemma 1. *The expected fraction of nodes that have just become m-order nodes at time k^+ is*

$$a_m(k) = \theta^2 \sum_{j=0}^{m-1} \ell_j(k-1)(1-\theta)^{m-j-1} \tag{2.17}$$

and the expected node distribution of age-gain at time k^+ is

$$\ell_m(k^+) = \begin{cases} (1-\theta)\ell_m(k-1) & m = 0 \\ (1-\theta)\ell_m(k-1) + a_m(k) & m \geq 1. \end{cases} \tag{2.18}$$

Proof. The proof is straightforward and delegated to Appendix A.7. □

We define $\hat{a}_m(k)$ as an estimate of $a_m(k)$, which can be obtained by (2.17) and (2.18) by replacing $\ell_m(k)$, $\ell_m(k^+)$ with $\hat{\ell}_m(k)$, $\hat{\ell}_m(k^+)$, respectively.

Stage 2: The threshold $\mathtt{T}(k)$ is determined based on $\{\hat{\ell}_m(k^+)\}_m$. We design $\mathtt{T}(k)$ such that the *effective arrival rate* of packets that have an age-gain above $\mathtt{T}(k)$ is close to $\frac{1}{e}$. In other words, we *thin* the arrival process using local age information. The critical point $\frac{1}{e}$ is the maximum sum arrival rate that ALOHA can support. So if the effective sum arrival rate falls below $\frac{1}{e}$, we do not use the full channel capacity[4] and if we operate above $\frac{1}{e}$, then we incur additional collisions and

[4]Here, capacity refers to the maximum achievable sum rate under ALOHA.

delay.

We define the effective arrival rate as the fraction of sources with new arrivals whose age-gain is larger than or equal to $\mathtt{T}(k)$. Recall that $\hat{a}_m(k)$ is the estimation of the expected fraction of nodes that have just become m-order nodes at time k^+ (coming from lower order nodes). So the total (estimated) fraction of nodes whose age-gain would, for the first time, pass the threshold $\mathtt{T}(k)$ is

$$\sum_{m \geq \mathtt{T}(k)} \hat{a}_m(k).$$

We propose to choose $\mathtt{T}(k)$ according to the following rule:

$$\mathtt{T}(k) = \max \left\{ t | \sum_{m \geq t} \hat{a}_m(k) \geq \frac{1}{eM} \right\}. \tag{2.19}$$

Remark 3. *We chose $\mathtt{T}(k)$ to be the maximum threshold value that does not bring an effective sum arrival rate below $\frac{1}{e}$. This is due to the integer nature of age and hence k. One can also time share between $\mathtt{T}(k)-1$ and $\mathtt{T}(k)$ to operate at an effective sum arrival rate (almost) equal to $\frac{1}{e}$. Thus, to simplify* (2.14)*, we can replace $M\theta$ by the effective arrival rate $\frac{1}{e}$ in* (2.14)*.*

Remark 4. *The threshold $\mathtt{T}(k)$ in* (2.19) *can also be applied to the regime $0 < \theta < \frac{1}{eM}$. In this regime, $\sum_{m=1}^{\infty} a_m(k) < \frac{1}{eM}$. Therefore, assuming that the estimates $\hat{a}_m(k)$ are accurate, the threshold is $\mathtt{T}(k) \leq 1$, reducing the proposed algorithm to the slotted ALOHA.*

Stage 3: Once the threshold $\mathtt{T}(k)$ is determined, each transmitter verifies locally if its age-gain is above the specified threshold. If so, it transmits its packet with probability $p_b(k)$ defined in (2.14) mimicking slotted ALOHA. If collisions happen or if all nodes abstain from transmitting, then AoI at the destination increases by 1 for all sources. If only one node transmits, its packet will be delivered successfully and the corresponding age at the destination drops to the source's AoI.

2.5.3. Estimating the node distribution

It remains to estimate $\hat{\ell}_m(k)$ at the end of time slot k, which will serve in computing $\mathtt{T}(k+1)$ in the next time slot. We assume that at the end of time slot k, all transmitters are provided with collision

feedback from the channel and we hence consider two cases separately: $c(k) = 0$ and $c(k) = 1$.

If collision has occurred, i.e., $c(k) = 1$, then the order of nodes will not change:

$$\hat{\ell}_m(k) = \hat{\ell}_m(k^+), \quad m \geq 0. \tag{2.20}$$

If there was no collision, i.e., $c(k) = 0$, then either a packet was delivered or no packet was delivered. Recall that we design $\mathtt{T}(k)$ to impose (in the limit of large M) an effective sum arrival rate almost equal to $\frac{1}{e}$. Following Lemma 15 in Appendix A.5, the two events of idle and successful delivery are almost equiprobable for large M:

$$\lim_{k\to\infty} \Pr\big(\sum_{i=1}^{M} d_i(k) = 1, c(k) = 0\big) \approx \frac{1}{e}$$
$$\lim_{k\to\infty} \Pr\big(\sum_{i=1}^{M} d_i(k) = 0, c(k) = 0\big) \approx \frac{1}{e}.$$

Thus, condition on $c(k) = 0$, a packet is delivered with probability $1/2$, i.e., the expected number of delivered packets is $1/2$ and by the inherent symmetry of the system, each active node has the same chance to deliver a new packet. For any $m \geq \mathtt{T}(k)$, a packet is delivered by m-order nodes with probability

$$r_m(k) = \frac{\ell_m(k^+)}{\sum_{t\geq \mathtt{T}(k)} \ell_t(k^+)}. \tag{2.21}$$

The expected number of m-order nodes is $M\ell_m(k^+)$ and the expected number of delivered packets by m-order nodes (condition on $c(k) = 0$) is $\frac{r_m(k)}{2}$. Note that m-order nodes can not deliver more than $M\ell_m(k^+)$ packets since the total number of m-order nodes is $M\ell_m(k^+)$ and the buffer size is 1, then

$$\frac{r_m(k)}{2} < M\ell_m(k^+). \tag{2.22}$$

In order to estimate the expected fraction of m-order nodes that have a successful delivery, we

simply plug in $\hat{\ell}_m(k^+)$ as an estimate for $\ell_m(k^+)$. Since (2.22) does not necessarily hold anymore using the estimates, we estimate the expected fraction of m-order nodes with a successful delivery as follows:

$$\frac{1}{M}\min\big(\frac{r_m(k)}{2}, M\hat{\ell}_m(k^+)\big),$$

where $r_m(k)$ is computed by (2.21) and replacing $\ell_m(k)$ with $\hat{\ell}_m(k)$. Consequently, the update rule of the node distribution of age, $\{\hat{\ell}_m(k)\}_m$, is given as follows:

$$\begin{aligned}
\hat{\ell}_0(k) =& \hat{\ell}_0(k^+) + \sum_{m=\mathtt{T}(k)}^{\infty} \min\big(\frac{r_m(k)}{2M}, \hat{\ell}_m(k^+)\big) \\
\hat{\ell}_m(k) =& \hat{\ell}_m(k^+), \quad 1 \le m \le \mathtt{T}(k)-1 \\
\hat{\ell}_m(k) =& \left(\hat{\ell}_m(k^+) - \frac{r_m(k)}{2M}\right)^+, \quad m \ge \mathtt{T}(k).
\end{aligned} \tag{2.23}$$

Collecting Stages 1 - 3, from (2.18), (2.16) can be re-written as

$$\begin{aligned}
\hat{\ell}_0(k) =& (1-\theta)\hat{\ell}_0(k-1) + 1_{\{c(k)=0\}} \sum_{m=\mathtt{T}(k)}^{\infty} \min\big(\frac{r_m(k)}{2M}, (1-\theta)\hat{\ell}_m(k-1) + \hat{a}_m(k)\big) \\
\hat{\ell}_m(k) =& (1-\theta)\hat{\ell}_m(k-1) + \hat{a}_m(k), \quad 1 \le m \le \mathtt{T}(k)-1 \\
\hat{\ell}_m(k) =& \left((1-\theta)\hat{\ell}_m(k-1) + \hat{a}_m(k) - 1_{\{c(k)=0\}}\frac{r_m(k)}{2M}\right)^+, \quad m \ge \mathtt{T}(k).
\end{aligned} \tag{2.24}$$

where $a_m(k)$ and $r_m(k)$ are defined in (2.17) and (2.21), respectively. Finally, in this case, the probability of transmission is calculated by (2.14), where $M\theta$ is replaced by the effective arrival rate $\frac{1}{e}$.

Algorithm 1 describes the proposed distributed age-based transmission policy. We numerically evaluate its age performance in Section 2.6 and analyze a stationary version of it when the threshold is fixed in Section 2.5.4. Comparing with the slotted ALOHA in (2.14), Algorithm 1 significantly reduces the NAAoI when the sum arrival rate is beyond $\frac{1}{e}$ (see Fig. 2.3a and Fig. 2.4a). It achieves this by carefully selecting and delivering packets with a large age-gain. The NAAoI under Algo-

rithm 1 decreases sharply when the arrival rate θ approaches 1 (see Figure 2.2a). In particular, the NAAoI it achieves at $\theta = 1$ is almost 1. Contrasting that with the lower bound in Proposition 1, one comes to the conclusion that the throughput achieved by Algorithm 1 is larger than that of a standard slotted ALOHA. This is because of the implicit coordination that is facilitated by estimating and utilizing the age gain distributions for decision-making.

Algorithm 1 Adaptive Age-based Thinning (AAT)

Set a large integer N and the time horizon K.
Set initial points: $h_i(0) = 1$, $w_i(0) = 0$ for $i = 1, 2, \cdots, M$; $c(0) = 0$; $\mathtt{T}(0) = 1$; $p_b(0) = 1$; $n(0) = 0$; $k = 1$.
repeat
Step 1: Calculate $\{\hat{\ell}_m(k^+)\}_{m=1}^N$ by (2.18).
Step 2: Calculate $\mathtt{T}(k)$ by (2.19).
Step 3: For transmitter i, $i = 1, \ldots, M$: compute $\delta_i(k^+) = h_i(k^+) - w_i(k^+)$; if $\delta_i(k^+) < \mathtt{T}(k)$, then it does not transmit packets; if $\delta_i(k^+) \geq \mathtt{T}(k)$, then it transmits a packet with probability $p_b(k)$[5].
Step 4: If $c(k) = 0$, calculate $\{\hat{\ell}_m(k)\}_{m=1}^N$ by (2.20), and if $c(k) = 1$, calculate $\{\hat{\ell}_m(k)\}_{m=1}^N$ by (2.23). Calculate $p_b(k+1)$ by (2.14) in which $M\theta$ is replaced by $\min(M\theta, e^{-1})$.
until $k = K$
Calculate J_K^{AAT} by (2.3).

Remark 5. *From* (2.14)*, to estimate the number of active nodes in each time slot, the number of nodes in the network is needed. We set M to be a pre-determined parameter, which is known to all nodes.*

Remark 6. *The estimates $\{\hat{\ell}_m(k)\}_m$ and $\{\hat{a}_m(k)\}_m$ in Algorithm 1 are not exactly accurate and this is due to the integer nature of the threshold. Assume that $\{\hat{\ell}_m(k_0)\}_m$ and $\{\hat{a}_m(k_0)\}_m$ are exactly accurate in time slot k_0. We may have $\sum_{m \geq \mathtt{T}(k_0)+1} \hat{a}_m(k_0) < \frac{1}{eM}$ but $\sum_{m \geq \mathtt{T}(k_0)} \hat{a}_m(k_0) > \frac{1}{eM}$ in which case the effective arrival rate $\left(= \sum_{m \geq \mathtt{T}(k_0)} \hat{a}_m(k_0)\right)$ would be larger than e^{-1}. But the steps in* (2.23) *are derived by assuming an effective arrival rate $\frac{1}{e}$ and this leads to inaccuracies in our estimates $\{\hat{\ell}_m(k)\}_m$ and $\{\hat{a}_m(k)\}_m$ as computed in Algorithm 1.*

Remark 7. *We updated $\{\hat{\ell}_m(k)\}_m$ as a function of $\{\hat{\ell}_m(k-1)\}_m$ and the collision feedback $c(k)$, hence the name adaptive. $\mathtt{T}(k)$ and $\{\hat{\ell}_m(k)\}_m$ are known at all sources and every source finds the same $\mathtt{T}(k)$. If we update $\{\ell_m(k)\}_m$ (not $\{\hat{\ell}_m(k)\}_m$) by the conditional expectation of $\{\ell_m(k)\}_m$,*

condition on $\{\ell_m(k^+)\}_m$ but not on $c(k)$, we will find a fixed limiting threshold $\mathtt{T}^*$ discussed next.

2.5.4. Fixed Threshold

A simple variant of the age-based thinning method is found when the threshold $\mathtt{T}(k) = \mathtt{T}^*$ is fixed throughout the transmission phase. In particular, we design $\mathtt{T}^*$ ahead of time based on the node distribution in the stationary regime. By doing so, we cannot benefit from the collision feedback to adaptively choose $\mathtt{T}(k)$. However, this framework is preferable for deriving analytical results.

We use the framework and derivation we developed for adaptive thinning in order to find a fixed "optimal" $\mathtt{T}^*$ that imposes an effective arrival rate approximately[6] equal to $1/e$. Note that a larger arrival rate implies further random thinning of the packets to meet the fundamental rate $1/e$ (as opposed to the selective nature of thinning by imposing age thresholding) and a smaller arrival rate corresponds to inefficient utilization of the channel.

The major difference between an adaptive threshold and a fixed threshold is in the update rules (2.20)-(2.23) because $c(k)$ is not known when $\mathtt{T}^*$ is designed. In particular, the update rule (2.20)-(2.23) is replaced by an average rule that weighs $c(k) = 1$ with probability $1 - \frac{2}{e}$ and $c(k) = 0$ with probability $\frac{2}{e}$ (following Lemma 15).

By the stationarity of the scheme, the limit of $\{\ell_m(k)\}_{m=0}^{\infty}$ and $\{\ell_m(k^+)\}_{m=0}^{\infty}$ exist as $k \to \infty$. Denote the two limits by $\{\ell_m^*\}_{m=0}^{\infty}$ and $\{\ell_m^{+*}\}_{m=0}^{\infty}$, respectively. Similar with (2.18), the update rule of Stage 1 implies

$$\begin{aligned} \ell_0^{+*} &= (1-\theta)\ell_0^* \\ \ell_m^{+*} &= (1-\theta)\ell_m^* + a_m^* \qquad m \geq 1 \end{aligned} \tag{2.25}$$

where

$$a_m^* = \theta^2 \sum_{j=0}^{m-1} \ell_j^* (1-\theta)^{m-j-1} \qquad m \geq 1. \tag{2.26}$$

[5] A packet with a large age-gain must have a packet ready to transmit.

[6] This approximation is due to the integer nature of the age threshold.

Since we let $\mathtt{T}(k) = \mathtt{T}(k-1) = \mathtt{T}^*$, the threshold proposed in Stage 2 is

$$\mathtt{T}^* = \max\Big\{t | \sum_{m \geq t} a_m^* \geq \frac{1}{eM}\Big\}. \tag{2.27}$$

Next, consider Stage 3. In contrast to Section 2.5.3, we do not utilize collision feedback in finding $\mathtt{T}(k)$. So estimating the fraction of m-order nodes at the end of time slot k will account for $c(k) = 1$ with probability $1 - \frac{2}{e}$ and $c(k) = 0$ with probability $\frac{2}{e}$ (see Lemma 15). We hence obtain

$$\begin{aligned} \ell_0^* &= \ell_0^{+*} + \frac{1}{eM} \\ \ell_m^* &= \ell_m^{+*}, \quad 1 \leq m \leq \mathtt{T}^* - 1 \\ \ell_m^* &= \ell_m^{+*} - \frac{r_m^*}{eM}, \quad m \geq \mathtt{T}^* \end{aligned} \tag{2.28}$$

where

$$r_m^* = \ell_m^{+*} / \sum_{i=\mathtt{T}}^{\infty} \ell_i^{+*}.$$

Putting together (2.25) - (2.28), we obtain

$$\begin{aligned} \ell_0^* &= (1-\theta)\ell_0^* + \frac{1}{eM} \\ \ell_m^* &= (1-\theta)\ell_m^* + a_m^*, \quad 1 \leq m \leq \mathtt{T}^* - 1 \\ \ell_m^* &= (1-\theta)\ell_m^* + a_m^* - \frac{r_m^*}{eM}, \quad m \geq \mathtt{T}^* \end{aligned} \tag{2.29}$$

and conclude the following lemma (see Appendix A.8 for the proof).

Lemma 2. *As $k \to \infty$, the stationary distributions $\{\ell_m^*\}_m$, $\{\ell_m^{+*}\}_m$ and $\{a_m^*\}_m$ satisfy the following*

properties:

$$\ell_m^* = \begin{cases} \frac{1}{eM\theta} & m = 0 \\ \frac{1}{eM} & 1 \le m \le \mathtt{T}^* - 1 \end{cases} \tag{2.30}$$

$$\ell_m^{+*} = \frac{1}{eM} \qquad 1 \le m \le \mathtt{T}^* - 1 \tag{2.31}$$

$$a_m^* = \frac{\theta}{eM} \qquad 1 \le m \le \mathtt{T}^*. \tag{2.32}$$

The closed form expression of the fixed threshold $\mathtt{T}^*$ is given below (see Appendix A.9 for the proof) and Algorithm 2 describes our stationary ge-based transmission policy.

Theorem 2. *The fixed threshold* $\mathtt{T}^*$ *in* (2.27) *has the following closed form expression:*

$$\mathtt{T}^* = \lfloor eM - \frac{1}{\theta} + 1 \rfloor.$$

Remark 8. *The threshold in Theorem 2 can be applied to the regime* $0 < \theta < \frac{1}{eM}$ *as well. In particular, in this regime, the threshold is* $\mathtt{T}^* \le 0$ *and the proposed algorithm reduces to the slotted ALOHA.*

Algorithm 2 Stationary Age-based Thinning (SAT)

Set the time horizon K.
Set initial points: $h_i(0) = 1$, $w_i(0) = 0$ for $i = 1, 2, \cdots, M$; $c(0) = 0$; $p_b(0) = 1$; $n(0) = 0$; $k = 1$.
Calculate $\mathtt{T}^* = \lfloor eM - \frac{1}{\theta} + 1 \rfloor$.
repeat
  Step 1: For transmitter i, compute $\delta_i(k) = h_i(k) - w_i(k)$, if $\delta_i(k) < \mathtt{T}^*$, then it does not transmit packets; if $\delta_i(k) \ge \mathtt{T}^*$, then it transmits a packet with probability $p_b(k)$.
  Step 2: Calculate $p_b(k+1)$ by (2.14) in which $M\theta$ is replaced by $\min(M\theta, e^{-1})$.
until $k = K$
Calculate J_K^{SAT} by (2.3).

We finally prove asymptotically (as $M \to \infty$) that the Stationary Age-based Thinning (SAT) policy described in Algorithm 2 significantly reduces age when $1/\theta = o(M)$. Recall that at $\theta = \frac{1}{eM}$, we have $\lim_{M\to\infty} J^{SA}(M) = e$. For larger arrival rates θ where $1/\theta = o(M)$, we prove that Algorithm 2 sharply reduces AoI from e to $\frac{e}{2}$.

Theorem 3. *For any* $\theta = \frac{1}{o(M)}$,

$$\lim_{M\to\infty} J^{SAT}(M) = \frac{e}{2}.$$

Remark 9. *The minimum NAAoI attained by a stabilized slotted ALOHA is (asymptotically)* e *and it is achieved at* $\theta = \frac{1}{eM}$ *(See Theorem 1 and Fig. 2.4a). Theorem 3 shows that our proposed SAT policy attains the NAAoI* $= \frac{e}{2}$ *(asymptotically) for* $\theta = \frac{1}{o(M)}$. *This provides a multiplicative factor of* 2 *compared to the minimum NAAoI under slotted ALOHA. Moreover, simulation results show that the AAT policy outperforms the SAT policy for* $\theta = \frac{1}{o(M)}$ *(see Fig. 2.4a).*

Proof. The proof is given in Appendix A.10. Here, we provide the road map of the proof. In every time slot k, the sources can be divided into two groups: 1) sources with $\delta_i(k) < \mathtt{T}^*$; 2) sources with $\delta_i(k) \geq \mathtt{T}^*$. The first group of sources has the main contribution to $J^{SAT}(M)$ (which is equal to $\frac{e}{2}$) when $M \to \infty$. The contribution of the second group of sources to $J^{SAT}(M)$ vanishes when $M \to \infty$. □

2.5.5. Extensions to Other Random Access Technologies

So far, we restricted attention to slotted ALOHA as the main random access technology. However, in the past decade, novel technologies such as Carrier Sensing Multiple Access (CSMA) technologies have emerged and led to significant improvements in terms of throughput. It is interesting to know how they perform with regard to age, especially since they are known to have large delays [26, 27, 158, 159, 160, 161, 162, 163]. In this regard, [26, 27] have proposed an efficient sleep-wake mechanism for wireless networks that attains the optimal trade-off between minimizing the AoI and energy consumption. In [161], a network with M sources (links) under the CSMA scheme was considered and the closed form of the average age of information was derived as a function of the back-off time and generation rate. In [162], the notion of the broadcast age of information was investigated in wireless networks with CSMA/CA technologies.

In this section, we outline how the age-based thinning method described in Section 2.5.4 (with a fixed

threshold) can be applied to other random access technologies. For this purpose, we consider any transmission policy π that does not employ coding across packets. All existing collision avoidance and resolution techniques such as ALOHA and CSMA [164, 165, 166, 167] fall into this class. Now develop a variant of the transmission policy π in which only the most recent packets of each transmitter are preserved and all older packets are discarded. Denote this policy by $\pi^{(1)}$. Define $C^{\pi^{(1)}}(M)$ as the maximum sum throughput when applying the transmission policy $\pi^{(1)}$ in a system with M sources, and denote the limit, when $M \to \infty$, by $C^{\pi^{(1)}}$. Consider the age-based thinning process in two steps: (i) the threshold $\mathtt{T}^*$ is calculated, (ii) all nodes with age-gains larger than or equal to $\mathtt{T}^*$ become active and transmit using the prescribed random access technology[7].

Consider M to be large, and suppose the expected number of delivered packets per time slot is around $\min(M\theta, C^{\pi^{(1)}})$. Therefore, (2.25) remains the same and (2.27) takes the following form:

$$\mathtt{T}^* = \max\left\{ t | \sum_{m \geq t} a_m^* \geq \frac{C^{\pi^{(1)}}}{M} \right\}. \tag{2.33}$$

Following a similar argument as in Section 2.5.4, the equations in (2.28) can be written more generally as follows:

$$\begin{aligned} \ell_0^* &= \ell_0^{+*} + \min(\theta, \frac{C^{\pi^{(1)}}}{M}) \\ \ell_m^* &= \ell_m^{+*}, \quad 1 \leq m \leq \mathtt{T}^* - 1 \\ \ell_m^* &= \ell_m^{+*} - r_m^* \min(\theta, \frac{C^{\pi^{(1)}}}{M}), \quad m \geq \mathtt{T}^* \end{aligned} \tag{2.34}$$

where

$$r_m^* = \ell_m^{+*} / \sum_{i=\mathtt{T}^*}^{\infty} \ell_i^{+*}.$$

[7]Here, "prescribed random access technology" refers to the specific transmission scheme which is applied to the random access channel.

Combining (2.25), (2.33), (2.34), we thus find

$$\ell_0^* = \begin{cases} \min(1, \frac{C^{\pi^{(1)}}}{M\theta}) & m = 0 \\ \min(\theta, \frac{C^{\pi^{(1)}}}{M}) & 1 \le m \le \mathtt{T}^* - 1 \end{cases} \tag{2.35}$$

$$\ell_m^* = \min(\theta, \frac{C^{\pi^{(1)}}}{M}) \qquad 1 \le m \le \mathtt{T}^* - 1 \tag{2.36}$$

$$a_m^* = \min(\theta^2, \frac{\theta C^{\pi^{(1)}}}{M}) \quad 1 \le m \le \mathtt{T}^*. \tag{2.37}$$

Moreover, the threshold T^* takes a simple closed-form expression as stated below (and proved in Appendix A.11).

Theorem 4. *The fixed threshold* $\mathtt{T}^*$ *in* (2.33) *has the following closed form expression:*

$$\mathtt{T}^* = \left\lfloor \frac{M}{C^{\pi^{(1)}}} - \frac{1}{\theta} + 1 \right\rfloor.$$

Using this result, Algorithm 3 proposes a decentralized age-based thinning method that can be applied to any given stationary random access technology.

Algorithm 3 Generalized Stationary Age-based Thinning (GSAT)

Set the time horizon K.
Set initial points: $h_i(0) = 1$, $w_i(0) = 0$ for $i = 1, 2, \cdots, M$; $c(0) = 0$; $p_b(0) = 1$; $n(0) = 0$; $k = 1$.
Calculate the threshold $\mathtt{T}(C^{\pi^{(1)}}) = \lfloor \frac{M}{C^{\pi^{(1)}}} - \frac{1}{\theta} + 1 \rfloor$.
repeat
  For the source node i, compute $\delta_i(k) = h_i(k) - w_i(k)$. If $\delta_i(k) < \mathtt{T}(C^{\pi^{(1)}})$ remain silent; If $\delta_i(k) \ge \mathtt{T}(C^{\pi^{(1)}})$, transmits according to the random access technology $\pi^{(1)}$.
until $k = K$
Calculate J_K^{GSAT} by (2.3).

We prove an analogue to Theorem 3, showing that the Generalized Stationary Age-based Thinning policy (GSAT) proposed in Algorithm 3 reduces age to $\frac{1}{2C^{\pi^{(1)}}}$ as θ increases.

Theorem 5. *For any* $\theta = \frac{1}{o(M)}$,

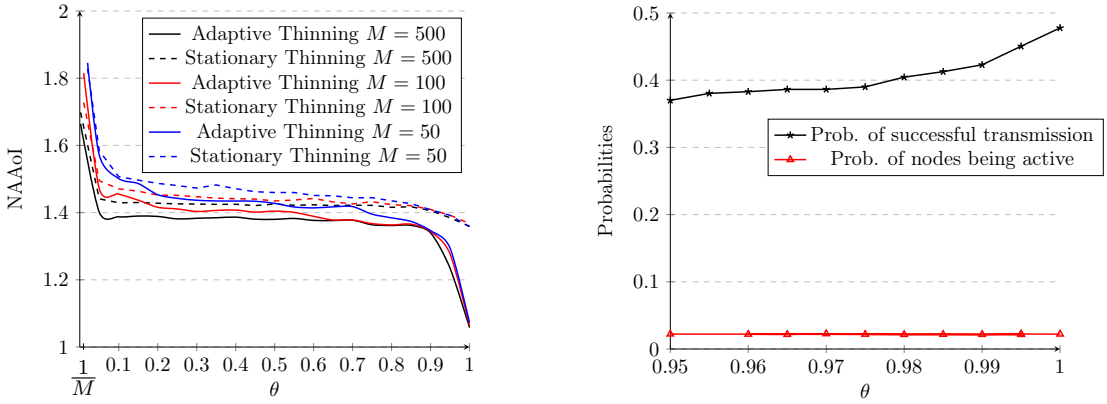
$$\lim_{M\to\infty} J^{GSAT}(M) = \frac{1}{2C^{\pi^{(1)}}}.$$

Proof. The proof of Theorem 5 is given in Appendix A.12. □

Remark 10. *The results in this section are stronger than [25] in three aspects: (i) we gave a simple and explicit expression for the threshold* $\mathtt{T}^*$*, while the threshold has to be computed numerically in [25]; (ii) we found the asymptotical NAAoI (*$\lim_{M\to\infty}\mathbb{E}[J^{GSAT}(M)]$*) analytically; (iii) the threshold in this section can be applied not only to CSMA, but also to any other transmission policy.*

Remark 11. *The framework we have built, particularly Algorithm 3 and Theorem 5, can be directly applied to other settings and multi-access technologies such as MAC with common information in [168] and a queue-length-based MAC in [169]. These technologies can achieve sum capacity* 1*, like CSMA, and their corresponding normalized age tends to* $\frac{1}{2}$ *as* M *gets large.*

2.6. Numerical Results

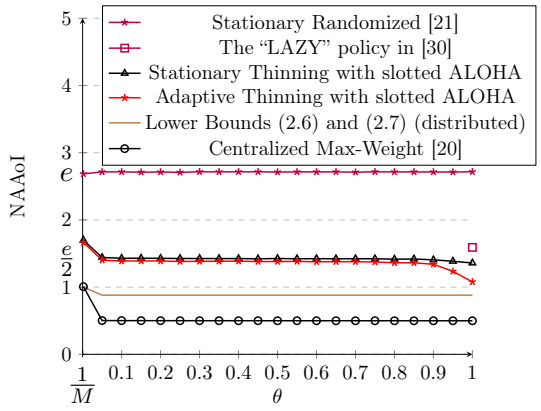


(a) NAAoI for stationary and adaptive age-based policies.

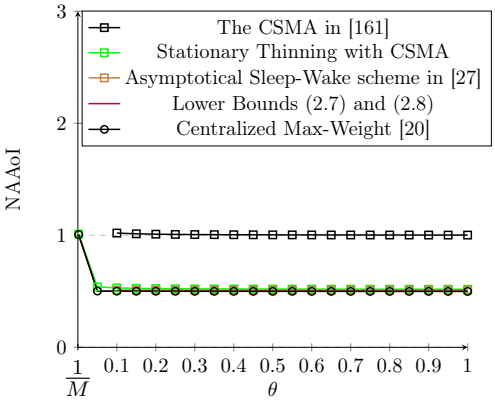
(b) Probabilities of successful transmission and being active in adaptive thinning policies.

Figure 2.2: NAAoI and success transmission probabilities.

In this section, we verify our findings through simulations. Figure 2.2a shows the normalized age under adaptive and stationary age-based transmission policies for $M = 50, 100, 500$. For stationary age-based policies, the normalized age converges to $\frac{e}{2}$ when M is large, validating our findings in

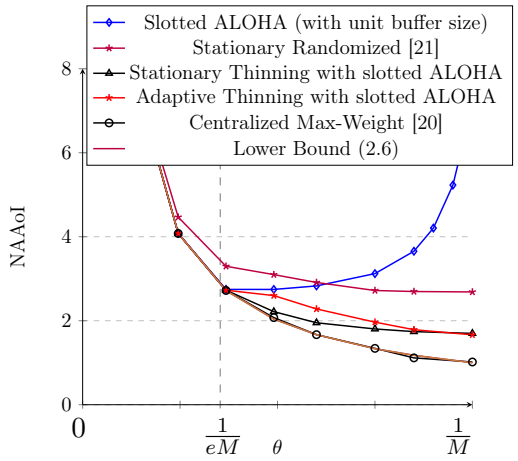


(a) NAAoI of Slotted ALOHA and the state-of-the-art.

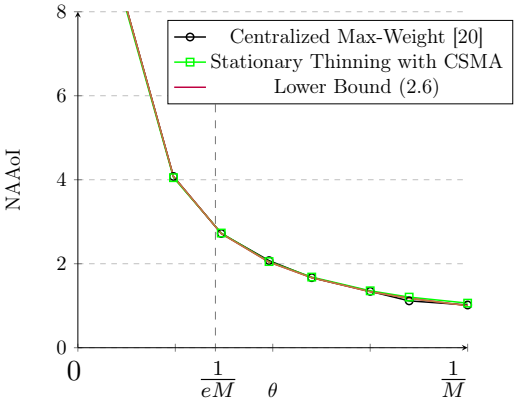


(b) NAAoI of CSMA and the state-of-the-art.

Figure 2.3: NAAoI when $M = 500$ v.s $\theta \in (\frac{1}{M}, 1]$.



(a) NAAoI of slotted ALOHA and the state-of-the-art.



(b) NAAoI of CSMA and the state-of-the-art.

Figure 2.4: NAAoI when $M = 500$ v.s $\theta \in (0, \frac{1}{M}]$.

Theorem 3.

The performance of the adaptive policy is better than that of the stationary age-based policy for $\theta > \frac{1}{M}$ and the efficacy (the gap between the two curves) increases with θ. Since the maximum sum throughput of slotted ALOHA is $\frac{1}{e}$, one may ask if this contradicts the lower bound of Proposition 2. To answer this question, we remark that the adaptive age-based transmission policy is *not* a slotted ALOHA scheme, and therefore the maximum throughput of slotted ALOHA would not apply. As a matter of fact, Fig. 2.2b shows that the throughput of the scheme increases beyond $\frac{1}{e}$ with

θ, supporting Proposition 1. This is because the AAT policy implicitly facilitates coordination among the transmitters as they utilize the (estimated) age-age distribution for decision-making. The throughput at $\theta = 1$, as seen in Fig. 2.2b, is close to .48 which is consistent with the known lower bound 0.4878 and upper bound 0.568 on the (information theoretic) channel capacity of random access channels with feedback [170, 157]. It is interesting that the AAT policy can both increase the throughput and decrease the AoI simultaneously when θ approaches 1.

One can also observe that the adaptive policy performs worse than the stationary policy for $\frac{1}{eM} \leq \theta \leq \frac{1}{M}$. The SAT policy is designed as the stationary version of the AAT policy, thus the SAT policy should not in principle outperform the AAT policy (assuming that our approximations of the estimates of the age-gain distribution are accurate). However, it is worthwhile to discuss this counter-intuitive phenomenon and we expand on the underlying reason: We consistently underestimate the threshold $\mathtt{T}(k)$ due to the integer nature of it, especially when θ is small. For example, consider $\frac{1}{eM-1} \leq \theta < \frac{1}{eM-2}$. Note that $\sum_{m\geq 1} a_m(k) = \theta$, which is close to $\frac{1}{eM}$. From (2.19), in some slots, the event $\{\sum_{m\geq 2} \hat{a}_m(k) < \frac{1}{eM}\}$ may occur (even if the estimate $\{\hat{a}_m(k)\}_m$ is exactly accurate), so the threshold $\mathtt{T}(k)$ under the AAT policy would be 1 because $\mathtt{T}(k)$ is always an integer. In these time slots, the AAT policy is reduced to the slotted ALOHA. Therefore, the fraction of active nodes becomes large, the throughput decreases, and the age increases. On the other hand, in the regime when $\frac{1}{eM} \leq \theta \leq \frac{1}{M}$, the threshold $\mathtt{T}^* = 2$ under the SAT policy in every time slot (see Theorem 2). Subsequently, the estimate of the age-gain distribution $\{\hat{\ell}_m(k)\}_m$ is imprecise (see Remark 6). Moreover, the imprecise estimates in (2.24) may aggravate the underestimation of the threshold, and the closed-loop worsens the performance of the AAT policy. The effect is more pronounced in the regime $\frac{1}{eM} \leq \theta \leq \frac{1}{M}$ where old packets are less frequently replaced with new packets.

Finally, the age performances of our proposed distributed age-based policies are compared with the lower bounds of Section 2.3, state-of-the-art distributed schemes such as [21], as well as centralized Max-Weight policies such as [20]. For clarity of exposition, we consider two regimes of θ: $\theta \in (\frac{1}{M}, 1]$ (see Fig. 2.3), and $\theta \in (0, \frac{1}{M}]$ (see Fig. 2.4). Fig. 2.4, in particular, shows that when $\theta \leq \frac{1}{eM}$, the

normalized age of slotted ALOHA coincides with centralized Max-Weight policies and the lower bound of Proposition 2. When θ increases beyond $\frac{1}{eM}$, our proposed age-based thinning methods provide significant gains compared to randomized stationary and slotted ALOHA schemes. When $\theta = \frac{1}{o(M)}$, the NAAoI of slotted ALOHA explodes, and we omit the curve of slotted ALOHA in Fig. 2.3a. Finally, we numerically observe that the normalized age of the centralized Max-Weight policy is approximately attained by stationary age-based thinning in perfect CSMA[8] (see the green square curve in Fig. 2.3b and Fig. 2.4b), where the length of one contention slot is set to be 1/100 [25, 171].

Next, we compare our proposed algorithms with policies in related works, such as a lazy version of slotted aloha in [30], and variants of CSMA in [161, 26, 27]. Different from [30], we considered a random access channel with re-transmission attempts for packets and used a slotted ALOHA with time-variant transmission probability, while in [30], a channel without re-transmission and a slotted ALOHA with a time-invariant transmission probability is investigated. More importantly, we proposed a policy where the best threshold is found in every time slot, while a predetermined threshold is given in [30]. Furthermore, we showed the performance analysis for arbitrary M sources under arbitrary generation/arrival rate (in $[0, 1]$), while [30] only provided the closed form of average AoI for the case when $M = 2$ and $\theta = 1$. Compared to the performance of the policy in [30], our proposed AAT and SAT policies outperform the lazy version of slotted aloha (see Fig. 2.3a, the purple square curve). To apply the policies in [161] on our model, we consider the generation/arrival rate is relatively large ($\theta \geq 0.1$) because under [161, Assumption 1], a transmitter sends a "fake" update if its buffer is empty. From Figure 2.3b, it is easy to see that the stationary thinning with CSMA outperforms the policy in [161] (Fig 2.3b, square black curve). References [26, 27] are concurrent works on optimizing peak AoI over random access channels with per-source battery lifetime constraints. Translating the introduced energy constraints to arrival rate θ, one can apply the sleep-waking schemes of [26, 27] to our problem when CSMA capabilities are available. Using the symmetry of our model along with [27, Eqns. (13), (16)], when M is sufficiently large, the fraction of time in which every source is in transmission mode is around $r/(Mr + 1)$ where r is the

[8]We consider perfect CSMA in simulations. In other words, no errors occur in carrier sensing.

sleep period parameter as proposed in [26, 27]. Using [27, Eqns. (13), (16)], one can argue that we need to consider the so-called energy-adequate regime introduced in [26, 27] which translates to relatively large θ, i.e., $\theta \geq 0.1$. The performance of the policy in [26, 27] (Fig 2.3b, red square point) is similar to that of the stationary thinning with CSMA, which is consistent with the optimality results presented in [26, 27].

It is worthwhile to mention that the proposed algorithms not only fully utilize channel capacity, but minimize NAAoI. If we only consider policies with maximum throughput (e.g. standard slotted ALOHA and its variants), the NAAoI explodes up with time for arrival rates above $\frac{1}{e}$. This is also observed in works such as [21] that adapt slotted ALOHA without packet management for age minimization.

2.7. Conclusion and Future Research

In this work, we investigated the AoI performance of a *decentralized* system consisting M source nodes communicating with a common receiver. We first derived a general lower bound on AoI. Then, we derived the analytical (normalized) age performance of (stabilized) slotted ALOHA in the limit of $M \to \infty$. As the sum arrival rate increases beyond $\frac{1}{e}$, slotted ALOHA becomes unstable. We show that by prioritizing transmissions that offer a significant reduction in AoI, we can increase the arrival rate and simultaneously decrease AoI. In particular, we proposed two age-based thinning policies: (i) Adaptive Age-based Thinning (AAT) and (ii) Stationary Age-based Thinning (SAT) and analyzed the age performance in the limit of $M \to \infty$. Finally, we demonstrated how our proposed thinning mechanism (SAT) is useful for other random access technologies. Numerical results showed that the proposed age-based thinning mechanisms make a significant contribution to the performance of age even for moderate values of M. Our framework can not be extended to generalized settings (such as [172, 173, 174, 175]) blindly. Appropriate adaptations related to different settings are necessary. For example, after applying Algorithm 3 to transmission schemes in [173, 174], they are reduced to stationary randomized policies. This is because we do not fully utilize the additional knowledge of the queue length that is provided in these settings. In the setting of [172], we should further assume that the set of nodes that transmit packets are known, and a

continuous-time version of the framework is needed in the setting of [175].

Future research includes generalization to accommodate 1) dynamic channels, i.e., the number of nodes M, or the arrival rates θ are time-variant 2) asymmetric channels, i.e., the arrival rates θ_i are different. In the first case, the method we proposed above can be applied directly. Suppose that the expressions of the number of nodes, $M(k)$, and the arrival rates, $\theta(k)$, are known. We cane replace M and θ by $M(k)$ and $\theta(k)$, respectively, in every time slot. Subsequently, the fixed threshold hold $\mathtt{T}^*$ is also a time-variant variable, $\mathtt{T}^*(k)$. In the second case, the method we proposed above can not be applied directly. This is because we use the profile of all sources as an estimate on any individual source. A more general estimation method should be proposed in the second case. In addition, note that slotted ALOHA algorithms are order optimal when M is sufficiently large and the generation/arrival rate is small ($\theta \leq \frac{1}{eM}$). An interesting extension is to consider a smarter decentralized age-based algorithm that can achieve a constant additive age gap from the optimum average age when θ is small.

CHAPTER 3

Beyond AoI: Real-time Sampling and Estimation

The Internet of Things (IoT) paradigm is changing our conception of communications. In the past decades, research has focused on various technologies to improve connectivity, rate, reliability, and/or latency [15, 18, 20, 23, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190]. But are such metrics representative of optimal multiple access system designs for future IoT and cyber-physical systems (CPS) applications or do we need to go beyond these metrics? Expanding on this question, it is important to note that in traditional designs, it is often assumed that information bits (or packets) are processed and stored at the sources, waiting to be reliably transmitted and replicated at the receiver node(s) with a high rate and low latency. However, in many IoT applications, these assumptions are no longer realistic. Oftentimes, information is to be collected and communicated in real-time. In such settings, rate, reliability, and latency may not directly be relevant.

In this chapter, we consider the problem of real-time sampling and estimation over reliable and unreliable random access channels with M Markov (autoregressive) physical processes (sources). These processes are to be observed, sampled, and communicated wirelessly with a fusion center for *timely estimation*. Considering applications in IoT and CPS, it is not realistic to assume a central scheduler that monitors all the sensors for decision-making; we, therefore, seek to design near-optimal decentralized sampling and communication strategies.

Towards understanding the problem of real-time sampling and estimation in the above setting, recent works [32, 35, 36, 37, 46] have proposed centralized and decentralized multiple access schemes to minimize the metric of AoI. Nonetheless, it has remained open to how such designs can inform near-optimal designs when the desired metric of performance is the real-time estimation error (rather than AoI as a proxy). The present work establishes this bridge and goes beyond AoI minimization.

Notations We use the notations $\mathbb{E}[\cdot]$ and $\Pr(\cdot)$ for expectation and probability, respectively. Scalars are denoted by lowercase letters, e.g. s, and random variables are denoted by capital

letters, e.g. S. The notation $A \sim B$ implies that A has the same distribution as B and $\mathcal{N}(\mu, \sigma^2)$ stands for the Gaussian distribution with mean μ and variance σ^2. The notations $O(\cdot)$ and $o(\cdot)$ represent the Big O and little o notations according to Bachmann-Landau notation, respectively.

3.1. Literature Review

Sampling: Remote estimation of physical processes requires efficient sampling and communication strategies that minimize not only the estimation error cost but also the sampling and transmission costs. With this viewpoint, prior works have studied optimal sampling strategies and their structural properties for various point-to-point scenarios. [38] designs optimal sampling strategies with limited measurements. [39] studies the problem for continuous sources. [40] proves the joint optimality of symmetric thresholding policies and Kalman-like estimators for autoregressive Markov processes. [41] formulates a two-player team problem and designs efficient iterative algorithms. Systems with energy harvesting sensors are considered in [42]. Noisy channels and packet drop channels are considered in [43, 44]. The above-mentioned works have all considered single-user channels and the developed methodologies do not generalize to random access networks with multiple sensors.

Reliable v.s. Timely communication: In estimation and control applications, timeliness of communication is key and that is why traditional rate-distortion frameworks and channel coding paradigms that propose asymptotic block coding solutions are not applicable. More importantly, it is oftentimes observed that as the rate and/or reliability of compression/communication schemes improve, their timeliness decrease. This aspect of sampling and remote estimation is barely studied in the estimation literature. One of the few existing works in this direction is [45] which proposes and optimizes a hybrid automatic repeat request (HARQ)-based remote estimation protocol and improves the performance of the remote estimation systems compared to conventional non-HARQ policies. Recently, tradeoffs between reliability/rate and the timeliness of communication have been looked at in the context of age. In channels with queue constraints, [35] establishes a tradeoff between AoI and rate. [46] finds the optimal blocklength of channel coding for minimizing AoI. [37] provides a centralized scheduling framework to attain tradeoffs between rate and AoI in broadcast channels. [36] proposes decentralized transmission strategies for random access channels that benefit from the availability of fresh packets and improve both communication rate and AoI. It is known

that AoI is closely related to the expected estimation error of schemes that are oblivious to the processes they monitor [32]. Non-oblivious sampling schemes are, however, signal-dependent and known to outperform oblivious schemes. In [32], threshold policies are shown to be optimal for point-to-point channels with a random delay, and closed-form solutions are found for the optimal threshold value. It is further shown that oblivious policies can be far from optimal.

Distributed decision making: In random access networks, a large number of sensors communicate with a single fusion center over a wireless channel. To avoid collisions, most works in this direction have considered centralized oblivious policies that do not observe the process realizations for decision-making (see, e.g., [47, 48, 49, 50, 51, 52, 53] and the references therein). In IoT applications, however, it is not realistic to assume a central scheduler that monitors all the sensors for decision-making. We seek decentralized solutions in which each sensor decides when to sample and transmit information based only on its local observations. In decentralized setups (and in the context of control, rather than estimation) [54], [55] consider wireless control architectures with multiple control loops over a random access channel and optimize the access rate of the sensors that randomly communicate. Policies that adapt to the state of the systems are proposed in [56]. The work [33] (which was carried out concurrently and independently) designs decentralized policies for the remote estimation of i.i.d processes over a collision channel. Decision-making in both [56, 33] is thresholding and based on the realization of the process (or a function of that). But since neither of the two works exploits channel collision feedback, adaptations of them (or other policies that impose fixed transmission probabilities on the channel) are far from optimal in our setup.

3.2. System Model

Consider a system with M statistically identical sensors and a fusion center. We often refer to the sensor nodes as nodes or transmitters and the fusion center as the receiver/destination. For analysis tractability, we focus on symmetric cases where all nodes have similar configurations. Let time be slotted. Each node i, $i = 1, 2, \cdots, M$, observes a process $\{X_i(k)\}_{k\geq 0}$ which is an autoregressive

process as follows

$$X_i(k+1) = \gamma X_i(k) + \Lambda_i(k) \tag{3.1}$$

where $W_i(k) \sim \mathcal{N}(0, \sigma^2)$ and $\gamma > 0$. The processes $\{X_i(k)\}_{k=0}^{\infty}$ are assumed to be mutually independent across i and for simplicity we let $X_i(0) = 0$.

At the beginning of each time slot, the nodes have the capability to sample the underlying process and decide whether or not to communicate the sample with the receiver. The communication medium is modeled by a collision channel: If two or more nodes transmit in the same time slot, then the packets interfere with each other (collide) and do not get delivered at the receiver. We assume a delay of one-time units in delivery for packets. At the end of time slot k, all transmitters are informed (through a low-rate feedback link) whether or not collision occurred, which is indicated by an indicator $c(k)$. If collisions happen in time slot k, then $c(k) = 1$; if a packet is delivered successfully at the receiver or no packet is transmitted, then $c(k) = 0$. We further assume that the buffer size of every transmitter is one packet and that new packets replace older undelivered packets at the transmitter. This assumption relies on the fact that the underlying processes that are monitored are Markovian.

The receiver estimates the process in every time slot based on the collection of the received samples. Denote by $\hat{X}_i(k)$ the estimate of $X_i(k)$ in time slot k. We define the following normalized average sum of estimation errors (NAEE) as our performance metric:

$$\begin{aligned} L^{\pi}(M) &= \lim_{K\to\infty} \mathbb{E}[L_K^{\pi}], \\ L_K^{\pi}(M) &= \frac{1}{M^2} \sum_{i=1}^{M} \frac{1}{K} \sum_{k=1}^{K} \left(X_i(k) - \hat{X}_i(k)\right)^2 \end{aligned} \tag{3.2}$$

where M is the number of sources, $\pi \in \Pi$ refers to the sampling and transmission policy in place, and Π is the set of all decentralized sampling and transmission policies. Note that the metric (3.2) is normalized by M. This allows us to study the asymptotic performance in the regime of large M. The minimum attainable NAEE is then denoted by $L(M)$: $L(M) = \min_{\pi\in\Pi} L^{\pi}(M)$. Our objective

is to design *decentralized* sampling and transmission mechanisms to attain $L(M)$.

Consider the i^{th} node. Let $\{k_\ell^{(i)}\}_{\ell\geq 0}$ be the sequence of time slots at the end of which packets are received at the destination from node i. In any time slot k, $k_{\ell-1}^{(i)} < k \leq k_\ell^{(i)}$, the latest sample from node i is received at the end of $k_{\ell-1}^{(i)}$ and since collisions may happen, then it is time-stamped at the beginning of time k' with $k' \leq k_{\ell-1}^{(i)}$. At the beginning of every time slot, the process of node i is updated. So the age of information with respect to node i, denoted by $h_i(k)$, is

$$h_i(k) = k - k_{\ell-1}^{(i)}. \tag{3.3}$$

Without loss of generality, assume $k_0^{(i)} = 0$. At the beginning of time slot k, the receiver knows the information of all packets delivered before time k, i.e., $\{X_i(k_t^{(i)})\}_{t=0}^{\ell-1}$ and reconstructs $\hat{X}_i(k)$ by the minimum mean square error (MMSE) estimator:

$$\hat{X}_i(k) = \mathbb{E}\left[X_i(k) | \left\{X_i\left(k_t^{(i)}\right)\right\}_{t=0}^{\ell-1}\right].$$

For the class of policies that we consider in this paper (oblivious policies and symmetric thresholding policies), the MMSE estimator reduces to a Kalman-like estimator:

$$\hat{X}_i(k) = \mathbb{E}[X_i(k) | X_i(k_{\ell-1}^{(i)})]. \tag{3.4}$$

In particular, from (3.1) and (3.3), we have

$$X_i(k) = \gamma^{h_i(k)} X_i\left(k_{\ell-1}^{(i)}\right) + \sum_{j=1}^{h_i(k)} \gamma^{j-1} W_i(k-j). \tag{3.5}$$

Note that $\mathbb{E}[W_i(k)] = 0$ for all i, k, then, Kalman-like estimator in (3.4) is:

$$\hat{X}_i(k) = \mathbb{E}[X_i(k) | X_i(k_{\ell-1}^{(i)})] = \gamma^{h_i(k)} X_i(k_{\ell-1}^{(i)}). \tag{3.6}$$

One of the major challenges in this problem arises from the decentralized nature of decision-making.

A decentralized policy is one in which the action of each node is only a function of its local observations and actions. In this setup, the action of node i at time k depends on the history of feedback and actions as well as casual observations of the process $\{X_i(j)\}_{j\leq k}$.

3.3. Error-based Thinning Policies

As discussed before, we consider the policies in which the nodes can observe their corresponding Markov processes for decision-making. Clearly, if all nodes try to transmit their samples at every time slot, no packet will go through due to collisions. The nodes, therefore, need to transmit packets at a lower rate. This means that they have to decide, in a decentralized manner, when to transmit. Motivated by the optimality of threshold policies in various point-to-point setups [32, 38, 42, 39], as well as their applications in age minimization over many-to-one random access channels [37], we propose threshold policies for decision making.

Define the *error process* $\psi_i(k)$ at node i as follows:

$$\psi_i(k) = |X_i(k) - \hat{X}_i(k)|. \tag{3.7}$$

Since the transmitters have access to collision feedback, they can calculate $\hat{X}_i(k)$, and hence $\psi_i(k)$, in each time slot and use this information for decision making. One way to understand $\psi_i(k)$ is as follows. In time k, if the sample of node i is successfully delivered, the estimation error will reduce by $\psi_i(k)$. So $\psi_i(k)$ quantifies the amount of instantaneous estimation error reduction upon successful delivery from transmitter i. With this viewpoint, we devise a threshold policy in which transmitters prioritize packets that have large $\psi_i(k)$. In particular, we design a fixed threshold β to distinguish and prioritize nodes that offer a high instantaneous gain.

The action of each node is thus as follows: node i becomes “active" if the error process $\psi_i(k)$ has crossed a pre-determined threshold β. Once a transmitter is active, it remains active until a packet is successfully delivered from that node. Active nodes transmit stochastically following Rivest's stabilized slotted ALOHA protocol [152, Chapter 4.2.3]. Denote the number of active nodes and an estimate of the number of active nodes in time slot k as $N(k)$, $\hat{N}(k)$, respectively. In particular, each

active node transmits its sample with probability $p_b(k)$ which is calculated adaptively as follows based on an estimate of the number of active nodes[9]:

$$p_b(k) = \min(1, \frac{1}{\hat{N}(k)})$$
$$\hat{N}(k) = \begin{cases} \hat{N}(k-1) + \hat{\lambda}(k) + (e-2)^{-1} & \text{if } c(k-1) = 1 \\ \hat{\lambda}(k) + \left(\hat{N}(k-1) - 1\right)^{+} & \text{if } c(k-1) = 0. \end{cases} \tag{3.8}$$

Here, $\hat{\lambda}(k)$ is an estimate of $\lambda(k)$, and $\lambda(k)$ is the sum arrival rate in time slot k. It is well-known that the maximum sum throughput of the slotted ALOHA is e^{-1} [152, Chapter 4.2.3] and the regime of interest is $\lambda(k) < e^{-1}$ when k is sufficiently large. In our setup, $\lambda(k)$ corresponds to the *expected* number of nodes that become *active* in time slot k (see Definition 3 ahead). We refer to $\lambda(k)$ as the activation rate or the effective arrival rate in time slot k.

So far, we have outlined a threshold policy in which a node decides to become active if its local error process is larger than a predetermined threshold value β. We call this procedure *Error-based Thinning* (EbT). The main underlying challenge is, however, in the design of the *optimal* β. In the rest of the paper, we will find an approximately optimal choice for β in different regimes of γ and analyze the corresponding NAEE. We start with some preliminaries.

3.3.1. Preliminaries

Consider node i and an inter-delivery interval $(k_{\ell-1}^{(i)}, k_{\ell}^{(i)}]$ (see Figure 3.1). The inter-delivery time $I_{\ell}^{(i)}$ is given by $I_{\ell}^{(i)} = k_{\ell}^{(i)} - k_{\ell-1}^{(i)}$. For any time slot k, $k_{\ell-1}^{(i)} < k \leq k_{\ell}^{(i)}$, based on (3.5), we can write the error process $\psi(k)$ as follows:

$$\psi_i(k) = |X_i(k) - \hat{X}_i(k)| = \Big| \sum_{j=1}^{h_i(k)} \gamma^{j-1} W_i(k-j) \Big|. \tag{3.9}$$

Note that the term on the right-hand side of (3.9) is the sum of $h_i(k)$ weighted independent Gaussian noise variables, and weights are the functions of γ. Indeed, (3.9) demonstrates that $\psi_i(k)$ contains both the information of sample values as well as the age with respect to source i.

[9]Since the sensors have unit buffer sizes, the number of "backlogged" nodes $N(k)$ in Rivest's algorithm is at most M. One notes that this has been incorporated in (3.8).

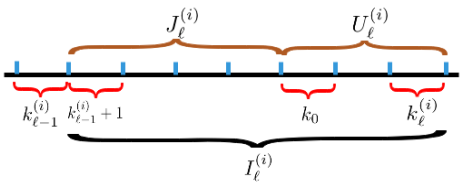


Figure 3.1: An example of $J_\ell^{(i)}$, $U_\ell^{(i)}$, and $I_\ell^{(i)}$: packets are generated at the beginning of every time slot, so $J_\ell^{(i)}$ arrivals/generations means $J_\ell^{(i)} - 1$ time slots.

We next define "active" nodes as follows.

Definition 3 (Active Nodes)**.** *If there exists a time slot $k_0 \in (k_{\ell-1}^{(i)}, k_\ell^{(i)}]$ such that (i) $\psi_i(j) < \beta$ for all $k_{\ell-1}^{(i)} < j < k_0$ and (ii) $\psi_i(k_0) \geq \beta$, then we say that node i is active in the entire interval $[k_0, k_\ell^{(i)}]$.*

Definition 4 (Silence Delay and Transmission Delay)**.** *Let k_0 be as defined in Definition 3. We define $J_\ell^{(i)} = k_0 - k_{\ell-1}^{(i)}$ as the silence delay, and $U_\ell^{(i)} = k_\ell^{(i)} - k_0 + 1$ as the transmission delay (see Figure 3.1).*

An active source becomes inactive immediately after a successful delivery. By the above two definitions, the inter-delivery time $I_\ell^{(i)}$ consists of two components – the silence delay $J_\ell^{(i)}$ and the transmission delay $U_\ell^{(i)}$:

$$I_\ell^{(i)} = J_\ell^{(i)} - 1 + U_\ell^{(i)}. \tag{3.10}$$

In this equation, $J_\ell^{(i)}$ is the first time slot after $k_{\ell-1}^{(i)}$ at which $\psi_i(k) \geq \beta$ (as defined in Definition 4). So $J_\ell^{(i)} - 1$ represents the number of time slots in which node i is not active, and $U_\ell^{(i)}$ represents the number of time slots in which node i is in active state. Recall that active nodes transmit with probability $p_b(k)$. So $U_\ell^{(i)}$ may be larger than 1 either because the node is active and it does not transmit or because the node transmits and experiences collision. By the stationarity of the transmission scheme, the processes $\{I_\ell^{(i)}\}_{i,\ell}$, $\{J_\ell^{(i)}\}_{i,\ell}$, and $\{U_\ell^{(i)}\}_{i,\ell}$ are statistically identical across i and ℓ. We define I_β, J_β, and U_β to have the same distributions as $\{I_\ell^{(i)}\}_{i,\ell}$, $\{J_\ell^{(i)}\}_{i,\ell}$, and $\{U_\ell^{(i)}\}_{i,\ell}$, respectively.

Let $a(k)$ denote the number of *newly* active nodes at time k, i.e., the number of nodes that become active from inactive states. We have $\mathbb{E}[a(k)] = \lambda(k)$, where $\lambda(k)$ is the expected sum arrival rate in time slot k (imposed by our sampling and transmission policy) [152, Chapter 4.2.3]. Now recall that in a traditional slotted Aloha-based random access channel, the maximum sum throughput is asymptotical e^{-1}. This is true also for the case with buffer size 1 where only the latest packets are stored, as discussed in [37, Appendix E]) and which applies to our setting here. Define $c(M)$ as the sum rate/throughput when the system contains M sources.

Definition 5. *The random access system is stabilized*[10] *if* $\lambda_m = \limsup_{k\to\infty} \lambda(k) < e^{-1}$.

We provide our analysis under the following two assumptions:

Assumption 1. *Under an optimal β, when M is sufficiently large, $\{a(k)\}_{k=1}^{\infty}$ are approximately independent.*

Assumption 2. *Under an optimal β, when M is sufficiently large, the random access system is stabilized, and $\lambda_m \approx e^{-1}$, $c(M) \approx e^{-1}$.*

Assumptions 1, 2 are given for analysis tractability, and we will verify them for our proposed β later. In the rest of the paper, let M be sufficiently large. We seek to find an optimal β under assumptions 1, 2. To transmit as many fresh samples as possible, β is designed such that $\lambda(k)$ is as large as possible. Thus, we focus on the regime where $\lambda(k)$ is close to e^{-1} when k is large, and from Assumption 2, $\lambda_m \approx e^{-1}$. For tractability in analysis, we let the estimate $\hat{\lambda}(k) = e^{-1}$ for all k. Specifically, we replace $\hat{\lambda}(k)$ with e^{-1} in (3.8).

Note that the system is stationary, so $U_\ell^{(i)}$ (or U_β) is a random variable and measurable. Recall from [191, Chapter 7.5.1, Lemma 7.5.1], J_β has finite moments of all orders. Therefore, I_β is measurable. Now, we first show that the (strong) law of large numbers holds for $\{I_\ell^{(i)}\}_\ell$. We remark that while $\{I_\ell^{(i)}\}_\ell$ is not independent, it is weakly correlated across ℓ, as we prove in Appendix B.1. We can

[10]Here, contrary to traditional slotted ALOHA schemes, the term "stabilized" does not refer to "stability of queues" in our problem setup. However, the term "stabilized" implies that the system is stationary when the sum arrival rate is less than e^{-1}.

thus conclude that the strong law of large numbers holds for $\{I_\ell^{(i)}\}_\ell$, see also [192].

Recall that $N(k)$ is the number of active nodes at the *beginning* of time slot k. The fraction of active nodes at the *beginning* of time slot k is hence $N(k)/M$.

Definition 6. *Define $\alpha_\beta(k)$ as the expected fraction of active nodes:*

$$\alpha_\beta(k) = \frac{\mathbb{E}[N(k)]}{M}. \tag{3.11}$$

If $\beta = 0$, then all nodes are active and $\alpha_0(k) = 1$; if $\beta = +\infty$, then all nodes are inactive and $\alpha_{+\infty}(k) = 0$. In the limit of $k \to \infty$, we denote the expected fraction of active nodes by α_β:

$$\alpha_\beta = \lim_{k\to\infty} \frac{\mathbb{E}[N(k)]}{M} = \lim_{k\to\infty} \mathbb{E}\Big[\frac{1}{M}\sum_{i=1}^{M} \mathbf{1}(\text{node } i \text{ is active at time } k)\Big]. \tag{3.12}$$

The limit in (3.12) exists because the transmission policy is stationary and hence the sequence in the expectation above is stationary in the steady state. Continuing from (3.12), we have

$$\begin{aligned} \alpha_\beta &= \lim_{K\to\infty} \mathbb{E}\Big[\frac{1}{MK}\sum_{k=1}^{K}\sum_{i=1}^{M} \mathbf{1}(\text{node } i \text{ is active at time } k)\Big] \\ &\overset{(a)}{=} \mathbb{E}\Big[\lim_{K\to\infty} \frac{1}{MK}\sum_{k=1}^{K}\sum_{i=1}^{M} \mathbf{1}(\text{node } i \text{ is active at time } k)\Big]. \end{aligned} \tag{3.13}$$

where step (a) holds by the dominated convergence theorem because the sequence in the expectation (3.13) is a fraction and bounded by 1. Utilizing the symmetry and stationarity with respect to various nodes (the system), we prove the following lemma in Appendix B.2, signifying that α_β represents the fraction of time that each node is active in the limit of $K \to \infty$, hence representing the probability of each node being active when the system is steady.

Lemma 3. *When the system is stabilized, α_β exists, and $\alpha_\beta = \frac{\mathbb{E}[U_\beta]}{\mathbb{E}[I_\beta]}$.*

Since α_β exists, then, when $k \to \infty$, the expected number of nodes that become active in every time

slot is $(1-\alpha_\beta)M\alpha_\beta$, and

$$(1-\alpha_\beta)M\alpha_\beta = \lim_{k\to\infty}\lambda(k) = \limsup_{k\to\infty}\lambda(k) = \lambda_m. \tag{3.14}$$

From Assumption 2, $\lambda_m \approx e^{-1} < 1$. Using (3.14), one sees that $M\alpha_\beta$ is an infinitesimal of a higher order than M. Now using Lemma 3, we can show that $\mathbb{E}[U_\beta]$ is an infinitesimal of a higher order than M, as discussed in the following lemma.

Lemma 4. *When the system is stabilized,*

$$\mathbb{E}[I_\beta] = \frac{M}{c(M)} \tag{3.15}$$

$$\mathbb{E}[U_\beta] = \frac{M}{c(M)}\alpha_\beta = o(M) \tag{3.16}$$

where α_β is the expected fraction of active nodes in the steady state as defined in (3.12).

Remark 12. *Lemma 4 coincides with one's intuition. Recall that the throughput of the channel is $c(M)$, so the throughput for each node is $\frac{c(M)}{M}$ (due to the symmetry). From the perspective of expectation, every successful delivery takes $\frac{M}{c(M)}$ time slots, i.e., $\mathbb{E}[I_\beta] = \frac{M}{c(M)}$. In addition, note that the expected number of active nodes is $M\alpha_\beta$, so the throughput of every active node is $\frac{c(M)}{M\alpha_\beta}$. Again, from the perspective of expectation, every successful delivery from active nodes takes $\frac{M}{c(M)}\alpha_\beta$ time slots, i.e., $\mathbb{E}[U_\beta] = \frac{M}{c(M)}\alpha_\beta$.*

Proof. The proof of Lemma 4 is given in Appendix B.3. □

3.4. Optimal Threshold in Regime of $\gamma = 1$

We start with a standard case where $\gamma = 1$. The error process $\psi(k)$ defined in (3.9) is reduced to:

$$\psi_i(k) = |X_i(k) - \hat{X}_i(k)| = \Big| \sum_{j=k_{\ell-1}^{(i)}}^{k-1} W_i(j) \Big|. \tag{3.17}$$

To provide a better picture of the solution of optimal β^*, we first consider a simpler class of policies, called *oblivious* policies, in Section 3.4.1. The actions in oblivious policies depend only on the history of feedback and actions at that node. Then, we design and analyze a general decentralized mechanism, called *non-oblivious* policies where decision-making depends on the observed processes, that outperform oblivious schemes in minimizing the expected average estimation error in the rest of the Section.

3.4.1. Oblivious Policies and Age of Information

We first define the *oblivious* policies as the policies where actions of each node depend only on the history of feedback and actions at that node. In particular, oblivious policies do not take into account the realization (value) of the samples, but only the time they were sampled, transmitted, and received (if successfully received). Note that oblivious policies are independent of the processes they observe and they are therefore less costly to implement. Moreover, they can still benefit from the channel collision feedback to (i) quantify how stale the information at the receiver has become (in order to decide when to sample and communicate) and (ii) adapt to the channel state (for communication purposes). In this sub-section, we show that minimizing NAEE in the class of oblivious policies is equivalent to minimizing the normalized average sum of AoI (NAAoI) as we have previously defined in [37].

Now, based on (3.17), we establish the following relationship between the expected estimation error and the expected age.

Lemma 5. *In oblivious policies, the expected estimation error associated with process i has the following relationship with the expected age function:*

$$\mathbb{E}[\big(X_i(k) - \hat{X}_i(k)\big)^2] = \mathbb{E}[h_i(k)]\sigma^2. \tag{3.18}$$

Proof. The proof of Lemma 5 is given in Appendix B.4. □

Remark 13. *Lemma 5 does not hold for non-oblivious policies. As a matter of fact, finding*

$\mathbb{E}[\big(X_i(k) - \hat{X}_i(k)\big)^2]$ *in closed-form is non-trivial and its numerical computation can be intractable when* M *is large. The reason is that even though the estimation error is the sum of* $h_i(k)$ *Gaussian noise variables, once we condition on* $h_i(k)$*, their distributions change because* $h_i(k)$ *can be dependent on the process that is being monitored.*

Lemma 5 is reminiscent of [193, Lemma 4]. Using Lemma 5, the metric NAEE in (3.2) can be re-written as follows:

$$L^{\pi}(M) = \lim_{K\to\infty} \sigma^2 J^{\pi}(M) \tag{3.19}$$

where

$$J^{\pi}(M) = \frac{1}{M^2} \sum_{i=1}^{M} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}[h_i^{\pi}(k)]. \tag{3.20}$$

Note that $J^{\pi}(M)$ is only a function of the age function $h_i^{\pi}(k)$. The metric in (3.20) is the NAAoI defined in [37] and, therefore, the decentralized threshold policies of [37] apply directly. Note that the generation rate of packets for every sensor is $\theta \in (0, 1]$ in [37], while θ should be set to 1 in the model defined in Section 3.2. This is because we assume that sensor i can observe the process $\{X_i(k)\}_k$ for every k. In particular, [37, Algorithm 2] outlines a stationary age-based thinning (SAT) policy in which a source transmits only when the corresponding AoI is larger than a predetermined threshold. Using this algorithm, it was shown that the following age performance can be achieved in the limit of large M:

$$\lim_{M\to\infty} J^{\mathrm{SAT}}(M) = \frac{e}{2} \tag{3.21}$$

$$\lim_{M\to\infty} L^{\mathrm{SAT}}(M) = \frac{e}{2}\sigma^2. \tag{3.22}$$

Results from [37, Proposition 1] also lead to the following lower bound on NAAoI $J^{\pi}(M)$ for any

decentralized policy π:

$$\lim_{M\to\infty} J^{\pi}(M) \geq .88. \tag{3.23}$$

Using (3.22) and (3.23), we arrive at the following proposition.

Proposition 4. *The minimum attainable NAEE in the class of oblivious policies is characterized by the following bounds*

$$.88\sigma^2 \leq \lim_{M\to\infty} \bar{L}(M) \leq \frac{e}{2}\sigma^2. \tag{3.24}$$

We compare the SAT policy in [37, Algorithm 2] with an oblivious centralized policy – the Max-Weight (MW) policy [20, 36, 37].

Definition 7. *At the beginning of each slot k, the MW policy chooses the action i^* such that*

$$h_{i^*}(k) = \max_i h_i(k). \tag{3.25}$$

Note that this policy is exactly the MW policy derived in [20] for age minimization. From Lemma 2 in [13, Section III], the policy defined in Definition 7 is optimal.

Proposition 5. *The MW policy in Definition 7 minimizes the one-slot Lyapunov Drift in each slot, and*

$$\lim_{M\to\infty} L^{MW}(M) = \frac{\sigma^2}{2}. \tag{3.26}$$

Proof. The proof of Proposition 5 is given in Appendix B.5. □

Comparing (3.22) with (3.26), we have

$$\lim_{M\to\infty} \frac{L^{SAT}(M)}{L^{MW}(M)} = e.$$

The NAEE of the decentralized SAT policy is e times that of the optimal centralized policy in the limit of large M. The conclusion coincides with one's intuition: the throughput of the decentralized SAT policy in [37] is e^{-1}, while the throughput of the centralized MW policy is 1, which implies the amount of delivered fresh packets in the centralized MW policy is e times that of the decentralized SAT policy. We illustrate their performances through simulations in Section 3.8.

3.4.2. Non-oblivious Policies

We now consider a more general class of policies in which the nodes can observe their corresponding Markov processes for decision-making. In other words, we seek to benefit from not only the AoI but also the process realization (in a casual manner).

Let $\{W_j\}_j$ be an i.i.d sequence with the same distribution as $\{W_j(k)\}_j$. Define

$$S_n = \sum_{j=1}^{n} W_j. \tag{3.27}$$

Using the definition of $h_i(k)$ in (2.1), and by the stationarity of $\{W_j\}_j$, we conclude that the error process in (3.17) has the following distribution,

$$\psi_i(k) \sim |S_{h_i(k)}|. \tag{3.28}$$

Recall that J_β has the same distribution as $J_\ell^{(i)}$. Then, J_β is the smallest time index at which $|S_n| \geq \beta$ in an inter-delivery interval. J_β is a stopping time for S_n. From [191, Chapter 7.5.1, Lemma 7.5.1], it follows that J_β has finite moments of all orders. Moreover, using [191, Chapter 7.5.2], we have

$$\mathbb{E}[S_{J_\beta}^2] = \sigma^2 \mathbb{E}[J_\beta]. \tag{3.29}$$

Finding an optimal β is non-trivial because β impacts both J_β and U_β. In the remainder of this section, we establish some useful expressions for the expectations of I_β and U_β in an optimal design.

3.4.3. The closed form of NAEE

We next derive a closed-form expression for the attained NAEE, $L^{EbT}(M)$. Using (3.28), we re-write (3.2) as follows.

$$L^{EbT}(M) = \lim_{K\to\infty} \mathbb{E}[\frac{1}{M^2K} \sum_{i=1}^{M} \sum_{k=1}^{K} S^2_{h_i(k)}]. \tag{3.30}$$

Define $\Delta_\ell^{(i)}$ as the sum of $S^2_{h_i(k)}$ in the interval $k \in (k^{(i)}_{\ell-1}, k^{(i)}_\ell]$:

$$\Delta_\ell^{(i)} = \sum_{k=k^{(i)}_{\ell-1}+1}^{k^{(i)}_\ell} S^2_{h_i(k)}. \tag{3.31}$$

Since $h_i(k)$ has the same distribution in the interval $[k^{(i)}_{\ell-1}+1, k^{(i)}_\ell]$ over i and ℓ, then $\Delta_\ell^{(i)}$ has the same distribution over i and ℓ. We define Δ_β to have the same distribution as $\Delta_\ell^{(i)}$. The next lemma shows that the expected time average in (3.30) takes a closed form expression in terms of $\mathbb{E}[\Delta_\beta]$ and $\mathbb{E}[I_\beta]$.

Lemma 6. *The proposed EbT policy attains the following NAEE:*

$$L^{EbT}(M) = \frac{1}{M} \frac{\mathbb{E}[\Delta_\beta]}{\mathbb{E}[I_\beta]}. \tag{3.32}$$

Proof. The proof of Lemma 6 is given in Appendix B.6. □

Similar to (3.31), Δ_β can be expressed as

$$\Delta_\beta = \sum_{j=1}^{I_\beta} S_j^2. \tag{3.33}$$

From (3.33), the NAEE in (3.32) can now be re-written as follows

$$L^{EbT}(M) = \frac{1}{M}\frac{\mathbb{E}\Big[\sum_{j=1}^{I_\beta} S_j^2\Big]}{\mathbb{E}[I_\beta]} = \frac{1}{M}\frac{\mathbb{E}\Big[\sum_{j=1}^{J_\beta+U_\beta-1} S_j^2\Big]}{\mathbb{E}[I_\beta]} \triangleq L_1^{EbT}(M) + L_2^{EbT}(M) \tag{3.34}$$

where

$$L_1^{EbT}(M) = \frac{1}{M}\frac{\mathbb{E}\Big[\sum_{j=1}^{J_\beta} S_j^2\Big]}{\mathbb{E}[I_\beta]} \tag{3.35}$$

$$L_2^{EbT}(M) = \frac{1}{M}\frac{\mathbb{E}\Big[\sum_{j=J_\beta+1}^{J_\beta+U_\beta-1} S_j^2\Big]}{\mathbb{E}[I_\beta]} = \frac{1}{M}\frac{2\mathbb{E}[J_\beta](\mathbb{E}[U_\beta]-1) + \mathbb{E}[U_\beta^2] - \mathbb{E}[U_\beta]}{2\mathbb{E}[I_\beta]}\sigma^2. \tag{3.36}$$

The equality in (3.36) is proved in Appendix B.7. Note that L^{EbT} is a function of the peak age I_β, the silence delay J_β, the transmission delay U_β, and the process realization through W_j.

3.4.4. Optimizing β Approximately

Finally, we find approximate closed-form expressions for $L_1^{EbT}(M)$ and $L_2^{EbT}(M)$. Let M be sufficiently large. Using (3.16) along with the the fact that $\mathbb{E}[J_\beta] \leq \mathbb{E}[I_\beta]$, one can simplify (3.36) in the limit of large M:

$$L_2^{EbT}(M) = \frac{1}{M}\cdot\frac{\mathbb{E}[U_\beta^2]}{2\mathbb{E}[I_\beta]}\sigma^2. \tag{3.37}$$

The following lemma comes in handy in our approximations.

Lemma 7. *Consider a Brownian motion B_t. Define $J = \inf\{t \geq 0, |B_t| \geq a\}$. The following holds:*

(1) [194, Chapter 7, Theorem 7.5.5, Theorem 7.5.9] $\mathbb{E}[J] = a^2$ and $\mathbb{E}[J^2] = \frac{5a^4}{3}$;

(2) $\mathbb{E}[\int_0^J B_t^2 dt] = \frac{1}{10}\mathbb{E}[J^2] = \frac{1}{6}a^4$.

Proof. The proof of Lemma 7 is given in Appendix B.8. □

For any j, $\frac{S_j}{\sigma}$ is Gaussian with mean zero and variance j. We propose to use B_j as an approximation

of $\frac{S_j}{\sigma}$. Letting $a = \beta/\sigma$ in Lemma 7, we obtain

$$\mathbb{E}\big[J_\beta] \approx \frac{\beta^2}{\sigma^2},\ \mathbb{E}[J_\beta^2] \approx \frac{5\beta^4}{3\sigma^4} \tag{3.38}$$

$$\mathbb{E}\big[\sum_{j=1}^{J_\beta} S_j^2\big] \approx \frac{\beta^4}{6\sigma^2} \approx \frac{1}{10}\mathbb{E}[J_\beta^2]. \tag{3.39}$$

The approximation error analysis is provided in Section 3.4.5.

Substituting (3.39) into (3.34), we find the following approximation for L^{EbT}:

$$\hat{L}^{EbT}(M) = \frac{\frac{1}{5}\mathbb{E}[J_\beta^2] + \mathbb{E}[U_\beta^2]}{2M\mathbb{E}[I_\beta]}\sigma^2. \tag{3.40}$$

Theorem 6. *Let M be sufficiently large. The optimal β^* is approximately given by*

$$\beta^* \approx \hat{\beta} = \sigma\sqrt{eM},$$

and

$$\hat{L}^{EbT} = \frac{e}{6}\sigma^2. \tag{3.41}$$

Proof. The detailed proof of Theorem 6 is given in Appendix B.9. Here, we only provide a roadmap of the proof. **(i)** After simplifying $\hat{L}^{EbT}(M)$ in (3.40) by using (3.10), (3.15), (3.16), (3.38), we find $\hat{L}^{EbT}(M) \approx \frac{\frac{M^2}{c(M)^2} - \frac{2\beta^4}{3\sigma^4} + Var(U_\beta)}{2\frac{M^2}{c(M)}}\sigma^2$. **(ii)** We show that the term $\frac{Var(U_{\beta^*})}{2\frac{M^2}{c(M)}}$ is negligible. **(iii)** We derive $\beta^* \approx \hat{\beta} = \sigma\sqrt{\frac{M}{c(M)}}$ as an (approximate) minimizer of NAEE. This leads to $\hat{L}^{EbT} = \frac{1}{6c(M)}\sigma^2$.

□

Finally, Assumptions 1, 2 are verified (approximately) for β^* when M is sufficiently large in Appendix B.10.

It is interesting to compare the performance of the proposed EbT policy with the oblivious decen-

tralized and centralized policies of Section 3.4.1. From (3.19), (3.20), and (3.21),

$$\lim_{M\to\infty} L^{SAT}(M) = \frac{e}{2}\sigma^2.$$

using (3.22) and (3.41), we obtain

$$\lim_{M\to\infty} \frac{L^{SAT}(M)}{\hat{L}^{EbT}(M)} \approx 3. \tag{3.42}$$

The NAEE of the oblivious SAT policy is around three times that of the EbT policy. From (3.24), the NAEE of the oblivious MW policy of Section 3.4.1 is asymptotic $\frac{\sigma^2}{2}$ and comparing with $\frac{e}{6}\sigma^2 = 0.455\sigma^2$ one concludes that the NAEE of the EbT policy is close to that of the oblivious MW policy. We remark that since $\hat{L}^{EbT}(M)$ is an estimate of $L^{EbT}(M)$, these comparisons are not exact. We will also compare the numerical performance of Algorithm 4 with oblivious policies as well as other state-of-the-art algorithms in Section 3.8. Algorithm 4 below summarizes the proposed decentralized error-based transmission policy.

Algorithm 4 Error-based Thinning (EbT)

Set the time horizon K.
Set $h_i(0) = 1$, $X_i(0) = \hat{X}_i(0) = 0$ for $1 \le i \le M$; $c(0) = \hat{N}(0) = 0$; $p_b(0) = k = 1$. Set $\beta^* = \sigma\sqrt{eM}$.
repeat
Step 1: For each node i, observe the collision feedback $c(k-1)$ at the end of time slot $k-1$, and update $k_\ell^{(i)}$'s and $\hat{X}_i(k)$, respectively.
Step 2: For each node i, observe $X_i(k)$ $\big($which evolves according to (3.1)$\big)$ and compute $\psi_i(k)$ by (3.7).
Step 3: If $\psi_i(k) < \beta^*$, then node i keeps silent; otherwise it transmits a packet with probability $p_b(k)$.
Step 4: Calculate $p_b(k)$ by (3.8) in which $\lambda(k) = e^{-1}$.
until $k = K$
Calculate

$$L_K^{EbT} = \frac{1}{M^2}\sum_{i=1}^{M}\frac{1}{K}\sum_{k=0}^{K}\psi_i^2(k).$$

3.4.5. Approximation Error Analysis

Note that approximations are used in (3.38) and (3.39), now we analyze the approximation error in terms of σ^2. The approximation error of L^{EbT} consists of (i) the approximation error in (3.38) and (ii) the approximation error in (3.39), both of which are incurred when approximating an autoregressive Markov process with a Wiener process. In other words, the approximation error is due to the discretization of the Wiener process. This discretization is analyzed by the Langevin dynamics in [195]. In particular, $\frac{S_n}{\sigma} = \sum_{i=1}^{n} W_i \approx B_n$ can be regarded as an overdamped Langevin dynamics with step size 1 to approximate the Brownian motion. The approximation error in each step remains constant due to the unit step size.

We first consider $\mathbb{E}[J_\beta]$. Substituting $\beta = \sigma\sqrt{eM}$ into $a = \beta/\sigma$ in Lemma 7, we find $a = \sqrt{eM}$ is constant. So the distribution of J in Lemma 7 does not change when σ changes. Thus, the approximation error in (3.38) keeps invariant when σ changes.

Then, we consider (3.39). J_β is an approximation of J, and

$$\sum_{j=1}^{J_\beta} S_j^2 = \sigma^2 \sum_{j=1}^{J_\beta} S_j^2/\sigma^2. \tag{3.43}$$

The distribution of J does not change with σ, nor does the distribution of J_β. The terms $\frac{S_j}{\sigma} \sim \mathcal{N}(0, j)$ inside the sum in (3.43) are independent of σ. The distribution of $\sum_{j=1}^{J_\beta} S_j^2/\sigma^2$ does not change with σ. Thus, the approximation error in (3.39) increases linearly with σ^2.

By Lemma 7 (2), $\mathbb{E}[J^2] = 10\mathbb{E}[\int_0^T B_t^2 dt]$. Recall that the approximation error in (3.39) increases linearly with σ^2, thus the approximation error in $\mathbb{E}[J^2]$ also increases linearly with σ^2. From (3.40), the approximation error in $L^{EbT}(M)$ increases linearly with σ^2.

3.5. Optimal Threshold in Regime of $\gamma < 1$

In this section, we consider the case when $0 < \gamma < 1$. From the definition of oblivious policies, decision-making is independent of observed processes. Then, the oblivious policies essentially investigate the minimization of age instead of sample values. So, the oblivious policies, in this case,

are the same as those in Section 3.4.1. We only consider the non-oblivious policies, i.e., the EbT policies. Recall that $\{W_j\}_j$ is an i.i.d sequence with the same distribution as $\{W_j(k)\}_j$. We re-define S_n in (3.27) as a general way

$$S_n = \sum_{j=1}^{n} \gamma^{j-1} W_{n-j}. \tag{3.44}$$

For simplicity, define $S_0 = W_0 = 0$. From (3.44), the recursion of S_n can be written as

$$S_n = W_n + \gamma S_{n-1}, \quad n \geq 1. \tag{3.45}$$

Then, from (3.9) and (3.44), we have $\psi_i(k) \sim |S_{h_i(k)}|$. From (3.45), $\{S_n\}_n$ is a stationary $AR(1)$ [196].

Due to the complexity of (3.45), it is complicated to obtain the closed-form of $\mathbb{E}[\sum_{j=1}^{J_\beta} S_j^2]$ or approximations, hence it is complicated to obtain $\hat{L}^{EbT}(M)$. To find (approximate) optimal β^*, we revisit Lemma 3 and Lemma 4, and start with the derivations of (3.14). Under an optimal threshold β^*, from Lemma 3, $\alpha_{\beta*} = \frac{\mathbb{E}[U_{\beta*}]}{\mathbb{E}[I_{\beta*}]}$, and from (3.14), we have

$$(1 - \frac{\mathbb{E}[U_{\beta*}]}{\mathbb{E}[I_{\beta*}]}) \frac{\mathbb{E}[U_{\beta*}]}{\mathbb{E}[I_{\beta*}]} = \frac{\lambda_m}{M},$$

which implies

$$\frac{\mathbb{E}[U_{\beta*}]}{\mathbb{E}[I_{\beta*}]} = \frac{1}{2} \cdot \Big(1 - \sqrt{1 - \frac{4\lambda_m}{M}}\Big), \tag{3.46}$$

and from (3.10), we have $\mathbb{E}[I_{\beta*}] = \mathbb{E}[J_{\beta*}] - 1 + \mathbb{E}[U_{\beta*}]$, hence

$$\mathbb{E}[J_{\beta*}] = \frac{1}{2} \cdot \Big(1 + \sqrt{1 - \frac{4\lambda_m}{M}}\Big) \mathbb{E}[I_{\beta*}] + 1. \tag{3.47}$$

From Lemma 4, $\mathbb{E}[J_{\beta*}]$ has the following closed form

$$\mathbb{E}[J_{\beta*}] = \frac{\left(1+\sqrt{1-\frac{4\lambda_m}{M}}\right)M}{2c(M)} + 1. \tag{3.48}$$

Remark 14. *From* (3.48), $\mathbb{E}[J_\beta]$ *can be represented as a function of channel characteristics, i.e., the maximum effective arrival rate* λ_m *and the channel throughput* $c(M)$*, instead of* β^**.*

The rest task is to find another closed form of $\mathbb{E}[J_\beta]$, which can be written as a function of the threshold β. Recall that J_β is the smallest time index at which $|S_n| \geq \beta$ in an inter-delivery interval. And note that S_n is a stationary $AR(1)$ with $W_n \sim \mathcal{N}(0, \sigma^2)$. From [197, Corollary 1], we have

$$\mathbb{E}[J_\beta] = \frac{1}{|\log\gamma|}\int_0^\infty \big(\mathbb{E}[\cosh(uS_{J_\beta})] - \cosh(uS_0)\big)\exp\Big(-\frac{u^2\sigma^2}{2(1-\gamma^2)}\Big)\cdot\frac{du}{u}, \tag{3.49}$$

where $\cosh(x) = \frac{e^x+e^{-x}}{2}$. Since we consider M is sufficiently large, then from Lemma 4, $\mathbb{E}[J_{\beta^*}]$ is sufficiently large, thus the optimal threshold β^* and $|S_{J_{\beta^*}}|$ are also large. From the definition of J_{β^*}, we have $|S_{J_{\beta^*}}| \geq \beta^*$, then $\frac{|S_{J_{\beta^*}}|}{\beta^*} \approx 1$. Recall that $S_0 = 0$, thus

$$\mathbb{E}[J_{\beta^*}] \approx \frac{1}{|\log\gamma|}\int_0^\infty \big(\cosh(u\beta^*) - 1\big)\exp\Big(-\frac{u^2\sigma^2}{2(1-\gamma^2)}\Big)\cdot\frac{du}{u}. \tag{3.50}$$

Now, we can calculate the optimal threshold approximately by the following theorem.

Theorem 7. *Given any* $0 < \gamma < 1$ *and* σ*, an approximate of the optimal threshold, i.e.,* β^**, is the solution of* (3.51)*,*

$$\int_0^\infty \big(\cosh(u\beta) - 1\big)\cdot\exp\Big(-\frac{u^2\sigma^2}{2(1-\gamma^2)}\Big)\cdot\frac{du}{u} = |\log\gamma|\Big(\frac{\left(1+\sqrt{1-\frac{4\lambda_m}{M}}\right)M}{2c(M)} + 1\Big). \tag{3.51}$$

Remark 15. *Note that* $\cosh(u\beta)$ *increases with* β *for any given* u*. So there exists a unique solution* β^* *in* (3.51)*.*

Proof. Combine with (3.48) and (3.50), we have the desired results. □

Corollary 1. *Given any* $0 < \gamma < 1$ *and* σ*, an approximate of the optimal threshold, i.e.,* β^**, is the solution of* (3.52)*,*

$$\int_0^\infty \big(\cosh(u\beta) - 1\big) \cdot \exp\Big(-\frac{u^2\sigma^2}{2(1-\gamma^2)}\Big) \cdot \frac{du}{u} = |\log\gamma| \Big(\frac{\Big(1+\sqrt{1-\frac{4}{eM}}\Big)eM}{2} + 1\Big). \tag{3.52}$$

Proof. From Appendix B.10, under an optimal β^*, Assumption 2 holds, so we have $\lambda_m \approx e^{-1}$, $c(M) \approx e^{-1}$. Substituting $\lambda_m \approx e^{-1}$, $c(M) \approx e^{-1}$ to (3.51), we have the desired results. □

3.6. Optimal Thereshold in Regime of $\gamma > 1$

Now, we consider the case where the processes are explosive, i.e., $\gamma > 1$. Similar to Section 3.5, we only consider the non-oblivious policies, i.e., the EbT policies. This is because the oblivious policies essentially investigate the minimization of age instead of sample values. So for the oblivious policies, we can use the results in Section 3.4.1 directly. From (3.44) and (3.45), $\{S_n\}_n$ is an explosive $AR(1)$ sequence when $\gamma > 1$, and there is no closed form for $\mathbb{E}[J_\beta]$ [198]. In this section, we find a numerical calculative method for $\mathbb{E}[J_\beta]$.

Firstly, we introduce a joint distribution. For any given n, the joint cumulative distribution function of $(S_1, S_2, \cdots, S_n)$, denoted by $\mathcal{P}(n, \beta, \sigma^2\Sigma_n)$, is

$$\begin{aligned} &\mathcal{P}(n, \beta, \sigma^2\Sigma_n) \\ &= \int_{O_n} \frac{1}{(2\pi)^{n/2}|\sigma^2\Sigma_n|} \exp\Big\{-\frac{1}{2}x^T(\sigma^2\Sigma_n)^{-1}x\Big\}dx, \end{aligned} \tag{3.53}$$

where x is a n-dimension vector, $O_n = (-\infty, \beta)^n$, and

$$\Sigma_n = \begin{bmatrix} 1 & \gamma & \cdots & \gamma^{n-1} \\ \gamma & 1 & \cdots & \gamma^{n-2} \\ \vdots & \vdots & \cdots & \vdots \\ \gamma^{n-1} & \gamma^{n-2} & \cdots & 1 \end{bmatrix}. \tag{3.54}$$

is the covariance matrix of $(S_1, S_2, \cdots, S_n)$. Then, the probability for $\{J_\beta = t\}$ can be calculated

as

$$\begin{aligned}\Pr\left(J_\beta = t\right) =& \Pr\{|S_1| < \beta, \cdots, |S_{t-1}| < \beta, |S_t| \geq \beta\} \\ =& \mathcal{P}(t-1, \beta, \sigma^2\Sigma_{t-1}) - \mathcal{P}(t, \beta, \sigma^2\Sigma_t).\end{aligned} \tag{3.55}$$

Note that Σ_n in (3.54) is not necessary positive definite when $\gamma > 1$, and $\mathcal{P}(n, \beta, \sigma^2\Sigma_n)$ in (3.53) can not even be calculated numerically when n is large. Therefore, we need to find an estimate of (3.55).

Now, we consider the random variable $\gamma^{-t}S_t$. From [198, Lemma 1], the limit of random variable $\gamma^{-t}S_t$ exists, and

$$\gamma^{-t}S_t \to L_0, \quad t \to \infty \text{ a.s.}, \tag{3.56}$$

where

$$L_0 = \sum_{i=1}^{\infty} \gamma^{-i} W_i \sim \mathcal{N}(0, \frac{\gamma^{-2}\sigma^2}{1-\gamma^{-2}}). \tag{3.57}$$

From (3.56), define $\tilde{S}_t = \gamma^t L_0$, then $S_t \to \tilde{S}_t$ a.s. when $t \to \infty$. We argue that given t is sufficiently large, $\Pr(J_\beta = t)$ in (3.55) can be approximated by,

$$\Pr(J_\beta = t) \approx 2\Big(\Phi\big(\frac{\beta\sqrt{1-\gamma^{-2}}}{\gamma^{t-1}\cdot(\sigma/\gamma)}\big) - \Phi\big(\frac{\beta\sqrt{1-\gamma^{-2}}}{\gamma^{t}\cdot(\sigma/\gamma)}\big)\Big). \tag{3.58}$$

In fact, since $S_t \to \tilde{S}_t$ a.s. when $t \to \infty$, then when t is sufficiently large, from (3.55),

$$\Pr\left(J_\beta = t\right) \approx \Pr\{|\tilde{S}_1| < \beta, \cdots, |\tilde{S}_{t-1}| < \beta, |\tilde{S}_t| \geq \beta\}. \tag{3.59}$$

For any given t, when β is sufficiently large, $\Pr(|\tilde{S}_t| \leq \beta) \approx 1$, and

$$\Pr\left(J_\beta = t\right) \approx \Pr\{|\tilde{S}_{t-1}| < \beta, |\tilde{S}_t| \geq \beta\}. \tag{3.60}$$

Note that $\tilde{S}_t = \gamma \tilde{S}_{t-1}$,

$$\begin{aligned}\Pr\left(J_\beta = t\right) &\approx 2\Pr\{\frac{\beta}{\gamma^t} \le L_0 < \frac{\beta}{\gamma^{t-1}}\} \\ &= 2\Big(\Phi\big(\frac{\beta\sqrt{1-\gamma^{-2}}}{\gamma^{t-1}\cdot(\sigma/\gamma)}\big) - \Phi\big(\frac{\beta\sqrt{1-\gamma^{-2}}}{\gamma^{t}\cdot(\sigma/\gamma)}\big)\Big).\end{aligned}$$

Note that we consider M is sufficiently large, then from Lemma 4, $\mathbb{E}[J_{\beta^*}]$ is sufficiently large, thus the optimal threshold β^* and $|S_{J_{\beta^*}}|$ are also large. To find the smallest index $J_{\beta*}$ such that $|S_{J_{\beta^*}}| \ge \beta^*$, it is approximately to have the following theorem.

Theorem 8. *Given any $0 < \gamma < 1$ and σ, an approximate of the optimal threshold, i.e., β^*, is the solution of* (3.61),

$$\sum_{t=1}^{\infty} 2t\Big(\Phi\big(\frac{\beta\sqrt{1-\gamma^{-2}}}{\gamma^{t-1}\cdot(\sigma/\gamma)}\big) - \Phi\big(\frac{\beta\sqrt{1-\gamma^{-2}}}{\gamma^{t}\cdot(\sigma/\gamma)}\big)\Big) = \frac{\Big(1+\sqrt{1-\frac{4\lambda_m}{M}}\Big)M}{2c(M)} + 1. \tag{3.61}$$

Remark 16. *The approximation in* (3.59) *holds when t is sufficiently large, and the approximation in* (3.60) *holds when β is sufficiently large. Thus, Theorem 8 is valid in a system with a huge number of sensors in a long time horizon.*

Proof. From (3.58), $\mathbb{E}[J_\beta] \approx \sum_{t=1}^{\infty} 2t\Big(\Phi\big(\frac{\beta\sqrt{1-\gamma^{-2}}}{\gamma^{t-1}\cdot(\sigma/\gamma)}\big) - \Phi\big(\frac{\beta\sqrt{1-\gamma^{-2}}}{\gamma^{t}\cdot(\sigma/\gamma)}\big)\Big)$. Combine with (3.48), we have the desired results. □

Corollary 2. *Given any $0 < \gamma < 1$ and σ, an approximate of the optimal threshold, i.e., β^*, is the solution of* (3.62),

$$\begin{aligned}&\sum_{t=1}^{\infty} 2t\Big(\Phi\big(\frac{\beta\sqrt{1-\gamma^{-2}}}{\gamma^{t-1}\cdot(\sigma/\gamma)}\big) - \Phi\big(\frac{\beta\sqrt{1-\gamma^{-2}}}{\gamma^{t}\cdot(\sigma/\gamma)}\big)\Big) \\ &= \frac{\Big(1+\sqrt{1-\frac{4}{eM}}\Big)eM}{2} + 1.\end{aligned} \tag{3.62}$$

Proof. From Appendix B.10, under an optimal β, Assumption 2 holds, so we have $\lambda_m \approx e^{-1}$, $c(M) \approx e^{-1}$. Substituting $\lambda_m \approx e^{-1}$, $c(M) \approx e^{-1}$ to (3.61), we have the desired results. □

3.7. Unreliable Random Access Channels

Finally, we generalize our model to account for unreliability in random access channels, i.e., erasure channels. Related works such as [199, 200] investigated Age of Information in unreliable channels, while optimal power allocation strategies in unreliable channels with respect to remote estimation have been considered in [201]. However, in this section, we aim to minimize NAEE defined in (3.2) under oblivious and non-oblivious policies in unreliable channels.

In the model defined in Section 3.2, sensors can deliver packets successfully if no collisions happen. Now, we assume that packets are erased with some probability even if no collisions happen in the channel. In particular, suppose that if the channel is not in collision, the packet can be delivered with probability $1-\epsilon$, where ϵ is the channel erasure probability. We do not introduce another feedback, i.e., we assume that only collision feedback (not the full feedback) can be transmitted to sensors. Same as Section 3.2, (active) sensors transmit packets through the slotted ALOHA algorithm (3.8). For clarity of exposition, we assume that packet erasure happens at the end of every time slot (after channel collisions).

From Section 3.3, in the limit of M, the channel throughput/rate is around e^{-1} when $\epsilon = 0$. Now, note that the channel erasure probability is ϵ, which implies when no collisions occur, every packet chosen by slotted ALOHA (3.8) is delivered with probability $1-\epsilon$. Thus, the throughput/rate is around $e^{-1}(1-\epsilon)$. In this section, we let $c(M) = e^{-1}(1-\epsilon)$. In the rest of the section, we find the optimal threshold in two regimes: a) $\gamma = 1$, and b) $\gamma \neq 1$.

$\gamma = 1$ Similar to Section 3.4, we first consider oblivious schemes. By a proof similar to that of Lemma 5, (3.18) and (3.19) still hold in our unreliable random access setting. Since the channel throughput/rate is around $e^{-1}(1-\epsilon)$, we use [37, Theorem 5] to obtain the following NAAoI:

$$\lim_{M\to\infty} J_{\epsilon}^{SAT}(M) = \frac{e}{2(1-\epsilon)}. \tag{3.63}$$

Note that (3.18) and (3.19) still hold in erasure channels. Thus, the normalized average estimation error is computed by

$$\lim_{M\to\infty} L_\epsilon^{SAT}(M) = \lim_{M\to\infty} L_\epsilon^{SAT}(M)\sigma^2 = \frac{e\sigma^2}{2(1-\epsilon)}. \tag{3.64}$$

Now, we consider non-oblivious schemes. The analysis in Section 3.3 can be generalized to yield the following theorem.

Theorem 9. *Let M be sufficiently large. An optimal β_ϵ^* is approximately given by*

$$\beta_\epsilon^* \approx \sigma\sqrt{eM/(1-\epsilon)},$$

and

$$\hat{L}_\epsilon^{EbT} = \frac{e}{6(1-\epsilon)}\sigma^2. \tag{3.65}$$

Remark 17. *The new threshold in Theorem 9 is larger than that in Theorem 6, i.e., $\beta_\epsilon^* \geq \beta^*$ for $0 \leq \epsilon < 1$. The expected number of newly active nodes is reduced. This is because (i) if a packet is erased, then the corresponding sensor is still active in the next time slot; (ii) the channel throughput/rate decreases to around $e^{-1}(1-\epsilon)$, not e^{-1}.*

Remark 18. *Comparing* (3.64) *and* (3.65)*, we still have $L_\epsilon^{SAT}/\hat{L}_\epsilon^{EbT} \approx 3$ in erasure channels.*

Proof. The proof of Theorem 9 is similar to that of Theorem 6. The only difference is replacing $c(M) \approx e^{-1}$ with $c(M) \approx e^{-1}(1-\epsilon)$. □

$\gamma \neq 1$ To find the approximate optimal β^* when $\gamma \neq 1$ on unreliable random access channels, one can straighforwardly submit $\lambda_m = e^{-1}$ and $c(M) = e^{-1}(1-\epsilon)$ to Theorem 7 (for the case when $0 < \gamma < 1$) and Theorem 8 (for the case when $\gamma > 1$), respectively.

3.8. Numerical Results

In this section, we verify our findings through simulations. The numerical results in the case when $\gamma = 1$, when $\gamma \neq 1$, and the unreliable random access channels are given in In Section 3.8.1, Section 3.8.2, and Section 3.8.3, respectively.

3.8.1. When $\gamma = 1$

Figure 3.2 compares the NAEE of our proposed policy with the state of the art for $M = 500$ under different σ^2. In this plot, the green (plus) curve corresponds to an optimal stationary randomized policy in which each node transmits with an optimal pre-determined probability. The performance of threshold policies like [33, 56] that impose an optimal (fixed) transmission rate for each sensor also coincides with this curve, i.e., the green (plus) one. These policies do not exploit the available feedback for decision-making. The purple (diamond) curve shows the performance of a standard pseudo-Bayesian slotted ALOHA. Slotted ALOHA does use feedback, but treats all packets similarly, independent of their corresponding sample values. The red (circle) and blue (squared) curves correspond to oblivious (age-based) policies [37, Algorithm 1] and [37, Algorithm 2], respectively. The black (star) curve shows the performance of our proposed decentralized policy in Algorithm 2 and the red (x) curve shows the approximation we find in (3.41). The gap between the two is small but increases linearly in σ^2 as discussed in Section 3.4.5. In this plot, we have also included an oblivious and a non-oblivious centralized policy. The former (green dashed curve) schedules the transmitter with the largest age Proposition 5 is optimal in the class of oblivious policies. The centralized non-oblivious policy that we have considered here (yellow smooth curve) schedules the transmitter with the largest estimation error. Both centralized oblivious and non-oblivious policies are often observed to be numerically very close to the optimal.

The numerical calculation and analytical approximation of $\mathbb{E}[J_\beta]$, $\mathbb{E}[\sum_{j=1}^{J_\beta} S_j^2]$ and $\mathbb{E}[U_\beta]$ are given in Figure 3.3a, Figure 3.3b and Figure 3.4a, respectively. Recall that $\mathbb{E}[J_\beta^2]$ is 10 times $\mathbb{E}\big[\sum_{j=1}^{J_\beta} S_j^2\big]$, so we only consider one of them. In order to offset the effect introduced by the number of nodes, we consider the normalized silence delay $\mathbb{E}[J_\beta]/M$, the normalized transmission delay $\mathbb{E}[U_\beta]/M$, and $\mathbb{E}[\sum_{j=1}^{J_\beta} S_j^2]/M$. The estimation error of the normalized silence delay $\mathbb{E}[J_\beta]/M$ is invariant of σ^2

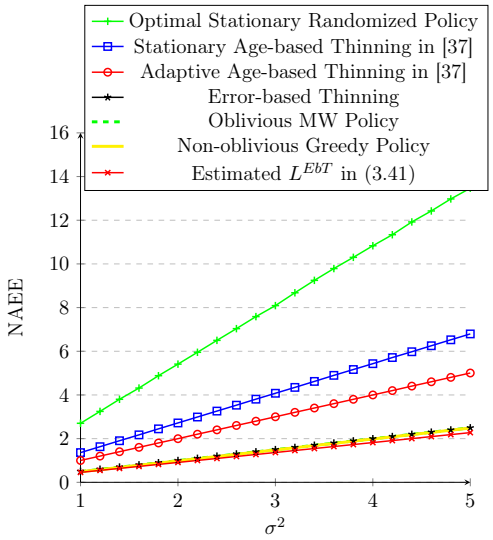
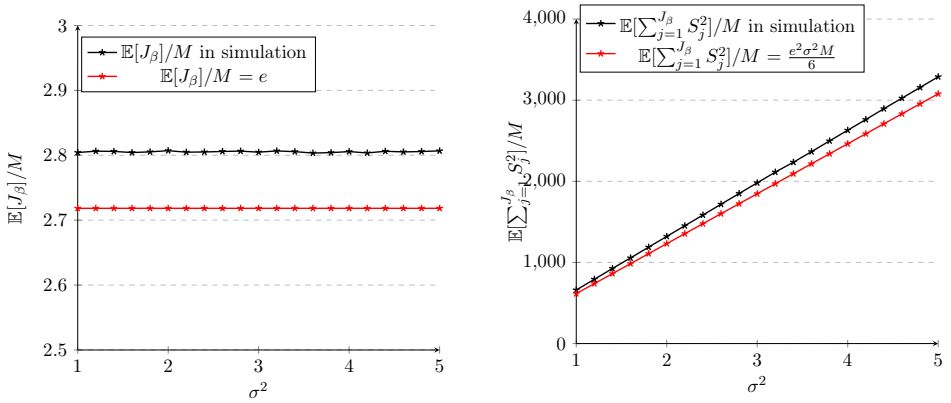


Figure 3.2: NAEE as a function of σ^2 for various state-of-the-art schemes with $M = 500$.

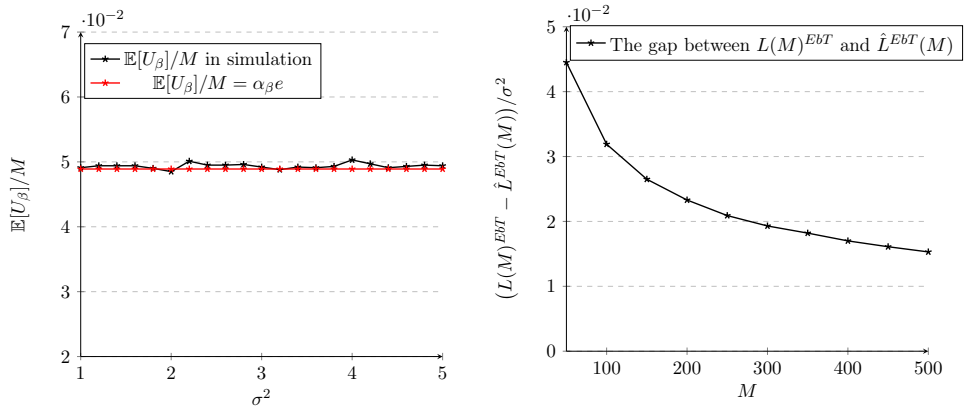


(a) $\mathbb{E}[J_\beta]/M$ as a function of σ^2. (b) $\mathbb{E}[\sum_{j=1}^{J_\beta} S_j^2]/M$ as a function of σ^2.

Figure 3.3: Simulated and analytical $\mathbb{E}[J_\beta]/M$ and $\mathbb{E}[\sum_{j=1}^{J_\beta} S_j^2]/M$.

(Figure 3.3a), while the estimation error of $\mathbb{E}[\sum_{j=1}^{J_\beta} S_j^2]/M$ increases linearly with σ^2 (Figure 3.3b). This coincides with the analysis in Section 3.4.5. In the simulation, we numerically find the expected fraction of active nodes to be $\alpha_\beta = 0.0173$. Substituting $\alpha_\beta = 0.0173$ into (3.16), we get $\mathbb{E}[U_\beta]$. From Figure 3.4a, we can see that normalized transmission delay $\mathbb{E}[U_\beta]$ coincides with analytical results in (3.16).

Next, we show in Figure 3.4b that the gap between $L^{EbT}(M)$ and $\hat{L}^{EbT}(M)$ decreases as M gets large. In other words, the influence of approximation error caused by Langevin dynamics in Algo-



(a) $\mathbb{E}[U_\beta]/M$ as a function of σ^2. (b) $\frac{L^{EbT}(M)-\hat{L}^{EbT}(M)}{\sigma^2}$ as a function of M.

Figure 3.4: Simulated and analytical $\mathbb{E}[U_\beta]/M$ and $\frac{L^{EbT}(M)-\hat{L}^{EbT}(M)}{\sigma^2}$.

rithm 2 weakens (but does not vanish) as M increases.

3.8.2. When $\gamma \neq 1$

In the region $0 < \gamma < 1$, we set $\gamma = 0.999$, while in the region $\gamma > 1$, we set $\gamma = 1.001$. Note that the closed from of β^* can not be obtained from (3.52) or (3.62), so we provide the numerical results for β^* in Table 3.1 ($\gamma = 0.999$) and Table 3.2 ($\gamma = 1.001$).

In Table 3.1, the channel throughput is close to but still less than e^{-1}. This is because we use two approximations in (3.50) and (3.51) to find β^*. Although approximations are utilized, the simulated $\mathbb{E}[J_\beta]$ is close to the right-hand side of (3.52), so the approximations are accurate and the relative error is less than 3%.

When $0 < \gamma < 1$, the comparison of NAEE among different policies is given in Figure 3.5a. Our proposed policy outperforms the decentralized state-of-the-art. These trends are similar to those in Section 3.8.1. It is interesting to observe that the EbT policy (decentralized scheme) performs better than the centralized oblivious MW policy. In this case, the AoI can not be a good representative of error. From (3.9), the previous error is compacted by the constant γ every time slot, which implies a large AoI may not result in a large error.

In Table 3.2, the channel throughput is less than e^{-1}, and relative errors between the simulated

$\mathbb{E}[J_\beta]$ and the right-hand side of (3.52) are larger than those in the case $0 < \gamma < 1$. As we mentioned in Remark 16, this is because the approximation in (3.60) holds only when t is relatively large, and it is not very accurate when t is small.

When $\gamma > 1$, from Figure 3.5b, our proposed policy outforms the decentralized state-of-the-art, i.e., SAT in [37]. In the figure, we do not plot the performance of the optimal stationary randomized policy, because the NAEE is explosive when the time horizon is large ($> 10^4$). Opposite to the case $0 < \gamma < 1$, the centralized oblivious MW policy performs better than the EbT policy. Note that the error in (3.9) is expanded by the constant γ in every time slot, so a large AoI results in a large error with a high probability.

At the end of the subsection, we investigate the impact of γ on the improvements of non-oblivious policies over oblivious policies. Specifically, we consider $\frac{L^{SAT}(M)}{L^{EbT}(M)}$ as a function of γ. From Table 3.3, we can see that $\frac{L^{SAT}(M)}{L^{EbT}(M)}$ increases with γ, which implies that the non-oblivious policies (EbT policies) provide more benefits in terms of the NAEE comparing to oblivious policies (SAT policies) when γ increases. Note that we consider $\frac{L^{SAT}(M)}{L^{EbT}(M)}$ instead of $\frac{L^{SAT}(M)}{\tilde{L}^{EbT}(M)}$, then $\frac{L^{SAT}(M)}{L^{EbT}(M)} \neq 3$ when $\gamma = 1$.

σ^2	1	1.5	2	2.5	3	3.5	4	4.5	5
β^*	30.9	37.9	43.8	48.9	53.6	57.9	61.9	65.7	69.2
throughput	0.349	0.347	0.347	0.346	0.345	0.345	0.350	0.347	0.350
simulated $\mathbb{E}[J_\beta]$	1386.0	1393.8	1394.9	1401.9	1404.4	1407.1	1384.2	1400.2	1383.1
error from (3.52)	1.9%	2.5%	2.6%	3.0%	3.2%	3.3%	1.8%	2.7%	1.7%

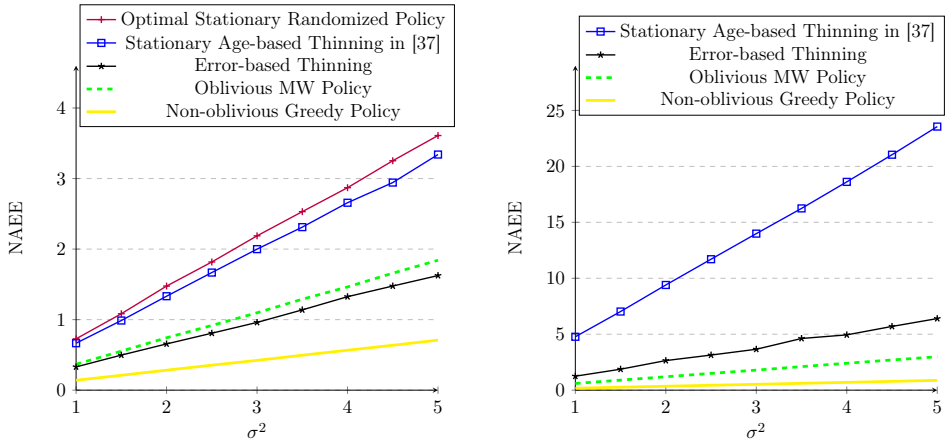
Table 3.1: Approximate optimal β^*, the channel throughput, and the simulated silent delay under different σ^2 when $\gamma = 0.999$.

σ^2	1	1.5	2	2.5	3	3.5	4	4.5	5
β^*	45.8	56.0	64.7	72.4	79.3	85.6	91.5	97.1	102.3
throughput	0.315	0.315	0.314	0.320	0.316	0.314	0.316	0.314	0.314
simulated $\mathbb{E}[J_\beta]$	1194.8	1190.2	1196.2	1199.5	1195.4	1188.8	1192.7	1193.6	1193.3
error from (3.52)	13.3%	13.9%	13.3%	13.1%	13.4%	14.0%	13.6%	13.6%	13.6%

Table 3.2: Approximate optimal β^*, the channel throughput, and the simulated silent delay under different σ^2 when $\gamma = 1.001$.

γ	0.995	0.996	0.997	0.998	0.999	1	1.001	1.002
$\sigma^2 = 1$	1.347	1.413	1.512	1.686	2.018	2.725	3.824	28.676
$\sigma^2 = 2$	1.352	1.401	1.515	1.712	2.032	2.724	3.555	26.060
$\sigma^2 = 3$	1.344	1.417	1.501	1.685	2.278	2.725	3.827	31.184

Table 3.3: $L^{SAT}(M)/L^{EbT}(M)$ v.s. γ when $M = 500$.



(a) NAEE as a function of σ^2 with $M = 500$ when $\lambda = 0.999$.

(b) NAEE as a function of σ^2 with $M = 500$ when $\lambda = 1.001$.

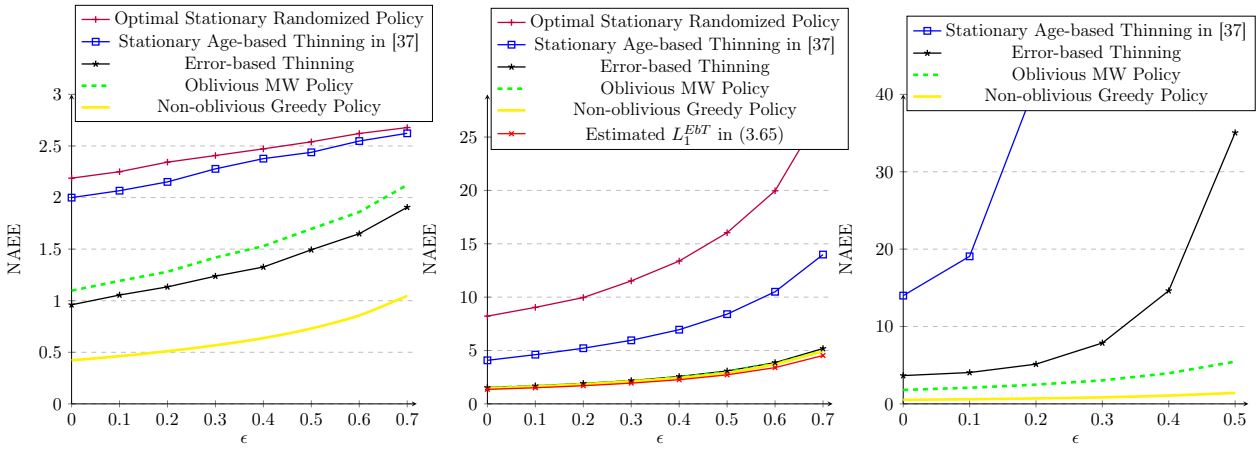
Figure 3.5: NAEE as a function of σ^2 for various state-of-the-art schemes under different γ.

3.8.3. Unreliable Random Access Channels

In this section, we show the performances of different policies in unreliable random access channels with erasure probability ϵ. Let $\sigma^2 = 3$. Figures 3.6a - 3.6c compares the NAEE of our proposed policy with the state of the art for $M = 500$ under different ϵ, γ. It is obvious to see that the EbT policy outperforms all other decentralized transmission mechanisms. In different regimes of γ, the NAEE increases with ϵ under any policy. This is because more packets are erased when the channel erasure probability is larger, hence a larger estimation error occurs.

3.9. Conclusion and Future Research

We considered the problem of real-time sampling and timely estimation over wireless collision channels with M independent and statistically identical first-order autoregressive processes (sources). We studied a normalized metric of estimation error which we termed the normalized average estimation error (NAEE), and focused on the regime of large M. We proposed two general classes of policies:



(a) $\lambda = 0.999$. (b) $\lambda = 1$. (c) $\lambda = 1.001$.

Figure 3.6: NAEE as a function of ϵ for various state-of-the-art schemes in reliable and unreliable channels under different γ.

oblivious policies and non-oblivious policies. When the Gauss-Markov processes are reduced to random walk processes, we showed that, in the former class of policies, minimizing the expected estimation error is equivalent to minimizing the expected age and consequently provided lower and upper bounds on the optimal estimation error. We then proposed and analyzed a (non-oblivious) threshold policy in which (1) nodes become active if their estimation error has crossed a threshold and (2) active nodes transmit stochastically with probabilities that adapt to the state of the channel (exploiting the collision feedback). Subsequently, we showed that the NAEE performance of oblivious (age-based) policies is at least twice better than the state-of-the-art schemes (which impose a fixed rate of transmission at the nodes) such as standard slotted ALOHA and optimal stationary randomized policy. Moreover, our proposed threshold policy offers a multiplicative gain of close to 3 compared to oblivious policies. When the processes are stationary or explosive, numerical optimal thresholds are obtained and the optimal threshold policies outperform all other decentralized schemes. In particular, when the processes are stationary, the optimal threshold policies outperform the centralized oblivious age-based ones. We observe numerically that the multiplicative gain offered by our proposed threshold policy increases with γ.

Then, we extended our framework to incorporate unreliable random access channels with erasure probability ϵ. The NAEE increases with ϵ no matter whether the processes are stationary, random

walk, or explosive. Specifically, under random walk processes, numerical results show that the multiplicative gain is 3 and independent of ϵ (which is consistent with Remark 18). Our findings suggest that designing optimal multiple access systems for future IoT and CPS applications requires going beyond traditional metrics of rate, reliability, latency, and age.

Future research includes generalizations to accommodate the following scenarios: 1) dynamic networks, i.e., the number of sensors changes with time; 2) asymmetric networks, i.e., the sensors are no longer statistically identical; 3) adaptive error-based thinning policies, i.e., the threshold $\beta(k)$ changes with time k; 4) correlated sources, i.e., sensors are no longer mutually independent. For the first scenario, we can simply replace M with $M(k)$ in every time slot. Subsequently, the error-based threshold is also a time-variant variable, $\beta(k)$. For the remaining three scenarios, the method we have proposed can not be applied directly. In particular, in the second scenario, we used the profile of all the sources to find an estimate on an individual source. This step fails in asymmetric settings. In the third scenario, the nodes need statistical inference about the distribution of error process $\{\psi_i(k)\}_i$ to decide which ones are of priority. In the fourth scenario, the policies should change to account for the correlation between the observations.

CHAPTER 4

Decentralized Machenisms in Ad-hoc Networks

In this chapter, we extend the problem of real-time sampling and estimation in random access channels to ad-hoc networks. There are M sources, and each source observes a physical process and is able to sample and communicate with other agents for timely estimation. We seek to design near-optimal *decentralized* sampling and communication strategies. Towards understanding the problem of real-time sampling and estimation under decentralized strategies, recent works have proposed decentralized multiple access schemes to minimize the AoI and estimation error [34, 37, 57]. Nonetheless, in these works, due to the essence of random access channels, sources do not communicate with each other but only communicate with the receiver. Network topologies indeed affect the transmission policies in ad-hoc networks. For example, consider a source with a large degree, i.e., it has a lot of neighboring sources. The transmission policy for this source is different from others even if all the sources are identical. This is because collisions with respect to it happen with high probability, which results in failure of receiving fresh/useful information from other sources. It has remained open to how such designs can inform near-optimal designs in the setting where network topology may affect the transmission policies.

4.1. Literature Review

Multi-agent reinforcement learning: The analytical solutions are not tractable when network topologies are incorporated. Thus, we use techniques from learning fields. Multi-agent reinforcement learning (MARL) algorithms have shown success in many applications, which use reinforcement learning (RL) techniques to co-train a set of agents in a multi-agent system [58]. MARL algorithms are utilized to train a set of agents simultaneously to achieve cooperative or uncooperative goals. A plethora of new MARL algorithms integrating deep learning techniques have appeared in recent years [58, 59, 60]. Generally speaking, MARL algorithms can be divided into three classes [58], (i) independent learning, each agent is trained by RL algorithms without consideration of the multi-agent structure [61]; (ii) centralized multi-agent policy gradient [62, 63]; (iii) value decom-

position algorithms [64, 65]. The last two follow the Centralised Training Decentralised Execution (CTDE) paradigm. MARL has recently witnessed successfully solving sequential decision-making problems in many wireless applications [66], such as link scheduling [67], user scheduling [68], resource management [69], power allocation [70], edge computing [71], fault-tolerant tracking control [72], anti-jamming [73], edge caching [74], and cellular offloading [75]. To our best knowledge, there is only one (weakly) related paper [76], which provided a modification of the Trust Region Policy Optimization (TRPO) algorithm, and frame problems as event-driven decision processes where agents make decisions asynchronously. However, in our setting, agents make decisions to transmit packets or not in every time slot, and the policies are not event-driven decision processes.

Graph Recurrent Neural Networks: Another difficulty that arises due to network topology is the input permutations. For example, consider two homogeneous agents and their actions and observations are A and B, respectively. Then, both AB and BA are valid and equivalent environments. Using a permuted input in classical deep nets will result in a different output [77]. To avoid this practical issue, we apply graph neural networks (GNNs) [78, 79, 80, 81], which have the permutation invariance property, rather than standard neural networks in MARL techniques. GNNs are a popular information processing architecture in graph signal processing [82, 83, 84, 85, 86], due to their great properties, i.e., invariance and stability, inheriting from graph convolutions [79]. Graph processes fundamentally contain a dimension of time, and recurrent neural networks (RNNs) can be utilized to describe the time dependencies in graph processes. The utilization of RNNs is a good choice when the data elements are Euclidean and time-dependent [87, 88, 89]. Some implementations of a graph recurrent architecture can be found in [90, 91, 92, 93]. In [81], GRNNs have been systematically proved to be permutation equivariant and they are stable to perturbations of the underlying graph support, and simulations have shown that GRNNs outperform GNNs and RNNs, which implies that it is essential to take both temporal and graph structures of a graph process into account for decision-making.

4.2. System Model

Consider a connected undirected graph with M statistically identical agents. Let the graph be time-invariant and time be slotted. We denote the graph as $\mathcal{G}_M = (\mathcal{V}_M, \mathcal{E}_M)$, where $\mathcal{V}_M = \{1, 2, \cdots, M\}$

denotes the set of sources and $\mathcal{E}_M \subset \mathcal{V}_M \times \mathcal{V}_M$ denotes the set of edges among sources. The i^{th} source and the j^{th} source are conntected with each other by an edge if $(i,j) \in \mathcal{E}_M$. Denote ∂_i as the neighbors of the i^{th} source, i.e., $\partial_i = \{j|(i,j) \in \mathcal{E}_M\}$. Every source $i,\ i = 1, 2, \cdots, M$, observes a physical process $\{X_i(k)\}_{k\geq 0}$,

$$X_i(k+1) = X_i(k) + \Lambda_i(k), \tag{4.1}$$

where $\Lambda_i(k) \sim \mathcal{N}(0, \sigma^2)$ are i.i.d for all i, k. The processes $\{X_i(k)\}_{k=0}^{\infty}$ are assumed to be mutually independent across i. By convention, $X_i(0) = 0$ for all $i \in \mathcal{V}_M$.

All sources can communicate with their neighbors, and each source communicates with at most one of its neighbors in a time slot. Specifically, the i^{th} source can transmit an update status (packet) to the j^{th} source in a time slot if $(i,j) \in \mathcal{E}_M$. If there is no edge between the i^{th} and the j^{th} sources, then they can not communicate with each other directly. The communication medium is modeled by a collision channel: if two or more sources in proximity transmit packets in the same time slot, then the packets interfere with each other (collide) and cause communication failures. We assume a delay of one-time units in delivery for packets. Let $c_i^j(k) = 1$ represent the event that collisions happen in the edge with the direction from the i^{th} source to the j^{th} source, otherwise $c_i^j(k) = 0$.

Each source can cache packets from others. For clear presentation, each source is assumed to have *M virtual queues*[11], and the queue with index $j \in \mathcal{V}_M$, denoted by $Q_{i,j}$, is utilized to cache the packets from the j^{th} source. We further assume that the buffer size of $Q_{i,j}$ is one and that a coming packet can either replace undelivered packets or be discarded. This assumption relies on the fact that the underlying processes that are monitored are Markovian. The indicator $q_i^j(k) = 1$ implies that $Q_{i,j}$ is occupied by a packet in time k otherwise $q_i^j(k) = 0$. In every time slot, every source can sample the underlying process (packets in $Q_{i,i}$) or transmit packets in $\{Q_{i,j}\}_{\{j\neq i\}}$ to one of its neighbors. Denote $d_i^{j,\ell}(k) = 1$ as the indicator such that the i^{th} source transmits a update status from the ℓ^{th} source (in $Q_{i,\ell}$) to the j^{th} source successfully in time k, otherwise $d_i^{j,\ell}(k) = 0$. Note that $d_i^{j,\ell}(k) = 0$ for all k, ℓ if $j \notin \partial_i$.

[11]Since the graph is connected, then every source can receive packets from every other source.

Now, we summarize the process of transmission: (i) At every time slot k, the i^{th} source observes the physical process $X_i(k)$ defined in (4.1) and updates the packet in $Q_{i,i}$. (ii) The i^{th} source chooses to transmit a packet or keep silent in the current time slot. (iii) If the i^{th} source chooses to transmit a packet, then he picks up a specific packet from $\{Q_{i,j}\}_{j\in\mathcal{V}_M}$ and transmits it to a specific neighbor. If the i^{th} source chooses to keep silent, then he will not transmit packets in this time slot. (iv) Once all sources determine their actions, the transmissions start. (v) Suppose that the receiver of the i^{th} source is the j^{th} source, and the transmitted packet is chosen from $Q_{i,\ell}$. If $c_i^j(k)=1$, then the packet can not be delivered to the j^{th} agent, otherwise the packet is delivered to the j^{th} source. (vi) Finally, if $Q_{j,\ell}$ is empty, then the j^{th} source will cache the transmitted packet; if $Q_{j,\ell}$ is non-empty, but the current packet in $Q_{j,\ell}$ is older (we will define "old" and "new" later) than the transmitted one, then the j^{th} source will replace the current packet with the transmitted one; otherwise, the transmitted packet (from $Q_{i,\ell}$) will be discarded.

Every source is assumed to estimate the process for every other source based on the collection of received samples. Denote by $\hat{X}_i^j(k)$ the estimate of $X_j(k)$ in time slot k from the perspective of the i^{th} source. In particular, $\hat{X}_i^i(k)=X_i^i(k)$ for all i and k. By convention, $\hat{X}_{i,0}^j=0$ for all $i,j\in\mathcal{V}_M$. We define the average sum of estimation errors (ASEE) as our performance metric:

$$\begin{aligned} L^{\pi}(M) &= \lim_{K\to\infty}\mathbb{E}[L_K^{\pi}] \\ L_K^{\pi}(M) &= \frac{1}{M^2K}\sum_{k=1}^{K}\sum_{i=1}^{M}\sum_{j=1}^{M}\left(\hat{X}_i^j(k)-X_j(k)\right)^2, \end{aligned} \tag{4.2}$$

where $\pi\in\Pi$ refers to a decentralized sampling and transmission policy in place, and Π is the set of all decentralized sampling and transmission policies. Thus, we need to solve the following optimization problem

$$\min_{\pi\in\Pi}\ L^{\pi}(M). \tag{4.3}$$

Let us provide a more precise definition of the policy appearing in (4.2) and (4.3) in the following definition.

Definition 8. *A decentralized sampling and transmission policy is the collection of $\mu_i(k)$, $\nu_i(k)$ for*

all $i \in \mathcal{V}_M$ *and* $k \geq 0$, *i.e.,* $\left\{\{\big(\mu_i(k), \nu_i(k)\big)\}_{i\in\mathcal{V}_M}\right\}_{k\geq 0}$, *such that:*

- *If* $\mu_i(k) \neq i$, *then the* i^{th} *agent transmits the packet in* $Q_{i,\nu_i(k)}$ *to the agent* $\mu_i(k)$ *in time slot* k. *It is implicity to assume that* $\mu_i(k) \in \partial_i$.

- *If* $\mu_i(k) = i$, *then the* i^{th} *agent keeps silent in time* k, *and we artificially set* $\nu_i(k) = i$.

The minimum attainable ASEE is then denoted by $L(M)$: $L(M) = \min_{\pi\in\Pi} L^{\pi}(M)$. Our objective is to design *decentralized* sampling and transmission mechanisms to attain $L(M)$. Every time k, every source chooses its action in a decentralized manner based on its current and past local observations, as well as its past actions. In this paper, we consider two general classes of policies are considered, namely *oblivious* policies and *non-oblivious* policies (as we defined in Chapter 3). In the former class, decision-making is independent of the processes that are monitored, while in the latter class, decision-making depends on the processes.

4.2.1. Age of Information and Oblivious Policies

Let $\tau_{i,j}$ be the generation time of the packet in $Q_{i,j}$. Note that the buffer size of $Q_{i,j}$ is 1, so the current packet is the latest one. The age of information (AoI) with respect to $Q_{i,j}$, denoted by $h_i^j(k)$, is defined as

$$h_i^j(k) = k - \tau_{i,j}. \tag{4.4}$$

Without loss of generality, let $h_i^j(0) = 0$. Let another packet associated with the j^{th} source is delivered to the i^{th} source and it may be cached in $Q_{i,j}$. Suppose that the delivered packet has generation time τ' and $\tau' < \tau_{i,j}$, i.e., the delivered packet is generated before the current packet in $Q_{i,j}$. If the packet is cached by the i^{th} source, then the AoI of $Q_{i,j}$ becomes $k - \tau' > k - \tau_{i,j}$, which implies that the delivered packet can not decrease the AoI of $Q_{i,j}$. In this case, we say the delivered packet is *older* than the current packet in $Q_{i,j}$, and the i^{th} source must discard the delivered (older) packet. From the discussions above, based on (4.4), $h_i^j(k)$ increases with time k, and it jumps to a

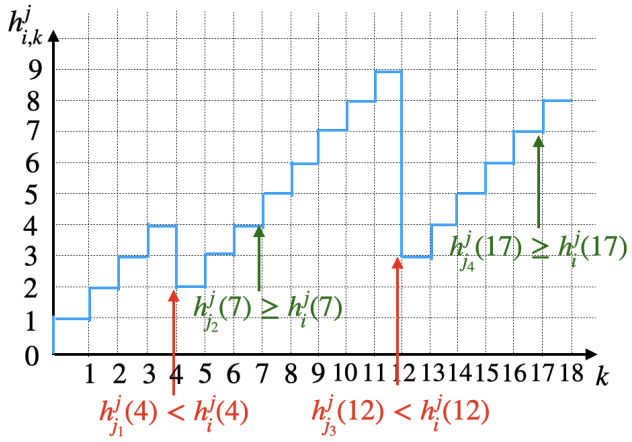


Figure 4.1: An example of trajectory of $h_i^j(k)$.

certain value when a newer packet is delivered. Then, the recursions of $h_i^j(k)$ is (see Figure 4.1),

$$h_i^j(k+1) = \begin{cases} h_u^j(k) + 1 & d_u^{i,j}(k) = 1,\ h_u^j(k) < h_i^j(k) \\ h_i^j(k) + 1 & \text{otherwise.} \end{cases} \tag{4.5}$$

At the beginning of time slot k, the i^{th} source knows the information of the packet cached in $Q_{i,j}$ before and including time k, i.e., $X_j(\tau_{i,j})$, and reconstructs $\hat{X}_i^j(k)$ by the minimum mean square error (MMSE) estimator. Since we assume that the buffer size of every agent is 1, the MMSE estimator reduces to a Kalman-like estimator:

$$\hat{X}_i^j(k) = \mathbb{E}\big[X_j(k)|X_j(\tau_{i,j})\big]. \tag{4.6}$$

In particular, from (4.1) and (4.4), we have

$$X_j(k) = X_j(\tau_{i,j}) + \sum_{\tau=1}^{h_i^j(k)} \Lambda_i(k-\tau). \tag{4.7}$$

Note that $\mathbb{E}[\Lambda_i(k)] = 0$ for all i, k, then, Kalman-like estimator in (4.6) is:

$$\hat{X}_i^j(k) = \mathbb{E}\big[X_j(k)|X_j(\tau_{i,j})\big] = X_j(\tau_{i,j}). \tag{4.8}$$

Based on (4.8), the recursion of estimates are

$$\hat{X}_i^j(k+1) = \begin{cases} \hat{X}_u^j(k) & d_u^{i,j}(k) = 1,\ h_u^j(k) < h_i^j(k) \\ \hat{X}_i^j(k) & \text{otherwise.} \end{cases} \tag{4.9}$$

Recall that our goal is to find an optimal decentralized sampling and transmission policy solving the optimization (4.3). One of the major challenges in this problem arises from the decentralized nature of decision-making. A decentralized policy is one in which the action of each source is only a function of its local observations and actions. In this setup, the action of source i at time k depends on the history of feedback and actions as well as casual observations of the process $\{\hat{X}_{i,k}^j\}_{j\in\mathcal{V}_M}$. To simplify the problem, we first consider *oblivious* policies. We define the *oblivious* policies as the policies where the actions of each source depend only on the history of feedback and actions at that node. In other words, oblivious policies do not take into account the realization (value) of the samples, but only the time they were sampled, transmitted, and received (if successfully received). Note that oblivious policies are independent of the processes they observe and they are therefore less costly to implement. In the rest of the subsection, we show that minimizing ASEE in the class of oblivious policies is equivalent to minimizing the average sum of AoI.

Lemma 8. *In oblivious policies, the expected estimation error associated with process i has the following relationship with the expected age function:*

$$\mathbb{E}[\big(X_j(k) - \hat{X}_i^j(k)\big)^2] = \mathbb{E}[h_i^j(k)]\sigma^2. \tag{4.10}$$

Proof. At the beginning of time slot k, the estimation error is

$$X_j(k) - \hat{X}_i^j(k) = X_j(k) - X_j(\tau_{i,j}) = \sum_{\tau=1}^{h_i^j(k)} \Lambda_i(k-\tau).$$

Now note that $h_i^j(k)$ is independent of $\{\Lambda_i(k)\}_{i,k}$ under oblivious policies. Therefore, using Wald's

equality, we find

$$\mathbb{E}[X_j(k) - \hat{X}_i^j(k)] = 0$$
$$\mathbb{E}[\big(X_j(k) - \hat{X}_i^j(k)\big)^2] = \mathbb{E}[h_i^j(k)]\sigma^2.$$

□

From Lemma 8, in the class of oblivious policies, minimizing the ASEE defined in (4.2) is equivalent to minimizing the normalized average sum of AoI defined in (4.11),

$$\begin{aligned} J^\pi(M) &= \lim_{K\to\infty} \mathbb{E}[J_K^\pi] \\ J_K^\pi(M) &= \frac{1}{M^2K} \sum_{k=1}^{K} \sum_{i=1}^{M} \sum_{j=1}^{M} h_i^j(k). \end{aligned} \tag{4.11}$$

Thus, in the class of oblivious policies, the optimization problem (4.3) is equivalent to

$$\min_{\pi\in\Pi'} \; J^\pi(M), \tag{4.12}$$

where Π' is a set of all possible oblivious policies. In the rest of the paper, we will propose a unified solution for both (4.2) and (4.12).

4.3. Preliminaries

4.3.1. Dec-POMDP

We give the definition of a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) [66]. A Dec-POMDP is a combination of a regular Markov Decision Process to model system dynamics with a hidden Markov model that connects unobservable system states probabilistically to observations.

Definition 9. *A Dec-POMDP can be described by a tuple of key elements*

$$(M, S, \{A_i\}, P_s, R, \{O_i\}_i, P_o, \gamma),$$

such that

- M*: the number of agents.*
- S*: the set of environmental states which are shared by all agents.*
- A_i*: the action spaces of the* i^{th} *agent.*
- P_s*:* $S \times \prod_{i \in \mathcal{V}_M} A_i \to [0,1]$ *denote the transition probability from a state* $s \in S$ *to a state* $s' \in S$ *after executing a (joint) action.*
- R*:* $S \times \prod_{i \in \mathcal{V}_M} A_i \times S \to \mathbb{R}$ *is a global reward function shared by all.*
- O_i*: the observation spaces of the* i^{th} *agent.*
- P_o*:* $S \times \prod_{i \in \mathcal{V}_M} A_i \times \prod_{i \in \mathcal{V}_M} O_i \to [0,1]$ *is the observation function which provides the probability of a joint observation after a joint action and transiting to a new state.*
- γ*: the discount factor that represents the value of time,* $\gamma \in [0,1]$.

In time slot k, the environment $s(k) \in S$ is unknown to all agents. However, at the current state $s(k)$, the i^{th} agent receives its observations $o_i \in O_i$ without knowing the other agents' observations. Every agent takes action $a_i(k) \in A_i$ $\big($resulting in a joint action $a(k) \in \prod_{i \in \mathcal{V}_M} A_i$ and $a(k) = a_1(k) \times a_2(k) \times a_M(k)\big)$, which causes the environment to transition to a new state $s'(k) \in S$ with transition probability $P_s\big(s'(k)|s(k), a(k)\big)$. Finally, every agent receives a reward $r \in R$. Here, we assume that all agents share the same reward. This is because all agents are homogeneous. This is because all agents are homogeneous.

4.3.2. Permutation Invariance

Once sources are located in a wireless network, there exists an inherent network topology among them. In this sense, the order of collecting observations and actions may significantly affect the transmission policies if we utilize a deep net critic which depends on all observations and actions. In other words, consider a deep net critic, a permuted input changes the output despite the environment

remaining identical.

To avoid the dilemma, [77] proposed a permutation invariant critic (PIC). Let all agents share a centralized critic [62, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211]. Denote $x(k)$ as the concatenation of $\{o_i(k)\}_{i\in\mathcal{V}_M}$, and denote $A(k)$ as the concatenation of $\{a_i(k)\}_{i\in\mathcal{V}_M}$. The PIC has the following permutation invariance, i.e.,

$$\text{PIC}(Ux(k), UA(k)|\theta_{\text{PIC}}) = \text{PIC}(U'x(k), U'A(k)|\theta_{\text{PIC}}), \tag{4.13}$$

where U and U' are any permutation matrices. To achieve (4.13), [77] proposed to use graph convolutional neural nets.

4.3.3. Graph Recurrent Neural Networks (GRNNs)

A recurrent neural network (RNN) can be defined as the following architecture [212],

$$z(t) = \rho_1\big(Bx(t) + Cz(t-1)\big), 1 \leq t \leq T, \tag{4.14}$$

$$\hat{y} = \rho_2\big(Dz(T)\big). \tag{4.15}$$

In (4.14), $\{x(t)\}_{1\leq t\leq T}$ with $x(t) \in \mathbb{R}^M$ is a sequence of M-dimensional data points, $z(t) \in \mathbb{R}^M$ is the hidden state extracted from the sequence $\{x(\tau)\}_{1\leq\tau\leq t}$, $B \in \mathbb{R}^{M\times M}$, $C \in \mathbb{R}^{M\times M}$ are linear operators, ρ_1 is a pointwise nonlinearity. Note that it is not necessary for $x(t)$ and $z(t)$ to share the same dimensions. In (4.15), $\hat{y} \in \mathbb{R}^{M'}$ is an estimate of $y \in \mathbb{R}^{M'}$, which is the target representation of $\{x(t)\}_{1\leq t\leq T}$, $D \in \mathbb{R}^{M\times M'}$ is the linear output map and ρ_2 is another pointwise nonlinearity. Given a training set $\{\{x(t)\}_{1\leq t\leq T}, y\}$, the optimal linear maps B, C and D are obtained by minimizing some loss function $\mathcal{L}(\rho(Dz(T)), y)$ over the training set.

Now, we extend x to a graph signal and let Ξ be a graph shit operator (GSO) [81]. Then, we define graph convolutions [78, 80, 81] as,

$$B(\Xi)x = \sum_{k=0}^{K-1} b_k \Xi^k x, \tag{4.16}$$

where $\Xi^k x = \Xi(\Xi^{k-1}x)$ and $b_k \in \mathbb{R}$ for all $0 \leq k \leq K-1$. In (4.16), $\Xi^k x$ represents the information contained in the k-hop neighborhood of each node. For $0 \leq k \leq K-1$, $[b_0, b_1, \cdots, b_{K-1}]$ is called filter taps. Then, combining (4.14), (4.15), and (4.16), the graph recurrent neural network (GRNN) architecture with $\{x(t)\}_{1\leq t\leq T}$ is defined as

$$z(t) = \rho_1\big(B(\Xi)x(t) + C(\Xi)z(t-1)\big), \tag{4.17}$$

$$\hat{y} = \rho_2\big(D(\Xi)z(T)\big), \tag{4.18}$$

where $x(t)$ and $z(t)$ in (4.17) and (4.18) are graph signals.

Suppose that each node has a vector of features. All feature vectors constitute a graph signal tensor, denoted by $V \in \mathbb{R}^{M\times F}$, where M is the number of nodes and F is the dimension of feature vectors. Each column $V_f \in \mathbb{R}^M$ is a graph signal corresponding to the values of feature f in all nodes. The linear transformation in (4.16) can be extended to an operator, denoted by $\mathcal{B}(\Xi)$,

$$\mathcal{B}(\Xi)V = \sum_{\tau=0}^{K-1} \Xi^\tau V B_\tau, \tag{4.19}$$

where $B_\tau \in \mathbb{R}^{F\times G}$ with $\tau = 0, 1, 2, \cdots, K-1$ are the filter taps, and G is the dimension of the output features.

Finally, denote the hidden state $Z(t)$ has the dimension $M \times H$. Consider the graph signal sequences $\{X(t)\}_{0\leq t\leq T}$, then (4.17) and (4.18) can be extended to

$$Z(t) = \rho_1(\mathcal{B}\big(\Xi)X(t) + \mathcal{C}(\Xi)Z(t-1)\big), \tag{4.20}$$

$$\hat{Y} = \rho_2\big(\mathcal{D}(\Xi)Z(T)\big), \tag{4.21}$$

where the filter taps are $B_\tau \in \mathbb{R}^{F\times H}$, $C_\tau \in \mathbb{R}^{H\times H}$, and $D_\tau \in \mathbb{R}^{G\times H}$, $\tau = 0, 1, 2, \cdots, K-1$. To compact (4.20) and (4.21), we write

$$\hat{Y} = \Phi(\mathcal{B}, \mathcal{C}, \mathcal{D}; \Xi, \{X(t)\}_{t=1}^T). \tag{4.22}$$

If $X(t) = X$ for all $1 \leq t \leq T$, (4.23) reduces to

$$\hat{Y} = \Phi(\mathcal{B}, \mathcal{C}, \mathcal{D}; \Xi, X, T). \tag{4.23}$$

4.3.4. Graphons

The descriptions of graphons are selected from [213]. A graphons is the limit of sequences of dense undirected graphs, which is defined as $W : [0,1]^2 \to [0,1]$, where W is *bounded*, *measurable* and *symmetric*. The graphon signal is defined as a function $X \in L^2([0,1])$. On one hand, an important interpretation of graphons (W) and graphon signals (X) are generative models for graphs and graph signals. A pair (Ξ_n, x_n) can be obtained from the pair (W, X) as follows: A point $u_i \in [0,1]$ is chosen to be the label of the i^{th} node with $i \in \mathcal{V}_n$. For $1 \leq i, j \leq n$,

$$[\Xi_n]_{ij} = \text{Bernoulli}(W(u_i, u_j)) \tag{4.24}$$

$$[x_n]_i = X(u_i). \tag{4.25}$$

For example, the *stochastic graphs* can be constructed by the following rule: Let $\{u_i\}_{i=1}^n$ be n points sampled independently and uniformly at random from $[0,1]$, and the n-node stochastic graph $\mathcal{G}_n$, whose GSO Ξ_n, is obtained from W by (4.24). On the other hand, graphons and graphon signals can be induced by graphs and graph signals, respectively. Let $\mathcal{G}_n$ be a graph with GSO Ξ_n and have node labels $\{u_i\}_{i=1}^n$, $u_i \in [0,1]$, and x_n is a graph signal. Denote $I_i = [u_i, u_{i+1})$ for $1 \leq i \leq n-1$ and $I_n = [u_n, 1] \cup [0, u_1)$. Then (W_{Ξ_n}, X_n) induced by (Ξ_n, x_n) is given by

$$W_{\Xi_n}(u, v) = \sum_{i=1}^{n} \sum_{j=1}^{n} [\Xi_n]_{ij} 1_{u \in I_i} 1_{v \in I_j} \tag{4.26}$$

$$X_n(u) = \sum_{i=1}^{n} [x_n]_i 1_{u \in I_i}. \tag{4.27}$$

The diffusion operator for graphon signals, denoted by T_W, is defined as

$$(T_W X)(v) = \int_0^1 W(u, v) X(u) du.$$

A graphon filter is an operator $T_{B,W}$: $L^2([0,1]) \to L^2([0,1])$, and is defined as follows,

$$\begin{aligned}(T_{B,W}X)(v) &= (\sum_{k=0}^{K-1} b_k T_W^{(k)} X)(v) \\ (T_W^{(k)}X)(v) &= \int_0^1 W(u,v)(T_W^{(k-1)}X)(u)du,\end{aligned} \tag{4.28}$$

where $T_W^{(0)} = I$ and $[b_0, b_1, \cdots, b_{K-1}]$ are the filter coefficients.

Similar to a GNN, a graphon neural network (WNN) is given as follows. The layer ℓ maps the incoming $F_{\ell-1}$ features from layer $\ell-1$ into F_ℓ features. Denote the features in layer $\ell-1$ as $X_{\ell-1}^g$[12], $1 \le g \le F_{\ell-1}$. Denote the features in layer ℓ as X_ℓ^f with $1 \le f \le F_\ell$. Then $X_\ell^f = \rho\big(\sum_{g=1}^{F_{\ell-1}} T_{B_\ell^{fg},W} X_{\ell-1}^g\big)$, where $\rho(\cdot)$ is a pointwise nonlinearity and $\{B_\ell^{fg}\}_{\ell,f,g} \triangleq \mathcal{B}$ is a tensor grouping the coefficient sets B_ℓ^{fg} for all features and all layers with $B_\ell^{fg} = [b_{\ell,0}^{fg}, b_{\ell,1}^{fg}, \cdots, b_{\ell,K-1}^{fg}]$. We abuse the notation X a little such that $X = \{X_0^g\}_{g=1}^{F_0}$, and denote $Y = \{X_L^f\}_{f=1}^{F_L}$. A WNN can be represented more compactly as the map

$$Y = T_{\mathcal{B},W} X. \tag{4.29}$$

4.3.5. Reinforcement Learning Algorithms

We introduce some commonly used reinforcement learning algorithms which will be used in Section 4.6, i.e., Independent Synchronous Advantage Actor-Critic (IA2C), Independent Proximal Policy Optimization (IPPO), Multi-Agent Advantage Actor-Critic (MAA2C), and Multi-Agent Proximal Policy Optimization (MAPPO) [58]. All these algorithms are variants of the commonly used A2C algorithm [214, 215].

The IA2C and IPPO are independent learning where each agent has an actor and a critic trained, in a decentralized manner, conditioned on the history of location observations, actions, and rewards, to minimize the loss function. Different from the IA2C and IPPO, in the MAA2C and MAPPO, the critic learns a joint state value function. It extends the existing on-policy actor-critic algorithm

[12] All features are graphon signals.

A2C by applying centralized critics conditioned on the state of the environment rather than the individual history of observations.

4.4. Graphical Reinforcement Learning Framework

Although every source decides, in a decentralized manner, when to sample, who to communicate with, and what to transmit, all sources minimize the ASEE $\big($see (4.2)$\big)$ in a cooperative setting.

Denote the adjacent matrix of $\mathcal{G}_M$ as Ξ_M. In each time slot k, the environment has a state s_k, including $\{X_i(k)\}_{i\in\mathcal{V}_M}$, $\{\hat{X}_j^i(k)\}_{i,j\in\mathcal{V}_M}$, $\{h_j^i(k)\}_{i,j\in\mathcal{V}_M}$, $\{c_i^j(k)\}_{i,j\in\mathcal{V}_M}$, $\{q_i^j(k)\}_{i,j\in\mathcal{V}_M}$, $\{d_i^{j,\ell}(k)\}_{i,j,\ell\in\mathcal{M}}$, and Ξ_M in time slot k, i.e.,

$$\begin{aligned} s(k) =&\Big\{\{X_i(k)\}_{i\in\mathcal{V}_M},\{\hat{X}_i^j(k)\}_{i,j\in\mathcal{V}_M},\{h_i^j(k)\}_{i,j\in\mathcal{V}_M},\\ &\{c_i^j(k)\}_{i,j\in\mathcal{V}_M},\{q_i^j(k)\}_{i,j\in\mathcal{V}_M},\{d_i^{j,\ell}(k)\}_{i,j,\ell\in\mathcal{V}_M},\Xi_M\Big\}. \end{aligned} \tag{4.30}$$

Each source can only observe its observations, so the observation of the i^{th} source is

$$\begin{aligned} o_i(k) =&\Big\{X_i(k),\{\hat{X}_i^j(k)\}_{j\in\mathcal{V}_M},\{h_i^j(k)\}_{j\in\mathcal{V}_M},\\ &\{c_i^j(k)\}_{j\in\mathcal{V}_M},\{q_i^j(k)\}_{j\in\mathcal{V}_M},\{d_i^{j,\ell}(k)\}_{j,\ell\in\mathcal{V}_M},\Xi_M\Big\} \end{aligned} \tag{4.31}$$

in time slot k. Here we assume that the network topology or adjacent matrix Ξ_M is *known by all sources*. After receiving the observations, the i^{th} source makes a decision $a_i(k) = (\mu_i(k), \nu_i(k))$. The joint actions $a(k) = \{a_i(k)\}_{i\in\mathcal{V}_M}$ and the state $s(k)$ makes the environment transition into the next state $s(k+1) \sim P_s\big(\cdot\,|s(k), a(k)\big)$.

Denote the policy neural network for the i^{th} source as $\pi(a_i(k)|o_i(k);\theta_i)$. Since all sources are identical and homogeneous, then $\pi(a_i(k)|o_i(k);\theta_i)$ is reduced to $\pi(a_i(k)|o_i(k);\theta)$, where $\theta_1 = \theta_2 = \cdots = \theta_M = \theta$. To find the optimal transmission policy in (4.2), we consider two commonly used frameworks, i.e., the A2C and PPO frameworks. We abuse the notation such that $a(k)$ and $o(k)$ represent the concatenations of $\{a_i(k)\}_{i\in\mathcal{M}}$ and $\{o_i(k)\}_{i\in\mathcal{M}}$, respectively. Under the A2C framework

[216, 214], we need to maximize

$$L^{A2C}(\theta) = \mathbb{E}\big[\log \pi(a(k)|o(k);\theta)\mathrm{Adv}(k)\big]$$

where $\mathrm{Adv}(k)$ is an estimator of the advantage function in time slot k. Under the PPO framework, we need to maximize [216]

$$L^{PPO}(\theta) = \mathbb{E}\Big[\min\big(\zeta(\theta)\mathrm{Adv}(k), \mathrm{clip}(\zeta(\theta), 1-\epsilon, 1+\epsilon)\mathrm{Adv}(k)\big)\Big],$$

where $\zeta(\theta) = \frac{\pi(a(k)|o(k);\theta)}{\pi(a(k)|o(k);\theta_{\mathrm{old}})}$. The PPO framework is a straightforward extension of the A2C framework.

4.4.1. Reward Function

At the end of every time slot, the environment returns a reward $r(k)$. We define the shared reward as

$$r(k) = -\frac{\sum_{i,j\in\mathcal{V}_M}\big(X_j(k) - \hat{X}_i^j(k)\big)^2}{M^2K}. \tag{4.32}$$

If we focus on the age minimization problem (4.12), then the reward can also be defined as $r(k) = -\frac{\sum_{i,j\in\mathcal{V}_M} h_i^j(k)}{M^2K}$. The return $\sum_{\tau=0}^{\infty}\gamma^\tau r(k+\tau)$ is the total accumulated return from time step k with discount factor γ.

4.4.2. Graph Actor

As mentioned in Section 4.3.2, there exists an inherent network topology among sources. By taking a graph perspective of the data, we structure it in a useful way for learning. Recall that we consider decentralized strategies, then every source has a graph with its observations for decision-making.

To avoid the impact of permutations of inputs, we consider graph convolution neural net. To fully utilize the information in the graph, we use L graph *recurrent* neural net layers (as we defined in Section 4.3.3). The actor is constructed by a GRNN. In time slot k, denote the graph associated with the i^{th} source as $\mathcal{G}_i(k)$. Note that it is assumed that the network topology is known for all sources. Then, $\mathcal{G}_i(k)$ has the same node set and edge set as $\mathcal{G}$. For every node in $\mathcal{G}_i(k)$, we embed a node

feature. Define the node feature of the j^{th} node in $\mathcal{G}_i(k)$ as $v_i^j(k)$, and the collection of node features in $\mathcal{G}_i(k)$ is $v_i(k) = \{v_i^j(k)\}_{j\in\mathcal{V}_M}$. Denote the dimension of node features as F_0. Then, we convert the collection of node features $v_i(k)$ to a graph signal tensor, denoted by $V_{i,0}$ and $V_{i,0} \in \mathbb{R}^{M\times F_0}$.

Consider the ℓ^{th} layer, $1 \le \ell \le L$. The input is the output from the $(\ell-1)^{th}$ layer, $V_{i,\ell-1} \in \mathbb{R}^{M\times F_{\ell-1}}$, and the output is $V_{i,\ell} \in \mathbb{R}^{M\times F_\ell}$. Similar to (4.23),

$$V_{i,\ell} = \Phi(\mathcal{B}_\ell, \mathcal{C}_\ell, \mathcal{D}_\ell; \Xi_M, V_{i,\ell-1}, T), \tag{4.33}$$

where T is the number of recurrent rounds.

4.4.3. Action Distribution

Denote the output of the actor as $V_{i,L} \in \mathbb{R}^{M\times F_L}$. Note that $V_{i,L}$ is node embeddings. Then, we introduce another operator ϕ to convert node embeddings to an action distribution, i.e., $\phi : \mathbb{R}^{M\times F_L} \to \mathbb{R}^{M\times M}$, with the following specific form,

$$\phi(V_{i,L}) :\triangleq V_{i,L}\Delta V_{i,L}', \tag{4.34}$$

where $\Delta \in \mathbb{R}^{F_L\times F_L}$. $a_{i,k}$ is sampled by,

$$a_{i,k} \sim F_{\text{softmax}}\big(\phi(V_{i,L})\big). \tag{4.35}$$

Remark 19. *Under the specific form* (4.34)*, the number of parameters in ϕ is independent of the number of agents M.*

Remark 20. *Note that all agents are homogeneous, so they share a common ϕ.*

4.4.4. Graph Critic

The construction of the graph critic is similar to that of the graph actor in Section 4.4.2. However, to reduce the complexity of our model, we remove the recurrent graph convolutional layers in the graph critic. In other words, the graph critic is a classical GNN.

4.5. Transferability of Action Distributions

Now, we provide a fundamental analysis of the graphical reinforcement learning framework in Section 4.4. We start with the permutation invariance property. Note that GNNs have the permutation invariance property, and a GRNN is a straightforward extension of a GNN combined with an RNN architecture, so GRNNs have the permutation invariance property.

Lemma 9. *([213, Proposition 1]) Let Ξ_n be a GSO and $\tilde{\Xi}_n = U^T \Xi_n U$ be a permutation of this GSO, for some permutation U. Let x_t be a graph signal and $\tilde{x}_t = U^T x_t$ be the permuted version of x_t for all $1 \leq t \leq T$. Then, it holds that*

$$\begin{aligned}\tilde{z}_t =& \rho_1(A(\tilde{\Xi})\tilde{x}_t + B(\tilde{\Xi})\tilde{z}_{t-1}) = U^T z_t, \quad 1 \leq t \leq T,\\ \tilde{y}_T =& \rho_2(C(\tilde{\Xi})\tilde{z}_T) = U^T y_T.\end{aligned}$$

In practice, it is impossible to learn GNNs for large networks. This is because (i) full knowledge of the graph should be known, which is hard to collect when a network's size is large, and (ii) the matrix multiplication operations are computationally complex when the network size is large. Fortunately, the filter taps in a GNN are independent of the network size. In other words, we can transfer the model trained on small or moderate graphs to large graphs [213]. Note that a GRNN is an extension of a GNN, so the transferability should hold in GRNNs. Suppose that y_{n_1} and y_{n_2} are outputs from a GNN or a GRNN with network size n_1 and n_2, respectively. The transferability means that the distortion between y_{n_1} and y_{n_2} are small when n_1 and n_2 are large. Since y_{n_1} and y_{n_2} have different dimensions, it is difficult to compare them directly, a tractable way is to compare y_{n_1} or y_{n_2} with their limit. Before delving into transferability, we first give some useful definitions and assumptions.

Definition 10. *Let W be a graphon and X be a graphon signal. Let $T_{B,W}$ be a graphon filter (see (4.28)), and $Y = T_{B,W}X$. Let $\mathcal{G}_n$ be n-node generic graph with node labels $\{u_i\}_{i=1}^n$. Suppose that (Ξ_n, x_n) is induced from (W, X) (see (4.24) and (4.25)), and (W_{Ξ_n}, X_n) is induced by (Ξ_n, x_n) (see (4.26) and (4.27)). We call that*

(i) $B(\Xi_n)$ *is a graph filter (see* (4.16)*) instantiated from* $T_{B,W}$ *on this graph.*

(ii) $Y_n = T_{B,W_{\Xi_n}} X_n$ *is induced by* $y_n = B(\Xi_n)x_n$.

(iii) $Y_n = T_{B_2,W_{\Xi_n}} \rho(T_{B_1,W_{\Xi_n}} X_n)$ *is induced by* $y_n = B_2(\Xi_n)\rho\big(B_1(\Xi_n)x_n\big)$, *where* ρ *is a pointwise nonlinearity, and* $T_{B_1,W}$ *and* $T_{B_2,W}$ *are graphon filters.*

(iv) $Y_n = T_{B_m,W}\rho_{m-1}\big(T_{B_{m-1},W}\cdots\rho_1(T_{B_1,W}X)\cdots\big)$ *is induced by the graph signal* $y_n = B_m(\Xi_n)\rho_{m-1}\Big(B_{m-1}(\Xi_n)\cdots\rho_1\big(B_1(\Xi_n)x_n\big)\cdots\Big)$, *where* $\rho_1,\cdots,\rho_{m-1}$ *are pointwise nonlinearities, and* $T_{B_1,W}$, $T_{B_2,W}$, $\cdots$, $T_{B_m,W}$ *are graphon filters.*

Definition 11. *Let* $T_{\mathcal{B},W}$ *be a WNN with* L *layers* $\big($*see* (4.29)$\big)$, $F_0 = 1$ *input feature,* $F_L = 1$ *output feature, and* $F_\ell = F$ *features per layer for* $1 \leq \ell \leq L-1$. *Let* $\mathcal{G}_n$, $\{u_i\}_{i=1}^n$, Ξ_n, W_{Ξ_n}, X, x_n, *and* X_n *be defined in Definition 10. We call that*

(i) $\mathcal{B}(\Xi_n)$ *is a linear transformation (see* (4.19)*) instantiated from* $T_{\mathcal{B},W}$ *on this graph.*

(ii) $Y_n = T_{\mathcal{B},W_{\Xi_n}} X_n$ *is induced by* $y_n = B(\Xi_n)x_n$.

(iii) $Y_n = T_{\mathcal{B}_2,W_{\Xi_n}} \rho(T_{\mathcal{B}_1,W_{\Xi_n}} X_n)$ *is induced by* $y_n = \mathcal{B}_2(\Xi_n)\rho\big(\mathcal{B}_1(\Xi_n)x_n\big)$, *where* ρ *is a pointwise nonlinearity, and* $T_{\mathcal{B}_1,W}$ *and* $T_{\mathcal{B}_2,W}$ *are WNNs.*

(iv) $Y_n = T_{\mathcal{B}_m,W}\rho_{m-1}\big(T_{\mathcal{B}_{m-1},W}\cdots\rho_1(T_{\mathcal{B}_1,W}X)\cdots\big)$ *is induced by the graph signal*

$$y_n = \mathcal{B}_m(\Xi_n)\rho_{m-1}\Big(\mathcal{B}_{m-1}(\Xi_n)\cdots\rho_1\big(\mathcal{B}_1(\Xi_n)x_n\big)\cdots\Big),$$

where $\rho_1,\rho_2,\cdots,\rho_{m-1}$ *are pointwise nonlinearities, and* $T_{\mathcal{B}_1,W}$, $T_{\mathcal{B}_2,W}$, $\cdots$, $T_{\mathcal{B}_m,W}$ *are WNNs.*

Definition 12. *Similar to* (4.20), (4.21), *we can define a graphon recurrent neural network (WRNN) as follows,*

$$Z_t = \rho_1\big(T_{\mathcal{B}_1,W}X_t + T_{\mathcal{B}_2,W}Z_{t-1}\big),\ 1 \leq t \leq T, \tag{4.36}$$

$$Y = \rho_2\big(T_{\mathcal{B}_3,W}Z_T\big), \tag{4.37}$$

or write (4.36) *and* (4.37) *compactly,*

$$Y = \Psi(\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3; W, \{X_t\}_{t=1}^T), \tag{4.38}$$

or

$$Y = \Psi(\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3; W, X, T), \tag{4.39}$$

if $X_t = X$ *for all* $1 \leq t \leq T$. *Let* $\mathcal{G}_n$, $\{u_i\}_{i=1}^n$, Ξ_n, W_{Ξ_n} *be defined in Definition 10. Suppose that* x_n *is induced from* X, *and* X_n *is induced by* x_n. *We call that* $Y_n = \Psi(\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3; \Xi_n, X_n, T)$ *is induced by* $y_n = \Phi(\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3; \Xi_n, x_n, T)$.

Definition 13. *([213, Definition 4]) The* ϵ*-band cardinality of a graphon* W, *denoted by* κ_W^ϵ, *is the number of eigenvalues* λ_i *of* T_W *with absolute value larger or equal to* ϵ, *i.e.,*

$$\kappa_W^\epsilon = \#\{\lambda_i : |\lambda_i| \geq \epsilon\}.$$

Definition 14. *([213, Definition 5]) For two graphons* W *and* W', *the* ϵ*-eigenvalue margin, denoted by* $\delta_{WW'}^\epsilon$, *is given by*

$$\delta_{WW'}^\epsilon = \min_{i, j \neq i} \{|\lambda_i(T_{W'}) - \lambda_j(T_W)| : |\lambda_i(T_{W'})| \geq \epsilon\},$$

where $\lambda_i(T_{W'})$ *and* $\lambda_i(T_W)$ *denote the eigenvalues of* $T_{W'}$ *and* T_W, *respectively.*

Assumption 3. *([213, AS1]) The spectral response of the convolutional filter of* $T_{B,W}$, *defined as* $b(\lambda) = \sum_{k=0}^{K-1} b_k \lambda^k$, *is* Ω_b*-Lipschitz in* $[-1, -\epsilon] \cup [\epsilon, 1]$ *and* ω_b*-Lipschitz in* $(-\epsilon, \epsilon)$, *with* $\omega_b < \Omega_b$. *Moreover,* $|b(\lambda)| < 1$.

Under Assumption 3, we can show that if a (large) graph filter is sampled from a graphon, then it

can approximate the graphon filter. Define

$$\Theta(\Omega,\omega)=(\Omega+\frac{\pi\kappa^{\epsilon}_{W_{\Xi_n}}}{\delta^{\epsilon}_{WW_{\Xi_n}}})\|W-W_{\Xi_n}\|+2\omega\epsilon. \tag{4.40}$$

Lemma 10. *([213, Theorem 1]) Let $T_{B_1,W}$ satisfy Assumption 3. Y and Y_n are given in Definition 10 (ii). For any $0<\epsilon\leq 1$, it holds that*

$$\|Y-Y_n\|\leq\Theta(\Omega_1,\omega_1)\|X\|+(\Omega_1\epsilon+2)\|X-X_n\|. \tag{4.41}$$

Based on Lemma 10, we can extend the approximation property of generic graphs to the case where multiple graphon filters are applied.

Assumption 4. *([213, AS5]) The activation functions are normalized Lipschitz, i.e., $|\rho(x)-\rho(y)|\leq|x-y|$, and $\rho(0)=0$.*

Theorem 10. *Let $T_{B_1,W}$ and $T_{B_2,W}$ satisfy Assumption 3, and ρ be an activation function satisfying Assumption 4. Y and Y_n are given in Definition 10 (iii). For any $0<\epsilon\leq 1$, it holds that*

$$\|Y-Y_n\|\leq\sum_{i=1}^{2}\Theta(\Omega_i,\omega_i)\|X\|+(\Omega_2\epsilon+2)\|X-X_n\|, \tag{4.42}$$

where $\{\Omega_i,\omega_i\}_{i=1}^2$ are Lipschitz constants in Assumption 3.

Proof. The proof is given in Appendix C.1. □

The results in Theorem 10 can be extended to any number of graphon filters.

Corollary 3. *Let $T_{B_1,W},T_{B_2,W},\cdots,T_{B_m,W}$ satisfy Assumption 3, and $\rho_1,\rho_2,\cdots,\rho_{m-1}$ satisfy Assumption 4. Y and Y_n are given in Definition 10 (iv). For any $0<\epsilon\leq 1$, it holds that*

$$\|Y-Y_n\|\leq\sum_{i=1}^{m}\Theta(\Omega_i,\omega_i)\|X\|+(\Omega_m\epsilon+2)\|X-X_n\|, \tag{4.43}$$

and $\{\Omega_i, \omega_i\}_{i=1}^m$ *are Lipschitz constants in Assumption 3.*

Proof. The proof is a straightforward extension of the proof in Appendix C.1. □

Lemma 11. *([81, Theorem 2]) Let the convolutional filters that make up the layers of* $T_{\mathcal{B}_1,W}$ *satisfy Assumption 3.* Y *and* Y_n *are given in Definition 11 (ii). For any* $0 < \epsilon \leq 1$*, it holds that*

$$\|Y - Y_n\| \leq LF^{L-1}\Theta(\Omega_1, \omega_1)\|X\| + (\Omega_1\epsilon + 2)\|X - X_n\|.$$

Similarly, based on Lemma 11, we can extend the approximation property of generic graphs to the case where multiple WNNs are applied.

Theorem 11. *Let the convolutional filters that make up the layers of* $T_{\mathcal{B}_1,W}$ *and* $T_{\mathcal{B}_2,W}$ *satisfy Assumption 3, and* ρ *be an activation function satisfying Assumption 4.* Y *and* Y_n *are given in Definition 11 (iii). For any* $0 < \epsilon \leq 1$*, it holds that*

$$\|Y - Y_n\| \leq LF^{L-1}\sum_{i=1}^{2}\Theta(\Omega_i, \omega_i)\|X\| + (\Omega_2\epsilon + 2)\|X - X_n\|, \tag{4.44}$$

where $\{\Omega_i, \omega_i\}_{i=1}^2$ *are Lipschitz constants in Assumption 3.*

Proof. The proof is given in Appendix C.2. □

Similar to Corollary 3, the results in Theorem 11 can be extended to the case where any number of WNNs are applied.

Corollary 4. *Let* $T_{\mathcal{B}_1,W}, T_{\mathcal{B}_2,W}, \cdots, T_{\mathcal{B}_m,W}$ *satisfy Assumption 3, and* $\rho_1, \rho_2, \cdots, \rho_{m-1}$ *satisfy Assumption 4.* Y *and* Y_n *are given in Definition 11 (iv). For any* $0 < \epsilon \leq 1$*, it holds that*

$$\|Y - Y_n\| \leq \sum_{i=1}^{m} LF^{L-1}\Theta(\Omega_i, \omega_i)\|X\| + (\Omega_m\epsilon + 2)\|X - X_n\|, \tag{4.45}$$

where $\{\Omega_i, \omega_i\}_{i=1}^m$ *are Lipschitz constants in Assumption 3.*

Proof. The proof is a straightforward extension of the proof in Appendix C.2. □

From Theorem 11 and Corollary 4, the transferability can be proved in GRNNs.

Theorem 12. *Let the convolutional filters that makeup of the layers of $T_{\mathcal{B}_1,W}$, $T_{\mathcal{B}_2,W}$, and $T_{\mathcal{B}_3,W}$ satisfy Assumption 3, and ρ_1 and ρ_2 be activation functions satisfying Assumption 4. Y and Y_n are given in Definition 12. For any $0 < \epsilon \leq 1$, it holds that*

$$\|Y - Y_n\| \leq LF^{L-1} \sum_{i=1}^{3} N_i \Theta(\Omega_i, \omega_i) \|X\| + (\Omega_3 \epsilon + 2)\|X - X_n\|, \tag{4.46}$$

where $\{\Omega_i, \omega_i\}_{i=1}^3$ are Lipschitz constants in Assumption 3, and N_1, N_2, N_3 are fixed constants only depends on T.

Proof. From the definition of WRNN in (4.36) and (4.37), to prove Theorem 12, we only need to apply Theorem 11 repeatedly. And the number of repetitions only depends on T, i.e., the number of recurrences. So, the coefficients N_1, N_2, and N_3 are fixed and only depend on T. We have the desired results. □

Remark 21. *It is straightforward to see that if Y_n and Y are obtained by* (4.38)*, and y_n is obtained by* (4.23)*, the transferability holds in a GRNN.*

Since transferability holds in GRNNs, then the transferability should hold in action distributions, which is given in (4.35). Note that the action distributions are discrete. To show the transferability, we borrow the idea from the definition of graphons, i.e., we compare the action distributions with their limit action distribution (which will be defined later).

In every learning step, we obtain a matrix Δ $\big($see (4.34)$\big)$. Although Δ is updated in the learning process, it is fixed within each learning step. Define a set of labels

$$f_i = \frac{i-1}{F_L}, \quad 1 \leq i \leq F_L. \tag{4.47}$$

Let $I_i = [f_i, f_{i+1})]$ for $i = 1, 2, \cdots, F_L - 1$ and $I_{F_L} = [f_{F_L}, 1] \cup [0, f_1)$. From (4.26), we can construct a graphon with respect to Δ as follows,

$$W_\Delta(u, v) = \sum_{i=1}^{F_L} \sum_{j=1}^{F_L} [\Delta]_{ij} 1_{u \in I_i} 1_{v \in I_j}. \tag{4.48}$$

The continous version of softmax function F_{softmax} $\big($see (4.35)$\big)$, denoted by $\tilde{F}_{\text{softmax}}$, is defined as follows,

$$(\tilde{F}_{\text{softmax}} X)(v) = \frac{e^{X(v)}}{\int_0^1 e^{X(u)} du}. \tag{4.49}$$

Similar to (4.49), we define a *continuous action distribution* as

$$A \sim \tilde{F}_{\text{softmax}} \mathcal{X}, \tag{4.50}$$

and $\mathcal{X}$ is defined as follows: let Y_1 and Y_2 be any two outputs of (4.39), i.e., $Y_1, Y_2 \in L^2([0, 1])$,

$$\mathcal{X}(Y_1, Y_2) = \langle Y_1, T_{W_\Delta} Y_2 \rangle, \tag{4.51}$$

where $\langle \cdot \rangle$ is the inner product.

Definition 15. *Consider a WRNN defined in Definition 12. Let $\mathcal{G}_n$ be a n-node generic graph with node labels $\{u_i\}_{i=1}^n$. Consider any two sequences of graphon signals $\{X_j\}_{j=1}^2$, $Y_j = \Psi(\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3; W, X_j, T)$. Suppose that $(\Xi_n, \{x_{n,j}\}_{j=1}^2)$ is induced from $(W, \{X_j\}_{j=1}^2)$, and $(W_{\Xi_n}, \{X_{n,j}\}_{j=1}^2)$ is induced by $(\Xi_n, \{x_{n,j}\}_{j=1}^2)$. Let $y_{n,j} = \Phi(\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3; \Xi_n, x_j, T)$ $\big($see (4.23)$\big)$, $Y_{n,j} = \Psi(\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3; \Xi_M, X_j, T)$ $\big($see (4.39)$\big)$, and $\mathcal{X}_n(Y_{n,1}, Y_{n,2}) = \langle Y_{n,1}, T_{W_\Delta} Y_{n,2} \rangle$. Then, we call that $A_n \sim \tilde{F}_{softmax} \mathcal{X}_n$ is induced by $V_{i,L} \Delta V'_{i,L}$ $\big($see (4.34) $\big)$.*

Theorem 13. *Let $T_{\mathcal{B}_1,W}$, $T_{\mathcal{B}_2,W}$, $T_{\mathcal{B}_3,W}$, ρ_1, and ρ_2 be in Theorem 12. X_1, X_2, Y_1, Y_2, $Y_{n,1}$, $Y_{n,2}$,*

$\mathcal{X}$, and $\mathcal{X}_n$ are given in Definition 15. For any $0 < \epsilon \leq 1$, it holds that

$$\begin{aligned}
&\|\mathcal{X}(Y_1, Y_2) - \mathcal{X}_n(Y_{n,1}, Y_{n,2})\| \\
&\leq \|T_{W_\Delta}\| \Big(C_N^2 \|X_1\| \|X_2 + (\Omega_3 + 2)^2 \|X_1 - X_{n,1}\| \|X_1 - X_{n,2}\| \\
&+ C_N(\Omega_3 + 2)\|X_1\| \|X_2 - X_{n,2}\| + C_N(\Omega_3 + 2)\|X_2\| \|X_1 - X_{n,1}\| \Big),
\end{aligned} \tag{4.52}$$

where $C_N = LF^{L-1} \sum_{i=1}^{3} N_i \Theta(\Omega_i, \omega_i)$, and N_1, N_2, N_3 are given in Theorem 12.

Remark 22. *From the definitions of W_{Ξ_n}, $X_{n,1}$ and $X_{n,2}$, for $\eta > 0$, we can choose large n and small ϵ, such that $\Theta(\Omega_i, \omega_i) < \eta$ for $i \in \{1, 2, 3\}$, $\|X_1 - X_{n,1}\| < \eta$, and $\|X_2 - X_{n,2}\| < \eta$. This implies $\|\mathcal{X}(Y_1, Y_2) - \mathcal{X}_n(Y_{n,1}, Y_{n,2})\|$ is bounded by a small scalar for any Y_1, Y_2, and graph $\mathcal{G}_n$.*

Remark 23. *It is straightforward to see that if $Y_{n,j}$ and Y_j are obtained by (4.38), and $y_{n,j}$ is obtained by (4.23), with $j \in \{1, 2\}$, the transferability still holds in the distribution of the actions.*

Proof. Refer to Appendix C.3. □

Theorem 14. *Let $T_{\mathcal{B}_1,W}$, $T_{\mathcal{B}_2,W}$, $T_{\mathcal{B}_3,W}$, ρ_1, and ρ_2 be in Theorem 12. A is given in (4.50) and A_n is given in Definition 15 (ii). For any small $\eta > 0$, there exists a $0 < \epsilon \leq 1$ and a large n, such that*

$$\|A - A_n\| \leq \Gamma \cdot \eta \tag{4.53}$$

where Γ is a constant independent of the WRNN $\Psi(\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, W_{\Xi_n}, \cdot, T)$ defined in (4.39).

Proof. Refer to Appendix C.4. □

Finally, we apply the results above on stochastic graphs in Appendix C.5.

4.6. Simulations

We verify our analysis through simulations. We first provide the constructions of actors and critics in Section 4.6.1, we then give 3 baselines in Section 4.6.2, and numerical results are given in

Section 4.6.3. Denote the adjacent matrix of $\mathcal{E}_M$ as Ξ_M.

4.6.1. Features in Graph Actors and Critics

Features in Graph Actor We first consider the graph actors. Since we investigate decentralized policies, then every node has a distinct graph for decision-making (as discussed in Section 4.4.2). For oblivious policies, we define the node feature $v_{i,k}^j$ (defined in Section 4.4.2) as,

$$v_{i,k}^j = [h_{i,k}^j, c_{i,k}^j, [\Xi_M]_{ij}, I_{i,k}^j], \tag{4.54}$$

where $I_{i,k}^j$ is an indicator such that $I_{i,k}^j = 1$ if the j^{th} agent caches the packet transmissted from the i^{th} agent in time k, otherwise $I_{i,k}^j = 0$. For non-oblivious policies, we define the node feature as,

$$v_{i,k}^j = [\big(X_{j,k} - X_{i,k}^j\big)^2, h_{i,k}^j, c_{i,k}^j, [\Xi_M]_{ij}, I_{i,k}^j], \tag{4.55}$$

In graph actors, edge features are not included. The information of edge is captured by the element $[\Xi_M]_{ij}$ in both (4.54) and (4.55).

Features in Graph Critic Recall that we consider two types of RL are considered, i.e., independent learning (e.g., IPPO and IA2C) and centralized training and decentralized execution (e.g., MAPPO and MAA2C). The critics in both types are different since every agent has a distinct critic in the former case, while all agents share a common critic in the latter case.

For independent learning, we let the graph critic has exactly the same formulation as that in the graph actor, and the features are defined in (4.54) and (4.55). For centralized training and decentralized execution, define the node features for the i^{th} agent as $\sum_{j=1}^M \Xi_M$, i.e., the degree of the i^{th} agent; and define the edge features[13] between i^{th} agent and the j^{th} agent as (4.54) $\big($respectively, (4.55)$\big)$ in the class of oblivious policies (respectively, non-oblivious policies).

[13]To fully utilize the information in the graph, we consider a "general edge": any pair of agents have an edge to connect each other. If $[\Xi_M]_{ij} = 0$, then the edge is a virtual edge. If $[\Xi_M]_{ij} = 1$, then the edge is a real edge.

4.6.2. Baselines

We consider three baselines: (i) Classical Reinforcement Learning, (ii) the Uniform Transmitting policy, and (iii) the Adaptive Age-based policy.

The classical reinforcement learning algorithms used as baselines are given in [58], i.e., IPPO, IA2C, MAPPO, and MAA2C. The main difference between classical reinforcement learning and graphical reinforcement learning (see Section 4.4) is that in the former case, the actor and critic are fully connected neural networks, while in the latter case, the actor and critic are graph convolutional layers.

Now, we define the Uniform Transmitting policy and the Adaptive Age-based policy. In each time slot, the i^{th} agent can be either silent or transmit packets. The number of packets cached at the side of the i^{th} agent is $\sum_{\ell=1}^{M} q_{i,k}^{\ell}$, and the number of receivers is $\sum_{j=1}^{M}[\Xi_M]_{ij}$. Thus, the total number of actions for the i^{th} agent in time k is $1+(\sum_{j=1}^{M} q_{i,k}^{j})\cdot(\sum_{j=1}^{M}[\Xi_M]_{ij})$.

Definition 16. *(Uniform Transmitting policy) In time slot k, the i^{th} agent chooses to keep silent with probability* $\frac{1}{1+(\sum_{\ell=1}^{M} q_{i,k}^{\ell})\cdot(\sum_{j=1}^{M}[\Xi_M]_{ij})}$, *and transmits the packet in $Q_{i,k}^{\ell}$ to the j^{th} agent with probability* $\frac{1_{\{q_{i,k}^{\ell}=1\}}\cdot 1_{\{[\Xi_M]_{ij}=1\}}}{1+(\sum_{\ell=1}^{M} q_{i,k}^{\ell})\cdot(\sum_{j=1}^{M}[\Xi_M]_{ij})}$.

If $h_{i,k}^{\ell}$ is large, then the packet in $Q_{i,\ell}$ is old in time k. To keep the system as fresh as possible, we can always let agents transmit packets in the queues with small AoI.

Definition 17. *(Adaptive Age-based policy) Given a fixed scalar $\epsilon>0$. In time slot k, the i^{th} agent chooses to keep silent with probability* $\frac{e^{\epsilon}}{e^{\epsilon}+\sum_{\ell=1}^{M} 1_{\{q_{i,k}^{\ell}=1\}}\cdot e^{1/(h_{i,k}^{\ell}+1)}}$[14], *and transmits the packet in $Q_{i,\ell}$ to the j^{th} agent with probability* $\frac{1}{\sum_{j=1}^{M}[\Xi_M]_{ij}}\cdot\frac{1_{\{q_{i,k}^{\ell}=1\}}\cdot 1_{\{[\Xi_M]_{ij}=1\}}\cdot e^{1/(h_{i,k}^{\ell}+1)}}{e^{\epsilon}+\sum_{\ell=1}^{M} 1_{\{q_{i,k}^{\ell}=1\}}\cdot e^{1/(h_{i,k}^{\ell}+1)}}$.

[14]We use $h_{i,k}^{\ell}+1$ instead of $h_{i,k}^{\ell}$ to avoid the case when $h_{i,k}^{\ell}=0$.

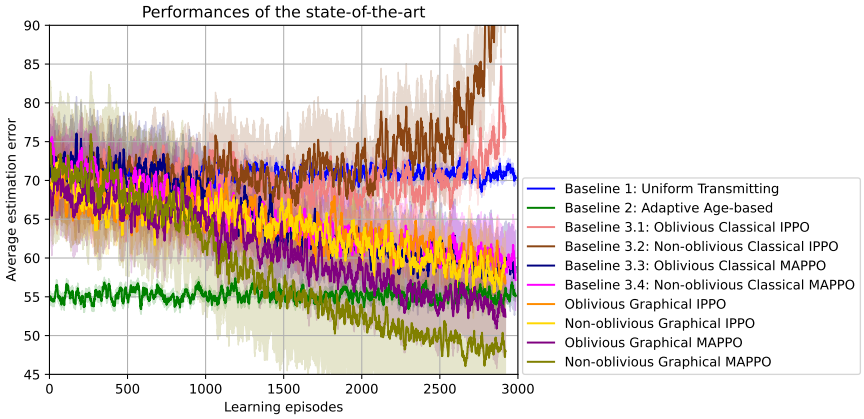


Figure 4.2: The performances of the state-of-the-art.

4.6.3. Numerical Results

For every learning episode, we let the time horizon be 1024 time slots. At the beginning of a learning episode, we generate a random Watts-Strogatz graph. The models are trained on a sequence of similar graphs. The performances of our proposed policies are given in Figure 4.2. In summary, our proposed policies outperform the state-of-the-art. We discuss the simulation results in the following aspects.

(i) The Graphical IPPOs (the dark orange and gold curves) outperform the classical IPPOs (the light coral and saddle brown curves), and the Graphical MAPPOs (the purple and olive curves) outperform the classical MAPPOs (the navy and fuchsia curves).

(ii) The Graphical MAPPO outperforms the Graphical IPPO, which implies that centralized learning and decentralized execution guarantee better performance than the independent learning in our setting.

(iii) Under the Graphical IPPO, the oblivious and non-oblivious curves are close, while under the graphical MAPPO, the non-oblivious policies are visibly better than the oblivious policies. This implies that centralized learning and decentralized execution techniques can make the most of information.

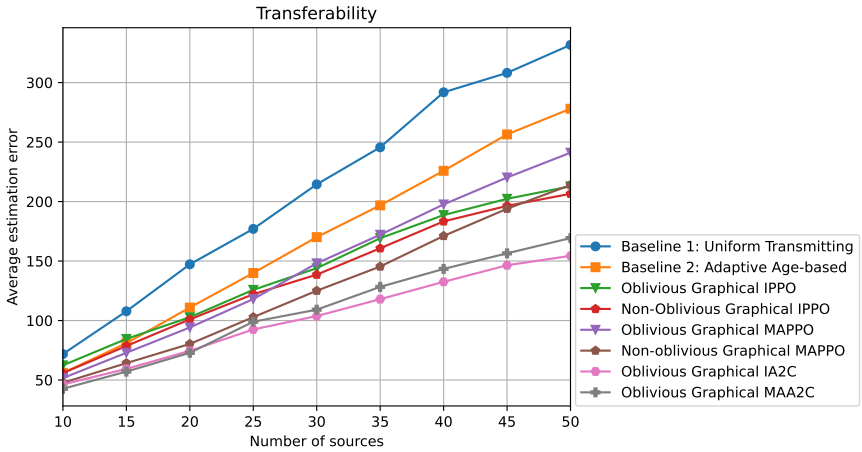


Figure 4.3: The transferability of proposed policies.

(iv) For classical IPPO (the dark orange and gold curves), the estimation error explodes with learning episodes, and the non-oblivious IPPO performs worse than policies IPPO. This is because independent learning techniques suffer from non-stationarity, and more information incurs more non-stationarity. Comparing the graphical IPPO and classical IPPO, we can observe that graphical reinforcement learning can withstand non-stationarity.

The transferability is provided in Figure 4.3. We see that the models trained on 10-node networks can be applied to larger networks. There are three interesting observations.

(i) The graphical IA2C and graphical MAA2C outperform the graphical IPPO and graphical MAPPO, respectively. This is because compared to the A2C framework, the PPO framework has an additional condition on the distribution of policy networks. Hence, the feasible region of the PPO framework is a subset of that in the A2C framework in every learning step.

(ii) The gaps between our proposed policies and baselines increase with the number of sources. We know that the transferability property keeps the sub-optimality. Then, the graphical reinforcement learning framework has more benefits when the number of agents is larger. In addition, when the number of sources is 10, the graphical IPPO performs worse than the adaptive age-based policy (Baseline 2), while it gradually outperforms this baseline as the number of agents increases.

(iii) The graphical IPPO (respectively, the graphical IA2C) transfers better than graphical MAPPO (respectively, MAA2C) after a certain network size. The transferability property holds only in the class of GNN architectures which are built graph filters [213]. In MAPPO and MAA2C, the critic GNN architectures are not built on graph filters. Therefore, the phenomenon happens because the critic GNN architecture violates the transferability. We believe that if we choose the critic GNN architectures built on graph filters, then the graphical MAPPO (respectively, MAA2C) should outperform the graphical IPPO (respectively, IA2C) in any number of agents.

4.7. Conclusion and Future Research

In this chapter, we investigated decentralized sampling and transmission policies to minimize the time-average estimation error and/or time-average age of information in ad-hoc networks with M sources. To capture the permutation equivalency of graphs, we proposed graphical reinforcement learning frameworks. We further proved that our proposed frameworks have transferability properties, i.e., the models trained in small or moderate networks can be applied to large-scale networks. Simulations showed that our proposed policies outperform the state-of-the-art, and verified the transferability hold in our policies.

We will continue the research in two directions. Firstly, providing complete theoretical guarantees for our proposed algorithms. For example, in some cases, the estimates have quantized error, then if we can show that the proposed framework has robustness. Secondly, the framework can be improved in an information-theoretic way, which makes our proposed framework more trustworthy and reliable.

CHAPTER 5

Age-Rate Tradeoffs in Broadcast Networks

In this chapter, we consider erasure networks and devise broadcast strategies that are efficient both in AoI and rate. In the system, there are M senders and M receivers, where every sender transmits packets to the intended receiver, and every receiver can cache all packets delivered. The inherent tradeoff can be explained as follows. On the one hand, a higher rate can correspond to a smaller delay/AoI (both in the sense that queues get emptied faster and that fewer uses of the network are needed in total to transmit a fixed number of information bits). On the other hand, to achieve high rates, we need to wait for the arrival of other packets and change transmission priorities to facilitate coding, and this can lead to a larger AoI. To shed light on the above tradeoff, we build on our previous work [36] and consider an erasure wireless network with M users. Motivated by the success of age-based scheduling in wireless networks, we propose a scheduling framework where we schedule various useful coding actions as opposed to the users. Within this framework, we can capture both rate efficiency and age efficiency. Coding is known to provide significant benefits compared to time sharing especially as the number of users M increases [153, 154, 155, 156]. Our work shows, for the first time, that coding also provides benefits in terms of age and the gain increases by M sharply, especially when the generation rate is small and/or the channel erasure probability is large. More generally, we design deterministic coding policies that minimize the average AoI under given rate constraints.

5.1. Literature Review

Timeliness is key for many applications and it has therefore emerged as a communication design criterion. There are, however, tradeoffs between timeliness and rate in broadcast networks that need to be more deeply revealed. Rate efficiency is often provided by channel coding schemes over multiple realizations of the network and it comes at the cost of large delays. It is, therefore, not clear a-priori what types of tradeoffs exist between rate and timeliness. Prior works have mainly studied point-to-point channels [6, 94, 95]. In erasure channels,[9] proves that when the source alphabet

and channel input alphabet have the same size, a Last-Come First-Serve (LCFS) policy with no coding is optimal. This is in contrast to channel coding schemes that provide rate efficiency by block coding. Considering erasure channels with FCFS M/G/1 queues, [46] finds an optimal block length for channel coding to minimize the average age and average peak age. In the context of broadcast packet erasure channels (BPECs) with feedback, coding is shown to be beneficial for age efficiency with two users [36]. In related work, [96, 97] design optimal precoding schemes to minimize AoI in a MIMO broadcast channel with multiple senders and receivers under FIFO channels without packet management. Reference [98] analyzes AoI in a multicast network with network coding.

5.2. System Model

We consider a model where time is slotted. Transmission occurs in a wireless network which we model by a Broadcast Packet Erasure Channel (BPEC) with M users. Let $[M] = \{1, 2, \cdots, M\}$. At the beginning of time slot k, a packet intended for user i is generated with probability θ_i. In each time slot k, the input of the channel is the packet $X(k)$ and the output at user i is:

$$Y_i(k) = \begin{cases} X(k) & \text{if } Z_i(k) = 1 \\ \Delta & \text{otherwise} \end{cases}$$

where $\{Z_i(k)\}_{k=1}^{\infty}$ is an iid Bernoulli process with probability $1-\epsilon_i$ modeling erasure at user $i \in [M]$ in time slot $k \in \{1, 2, \cdots\}$ and Δ is the symbol denoting erasure. Due to the available feedback, the encoder has the knowledge of $\{\big(Y_1(\ell), Y_2(\ell), \cdots, Y_M(\ell)\big)\}_{\ell=1}^{k}$ in (the beginning of) time $k+1$.

A packet can be cached by other user(s) (with some probability) if it is not delivered to its intended user. Using the available feedback, the encoder can track the cached packets and exploit them as side information in the code design to form more efficient coded packets that are simultaneously useful for multiple users. Such code designs have also appeared in [153, 154, 155, 156] with a focus on rate maximization. Note that the pairs $\big\{(Z_1(k), Z_2(k), \cdots, Z_M(k))\big\}_k$ are iid across time but potentially correlated across $i \in [M]$. Let $\{\big(Z_1, Z_2, \cdots, Z_M\big)\}$ have the same distribution with

$\{\big(Z_1(k), Z_2(k), \cdots, Z_M(k)\big)\}$ for any k. Define

$$\epsilon_i := \Pr\big(Z_i = 0\big), \quad i \in [M] \tag{5.1}$$

$$\sigma(\mathcal{I}) := \Pr\big(Z_i = 0, Z_j = 1, i \in \mathcal{I}, j \in [M]\backslash\mathcal{I}\big) \tag{5.2}$$

$\sigma(\mathcal{I})$ represents the probability such that a packet is erased by users in $\mathcal{I}$, and is cached by users in $[M]\backslash\mathcal{I}$. Note that erasure events at multiple users can be dependent in general, from (5.1) and (5.2), $\sigma(\{i\}) <= \epsilon_i$. In particular, if the channels to individual users are independent and symmetric (with $\epsilon_1 = \cdots = \epsilon_M = \epsilon$), then $\sigma(\mathcal{I}) = (1-\epsilon)^{M-|\mathcal{I}|}\epsilon^{|\mathcal{I}|}$.

In this work, we consider that several packets can be coded by linear network coding through XOR operations only. This is because for broadcast erasure channels with multiple unicast traffic, coding over larger finite fields may impose a larger decoding delay and is often practically less desirable, but such simple coding operations (XOR operations) are practical and lead to low decoding delay [217]. The *transmission delay* is assumed to be a unit time slot. After each transmission, the transmitter receives ACK/NACK feedback from all receivers and can thus calculate and track the aging of information at each user.

Now, we provide the following definitions to concrete our presentation of the paper.

Definition 18. *A packet is called coded if it is formed by linearly combining more than one packet; otherwise, it is called an uncoded packet.*

Definition 19. *Suppose that a coded packet x contains uncoded packets for user i. Then, user i is called a destination (of packet x) if one can instantaneously decode x upon successful delivery (possibly using its locally cached packets).*

Definition 20. *A coded packet x is fully decodable at user i if, upon successful delivery, user i can recover all of the (uncoded) source packets that formed x (possibly from its local cache).*

Note that even if packet x is fully decodable at user i, one may not be a destination of packet x.

Finally, we introduce the indicator $d_{\mathcal{S}}(k)$. Let $d_{\mathcal{S}}(k) = 1$ if users in $\mathcal{S}$ decodes a (coded) packet in time k, and $d_{\mathcal{S}}(k) = 0$ otherwise. In particular, if $\mathcal{S} = \{i\}$, then $d_i(k) = 1$ represents that user i decodes a packet in time k.

Depending on the available caching and coding capabilities, we consider three classes of policies:

(i) policies that benefit from coding by caching *uncoded* packets;

(ii) policies that benefit from coding by caching more general *linearly coded* packets;

(iii) policies that schedule different users and perform no caching/coding [13, 14] (time-sharing policies).

We investigate class (i) policies in Section 5.3, and refer to them as coding policies with uncoded caching. The second class, referred to as coding policies with coded caching, is investigated in Section 5.4. Time-sharing policies [13, 14] form benchmarks in our simulations and are discussed in Appendix D.1.

5.2.1. Coding Actions

The idea of caching and coding on the fly is to cache overheard packets at the users and track them using feedback at the encoder through a *network of virtual queues*. In the rest of the section, we only consider class (i) policies where cached packets are uncoded, while in Section 5.4, we extend the framework to (ii) policies.

Let Q_i denote the queue of incoming packets for user i. If a packet chosen from Q_i is transmitted and received by its intended user, it is removed from the queue. If it is cached by non-intended users, then the packet is removed from Q_i and tracked at another virtual queue(s) by the encoder. Define $Q_{i,\mathcal{S}}$ as the virtual queue that tracks, *at the encoder*, uncoded packets for user i that are cached *only* by the users in $\mathcal{S}$, where $\mathcal{S} \subset [M]\backslash i$. When $\mathcal{S} = \varnothing$, we recover $Q_{i,\varnothing} = Q_i$. Note that the queues $Q_{i,\mathcal{S}}$ are defined so that the set of packets in them are disjoint. Queue $Q_{i,\mathcal{S}}$ contains two types of packets: packets from $Q_{i,\varnothing}$ that are cached by the users in $\mathcal{S}$ (see Example 1), and/or uncoded packets combined within coded packets which are fully decoded. Packets stored in the

virtual queues at the encoder can form efficiently coded packets that are simultaneously useful for multiple users.

Example 1. *Consider users* 1, 2, *and* 3. *Packets* $\{a_i\}_{i\geq 1}$ $\big($*respectively,* $\{b_i\}_{i\geq 1}$ *and* $\{c_i\}_{i\geq 1}\big)$ *are intended for user* 1 $\big($*respectively, user* 2 *and user* 3$\big)$. *In some time slot, consider* 3 *packets* $a_1 \in Q_{1,\{2\}}$, $b_1 \in Q_{2,\{1,3\}}$, *and* $c_1 \in Q_{3,\{2\}}$. *In the next time slot, suppose that the coded packet* $x = b_1 \oplus c_1$ *is transmitted, and is received by user* 1. *Since* $b_1 \in Q_{2,\{1,3\}}$, *then* x *is fully decodable at user* 1, *hence* c_1 *is cached at user* 1.

More generally, consider a set of non-empty queues $\{Q_{\tau_i,\mathcal{S}_{\tau_i}}\}_{i=1}^{\ell}$ where τ_i is a user index ($\tau_i \in [M]$) and $\mathcal{S}_{\tau_i}$ is a subset of $[M]\backslash\tau_i$. Suppose the following condition holds:

$$\mathcal{S}_{\tau_i} \supset \{\cup_{j=1,j\neq i}^{\ell}\tau_j\} \qquad \forall i = 1,\dots \ell. \tag{5.3}$$

Then XORing packets

$$p_i \in Q_{\tau_i,\mathcal{S}_{\tau_i}}$$

forms a coded packet x as follows

$$x = \bigoplus_{i=1}^{\ell} p_i \tag{5.4}$$

which is simultaneously decodable at all users $\{\tau_1,\dots,\tau_\ell\}$. To view condition (5.3) alternatively, draw a side information graph $\mathcal{G}$ [218, 219] with nodes $V = \{1,\dots,M\}$. Add an edge between nodes (i,j) if $Q_{i,\mathcal{S}_i}$ is non-empty for some set $\mathcal{S}_i$ that has j as an element. For example, if $Q_{1,\{2,3,4\}}$ is non-empty, then a directed edge (oriented away node 1) between node 1 and node 2 (respectively, node 3, node 4) exists. On this graph, condition (5.3) corresponds to the subgraph induced by nodes $\{\tau_1,\dots,\tau_\ell\}$ forming a cycle of size ℓ. An example with a cycle of size 3 is shown in Figure 5.1 with 4 users. Consider packets $p_1 \in Q_{1,\{2,3,4\}}$, $p_2 \in Q_{2,\{14\}}$ and $p_4 \in Q_{4,\{12\}}$. Cycle $1 \leftrightarrow 2 \leftrightarrow 4 \leftrightarrow 1$ corresponds to the coded packet $x = p_1 \oplus p_2 \oplus p_4$.

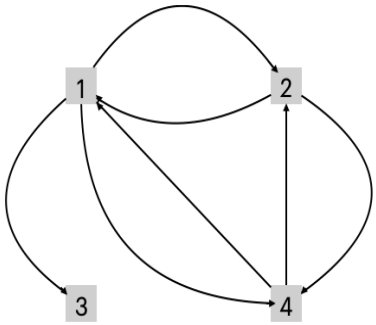


Figure 5.1: Each cycle corresponds to a coded packet.

The subset of nodes $\{i_1, i_2, \cdots, i_j\}$ is defined as a *maximal cliques*[15] if (i) $\{i_1, i_2, \cdots, i_j\}$ forms a cycle, and (ii) $\{i_1, i_2, \cdots, i_j, l\}$ can not form a cycle for any $l \in V \backslash \{i_1, i_2, \cdots, i_j\}$. The *coding actions* we consider in this section correspond to cycles on the side information graph (which has to be updated on the fly after each transmission). In this class, it is sufficient to only consider the maximal cliques because sending a coded packet that corresponds to a subset of a cycle is at most as useful (in terms of the users at which the corresponding packet is decodable) as a coded packet that corresponds to the maximal clique. The number of maximal cliques is of the order of $3^{\frac{M}{3}}$ [220]. Among all possible maximal cliques, we aim to choose (schedule) one that leads to a coding action with the most benefit in terms of information freshness and rate.

Note that each coding action can be described by a set of queues, each storing multiple packets. We allow packet management to form coded packets as it reduces age without impacting the rate. Recall that $Q_{i,\mathcal{S}}$ is the queue that contains packets of user i that are decoded only by the users in $\mathcal{S}$. Thus, if $p_i \in Q_{i,\mathcal{S}}$, for any $\mathcal{S}' \subset \mathcal{S}$ and $\mathcal{S}' \neq \mathcal{S}$, $p_i \notin Q_{i,\mathcal{S}'}$. In addition, if $p_i \in Q_{i,\varnothing}$, then $p_i \notin Q_{i,\mathcal{S}}$ for all $\mathcal{S} \neq \varnothing$. So the map from packets to queues is a surjection. The encoder decides among the following actions, denoted by $A(k)$, and defined below:

- $A(k) = Q_{i,\varnothing}$: a packet is transmitted from $Q_{i,\varnothing}$;
- $A(k) = \oplus_{j=1}^{l} Q_{\tau_j,\mathcal{S}_{\tau_j}}$: a coded packet is transmitted that is formed by an XOR of l packets, one from each of the queues $Q_{\tau_1,\mathcal{S}_{\tau_1}}, Q_{\tau_2,\mathcal{S}_{\tau_2}}, \cdots, Q_{\tau_l,\mathcal{S}_{\tau_l}}$, where $S_{\tau_l} \not\ni \tau_l$ and users $\tau_1, \tau_2, \cdots, \tau_l$

[15]Note that the term "maximal cliques" is defined in undirected graphs. Here, to convey the idea (condition (5.3)) clearly, we abuse the term in directed graphs (side information graphs)

form a *maximal* clique on the side information graph.

Finally, we define an indicator function $t_{i,\mathcal{S}}(k)$ as follows: $t_{i,\mathcal{S}}(k) = 1$ if the latest packet in $Q_{i,\mathcal{S}}$ is encoded and transmitted in time slot k, and $t_{i,\mathcal{S}}(k) = 0$ otherwise. In Figure 5.1, if the encoder transmits $p = a_1 \oplus a_2 \oplus a_4$ in time slot k, then $t_{1,\{2,3,4\}}(k) = 1$, $t_{2,\{1,4\}}(k) = 1$ and $t_{4,\{1,2\}}(k) = 1$.

5.2.2. Age and Rate Efficiency

To capture the freshness of information, we use the metric of *Age of Information* (AoI) defined in [221]. Denote $h_i(k)$ as the AoI of user i in time slot k. The age function $h_i(k)$ increases linearly in time when no delivery for user i occurs and drops with every delivery to a value that represents how old the received packet is. If an outdated packet (for user i) is received (meaning that a more recently generated packet is previously received at user i), then the outdated packet does not offer age reduction, and $h_i(k)$ keeps increasing linearly. Note that we consider a time-discrete model, then $h_i(k)$ increases by 1 after a one-time slot.

Definition 21. *Denote the generation time of the packet received by user i in time slot k as $v_i(k)$. Assuming the initial state $h_i(0) = 1$, the age function $h_i(k)$ evolves as follows:*

$$h_i(k) = \begin{cases} \min\{h_i(k-1) + 1, k - v_i(k)\} & d_i(k) = 1 \\ h_i(k-1) + 1 & d_i(k) = 0. \end{cases}$$

The expected weighted sum of AoI (EAoI) at the users is given by $\mathbb{E}[J_K^\pi]$ where

$$J_K^\pi := \frac{1}{MK} \sum_{k=1}^{K} \sum_{i=1}^{M} \alpha_i h_i^\pi(k), \tag{5.5}$$

the scalars $\alpha_1, \alpha_2, \cdots, \alpha_M$ are weights and the superscript π represents the communication policy. We are interested in minimizing EAoI under given constraints on the rate of communications.

Define the *communication rate* to user i as the number of decoded packets (intended for user i) per time slot in the limit of time. The larger the rate, the fewer packet in the network of virtual queues at the encoder. Let q_i be a strictly positive real value that represents the minimum rate requirement

of user i. Without loss of generality, we assume that $\underline{q} = (q_1, q_2, \cdots, q_M)$ is in the capacity region for which inner and outer bounds are known [14, 20, 153, 154, 155]. Similar to [14], we define the communication rate of user i when policy π is employed as

$$r_i^\pi := \lim_{K\to\infty} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}\big[d_i^\pi(k)\big]. \tag{5.6}$$

The *minimum rate constraint* of each user is thus expressed by

$$r_i^\pi \geq q_i, \quad i = 1, 2, \cdots, M. \tag{5.7}$$

Ultimately, we seek to *schedule the coding actions* to achieve a judicious tradeoff between the EAoI and communication rate, as outlined below. Combining (5.5), (5.6) and (5.7), the objective is given by the following optimization problem:

$$J(\underline{q}) := \min_{\substack{\pi \\ r_i^\pi \geq q_i,\ i\in[M]}} \lim_{K\to\infty} \mathbb{E}[J_K^\pi] \tag{5.8}$$

5.2.3. Notation

We use the notations $\mathbb{E}(\cdot)$ and $\Pr(\cdot)$ for expectation and probability, respectively. We denote scalars by lowercase letters, e.g. s, vectors by underlined lowercase letters, e.g., $\underline{s}$, and random variables by capital letters, e.g. S. Sets are denoted by calligraphic letters, e.g. $\mathcal{S}$. We use M to denote the number of users, and K to denote the time horizon. The notation $[n]$ denotes the set $\{1, 2, \cdots, n\}$. For two sets $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A} \subset \mathcal{B}$ represents that $\mathcal{A}$ is a subset of $\mathcal{B}$. We summarize the notations in Table 5.1.

5.3. Scheduling Coding Actions With Uncoded Caching

In this section, we consider coding policies with uncoded caching, i.e., all cached packets are uncoded. We develop and analyze deterministic policies that schedule the coding actions to optimize (5.8). Our framework can be generalized to the case where we allow the caching of coded packets as outlined in Section 5.4.

M	The number of sources
K	The time horizon
θ_i	The generation/arrival rate of new packets for user i
ϵ_i	The channel erasure probability for user i
$d_{\mathcal{S}}(k)$	The indicator of decoding at users in $\mathcal{S}$
$A(k)$	Coding actions of the encoder
$h_i(k)$	The destination's AoI at time k w.r.t source i
α_i	The weight for user i
π	A specific transmission policy
J_K^π	Expected weighted sum of AoI (EAoI)
r_i^π	The communication rate of user i
q_i	The minimum rate requirement of user i
$w_{i,\mathcal{S}}(k)$	The AoI of $Q_{i,\mathcal{S}}$ in time slot k

Table 5.1: Useful notations in Chapter 5.

5.3.1. Encoder's Age of Information

In order to optimize for age, we first define the AoI of queues $Q_{i,\mathcal{S}}$ in the virtual network of queues (at the encoder) and explain their time evolution. The following Lemma is proved in Appendix D.2

Lemma 12. *If $p_j \in Q_{i,\mathcal{S}}$ has the generation time k_j, $j \in \{1,2\}$, and $k_2 > k_1$, then (encoding and) transmitting p_2 can not be worse than (encoding and) transmitting p_1 in terms of AoI.*

If $\mathcal{S} = \varnothing$, denote the AoI of $Q_{i,\varnothing}$ as $w_{i,\varnothing}(k)$, and the generation time of latest packet in $Q_{i,\varnothing}$ as k'. Note that all new packets are generated in $Q_{i,\varnothing}$.

Definition 22. *In time slot k,*

$$w_{i,\varnothing}(k) = k - k'. \tag{5.9}$$

By convention, $w_{i,\varnothing}(0) = h_i(0)$.

The evolution of the AoI at the queue $Q_{i,\varnothing}$ is as follows: $w_{i,\varnothing}(k)$ drops to 0 if a new packet is

generated; otherwise it increases by 1. Thus, the recursion of $w_{i,\varnothing}(k)$ is

$$w_{i,\varnothing}(k+1) = \begin{cases} 0 & \text{a new packet is generated} \\ w_{i,\varnothing}(k)+1 & \text{otherwise.} \end{cases} \tag{5.10}$$

Now we consider the AoI of $Q_{i,\mathcal{S}}$ with $\mathcal{S} \neq \varnothing$. Let the generation time of the latest packet in $Q_{i,\mathcal{S}}$ be k'. Similar with Definition 22,

Definition 23. *In time slot k, $w_{i,\mathcal{S}}(k) = \min\{k-k', h_i(k)\}$. By convention, $w_{i,\mathcal{S}}(0) = h_i(0)$.*

From Definition 23, if $k - k' > h_i(k)$, i.e., an older packet is cached in $Q_{i,\mathcal{S}}$, then the cached one can not provide fresh information, thus let the AoI of $Q_{i,\mathcal{S}}$ equal to the AoI of user i.

One can see that $w_{i,\mathcal{S}}(k)$ increases by 1 unless (i) a fresher packet enters into the virtual queue $Q_{i,\mathcal{S}}$; (ii) the latest packet in $Q_{i,\mathcal{S}}$ moves to other virtual queues at the encoder[16]. Let $\mathcal{S}' \subset \mathcal{S}$. The recursion of $w_{i,\mathcal{S}}(k)$ is

$$w_{i,\mathcal{S}}(k+1) = \begin{cases} \min\{w_{i,\varnothing}(k)+1, h_i(k)+1\} & \mathcal{P}^{(1)}_{i,\mathcal{S}}(k) \\ \min\{w_{i,\mathcal{S}'}(k)+1, h_i(k)+1\} & \mathcal{P}^{(2)}_{i,\mathcal{S}}(k) \\ \min\{w_{i,\mathcal{S}}(k)+1, h_i(k)+1\} & \text{otherwise} \end{cases}. \tag{5.11}$$

where

$$\mathcal{P}^{(1)}_{i,\mathcal{S}}(k) = \{d_i(k) = 0, d_{\mathcal{S}}(k) = 1, t_{i,\varnothing}(k) = 1\}$$

and

$$\mathcal{P}^{(2)}_{i,\mathcal{S}}(k) = \{\mathcal{S}' \subset \mathcal{S}, d_i(k) = 0, d_{\mathcal{S}\setminus\mathcal{S}'}(k) = 1, t_{i,\mathcal{S}'}(k) = 1\}.$$

[16]For example, If packet $p_i \in Q_{i,\mathcal{S}}$ is recovered by some other users in $\mathcal{I}$, $\mathcal{I} \cap \mathcal{S} = \varnothing$, then $p_i \in Q_{i,\mathcal{I}\cup\mathcal{S}}$ and $p_i \notin Q_{i,\mathcal{S}}$.

From (5.10) and (5.11), the recursion of $h_i(k)$ is

$$h_i(k+1) = \begin{cases} w_{i,\mathcal{S}}(k) + 1 & t_{i,\mathcal{S}}(k) = 1, d_i(k) = 1 \\ h_i(k) + 1 & \text{otherwise} \end{cases}. \tag{5.12}$$

5.3.2. Age-based Scheduling in Two-user Networks

In this section, we consider a representative example of scheduling coding actions with uncoded caching – age-based scheduling in two-user broadcast packet erasure channels.

To simplify the model in Section 5.2, we ignore the impact of communication rate, and consider the users are scheduled with the goal of minimizing age. Furthermore, we simplify the model where new packets are generated for the users ($\theta_i = 1$ for $i \in [2]$) at the beginning of every time slot, and they replace any undelivered packets from previous time slots. In each time slot k, the encoder decides between the following three actions, denoted by $A(k) \in \{1, 2, 3\}$ and defined below:

- $A(k)=1$: a packet is transmitted from Q_1;
- $A(k)=2$: a packet is transmitted from Q_2;
- $A(k)=3$: a coded packet is transmitted from $Q_{1,\{2\}}$, $Q_{2,\{1\}}$.

From Lemma 12, to attain the optimal age in the above class of 3−action coding algorithms, we can assume, without loss of generality, that all queues are of buffer size 1 (Please refer to [36, Lemma 1] for the detailed proof).

Then, we devise deterministic policies using techniques from Lyapunov Optimization. Denote $\vec{h}(k) = (h_1(k), h_2(k))$ and $\vec{s}(k) = (h_1(k), h_2(k), w_{1,\{2\}}(k), w_{2,\{1\}}(k))$. Define the Lyapunov function

$$L(\vec{h}(k)) = \sum_{i=1}^{2} \beta_i h_i(k), \tag{5.13}$$

and the one-slot Lyapunov Drift

$$\Theta(\vec{h}(k)) = \mathbb{E}[L(\vec{h}(k+1)) - L(\vec{h}(k))|\vec{s}(k)]. \tag{5.14}$$

Here, to coincide with (5.5), we can set $\beta_i = \alpha_i$ for $i \in [2]$.

Definition 24. *In each slot k, the MW policy chooses the action that has the maximum weight as shown in Table 5.2:*

$A(k)$	Weights
1	$\beta_1(1-\epsilon_1)h_1(k)$
2	$\beta_2(1-\epsilon_2)h_2(k)$
3	$\sum_i \beta_i(1-\epsilon_i)1_{\{w_{i,[2]\setminus\{i\}}(k)>0\}}h_i(k)$

Table 5.2: Coding actions and their weights in 2-user networks.

Theorem 15. *The MW policy defined in Definition 24 minimizes the one-slot Lyapunov Drift in each slot.*

Proof. The proof can be found in [36, Theorem 1]. □

5.3.3. Age-Rate Max-Weight Scheduling

It is well established that coding can enhance the communication rate in erasure broadcast channels [153], and may incur additional delays. To seek efficiency in both AoI and communication rate, similar to [14, 36, 222], we propose Age-Rate Max-Weight (ARM) policies to minimize EAoI in (5.8) under rate constraints.

We define the *age-gain* of queue $Q_{i,\mathcal{S}}$ (for user i), where $\mathcal{S} \subset [M]\setminus i$ as follows:

$$\delta_{i,\mathcal{S}}(k) = h_i(k) - w_{i,\mathcal{S}}(k). \tag{5.15}$$

The term $\delta_{i,\mathcal{S}}(k)$ quantifies how much the instantaneous age of information of user i reduces upon successful delivery from the encoder's virtual queue $Q_{i,\mathcal{S}}$. If $Q_{i,\mathcal{S}}$ is empty or contains old packets,

then by the definition of $w_{i,\mathcal{S}}(k)$, $\delta_{i,\mathcal{S}}(k) = 0$.

Let $y_i(k)$ be the *throughput debt* associated with node i at the beginning of slot k [14, Eqn (35)]. It evolves as follows:

$$y_i(k+1) = kq_i - \sum_{\tau=1}^{k} d_i^{\pi}(\tau). \tag{5.16}$$

The value of kq_i is the minimum average number of packets that user i should have decoded by slot k and $\sum_{\tau=0}^{k} d_i^{\pi}(\tau)$ is the total number of recovered packets in the same interval. In fact, the strong stability of the process $y_i^+(k)$ is sufficient to establish that the minimum rate constraint, $r_i^{\pi} \geq q_i$, is satisfied [14], [223, Theorem 2.8].

Define the encoder's state in time slot k as $\vec{s}(k) = \Big(\{h_i(k)\}_i, \{w_{i,\mathcal{S}}(k)\}_{i,\mathcal{S}}, \{x_i(k)\}_i\Big)$, and the Lyapunov function $L\big(\vec{s}(k)\big)$ as

$$L\big(\vec{s}(k)\big) = \sum_{i=1}^{M} \beta_i h_i(k) + \lambda \sum_{i=1}^{M} \big(y_i^+(k)\big)^2 \tag{5.17}$$

where $\beta_i, \lambda > 0$ are super-parameters. Here, the quadratic function for $y_i(k)$ is to maximize the rate [14, 20, 222], and the linear function for $h_i(k)$ is to simplify the derivation. The one-slot Lyapunov Drift is defined as

$$\Theta(k) = \mathbb{E}\Big[L\big(\vec{s}(k+1)\big) - L\big(\vec{s}(k)\big)|\vec{s}(k)\Big]. \tag{5.18}$$

Define the *rate-gain* of user i in time slot k as follows:

$$f_i(k) = \Big(\big(y_i(k) + q_i\big)^+\Big)^2 - \Big(\big(y_i(k) + q_i - 1\big)^+\Big)^2. \tag{5.19}$$

Definition 25. *In each slot k, the ARM policy chooses the action that has the maximum weight in Table 5.3.*

Remark 24. *When AoI is the only metric in decision making (i.e., $q_i = 0$ for all i) (see Lemma 12),*

$A(k)$	Weights
$Q_{i,\varnothing}$	$(1-\epsilon_i)\Big(\beta_i\delta_{i,\varnothing}(k)+\lambda f_i(k)\Big)$
$\oplus_{u\in[l]}Q_{\tau_u,\mathcal{S}_{\tau_u}}$	$\sum_{u=1}^{l}\beta_{\tau_u}\delta_{\tau_u,\mathcal{S}_{\tau_u}}(k)(1-\epsilon_{\tau_u})+\lambda\sum_{u=1}^{l}(1-\epsilon_{\tau_u})f_{\tau_u}(k)$

Table 5.3: Coding actions and their weights in M-user networks.

we can assume that the buffer size of every queue is 1 *and the stability region is* $\{\theta_i \leq 1, i = 1, 2, \cdots, M\}$.

Remark 25. *We have observed in simulations that a good approximation of the above ARM policy is obtained by choosing the maximal clique size* l *to be* 2.

Theorem 16. *The ARM policy defined in Definition 25 minimizes the one-slot Lyapunov Drift in each slot.*

Proof. The proof is given in Appendix D.3. □

Now we set to obtain an upper bound on AoI under the rate constraints. In this paper, we consider an upper bound with $M = 3$, one can generalize to an upper bound with arbitrary M using the same idea straightforwardly and systematically. Let $C^{uncoded}$ be the set of all tuples $\underline{q} = (q_1, q_2, q_3)$ for which $\{y_i^+(k)\}_{i=1}^3$ is strongly stabilized using the coding actions $A(k)$ defined in Section 5.2.1.

Generally speaking, the ARM policy (a centralized policy) outperforms the randomized policies (decentralized policies) in terms of EAoI. To obtain an upper bound on EAoI, we introduce a stationary randomized policy in which each action is chosen with a probability as described below:

$$\Pr(A(k) = Q_{i,\varnothing}) = \mu_{i,\varnothing} \tag{5.20}$$

$$\Pr\{A(k) = \oplus_{j\in[l]}Q_{\tau_j,\mathcal{S}_{\tau_j}}\} = \mu_{(\tau_1,\mathcal{S}_{\tau_1},\cdots,\tau_l,\mathcal{S}_{\tau_l})}. \tag{5.21}$$

In (5.21), we do not distinguish different permutations on $\{\tau_j, \mathcal{S}_{\tau_j}\}_j$. For example, $\mu_{1,\{3\},3,\{1\}}$ and

$\mu_{3,\{1\},1,\{3\}}$ are exactly the same. Clearly, we have

$$\sum_{j=1}^{3}\mu_{i,\varnothing}+\sum_{l=2}^{3}\sum_{\tau_1,\mathcal{S}_{\tau_1},\cdots,\tau_l,\mathcal{S}_{\tau_l}}\mu_{(\tau_1,\mathcal{S}_{\tau_1},\cdots,\tau_l,\mathcal{S}_{\tau_l})}=1. \tag{5.22}$$

To simplify the analysis, we assume that the channel to individual users is independent and symmetric, i.e., $\epsilon_i=\epsilon$, $\theta_i=\theta$, and $q_i=q$ for $i\in[3]$. Let τ_1,τ_2,τ_3 be the user index, and

$$\mu_{i,\varnothing}=\mu,\ \mu_{\tau_1,\{\tau_2\},\tau_2,\{\tau_1\}}=\zeta_1,\ \mu_{\tau_1,\{\tau_2\},\tau_2,\{\tau_1,\tau_3\}}=\zeta_2$$
$$\mu_{\tau_1,\{\tau_2,\tau_3\},\tau_2,\{\tau_1,\tau_3\}}=\zeta_3,\ \mu_{\tau_1,\{\tau_2,\tau_3\},\tau_2,\{\tau_1,\tau_3\},\tau_3,\{\tau_1,\tau_2\}}=\zeta_4.$$

Then, (5.22) is reduced to $3\mu+3\zeta_1+6\zeta_2+3\zeta_3+\zeta_4=1$. Upper bounding $J(\underline{q})$ for the ARM policy, we prove the following result in Appendix D.4.

Theorem 17. *For any $\underline{q}\in C^{uncoded}$, we have the following upper bound on $J(\underline{q})$:*

$$\begin{aligned}\min_{\mu}\quad & \frac{\frac{1}{3}\sum_{i=1}^{3}\alpha_i}{\theta}+\frac{\frac{1}{3}\sum_{i=1}^{3}\alpha_i}{\mu(1-\epsilon)}+\lambda\\ s.t\quad & 3\mu+3\zeta_1+6\zeta_2+3\zeta_3+\zeta_4=1\\ & \mu(1-\epsilon^3)\geq q,\ (\mu+\zeta_2)(1-\epsilon^2)+\zeta_1(1-\epsilon)\geq q\\ & \mu(1-2\epsilon^2+\epsilon^3)+2\zeta_1(1-\epsilon)+2\zeta_2(1-\epsilon^2)\geq q\\ & \big(\mu+2\zeta_1+4\zeta_2+2\zeta_3+\zeta_4\big)(1-\epsilon)\geq q,\ \mu\geq 0,\ \zeta_j\geq 0,\ j\in[4].\end{aligned} \tag{5.23}$$

Remark 26. *On the right-hand side of the objective function in* (5.23)*, the parameter λ shows the influence of rate constraints on the upper bound. The larger λ, the more important role of communication rate in decision-making, which results in the larger AoI. Hence, the upper bound in* (5.23) *increases according to the increase of λ.*

5.4. Practical Scheduling Coding Actions with Coded Caching

In this section, we consider the coding policies with coded caching, where coded packets can be cached by users.

5.4.1. Coding Actions

First of all, we introduce a new class of queues, denoted by $Q_{\mathcal{I},\mathcal{K}}$: *coded packets are formed by uncoded packets intended for users in $\mathcal{I}$ only, and cached by users in $\mathcal{K}$ only* (see Example 2-(1)). A coded packet in $Q_{\mathcal{I},\mathcal{K}}$ may contain more than one uncoded packet intended for the same user (see Example 2-(2)). In addition, note that the intersection of $\mathcal{I}$ and $\mathcal{K}$ can be non-empty. It is obvious that the coding situation is extremely sophisticated when the number of users is large. Thus, the complexity of coding actions becomes extremely large because of coded caching.

Example 2. *Suppose we have users 1, 2, 3, and 4. Packets $\{a_i\}_{i\geq 1}$ (respectively, $\{b_i\}_{i\geq 1}$, $\{c_i\}_{i\geq 1}$, and $\{d_i\}_{i\geq 1}$) are intended for user 1 (respectively, 2, 3, and 4). Let $b_1 \in Q_{2,\{3,4\}}$, $c_1 \in Q_{3,\{4\}}$, and $d_1 \in Q_{4,\{3\}}$.*

(1) If the packet $x = b_1 \oplus c_1 \oplus d_1$ is transmitted and received by users 1 and 2[17], both users can not decode it, then $\mathcal{I} = \{2,3,4\}$ and $\mathcal{K} = \{1,2\}$.

(2) Let $b_2 \in Q_{2,\{3,4\}}$. The coding action $x = b_1 \oplus b_2 \oplus c_1 \oplus d_1$ is a valid action and the coded packet x contains two packets (b_1, b_2) intended for user 2.

To simplify the analysis, we specialize our coding policies by the following assumptions.

Assumption 5. *(1) $\mathcal{I} \cap \mathcal{K} = \varnothing$.*

(2) If $x \in Q_{\mathcal{I},\mathcal{K}}$ and $x = \oplus_{j\in[l]} p_j$, then $p_1, p_2, \cdots, p_l$ are intended for distinct users in $\mathcal{I}$.

Assumption 5 (1) implies that one can not cache a coded packet that contains an uncoded packet intended for oneself. In other words, a destination user can extract the uncoded packet (intended for oneself) from coded packets upon successful delivery. Under Assumption 5 (2), two uncoded

[17]Note that we consider the coded caching, then coding action $b_1 \oplus c_1 \oplus d_1$ is a valid action.

packets for an identical user can not be encoded in one coded packet. Finally, we define the *length* of a coded packet as the number of uncoded packets encoded within it. From Assumption 5 (2), the length of a coded packet is at most M.

Similar to Section 5.2.1, we consider a set of non-empty queues $\{Q_{\tau_i,\mathcal{S}_{\tau_i}}\}_{i=1}^{\ell_1}$ and $\{Q_{\mathcal{I}_j,\mathcal{K}_j}\}_{j=1}^{\ell_2}$, suppose the following conditions hold:

$$\begin{aligned} &\mathcal{S}_{\tau_m} \supset \{\tau_i\}_{i=1,i\neq m}^{\ell_1} \cup \{\mathcal{I}_j\}_{j=1}^{\ell_2}, \quad \forall m = 1,\dots \ell_1 \\ &\mathcal{K}_m \supset \{\tau_i\}_{i=1}^{\ell_1} \cup \{\mathcal{I}_j\}_{j=1,j\neq m}^{\ell_2}, \quad \forall m = 1,\dots \ell_2 \\ &\tau_i \cap \mathcal{I}_j = \varnothing, \quad \mathcal{I}_i \cap \mathcal{I}_j = \varnothing. \end{aligned} \tag{5.24}$$

The last condition in (5.24) comes from Assumption 5 (2) directly. Then XORing packets

$$p_i \in Q_{\tau_i,\mathcal{S}_{\tau_i}}, \quad x_j \in Q_{\mathcal{I}_j,\mathcal{K}_j}$$

forms a coded packet x as follows

$$x = \bigoplus_{i=1}^{\ell_1} p_i \bigoplus_{j=1}^{\ell_2} x_j \tag{5.25}$$

which is simultaneously decodable at all users $\{\tau_1,\dots,\tau_{\ell_1}\} \cup \{\mathcal{I}_j\}_{j=1}^{\ell_2}$. To view condition (5.24) alternatively, draw a side information graph $\mathcal{G}'$ with nodes $V' = \{1,\dots,M\} \cup \{\mathcal{I}_i\}_i$, where $\mathcal{I}_i$ can be regarded as a "*big*" node. Add an edge between nodes (i,m) $\big($respectively, $(\mathcal{I}_j,m)\big)$ if $Q_{i,\mathcal{S}_i}$ $\big($respectively, $Q_{\mathcal{I}_j,\mathcal{K}_j}\big)$ is non-empty for some set $\mathcal{S}_i$ $\big($respectively, $\mathcal{K}_j\big)$ that has m as an element. On this graph, condition (5.24) corresponds to the subgraph induced by a clique of size $\ell_1 + \ell_2$.

Note that the number of coding actions increases exponentially due to caching coded packets. For clarity of ideas, we explain and derive for $M = 3$, one can extend in a straightforward manner. Denote the packets intended for user 1, user 2 and user 3 as $\{a_t\}_{t\geq 1}$, $\{b_t\}_{t\geq 1}$ and $\{c_t\}_{t\geq 1}$, respectively. From Assumption 5, the length of coded packets is at most 3. Coded packets with length 3 must have forms as $a_{t_1} \oplus b_{t_2} \oplus c_{t_3}$, and it must be decoded by all users $\big($by Assumption 5 (1)$\big)$. In other

words, a coded packet that is cached can only have length 2. Let $\mathcal{I} \in \mathcal{X} = \big\{\{1,2\},\{1,3\},\{2,3\}\big\}$ and $u = [3]\backslash\mathcal{I}$. Then the notation $Q_{\mathcal{I},\mathcal{K}}$ is reduced to $Q_{\mathcal{I},u}$. In $Q_{\mathcal{I},u}$, coded packets for users in $\mathcal{I}$ are cached by user u, *and packets in $Q_{\mathcal{I},u}$ can be decoded by users in $\mathcal{I}$ (by Assumption 5 (1)).* It is obvious to show that if $u_1 \neq u_2$, then the encoder would not encode packets from $Q_{\mathcal{I}_1,u_1}$, $Q_{\mathcal{I}_2,u_2}$. In fact, suppose that $x_1 \in Q_{\mathcal{I}_1,u_1}$, $x_2 \in Q_{\mathcal{I}_2,u_2}$, then the length of coded packet $x = x_1 \oplus x_2$ is 4, which contradicts with Assumption 5 (2). In each time slot k, the encoder decides among the following actions, denoted by $A(k)$, and defined below:

- $A(k) = Q_{i,\varnothing}$: a packet is transmitted from $Q_{i,\varnothing}$;
- $A(k) = \oplus_{j=1}^{l} Q_{\tau_j,\mathcal{S}_{\tau_j}}$: a coded packet is transmitted from $Q_{\tau_1,\mathcal{S}_{\tau_1}},\cdots,Q_{\tau_l,\mathcal{S}_{\tau_l}}$, where users $\tau_1,\cdots,\tau_l$ form a maximal clique on the side information graph, $l = 2,3$.
- $A(k) = Q_{\tau_i,\mathcal{S}_{\tau_i}} \oplus Q_{\mathcal{I},u}$: a coded packet is transmitted from $Q_{\tau_i,\mathcal{S}_{\tau_i}}, Q_{\mathcal{I},u}$, where $\tau_i, \mathcal{I}$ form a (general) maximal clique on the side information graph.

5.4.2. Age-Rate Max-Weighted Scheduling

Similar to Section 5.3.3, we propose an Age-Rate Max-Weighted policy, and a corresponding upper bound. Before obtaining the Age-Rate Max-Weighted policy, we first need to define the AoI of $Q_{\mathcal{I},u}$. Without loss of generality, we fix $\mathcal{I} = \{1,2\}$, $u = 3$. Let $x_1, x_2 \in Q_{\{1,2\},3}$, and $x_1 = a_{t_1} \oplus b_{t_2}$, $x_2 = a_{t_3} \oplus b_{t_4}$. By Assumption 5 (1), it suffices to assume that x_1, x_2 both can be decoded by users 1 and 2. Denote the generation time of a_{t_1}, a_{t_3}, b_{t_2}, b_{t_4} as $e(t_1)$, $e(t_3)$, $v(t_2)$, $v(t_4)$, respectively. Let α_1, α_2, and α_3 be defined in (5.5). If $\alpha_1 e(t_1) + \alpha_2 v(t_2) > \alpha_1 e(t_3) + \alpha_2 v(t_4)$, then (encoding and) transmitting x_2 can not be worse than (encoding and) transmitting x_1 in terms of AoI (see Appendix D.5).

For $\mathcal{I} = \{i,j\}$ and any $x \in Q_{\mathcal{I},u}$, denote the corresponding generation time vector of uncoded packets within x as (e_i, v_j). Suppose that packet $x_0 \in Q_{\mathcal{I},u}$ has the maximum weighted sum of generation time[18], i.e., $x_0 = \max\{x \in Q_{\mathcal{I},u} | \alpha_i e_i + \alpha_j v_j\}$, and denote the corresponding generation

[18]Packet x_0 is not always the latest packet in $Q_{\mathcal{I},u}$.

time vector as $(e_{i,0}, v_{j,0})$. Now, we define the AoI of $Q_{\mathcal{I},l}$ as

$$w_{\mathcal{I},u}(k) = \big(w^{(i)}_{\mathcal{I},u}(k), w^{(j)}_{\mathcal{I},u}(k)\big) \tag{5.26}$$

where

$$w^{(i)}_{\mathcal{I},u}(k) = \min\{k - e_{i,0}, h_i(k-1)+1\},\ w^{(j)}_{\mathcal{I},u}(k) = \min\{k - v_{j,0}, h_j(k-1)+1\}$$

and $w_{\mathcal{I},u}(0) = (h_i(0), h_j(0))$. Similar with (5.15), we can define *age-gain of* $Q_{\mathcal{I},u}$ as

$$\delta_{\mathcal{I},u}(k) = h_{\mathcal{I}}(k) - w_{\mathcal{I},u}(k) \triangleq (\delta^{(i)}_{\mathcal{I},u}(k), \delta^{(j)}_{\mathcal{I},u}(k)) \tag{5.27}$$

where $h_{\mathcal{I}}(k) = \big(h_i(k), h_j(k)\big)$. Define $\vec{b}(k) = \Big(\{h_i(k)\}_i, \{w_{i,\mathcal{S}}(k)\}_{i,\mathcal{S}}, \{w_{\mathcal{I},u}(k)\}_u, \{x_i(k)\}_i\Big)$. The Lyapunov function $L\big(\vec{b}(k)\big)$ is defined as

$$L\big(\vec{b}(k)\big) = \sum_{i=1}^{M} \beta_i h_i(k) + \lambda \sum_{i=1}^{M} \big(y_i^+(k)\big)^2, \tag{5.28}$$

where $\lambda, \beta_i > 0$ are positive super-parameters. We also define the one-slot Lyapunov Drift as

$$\Theta(k) = \mathbb{E}\Big[L\big(\vec{b}(k+1)\big) - L\big((\vec{b}(k)\big)|\vec{b}(k)\Big]. \tag{5.29}$$

Definition 26. *Let* $f_i(k)$ *be defined in* (5.19)*. The ARM policy chooses the action that has the maximum weight as follows:*

$A(k)$	Weights
$Q_{i,\varnothing}$	$(1-\epsilon_i)\Big(\beta_i \delta_{i,\varnothing}(k) + \lambda f_i(k)\Big)$
$\oplus_{u\in[l]} Q_{\tau_u,\mathcal{S}_{\tau_u}}$	$\sum_{u=1}^{l} \beta_{\tau_u}\delta_{\tau_u,\mathcal{S}_{\tau_u}}(k)(1-\epsilon_{\tau_u}) + \lambda \sum_{u=1}^{l}(1-\epsilon_{\tau_u}) f_{\tau_u}(k)$
$Q_{u,i} \oplus Q_{u,\mathcal{I}}$	$\beta_u \delta_{u,i}(k)(1-\epsilon_u) + \lambda(1-\epsilon_u) f_u(k) + \beta_i \delta^{(i)}_{\mathcal{I},u}(k)(1-\epsilon_i) + \lambda(1-\epsilon_i) f_i(k)$
$Q_{u,\mathcal{I}} \oplus Q_{\mathcal{I},u}$	$\beta_u \delta_{u,\mathcal{I}}(k)(1-\epsilon_u) + \sum_{\mathcal{I}: i\in\mathcal{I}} \beta_i \delta^{(i)}_{\mathcal{I},u}(k)(1-\epsilon_i) + \lambda \sum_{i\in[3]}(1-\epsilon_i) f_i(k)$

Theorem 18. *The ARM policy defined in Definition 26 minimizes the one-slot Lyapunov Drift in*

each slot.

Proof. The proof is very similar to that of Theorem 16 in Appendix D.3. □

Finally, we set to obtain an upper bound on AoI under the rate constraints. Let C^{coded} be the set of all tuples $\underline{q} = (q_1, q_2, q_3)$ for which $\{y_i^+(k)\}_{i=1}^3$ is strongly stabilized using coding actions $A(k)$ defined in Section 5.4.1. Similar to Section 5.3, we introduce a stationary randomized policy in which

$$\Pr(A(k) = Q_{i,\varnothing}) = \mu_{i,\varnothing} \tag{5.30}$$

$$\Pr(A(k) = \oplus_{j\in[l]} Q_{\tau_j,\mathcal{S}_{\tau_j}}) = \mu_{\tau_1,\mathcal{S}_{\tau_1},\cdots,\tau_l,\mathcal{S}_{\tau_l}} \tag{5.31}$$

$$\Pr(A(k) = Q_{\tau_i,\mathcal{S}_{\tau_i}} \oplus Q_{\mathcal{I},u}) = \mu_{\tau_i,\mathcal{S}_{\tau_i},\mathcal{I},u} \tag{5.32}$$

In (5.31) $\big($respectively, (5.32)$\big)$, we do not distinguish different permutations on $\{\tau_j, \mathcal{S}_{\tau_j}\}_j$ $\big($respectively, $\{\tau_i, \mathcal{S}_{\tau_i}\}$ and $\{\mathcal{I}, u\}\big)$. Then, to simplify the complicated coding situation, we consider a stationary randomized policy in an independent and symmetric channel, i.e., $\epsilon_i = \epsilon$, $\theta_i = \theta$, and $q_i = q$ for $i \in [3]$, and $\sigma(\mathcal{I}) = \epsilon^{|\mathcal{I}|}(1-\epsilon)^{3-|\mathcal{I}|}$. By symmetry, we define

$$\mu_{i,\varnothing} = \mu,\ \mu_{\tau_1,\{\tau_2\},\tau_2,\{\tau_1\}} = \zeta_1,\ \mu_{\tau_1,\{\tau_2\},\tau_2,\{\tau_1,\tau_3\}} = \zeta_2,\ \mu_{\tau_1,\{\tau_2,\tau_3\},\tau_2,\{\tau_1,\tau_3\}} = \zeta_3,$$

$$\mu_{\tau_1,\{\tau_2,\tau_3\},\tau_2,\{\tau_1,\tau_3\},\tau_3,\{\tau_1,\tau_2\}} = \zeta_4,\ \mu_{\{\tau_1,\tau_2\},\tau_3,\tau_3,\{\tau_1\}} = \zeta_5,\ \mu_{\{\tau_1,\tau_2\},\tau_3,\tau_3,\{\tau_1,\tau_2\}} = \zeta_6.$$

Upper bounding $J(\underline{q})$ for the ARM policy, we prove the following result in Appendix D.6.

Theorem 19. *For any $\underline{q} \in C^{coded}$, we have the following upper bound on $J(\underline{q})$:*

$$\begin{aligned}
\min_{\mu} \quad & \frac{\frac{1}{3}\sum_{i=1}^{3}\alpha_i}{\theta} + \frac{\frac{1}{3}\sum_{i=1}^{3}\alpha_i}{\mu(1-\epsilon)} + \lambda \\
s.t \quad & 3\mu + 3\zeta_1 + 6\zeta_2 + 3\zeta_3 + \zeta_4 + 6\zeta_5 + 3\zeta_6 = 1 \\
& \mu(1-\epsilon^3) \geq q,\ (\mu+\zeta_2)(1-\epsilon^2) + \zeta_1(1-\epsilon+\epsilon^2-\epsilon^3) \geq q \\
& \mu(1-2\epsilon^2+\epsilon^3) + 2\zeta_1(1-\epsilon+\epsilon^2-\epsilon^3) + 2\zeta_2(1-\epsilon^2) \geq q \\
& \big(\mu + 2\zeta_1 + 4\zeta_2 + 2\zeta_3 + \zeta_4\big)(1-\epsilon) + 2\zeta_1(\epsilon^2-\epsilon^3) \geq q \\
& \big(\mu + 2\zeta_1 + 4\zeta_2 + 2\zeta_3 + \zeta_4 + 2\zeta_5 + \zeta_6\big)(1-\epsilon) + \zeta_1(\epsilon^2-\epsilon^3) \geq q \\
& \big(\mu + 2\zeta_1 + 4\zeta_2 + 2\zeta_3 + \zeta_4 + 4\zeta_5 + 2\zeta_6\big)(1-\epsilon) \geq q,\ \mu \geq 0,\ \zeta_j \geq 0,\ j \in [6].
\end{aligned} \tag{5.33}$$

5.5. Lower Bound

In prior works [14, 36, 37], lower bounds were found on AoI as a function of the communication rate. Similar to [37, Section III], we derive two lower bounds on the achievable age. The first lower bound is derived by assuming that there is always a fresh packet to be delivered. The second one assumes that all packets are delivered instantaneously upon arrivals.

Theorem 20. *For any policy π with communication rate r_i^π, we have the following lower bounds on $J^\pi(\underline{q})$ in (5.8):*

$$J^\pi(\underline{q}) \geq \frac{M}{2\sum_{i=1}^{M} r_i^\pi/\alpha_i} + \sum_{i=1}^{M}\frac{\alpha_i}{2M} \tag{5.34}$$

$$J^\pi(\underline{q}) \geq \frac{1}{M}\sum_{i=1}^{M}\frac{\alpha_i}{\theta_i} \tag{5.35}$$

Proof. The proof of (5.34) can be found in [37, Appendix B], and the proof of (5.35) can be found in [37, Appendix C]. □

In (5.34), as the communication rate increases, the lower bound on EAoI decreases. The rate terms r_i^π in (5.34) satisfy $r_i^\pi \geq q_i$, but it is not clear if we can replace them with q_i because (5.8) may

admit its optimal solution at rates larger than the target values q_i. The high rate of communication is indeed useful for age minimization. This is the reason why coding and caching can ultimately reduce age as shown in our work. In (5.35), as θ_i increases, i.e., more fresh packets are generated, the lower bound on EAoI decreases capturing the importance of frequent updating. This bound is active in regimes where new packets are generated less frequently.

Corollary 5. *For symmetric networks with independent erasure events, the lower bound in* (5.34) *leads to*

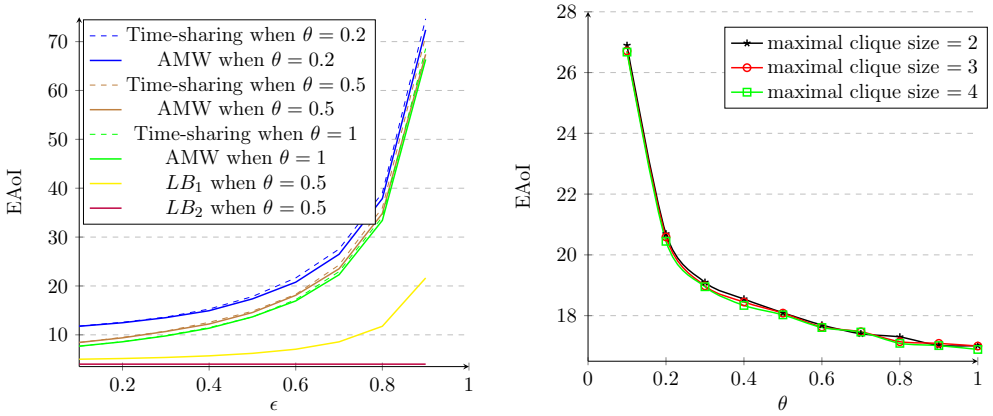
$$J^{\pi}(\underline{q}) \geq \frac{M}{2\sum_{j=1}^{M}\frac{1}{1-\epsilon^j}\sum_{i=1}^{M}\frac{1}{\alpha_i}} + \sum_{i=1}^{M}\frac{\alpha_i}{2M}.$$

Proof. The proof of Corollary 5 is given in Appendix D.7. □

5.6. Numerical Results

Finally, we seek to answer the questions that we raised at the very beginning of this chapter through simulations. We consider symmetric networks with $\epsilon_i = \epsilon$, $q_i = q$, and $\theta_i = \theta$ for $i \in [M]$.

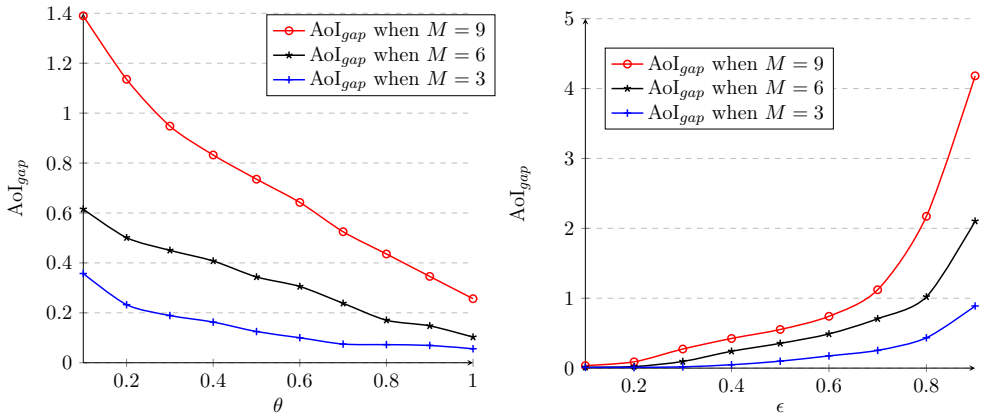
5.6.1. Benefits of Coding



(a) EAoI and the lower bounds. (b) EAoI under different maximal cliques.

Figure 5.2: EAoI as a function of ϵ and θ when $M = 6$, the upper bound, and the lower bounds (left). EAoI under ARM policies with different maximal cliques (right).

We first consider the benefits of coding. The ARM policy and the Time-sharing policy are compared in Figure 5.2a and Figure 5.3. To eliminate the impact of rate, we consider the case defined in



(a) AoI$_{gap}$ v.s. θ when $\epsilon = 0.6$. (b) AoI$_{gap}$ v.s. ϵ when $\theta = 0.2$.

Figure 5.3: AoI$_{gap}$ as a function of different parameters.

Remark 24, i.e., the buffer size of every queue is 1 and the stability region is $\{\theta \leq 1\}$. We have set $\lambda = 0$ and $\alpha_i = \min\big\{i, \max\{0, i-3\}\big\}$. Figure 5.2a plots the EAoI for $M = 6$ users under the ARM and time-sharing policies, and against the lower bound in (5.34). We observe that coding is indeed beneficial when the erasure probability ϵ is relatively large (≥ 0.6) and/or the arrival rate θ is relatively small (≤ 0.5). When θ is fixed, EAoI increases with ϵ.

Next, we define AoI$_{gap}$ as the gap between the EAoI under the ARM and time-sharing policies. The relationship between AoI$_{gap}$ and θ (resp. ϵ) is provided in Figure 5.3a (resp. Figure 5.3b). In Figure 5.3a, we set $\epsilon = 0.6$. We observe that AoI$_{gap}$ (the benefit of coding) decreases with the arrival rate θ. This is because the (expected) number of newly incoming packets increases with θ and the availability of fresh uncoded packets weakens the impact of coding actions.

In Figure 5.3b, we set $\theta = 0.2$. AoI$_{gap}$ increases with ϵ. This is because erased packets can be cached and provide more coding opportunities. AoI$_{gap}$ increases slowly when ϵ is small, and sharply when ϵ is large. In addition, from Figure 5.3a and Figure 5.3b, the benefits of coding increase with M.

5.6.2. Impact of Maximal Clique Size

The impact of maximal clique size is captured in Figure 5.2b. Let the buffer size of all (virtual) queues be 1 and set $M = 6$, $\lambda = 0$, $\epsilon = 0.6$ and $\alpha_i = \lceil \frac{i}{2} \rceil$. ARM in Definition 25 with maximal clique sizes $= 2, 3, 4$ are compared in Figure 5.2b. From Figure 5.2b, we can see that the ARM policy

with maximal clique size of 2 is a good approximation. This is useful as it significantly reduces the number of coding actions.

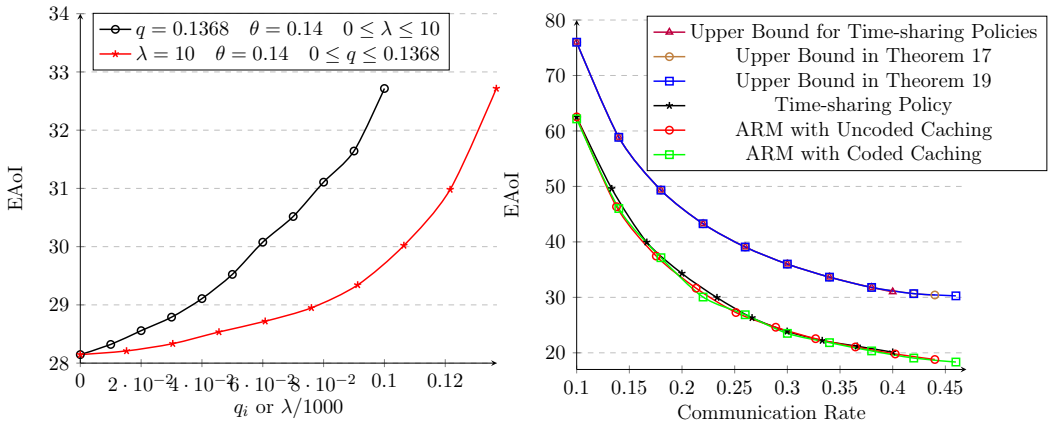
5.6.3. Tradeoff between Age and Rate

We finally investigate the tradeoff between the AoI and rate. Set $M = 3$. The maximum sum rate achievable with uncoded caching is around 0.44, and the channel capacity is around 0.46. Setting $\alpha_i = 3$ and $\epsilon = 0.6$ in the ARM policy, we first investigate the relationship between q and EAoI (the red star curve in Figure 5.4a). Now set $\theta = 0.14$, $\lambda = 10$, $q \in [0, 0.1368]$. EAoI increases with q implying that if the minimum required throughput becomes larger, the system has to sacrifice EAoI to satisfy the rate constraints. Next, the relationship between λ and EAoI is investigated (the black circle curve in Figure 5.4a). Let $\theta = 0.14$, $q = 0.1368$, $\lambda \in [0, 10]$. EAoI increases with λ. In other words, if the rate constraints become more important, then EAoI increases.

Finally, in Figure 5.4b, the EAoI is plotted as a function of the communication rate under the time-sharing policy as well as the ARM policy with uncoded and coded caching. This plot is obtained by setting $\alpha_i = i$, $\theta = q$, and $\lambda = 1$. We observe that EAoI decreases as the rate increases. From the viewpoint of expectation, almost all packets are successfully delivered. The increase in theta implies more fresh packets are generated, and the increase in q (which equals θ) implies more fresh packets are delivered. Thus, the EAoI decreases. The three policies have similar performances up to the rate they support. It appears that ARM with coded caching outperforms the other policies when the rate close to the boundary of the capacity region.

5.7. Conclusion

We investigated the benefits of coding in memoryless broadcast channels with M users. A scheduling framework is proposed in which coding actions, as opposed to users, are scheduled to attain desired tradeoffs between rate and age of information (AoI). Two general lower bounds and the upper bound for the proposed MW policies are obtained. The simulation showed that (i) coding is beneficial, and the benefits increase with the number of users; (ii) the tradeoff between rate and AoI exists, which implies that the system has to sacrifice AoI to achieve a higher rate; (iii) a good approximation of proposed policies is obtained based on maximum clique size of information graph.



(a) AoI_{gap} v.s. θ (b) EAoI v.s rate under different policies.

Figure 5.4: AoI_{gap} v.s. θ when $\epsilon = 0.6$ (left). EAoI v.s rate (when $M = 3$) under time-sharing, ARM with uncoded and coded caching policies when $\epsilon = 0.6$ (right).

CHAPTER 6

Exploitation and Exploration in Sequential Learning

We consider learning and decision-making in networked systems for processes that evolve both temporally and spatially. An important example in this class of processes is COVID-19 infection. It evolves in time (e.g. through different stages of the disease for an infected individual) and over a contact network and its spread can be contained by testing and isolation. Public health systems need to judiciously decide who should be tested and isolated in the presence of limitations on the number of individuals who can be tested and isolated on a given day.

The following challenges arise in the design of intelligent testing strategies if one seeks to exploit the spatiotemporal evolution of the disease process and comply with limited testing budgets. Observing the state of a node at time t will provide information about the state of (i) the node in time $t+1$ and (ii) the neighbors of the node at time $t, t+1, \ldots$. This is due to the inherent correlation that exists between states of neighboring nodes because an infectious disease spreads through contact. Thus, testing has a dual role. It has to both detect/isolate infected nodes and learn the spread in various localities of the network. The spread can often be silent: an undetected node (that may not be particularly likely to be infected based on previous observations) can infect its neighbors. Thus, testing nodes that do not necessarily appear to be infected may lead to the timely discovery of even larger clusters of infected nodes waiting to explode. In other words, there is *an intrinsic tradeoff between the exploitation of knowledge vs. exploration of the unknown.*

6.1. Literature Review

SIR and Variants: Most existing have investigated the spread of COVID-19 through dynamic systems such as SIR models and their variants [99, 100, 101, 102, 103, 104]. These models are made more complex to fit the real data in [105, 106, 107, 108, 109, 110]. Estimation of the model parameters by learning-based methods are considered and verified by real data in [111, 112, 113, 114, 115, 116]. Other attributes such as lockdown policy [117], multi-wave prediction [118], herd immunity threshold [119] are also considered by data-driven experiments. These works mostly

focus on the estimation of model parameters through real data, and aim to make a more accurate prediction of the spread. None of them, however, consider testing and isolation policies. Our work complements these investigations by designing sequential testing and isolation policies in order to minimize cumulative infections. For this purpose, we have assumed full statistical knowledge of the spread model and the underlying contact network and we are not concerned with prediction and estimation of model parameters.

Mean-field Approximations: Designing optimal testing and control policies in dynamic networked systems often involves computational challenges. These challenges have been alleviated in control literature by capturing the spread through differential equations [120, 121, 122, 123, 124]. The differential equations rely on classical mean-field approximations, considering neighbors of each node as "socially averaged hypothetical neighbors". Refinements of the mean-field approximations such as pair approximation [125], degree-based approximation [126], meta-population approximation [127] etc, all resort to some form of averaging of neighborhoods or more generally groups of nodes. The averaging does not capture the heterogeneity of a real-world complex social network and in effect disregards the contact network topology. But, in practice, the contact network topologies are often partially known, for example, from contact tracing apps that individuals launch on their phones. Thus testing and control strategies must exploit the partial topological information to control the spread.

Contact Tracing: The most widely deployed testing and control policy, the (forward and backward) contact tracing (and its variants) [128, 129, 130, 131, 132, 133, 132, 134, 135], relies on partial knowledge of the network topology (ie, the neighbors of infectious nodes who have been detected), and therefore does not lend itself to mean-field analysis. Contact tracing policies are in a sense exploitation policies: upon finding positive nodes, they exploit that knowledge and trace the contacts. While relatively practical, they have two main shortcomings, as implemented today: (i) They are not able to prioritize nodes based on their likelihood of being infected (beyond the coarse notion of contact or lack thereof). For example, consider an infectious node that has two neighbors, with different degrees. Under current contact tracing strategies, both neighbors have the same status. But in order to contain the spread as soon as possible, the node with a large degree should be

prioritized for testing. A similar drawback becomes apparent if the neighbors themselves have a different number of infectious neighbors; one with a larger number of infectious neighbors should be prioritized for testing, but current contact tracing strategies accord both the same priority. (ii) Contact tracing strategies do not incorporate any type of exploration. This may be a fundamental limitation of contact tracing. [135] has shown that, with high cost, contact tracing policies perform better when they incorporate exploration (active case finding). In contrast, our work provides a probabilistic framework to not only allow for exploitation in a fundamental manner but also to incorporate exploration in order to minimize the number of infections.

Exploration vs. Exploitation Tradeoffs: Exploration vs. exploitation tradeoffs were originally studied in classical multi-armed bandit (MAB) problems where there is the notion of a single optimal arm that can be found by repeating a set of fixed actions [136, 137, 138]. MAB testing strategies have also been designed for exploring partially observable networks [139]. Our problem differs from what is mainly studied in the MAB literature because (i) the number of arms (potential infected nodes) is time-variant and actions cannot be repeated; (ii) the exploration vs. exploitation tradeoff in our context arises due to lack of knowledge about the time-evolving set of infected nodes, rather than lack of knowledge about the network or the process model and its parameters. Our problem is also related to active search in graphs where the goal is to test/search for a set of (fixed) target nodes under a set of given (static) similarity values between pairs of nodes [140, 141, 142, 143]. But the target nodes in these works are assumed fixed, whereas the target is dynamic in our setting because the infection spreads over time and space (i.e., over the contact network). Thus, a node may need to be tested multiple times. The importance of exploitation/exploration is also known, implicitly and/or explicitly, in various reinforcement learning literature [144, 145, 146].

We now distinguish our work from testing strategies that combine exploitation and exploration in some form [147, 148, 149]. Through a theoretical approach, [147] models the testing problem as a partially observable Markov decision process (POMDP). An optimal policy can, in principle, be formulated through POMDP, but such strategies are intractable in their general form (and heuristics are often far from optimal) [150, 151]. [147] devises tractable approximate algorithms with a

significant caveat: In the design, analysis, and evaluation of the proposed algorithms, it is assumed that at each time the process can spread only on a single random edge of the network. This is a very special case that is hard to justify in practice and it is not clear how one could go beyond this assumption. On the other hand, [148] proposes a heuristic by implementing classical learning methods such as Linear support vector machine (SVM) and Polynomial SVM to rank nodes based on a notion of risk score (constructed by real data) while reserving a portion of the test budget for random testing which can be understood as exploration. No spread model or contact network is assumed. [149] and this work was done concurrently. In [149], a tractable scheme to control dynamical processes on temporal graphs was proposed, through a POMDP solution with a combination of Graph Neural Networks (GNN) and Reinforcement Learning (RL) algorithm. Nodes are tested based on some scores obtained by the sequential learning framework, but no fundamental probabilities of the states of nodes were revealed. Different from [148, 149], our approach is model-based and we observe novel exploration-exploitation tradeoffs that arise not due to a lack of knowledge about the model or network, but rather because the set of infected nodes is unknown and evolves with time. We can also utilize knowledge about both the model and the contact network to devise a probabilistic framework for decision-making.

Finally, we summarize the contribution of some significant works that consider only exploitation and do not utilize any exploration [132, 134, 135]. [132, 134] have considered a combination of isolation and contact tracing sequential policies, and [132] has shown that the sequential strategies would reduce transmission more than mass testing or self-isolation alone, while [134] has shown that the sequential strategies can reduce the amount of quarantine time served and cost, hence individuals may increase participation in contact tracing, enabling less frequent and severe lockdown measures. [135] have proposed a novel approach to modeling multi-round network-based screening/contact tracing under uncertainty.

6.2. System Model

To describe a spreading process, we use a *discrete-time compartmental model* [224]. Over decades, compartmental models have been key in the study of epidemics and opinion dynamics, albeit often

disregarding the network topology. In this work, we capture the spread on a given contact network. For clarity of presentation, we focus on a model for the spread of COVID-19. The ideas can naturally be generalized to other applications. The main notations in the full paper are given in Table 6.1.

Notations	Definitions
β	transmission probability
$1/\gamma$	mean duration in the latent state
$1/\lambda$	mean duration in the infectious state
$\sigma_i(t)$	state of node i at time t, $\sigma_i(t) \in \{I, S, L, R\}$
$\mathcal{G}(t)$	contact network at time t
$\mathcal{V}(t)$	set of nodes at time t
$\mathcal{E}(t)$	set of edges at time t
$N(t)$	cardinality of $\mathcal{V}(t)$
N	$N = N(0)$
$\partial_i(t)$	neighbors of node i at time t
$\partial_i^+(t)$	$\{i\} \cup \partial_i(t)$
$Y_i(t)$	testing result of node i at time t
$\mathcal{O}(t)$	set of nodes tested at time t
$\underline{Y}(t)$	$\{Y_i(t)\}_{\{i \in \mathcal{O}(t)\}}$
$B(t)$	testing budget at time t
π	a testing and isolation policy
$C^\pi(t)$	cumulative infections at time t
$\mathcal{K}^\pi(t)$	set of nodes tested at time t (under policy π)
$K^\pi(t)$	$K^\pi(t) = \lvert\mathcal{K}^\pi(t)\rvert$
T	time horizon
$\underline{v}_i(t)$	true probability vector of node i
$\underline{u}_i(t)$	prior probability vector of node i
$\underline{w}_i(t)$	posterior probability vector of node i
$\underline{e}_i(t)$	updated posterior probability vector of node i
$r_i(t)$	rewards of selecting node i at time t
$\hat{r}_i(t)$	estimated rewards of node i at time t
$\Psi_i(t)$	$\Psi_i(t) = \mathcal{O}(t) \cap \partial_i^+(t-1)$
$\Phi_i(t)$	$\Phi_i(t) = \{j \lvert j \in \partial_k^+(t-1), k \in \Psi_i(t)\} \backslash \{i\}$
$\theta_i(t)$	$\theta_i(t) = \sigma_i(t)\lvert_{\{\underline{Y}(\tau)\}_{\tau=1}^{t-1}}$, $\theta_i(t) \in \{I, S, L, R\}$
$\zeta_i(t)$	$\zeta_i(t) = \sigma_i(t)\lvert_{\{\underline{Y}(\tau)\}_{\tau=1}^{t}}$, $\zeta_i(t) \in \{I, S, L, R\}$

Table 6.1: Useful notations in Chapter 6.

We model the progression of Covid-19 per individual, in time, through four stages or states: *Susceptible (S)*, *Latent (L)*, *Infectious (I)*, and *Recovered (R)*. Per contact, an infectious individual infects a susceptible individual with transmission probability β. An infected individual is initially

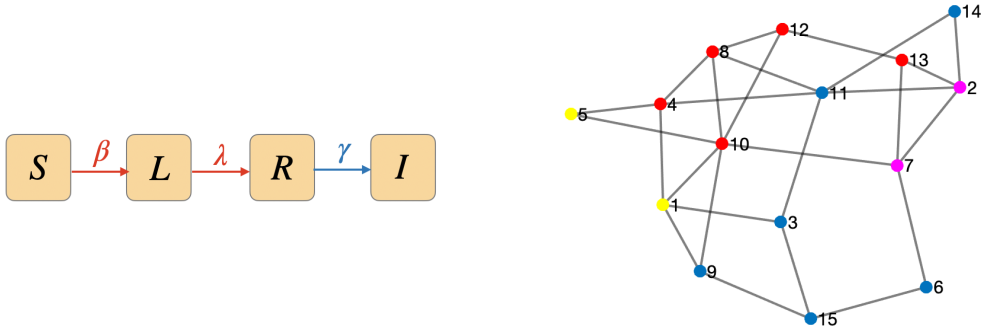


Figure 6.1: Time evolution of the process per individual nodes (left). A contact network with nodes in states (blue) susceptible, (pink) latency, (red) infectious, (yellow) recovered (right).

in the latent state L, subsequently, he becomes infectious (state I), and finally, he recovers (state R). Fig. 6.1 (left) depicts the evolution. The durations in the latent and infectious states are geometrically distributed, with means $1/\lambda, 1/\gamma$ respectively. We represent the state of node i at time t by random variable $\sigma_i(t)$ and its support set $\mathcal{X} = \{S, L, I, R\}$. We assume that the parameters β, λ, and γ are known to the public health authority. This is a practical assumption because the parameters can be estimated by the public health authority based on the pandemic data collected [225, 226, 227].

Let $\mathcal{G}(t) = (\mathcal{V}(t), \mathcal{E}(t))$ denote the contact network at time t, where $\mathcal{V}(t)$ is the set of nodes/individuals, of cardinality $N(t)$, and $\mathcal{E}(t)$ is the set of edges between the nodes, describing interactions/contacts on day t. Let $\mathcal{V} = \mathcal{V}(0)$, $\mathcal{E} = \mathcal{E}(0)$, $\mathcal{G} = \mathcal{G}(0)$, and $N = N(0)$. The network is time-dependent not only because interactions change on a daily basis, but also because nodes may be tested and isolated. If a node is tested positive on any day t, it will be isolated immediately. If a node is isolated on any day t, we assume that it remains in isolation until it recovers. We assume that a recovered node can not be reinfected again. Thus a node that is isolated on any day t has no impact on the network from then onwards. Such nodes can be regarded as "removed". Therefore, it is removed from the contact network for all subsequent times $t, t+1, \ldots$. Fig. 6.1 (right) depicts a contact network at a given time t. We assume that a public health authority knows the entire contact network and decides who to test based on this information. This assumption has been made in several other works in this genre eg in [135].

Denote the set of neighbors of node i, on day t, by $\partial_i(t)$. The state of each node at time $t+1$ depends on the state of its neighbors $\partial_i(t)$, as well as its own state in day t, as given by the following conditional probability:

$$\Pr\left(\sigma_i(t+1)|\{\sigma_j(t)\}_{j\in\partial_i^+(t)}\right)$$

where $\partial_i^+(t) = \partial_i(t) \cup \{i\}$.

Node i is tested positive on the day t if it is in the infectious state (I)[19]. Let $Y_i(t)$ denote the test result:

$$Y_i(t) = \begin{cases} 1 & \sigma_i(t) = I \\ 0 & \sigma_i(t) \in \{S, L, R\}. \end{cases} \tag{6.1}$$

We do not assume any type of error in testing and $Y_i(t)$ is hence a deterministic function of $\sigma_i(t)$. Let $\mathcal{O}(t)$ be the set of nodes that have been tested (observed) in day t and denote the *network observations* at time t by $\underline{Y}(t) = \{Y_i(t)\}_{i\in\mathcal{O}(t)}$.

Our goal in this paper is to design testing and isolation strategies in order to contain the spread and minimize cumulative infections. Naturally, testing resources (and hence observations) are often limited and such constraints make decision-making challenging. Let $B(t)$ be the maximum number of tests that could be performed in day t, called the *testing budget*. $B(t)$ can evolve based on the system necessities, e.g., in contact tracing that is widely deployed for COVID-19, the number of tests is chosen based on the history of observations[20]. Also, governments often upgrade testing infrastructure as the number of cases increases. Our framework captures both fixed and time-dependent budget $B(t)$, but we focus on time-dependent $B(t)$ for simulations.

Define the cumulative infections on the day t, denoted by $C^\pi(t)$, as the number of nodes who have been infected before and including day t, where π is the testing and isolation policy. Let $\mathcal{K}^\pi(t)$

[19]We assume that a node in the latent state L is infected, but not infectious. We further assume that latent nodes test negative.

[20]In practical implementations, scheduling constraints do play a role but we disregard that in this work.

denote the set of tests π performs on day t. Given a large time horizon T, our objective is:

$$\begin{aligned} \min_{\pi} \quad & \mathbb{E}[C^{\pi}(T)] \\ s.t. \quad & |\mathcal{K}^{\pi}(t)| \leq B(t),\ 0 \leq t \leq T-1. \end{aligned} \tag{6.2}$$

Recall that $\sigma_i(t)$, the state of node i on day t, is a random variable and unknown. For each node i, define a probability vector $\underline{v}_i(t)$ of size $|\mathcal{X}|$, where each coordinate is the probability of the node being in a particular state at the end of time t. The coordinates of $\underline{v}_i(t)$ follow the order (I, L, R, S) and we have

$$\underline{v}_i(t) = \big[v_x^{(i)}(t)\big]_{x\in\mathcal{X}}, \qquad v_x^{(i)}(t) = \Pr\big(\sigma_i(t) = x\big). \tag{6.3}$$

For example, $v_S^{(i)}(t)$ represents the probability of node i being in state S in time t. We now define $F_i(\mathcal{D}; t)$ to be the *conditional* probability of node i being infected by nodes in $\mathcal{D}$ (for the first time) at day t, as a function of the nodes' states $\{\sigma_i(t)\}_{i\in\mathcal{V}(t)}$. We have

$$F_i(\mathcal{D}; t) = 1_{\{\sigma_i(t)=S\}} \cdot \prod_{j\in\partial_i(t)\setminus\mathcal{D}} \big(1 - \beta 1_{\{\sigma_j(t)=I\}}\big) \cdot \big(1 - \prod_{j\in\mathcal{D}\cap\partial_i(t)} (1 - \beta 1_{\{\sigma_j(t)=I\}})\big). \tag{6.4}$$

Equation (6.4), captures the impact that the nodes in $\mathcal{D}$ have on infecting node i at day t. In this equation, we assume that the infections from different nodes are independent. The same assumption has also been made in several other papers in this genre, eg in [125, 126, 127]. Then, we find the expectation (with respect to $\{\sigma_i(t)\}_{i\in\mathcal{V}(t)}$) of (6.4) as follows:

$$\mathbb{E}_{\{\sigma_i(t)\}_{i\in\mathcal{V}(t)}}\left[F_i(\mathcal{D}; t)\right] = v_S^{(i)}(t) \cdot \big\{ \prod_{j\in\partial_i(t)\setminus\mathcal{D}} (1 - \beta v_I^{(j)}(t))\big\} \cdot \big\{1 - \prod_{j\in\mathcal{D}\cap\partial_i(t)} (1 - \beta v_I^{(j)}(t))\big\}. \tag{6.5}$$

It is worth noting that (6.4) is a probability conditioned on $\{\sigma_i(t)\}_{i\in\mathcal{V}(t)}$, while (6.5) is an unconditional probability. To obtain (5), we have indeed assumed that the states of the nodes are independent. This assumption does not hold in general and we only utilize it here to obtain a simple expression in (5) in terms of the infection probabilities. We do not use this independence

assumption in the rest of the paper. Define

$$S\big(\mathcal{D};t\big) = \sum_{i\in\mathcal{V}(t)} \mathbb{E}\left[F_i(\mathcal{D};t)\right]. \tag{6.6}$$

Here, $S\big(\mathcal{D};t\big)$ represents the (expected) number of newly infectious nodes incurred by nodes in $\mathcal{D}$ at day t. Recall that $\mathcal{K}^\pi(t)$ is the set of nodes that are tested at time t. We show the following result in Appendix E.1.

Lemma 13. $\mathbb{E}\big[C^\pi(t+1) - C^\pi(t)\big] = S\big(\mathcal{V}(t)\backslash\mathcal{K}^\pi(t);t\big)$.

6.2.1. Supermodularity

It is complex to solve (6.2) globally, especially if one seeks to find solutions that are optimal looking into the future. We thus simplify the optimization (6.2) for policies that are myopic in time as follows. First, note that $C^\pi(T)$ can be re-written as follows through a telescopic sum:

$$C^\pi(T) = \sum_{t=0}^{T-1} C^\pi(t+1) - C^\pi(t). \tag{6.7}$$

Then, we restrict attention to myopic policies that at each time minimize $\mathbb{E}\left[C^\pi(t+1) - C^\pi(t)\right]$. We then show how $\mathbb{E}\left[C^\pi(t+1) - C^\pi(t)\right]$ can be expressed in terms of a supermodular function.

Using (6.7) along with Lemma 13, we seek to solve the following optimization sequentially in time for $0 \le t \le T-1$:

$$\min_{|\mathcal{K}^\pi(t)|\le B(t)} \quad S\big(\mathcal{V}(t)\backslash\mathcal{K}^\pi(t);t\big). \tag{6.8}$$

We now prove some desired properties for the set function $S(\mathcal{K}^\pi(t);t)$ (see Appendix E.2).

Theorem 21. *$S\big(\mathcal{K}^\pi(t);t\big)$ defined in (6.6) is a supermodular[21] and increasing monotone function on $\mathcal{K}^\pi(t)$.*

On day t, and given the network, the probability vectors of all nodes, and $\mathcal{K}_1^\pi(t) \subset \mathcal{K}_2^\pi(t)$, for any

[21] Let $\mathcal{X}$ be a finite set. A function $f : 2^{\mathcal{X}} \to \mathbb{R}$ is supermodular if for any $\mathcal{A} \subset \mathcal{B} \subset \mathcal{X}$, and $x \in \mathcal{X}\backslash\mathcal{B}$, $f(\mathcal{A}\cup\{x\}) - f(\mathcal{A}) \le f(\mathcal{B}\cup\{x\}) - f(\mathcal{B})$.

node $i \notin \mathcal{K}_2^\pi(t)$, node i will incur larger increment of newly infectious nodes under $\mathcal{K}_2^\pi(t)$ than that under $\mathcal{K}_1^\pi(t)$. This is because node i may have common neighbors with nodes in $\mathcal{K}_2^\pi(t)$. So, supermodularity holds in Theorem 21.

The optimization (6.8) is NP-hard [228]. However, using the supermodularity of $S\big(\mathcal{V}(t)\backslash\mathcal{K}^\pi(t);t\big)$, we propose Algorithm 5 based on [229, Algorithm A] to greedily optimize (6.8) in every day t. Denote the optimum solution of (6.8) as OPT. As proved in [229], on every day t, Algorithm 5 attains a solution, denoted by $\tilde{\mathcal{K}}^\pi(t)$, such that $\big(\mathcal{V}(t)\backslash\tilde{\mathcal{K}}^\pi(t);t\big) \leq \big(1+\epsilon(t)\big)\cdot \text{OPT}$, i.e., the solution $\tilde{\mathcal{K}}^\pi(t)$ is an $\epsilon(t)$-approximation of the optimum solution. Here, on day t, the constant $\epsilon(t)$, which is the steepness of the set function $S(\cdot;t)$ as described in [229], can be calculated as follows, $\epsilon(t) = \frac{\epsilon'}{4(1-\epsilon')}$ and $\epsilon' = \max_{a\in\mathcal{V}(t)} \frac{S(\mathcal{V}(t);t)-S(\mathcal{V}(t)\backslash\{a\};t)-S(\{a\};t)}{S(\mathcal{V}(t);t)-S(\mathcal{V}(t)\backslash\{a\};t)}$.

In Algorithm 5, on every day t, in every step, we choose the node that provides the minimum increment on $S(\cdot;t)$ based on the results in the previous step, and then remove the node from the current node set. Algorithm 5 is stopped when $K^\pi(t)$ nodes have been chosen. The complexity of this algorithm is discussed in Appendix E.3.

Algorithm 5 Greedy Algorithm

Step 0: On day t, input $\{\underline{v}_i(t)\}_{i\in\mathcal{V}(t)}$, set $\mathcal{A}_0 = \mathcal{V}(t)$.
repeat
Step i: Let $\mathcal{A}_i = \mathcal{A}_{i-1}\backslash\{a_i\}$, where

$$a_i = \arg\min_{a\in\mathcal{A}_{i-1}} S\big(\{a\}\cup\{a_1,\cdots,a_{i-1}\};t\big).$$

until $i = N(t) - |\mathcal{K}^\pi(t)|$, and return $\mathcal{K}^\pi(t) = \mathcal{A}_i$.

6.3. Exploitation and Exploration

In Section 6.2.1, we proposed a near-optimal greedy algorithm to sequentially (in time) select the nodes to test. However, Algorithm 5 has two shortcomings. (i) The computation is costly when N and/or T are large (see Appendix E.3). (ii) The objective function $S\big(\mathcal{V}(t)\backslash\mathcal{K}^\pi(t)\big)$ is dependent on $\{\underline{v}_i(t)\}_{i\in\mathcal{V}(t)}$ which is unknown, even though the network and the process are stochastically fully given (see Section 6.2). This is because the set of infected nodes is unknown and time-evolving.

To overcome the first shortcoming, we propose a simpler reward maximization policy by minimizing

an upper bound on the objective function in (6.8). To overcome the second shortcoming, we estimate $\{\underline{v}_i(t)\}_{i\in\mathcal{V}(t)}$ using the history of test observations $\{\underline{Y}(\tau)\}_{\tau=0}^{t}$ (as presented in Section 6.4). we refer to the estimates as $\{\underline{u}_i(t)\}_{i\in\mathcal{V}(t)}$. Both the greedy policy and its reward-based variant that we will propose in this section thus need to perform decision-making based on the estimates $\{\underline{u}_i(t)\}_{i\in\mathcal{V}(t)}$ and we refer to them as "exploitation" policies.

It now becomes clear that testing has two roles: to find the infected in order to isolate them and contain the spread, and to provide better estimates of $\{\underline{v}_i(t)\}_{i\in\mathcal{V}(t)}$. This leads to interesting tradeoffs between exploitation and exploration as we will discuss next. Under exploitation policies, we test nodes deterministically based on a function of $\{\underline{v}_i(t)\}_{i\in\mathcal{V}(t)}$, (which is called "reward", and will be defined later); while under exploration policies, nodes are tested according to a probabilistic framework (based on rewards of all nodes).

To simplify the decision-making into reward maximization, we first derive an upper bound on $S\big(\mathcal{V}(t)\backslash\mathcal{K}^{\pi}(t);t\big)$. Define

$$r_i(t) = S\big(\{i\};t\big). \tag{6.9}$$

Using the supermodularity of the function $S(\cdot)$, we prove the following lemma in Appendix E.4.

Lemma 14. $S\big(\mathcal{V}(t)\backslash\mathcal{K}^{\pi}(t);t\big) \leq S\big(\mathcal{V}(t);t\big) - \sum_{i\in\mathcal{K}^{\pi}(t)} r_i(t)$.

Remark 27. *Recall that $S(\cdot;t)$ is a supermodular function, then the amount of newly infectious nodes incurred by the set $\mathcal{K}^{\pi}(t)$, $S(\mathcal{K}^{\pi}(t);t)$, is larger than the sum of the amount of newly infectious nodes by every individual node in $\mathcal{K}^{\pi}(t)$, i.e., $\sum_{i\in\mathcal{K}^{\pi}(t)} r_i(t)$. Thus, $S\big(\mathcal{V}(t)\backslash\mathcal{K}^{\pi}(t);t\big)$ is upper bounded by $S\big(\mathcal{V}(t);t\big) - \sum_{i\in\mathcal{K}^{\pi}(t)} r_i(t)$.*

We propose to minimize the upper bound in Lemma 14 instead of $S\big(\mathcal{V}(t)\backslash\mathcal{K}^{\pi}(t);t\big)$. Since $\mathcal{V}(t)$ is known and $S\big(\mathcal{V}(t);t\big)$ is hence a constant, the problem reduces to solving:

$$\max_{|\mathcal{K}^{\pi}(t)|\leq B(t)} \quad \sum_{i\in\mathcal{K}^{\pi}(t)} r_i(t). \tag{6.10}$$

Given probabilities $\{\underline{v}_i(t)\}_{i\in\mathcal{V}(t)}$, the solution to (6.10) is to pick the nodes associated with the $B(t)$ largest values $r_i(t)$. We thus refer to $r_i(t)$ as the *reward* of selecting node i.

Let $\{\underline{u}_i(t)\}_{i\in\mathcal{V}(t)}$ be an estimate for $\{\underline{v}_i(t)\}_{i\in\mathcal{V}(t)}$ found by estimating the conditional probability of the state of node i given the history of observations $\{\underline{Y}(\tau)\}_{\tau=0}^{t-1}$. Our proposed reward-based Exploitation (RbEx) policy follows the same idea of selecting the nodes with the highest rewards. Note that $\{\underline{v}_i(t)\}_{i\in\mathcal{V}(t)}$ is unknown to all nodes. Instead of using the true probabilities $\{\underline{v}_i(t)\}_{i\in\mathcal{V}(t)}$, we consider the estimates of it which we sequentially update by computing the prior probabilities $\{\underline{u}_i(t)\}_{i\in\mathcal{V}(t)}$ and the posterior probabilities $\{\underline{w}_i(t)\}_{i\in\mathcal{V}(t)}$. In particular, $\{\underline{u}_i(0)\}_{i\in\mathcal{V}(0)}$ and $\{\underline{w}_i(0)\}_{i\in\mathcal{V}(0)}$ are the prior probabilities and the posterior probabilities on the initial day, respectively. Hence, we calculate the estimate of rewards, denoted by $\hat{r}_i(t)$, by replacing $\{\underline{v}_i(t)\}_{i\in\mathcal{V}(t)}$ with $\{\underline{u}_i(t)\}_{i\in\mathcal{V}(t)}$ in (6.6) and (6.9).

Algorithm 6 Reward-based Exploitation (RbEx) Policy

Input $\{\underline{w}_i(0)\}_{i\in\mathcal{V}(t)}$, $\{\underline{u}_i(0)\}_{i\in\mathcal{V}(0)}$, $\underline{Y}(0)$, and $t=0$.
Repeat for $t=1,2,\cdots,T-1$.
Step 1: Calculate $\{\hat{r}_i(t)\}_{i\in\mathcal{V}(t)}$ based on $\{\underline{u}_i(t)\}_{i\in\mathcal{V}(t)}$ and (6.9).
Step 2: Re-arrange the sequence $\{\hat{r}_i(t)\}_{i\in\mathcal{V}(t)}$ in descending order, and test the first $K(t)$ nodes. Get the new observations $\underline{Y}(t)$.
Step 3: Based on $\underline{Y}(t)$, update $\{\underline{u}_i(t+1)\}_{i\in\mathcal{V}(t+1)}$ by Algorithm 8 (Step 0 $\sim$ Step 2) in Section **??**.

The shortcoming of Algorithm 6 is that it targets maximizing the estimated sum rewards, even though the estimates may be inaccurate. In this case, testing is heavily biased towards the history of testing and it does not provide opportunities for getting better estimates of the rewards. For example, consider a network with several clusters. If one positive node is known by Algorithm 6, then it may get stuck in that cluster and fail to locate more positives in other clusters.

In Section 6.4.3, we will prove, in a line network, that the exploitation policy described in Algorithm 6 can be improved by a constant factor (in terms of the resulting cumulative infections) if a simple form of exploration is incorporated.

We next propose an exploration policy. Our proposed policy is probabilistic in the sense that the

nodes are randomly tested with probabilities that are proportional to their corresponding estimated rewards. This approach has similarities and differences to Thompson sampling and more generally posterior sampling. The similarity lies in the probabilistic nature of testing using posterior probabilities. The difference is that in our setting decision-making depends on the distributions of decision variables, but not samples of the decision variables.

More specifically, at time t, node i is tested with probability $\min\{1, \frac{B(t)\hat{r}_i(t)}{\sum_{j\in\mathcal{V}(t)}\hat{r}_j(t)}\}$, which depends on the budget $B(t)$. Note that each node is tested with probability at most 1; so if $\frac{B(t)\hat{r}_i(t)}{\sum_{j\in\mathcal{V}(t)}\hat{r}_j(t)} > 1$ for some node i, then we would not fully utilize the budget. The unused budget is thus

$$c(t) = \sum_{i\in\mathcal{V}(t)} \big(\frac{B(t)\hat{r}_i(t)}{\sum_{j\in\mathcal{V}(t)}\hat{r}_j(t)} - 1\big)^+ \tag{6.11}$$

and can be used for further testing[22]. Algorithm 7 outlines our proposed Reward-based Exploitation-Exploration (REEr) policy.

Algorithm 7 Reward-based Exploitation-Exploration (REEr) Policy

Input $\{\underline{w}_i(0)\}_{i\in\mathcal{V}(t)}$, $\{\underline{u}_i(0)\}_{i\in\mathcal{V}(0)}$, $\underline{Y}(0)$, and $t = 0$.
Repeat for $t = 1, 2, \cdots, T-1$
Step 1: Calculate $\{\hat{r}_i(t)\}_{i\in\mathcal{V}(t)}$ based on $\{\underline{u}_i(t)\}_{i\in\mathcal{V}(t)}$ and (6.9).
Step 2: Test node i with probability $\min\{1, \frac{B(t)\hat{r}_i(t)}{\sum_{j\in\mathcal{V}(t)}\hat{r}_j(t)}\}$. After that, randomly select $c(t)$ $\big($defined in (6.11)$\big)$ further nodes to test (see Footnote 22). Get the new observations $\underline{Y}(t)$.
Step 3: Based on $\underline{Y}(t)$, update $\{\underline{u}_i(t+1)\}_{i\in\mathcal{V}(t+1)}$ by Algorithm 8 (Step 0 $\sim$ Step 2) in Section 6.4.

6.4. Message-Passing Framework

As discussed in Section 6.3, the probabilities $\{\underline{v}_i(t)\}_i$ are unknown. In this section, we develop a message-passing framework to sequentially estimate $\{\underline{v}_i(t)\}_i$ based on the network observations and the dynamics of the spread process. We refer to these estimates as $\{\underline{u}_i(t)\}_i$.

When node i is tested on the day t, an observation $Y_i(t)$ is provided about its state. Knowing the state of node i provides two types of information: (i) it provides information about the state of the neighboring nodes in future time slots $t+1, t+2, \ldots$ (because of the evolution of the spread in time

[22]Note that $c(t)$ is not always an integer. Instead of $c(t)$, we use $\texttt{Int}\big(c(t)\big)$ with probability $|\texttt{Int}\big(c(t)\big) - c(t)|$ where $\texttt{Int}(\cdot) \in \{\lfloor\cdot\rfloor, \lceil\cdot\rceil\}$.

and on the network), and (ii) it also provides information about the past of the spread, meaning that we can infer about the state of the (unobserved) nodes at previous time slots. For example, if node i tested positive in time t, we would know that (i) its neighbors are more likely to be infected in time $t+1$ and (ii) some of its neighbors must have been infected in a previous time for node i to be infected now. This forms the basis for our backward-forward message-passing framework.

Given the spread model of Section 6.2, we first describe the forward propagation of belief. Suppose that at time t, the probability vector $\underline{v}_i(t)$ is given for all i. The probability vector $\underline{v}_i(t+1)$ can be computed as follows (see Appendix E.5):

$$\underline{v}_i(t+1) = \underline{v}_i(t) \times \mathtt{P}_i\big(\{\underline{v}_j(t)\}_{j\in\partial_i^+(t)}\big) \tag{6.12}$$

where $\mathtt{P}_i\big(\{\underline{v}_j(t)\}_{j\in\partial_i^+(t)}\big)$ is a local transition probability matrix given in Appendix E.5.

Recall that $\underline{Y}(t)$ denotes the collection of network observations on day t. The history of observations is then denoted by $\{\underline{Y}(\tau)\}_{\tau=1}^{t-1}$. Based on these observations, we wish to find an estimate of the probability vector $\underline{v}_i(t)$ for each $i \in \mathcal{V}(t)$. We denote this estimate by $\underline{u}_i(t) = (u_x^{(i)}(t), x \in \mathcal{X})$ and refer to it, in this section, as the *prior probability* of node i in time t. We further define the *posterior probability* $\underline{w}_i(t) = (w_x^{(i)}(t), x \in \mathcal{X})$ of node i in time t (after obtaining new observations $\underline{Y}(t)$). In particular,

$$u_x^{(i)}(t) = \Pr\big(\sigma_i(t) = x | \{\underline{Y}(\tau)\}_{\tau=1}^{t-1}\big)$$
$$w_x^{(i)}(t) = \Pr\big(\sigma_i(t) = x | \{\underline{Y}(\tau)\}_{\tau=1}^{t}\big).$$

Here, the prior probability is defined *at the beginning of* every day, and the posterior probability is defined *at the end of* every day. Conditioning all probabilities in (6.12) on $\{\underline{Y}(\tau)\}_{\tau=1}^{t}$, we obtain the following *forward-update rule* (see Appendix E.6)

$$\underline{u}_i(t+1) = \underline{w}_i(t) \times \mathtt{P}_i\big(\{\underline{w}_j(t)\}_{j\in\partial_i^+(t)}\big). \tag{6.13}$$

Remark 28. *Following (6.13), we need to utilize the observations $\underline{Y}(t)$ and the underlying dependency among nodes' states to update the posterior probabilities $\{\underline{w}_i(t)\}_i$, and consequently update $\{u_i(t+1)\}_i$ based on the forward-update rule (6.13). This is however non-trivial. A Naive approach would be to locally incorporate node i's observation $Y_i(t)$ into $w_i(t)$ and obtain $\underline{u}_i(t+1)$ using* (6.13)*. This approach, however, does not fully exploit the observations and it disregards the dependency among nodes' states, as caused by the nature of the spread (An example is provided in Appendix E.8).*

6.4.1. Backward Propagation of Belief

To capture the dependency of nodes' states and thus best utilize the observations, we proceed as follows. First, denote

$$\underline{e}_i(t-1) = (e_x^{(i)}(t-1), x \in \mathcal{X})$$
$$e_x^{(i)}(t-1) = \Pr\left(\sigma_i(t-1) = x | \{\underline{Y}(\tau)\}_{\tau=1}^{t}\right).$$

Vector $\underline{e}_i(t-1)$ is the posterior probability of node i at time $t-1$, after obtaining the history of observations up to and including time t. By computing $\underline{e}_i(t-1)$, we are effectively correcting our belief on the state of the nodes in the previous time slot by *inference* based on the observations acquired at time t. This constitutes the *backward step* of our framework and we will expand on it shortly. The backward step can be repeated to correct our belief also in times $t-2$, $t-3$, etc. For clarity of presentation and tractability of our analysis and experiments, we truncate the backward step at time $t-1$ and present assumptions under which this truncation is theoretically justifiable. Considering larger truncation windows is straightforward but out of the scope of this paper.

Once our belief about nodes' states is updated in prior time slots (e.g., $\underline{e}_i(t-1)$ is obtained), it is propagated forward in time for *prediction* and to provide a more accurate estimate of the nodes' posterior and prior probabilities. More specifically, consider (6.12) written for time t (rather than $t+1$) and condition all probabilities on $\{\underline{Y}(\tau)\}_{\tau=1}^{t}$. We obtain the following update rule (see

Appendix E.6):

$$\underline{w}_i(t) = \underline{e}_i(t-1) \times \tilde{\mathtt{P}}_i\big(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)}\big) \tag{6.14}$$

where $\tilde{\mathtt{P}}_i\big(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)}\big)$ is given in Appendix E.6. *Note that the local transition matrix in (6.14) is not the same as (6.13). This is because "future" observations were available in* $\tilde{\mathtt{P}}_i\big(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)}\big)$. The probability vectors $\{\underline{e}_i(t-1)\}_i$ provide better estimates for $\{\underline{w}_i(t)\}_i$ through (6.14) and the prior probabilities $\{\underline{u}_i(t+1)\}_i$ are then computed using (6.13) to be used for decision making in time $t+1$. The block diagram in Fig. 6.3 depicts the high-level idea of our framework. It is worth noting that $\mathtt{P}_i\big(\{\underline{w}_j(t)\}_{j\in\partial_i^+(t)}\big)$ in (6.13) and $\tilde{\mathtt{P}}_i\big(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)}\big)$ in (6.14) both depend on the observations, $\{\underline{Y}(\tau)\}_{\tau=1}^t$.

We next discuss how $\underline{e}_j(t-1)$ can be computed, starting with some notations. Denote by

$$\zeta_i(t) = \sigma_i(t)|_{\{\underline{Y}(\tau)\}_{\tau=1}^t}, \quad \theta_j(t) = \sigma_i(t)|_{\{\underline{Y}(\tau)\}_{\tau=1}^{t-1}}, \tag{6.15}$$

the state of the nodes in the posterior probability spaces conditioned on the observations $\{\underline{Y}(\tau)\}_{\tau=1}^t$ and $\{\underline{Y}(\tau)\}_{\tau=1}^{t-1}$, respectively. We further define $\Psi_i(t)$ to be the set of those neighbors of node i at time $t-1$, including node i, who are observed/tested at time t. This set consists of all nodes whose posterior probabilities will be updated at time $t-1$ (given a new observation $Y_i(t)$). The set of all neighbors (except node i) of the nodes in $\Psi_i(t)$ then defines $\Phi_i(t)$. The set $\Phi_i(t)$ consists of all nodes whose posterior probabilities at time t are updated by the observation $Y_i(t)$. More precisely, we have

$$\begin{aligned}
\Psi_i(t) &= \mathcal{O}(t) \cap \partial_i^+(t-1),\\
\Phi_i(t) &= \{j | j \in \partial_k^+(t-1), k \in \Psi_i(t)\}\backslash\{i\},\\
\Theta_i(t) &= \{j | j \in \partial_k^+(t-1), k \in \mathcal{O}(t)\}\backslash\{i\}
\end{aligned}$$

where $\mathcal{O}(t)$ is the set of observed nodes at time t (see Figure 6.2). In Appendix E.7, we show

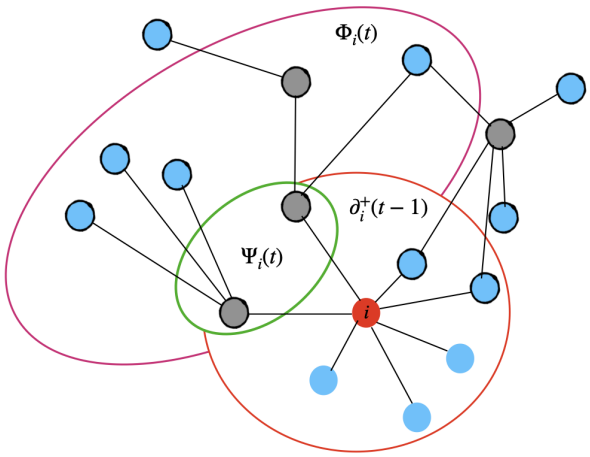


Figure 6.2: An example of $\Psi_i(t)$, $\Phi_i(t)$ and $\Theta_i(t)$. Node i is marked in red, and its neighborhood $\partial_i^+(t-1)$ is shown by the red contour. Suppose that the gray nodes are tested on the day $t-1$, then $\Psi_i(t)$ is the set of nodes within the green contour, and $\Phi_i(t)$ consists of the nodes in the purple contour. Finally, nodes in $\Theta_i(t)$ are marked with a bold black border

.

$$e_x^{(i)}(t-1) = \frac{\Pr\left(\underline{Y}(t)|\zeta_i(t-1)=x\right)\ w_x^{(i)}(t-1)}{\Pr\left(\underline{Y}(t)\right)}. \tag{6.16}$$

It suffices to find $\Pr\left(\underline{Y}(t)|\zeta_i(t-1)=x\right)$. The denominator $\Pr\left(\underline{Y}(t)\right)$ is then found by normalization of the enumerator in (6.16). Let $\{x_j\}_{j\in\mathcal{O}(t)}$ be a realization of $\{\theta_j(t)\}_{j\in\mathcal{O}(t)}$ and $\{y_l\}_{l\in\Theta_i(t)}$ be a realization of $\{\zeta_l(t-1)\}_{l\in\Theta_i(t)}$. We prove the following in Appendix E.7 under a simplifying truncation assumption (see Assumption 9 in Appendix E.7) where the backward step is truncated in time $t-1$:

$$\begin{aligned}
&\Pr\left(\underline{Y}(t)|\zeta_i(t-1)=x\right) = \Pr\left(\{Y_j(t)\}_{j\in\Psi_i(t)}|\zeta_i(t-1)=x\right)\\
&= \sum_{\{x_j\}_{j\in\Psi_i(t)}} \prod_{j\in\Psi_i(t)} \Pr\left(Y_j(t)|\theta_j(t)\right) \times \sum_{\{y_l\}_{l\in\Phi_i(t)}} \prod_{j\in\Psi_i(t)} \Pr\left(x_j|\{y_l\}_{l\in\partial_j^+(t-1)\setminus\{i\}}, x\right) \times \prod_{l\in\{\Phi_i(t)\}} w_{y_l}^{(i)}(t-1).
\end{aligned} \tag{6.17}$$

We finally present our *Backward-Forward Algorithm* to sequentially compute estimates $\{\underline{u}_i(t)\}_i$ in Algorithm 8. The process of Algorithm 8 is given in Fig 6.3, and we also give a simple example to show the process of Algorithm 8 in Appendix E.8.

6.4.2. Necessity of Backward Updating

Now we provide an example that illustrates the necessity of backward updating.

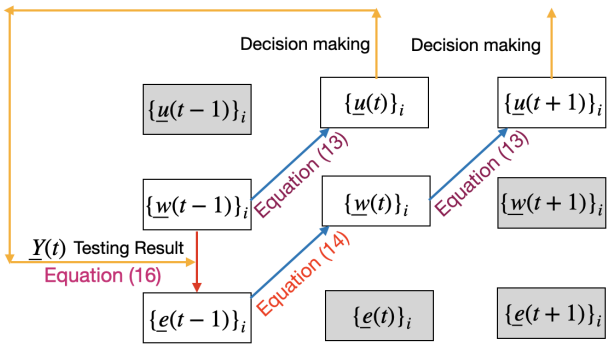


Figure 6.3: The process of Algorithm 8. One example of a complete process is given in unshaded blocks. Recall that $\underline{u}_i(\tau)$, $\underline{w}_i(\tau)$, and $\underline{e}_i(\tau)$, where $\tau \in \{t-1, t, t+1\}$, are the prior probabilities, the posterior probabilities, and the updated posterior probabilities, respectively.

Algorithm 8 Backward-Forward Algorithm

Input $\underline{Y}(0)$, $\{\underline{e}_i(0)\}_{i\in\mathcal{V}(0)}$, $\{\underline{w}_i(0)\}_{i\in\mathcal{V}(0)}$, $\{\underline{u}_i(0)\}_{i\in\mathcal{V}(0)}$.
Repeat for $t = 1, 2, \cdots, T-1$
Step 0: Based on $\underline{Y}(t)$, get $\mathcal{V}(t)$ from $\mathcal{V}(t-1)$.
Step 1: Backward step. Update $\underline{e}_i(t-1)$ by (6.16), (6.17), and then compute $\underline{w}_i(t)$ by (6.14).
Step 2: Forward step. Compute $\underline{u}_i(t+1)$ by (6.13).

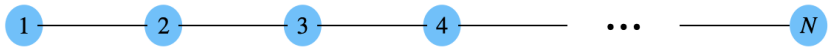


Figure 6.4: The line network in Example 3.

Example 3. *Consider a line network with the node set* $\mathcal{V} = \{1, 2, \ldots, N\}$ *and the edge set* $\mathcal{E} = \{(i, i+1), 1 \le i \le N-1\}$ *(see Figure 6.4). On the initial day, we assume that each node is infected independently with probability* $1/N$. *Let* $\beta = 1$, $\lambda = 0$, $\gamma = 0$[23], *and* $B(t) = 1$. *We further assume that there is no isolation when a positive node is tested.*

Based on Example 3, we show that the naive approach of Remark 28 (i.e., forward-only updating) will cause the estimated probabilities to never converge to the true probabilities of infection. Nonetheless, if we use the Backward-Forward Algorithm 8, the estimated probabilities converge to the true probabilities after a certain number of steps. Formally, we prove the following result in Appendix E.10.

[23]Here, $\lambda = 0$ implies there is no latent state, and $\gamma = 0$ implies that nodes never recover.

Theorem 22. *For any testing policy that sequentially computes* $\{\underline{u}_i(t)\}_i$ *based on (6.13) (see Remark 28), with probability (approximately)* $\frac{1}{e}$*, we have* $\sum_{i=1}^{N} \|\underline{v}_i(t) - \underline{u}_i(t)\| \overset{t\to\infty}{\to} \Theta(N)$*, for large* N [24]*. On the other hand, there exists a testing policy that sequentially updates* $\{\underline{u}_i(t)\}_i$ *based on Algorithm 8 and attains* $\sum_{i=1}^{N} \|\underline{v}_i(t) - \underline{u}_i(t)\| = 0,\ t \geq 2N$.

Roadmap of proof: Consider a simple case where every node is susceptible. Since each node is infected with probability $1/N$*, then the case occurs with probability* $\simeq 1/e$.

Under the case above, consider any testing policy based on the algorithm in Remark 28. If a node is tested on the day t*, then the policy "clears" the tested node. Since the updating rule of the algorithm can not go back to the information on the day* $t-1$*, then it can not "clear" any neighbors of the tested node and its probability of infection updates to a non-zero value the next day. Furthermore, we show that almost all nodes have a significantly large probability of infection when the time horizon is sufficiently large, hence* $\sum_{i=1}^{N} \|\underline{v}_i(t) - \underline{u}_i(t)\| \overset{t\to\infty}{\to} \Theta(N)$.

On the other hand, we can propose a specific testing policy. Note that there is no infection, if Algorithm 8 is used to update probabilities, then it can reveal the states of all nodes under the specific testing policy after at most $2N$ *days. So we have* $\sum_{i=1}^{N} \|\underline{v}_i(t) - \underline{u}_i(t)\| = 0,\ t \geq 2N$.

In Theorem 22, we illustrate the necessity of backward updating when testing is limited. In essence, we want to "clear" the graph and confirm that there are no infections. If the number of tests is limited, we have mathematically shown that no algorithm can correctly estimate the nodes' infection probabilities if it does not use the backward (inference) step. On the contrary, there is an algorithm that uses the backward step along with the forward step and the estimates that it provides for the nodes' infection probabilities converge to the true probabilities of the nodes after some finite steps. Even though the considered graph is simple but the phenomena it captures are general.

As discussed in Theorem 22, backward updating is necessary. However, backward updating can be computationally expensive in large dense graphs. To trade off the impact of backward updating and

[24] Theorem 22 holds for all kinds of noem due to the equivalence of norms. In addition, the convergence is topological convergence.

the reduction of computation complexity, we propose an α-linking backward updating algorithm in Appendix E.11, where Algorithm 8 is applied on a random subgraph with fewer edges.

6.4.3. Necessity of Exploration

Note that in reality we have no information for $\{\underline{v}_i(t)\}_i$, and only have the estimates $\{\underline{u}_i(t)\}_i$. One may wonder if exploitation based on wrong initial estimated probability vectors, i.e., $\{\underline{u}_i(0)\}_i$, misleads decision-making by providing poorer and poorer estimates of the probabilities of infection. If so, exploration may be necessary.

Example 4. *Consider $\mathcal{V} = \{1, 2, \ldots, N\}$ and edges $\mathcal{E} = \{(i, i+1), 1 \leq i \leq N-1\}$ (see Figure 6.4). Let $N \gg 10$, $\beta = 1$, $\lambda = 0$, $\gamma = 0$, and $B(t) = 10$. Suppose that on the initial day, node 1 is infected and all other nodes are susceptible. Consider a wrong initial estimate: $w_I^{(i)}(0) = u_I^{(i)}(0) = 0$ if $i \leq \frac{9N}{10}$, and $w_I^{(i)}(0) = u_I^{(i)}(0) = \frac{10\epsilon}{N}$ otherwise, where $\epsilon > 0$. With this initial belief, we have $\sum_{i=1}^{N} \|\underline{w}_i(0) - \underline{v}_i(0)\| = O(1+\epsilon)$.*

Different from Example 3, here we consider the isolation of nodes that tested positive. In Example 4, suppose that a specific exploration policy is applied: 1 (out of 10) test is done randomly, and the other 9 tests are done following exploitation. Now, in Appendix E.12, we show that under the RbEx policy, the cumulative infection is at least aN for a constant a, while under the exploration policy defined above, the cumulative infection is at most bN with very high probability, and the ratio a/b can be any constant for a large enough N. More formally, we have the following theorem. Let p_0 be a large probability and consider a large time horizon T. Denote the cumulative infections under the RbEx policy by $C^{RbEx}(T)$ and under the specific exploration policy defined above by $C^{exp}(T)$. We prove the necessity of exploration in the following Theorem.

Theorem 23. *With probability $p_0 \geq \frac{99}{100}$, $\frac{C^{RbEx}(T)}{C^{exp}(T)} \geq c(N, p_0)$, where $c(N, p_0)$ is a constant only depending on N and p_0.*

Roadmap of proof: Under the RbEx policy, we test nodes based on their predicted probabilities. Since the nodes that are located towards the end of the line (right side in Fig. 6.4) have non-zero probabilities, they are tested first while the disease spreads on the other end of the network (left side

in Fig. 6.4). Mathematically, suppose that for the first time, an infectious node is tested at day $t = aN$*, then there are at least* $\min\{aN, N\}$ *infectious nodes before the spread can be contained.*

Under the specific exploration policy described above, consider the event that, for the first time, an infectious node is explored on the day $t = b'N$ *(*$b' < a$*). We argue that with probability* p_0*, the exploration policy catches at least two new infections at each step after* $t = b'N$*. After* $2t$*, the algorithm catches all the infections, and we have at most* $2b'N$ *infections. Let* $b = 2b'$*. This is an improvement by a factor of at least* $\frac{a}{b}$ *in comparison to the RbEx strategy. Factor* $\frac{a}{b}$ *depends on the values of* N *and* p_0*.*

In Theorem 23, we show the necessity of exploration when our initial belief is slightly wrong, i.e., it is slightly biased toward the other end of the network (In general, this could be due to a wrong belief, prior test results, etc). We have formally proved that when the testing capacity is limited, exploration can significantly improve the cumulative infections, i.e., contain the spread. This motivates the design of exploration policies. Even though the setting is simple, the phenomena it captures are much more general.

6.5. Numerical Results

6.5.1. Overview

In this section, we use simulations to study the performance of the proposed exploitation and exploration policies for various synthetic and real-data networks. Towards this end, we define some metrics that quantify how different metrics perform and key network parameters and attributes that determine the values of these metrics and thereby how exploitation and exploration compare. We also identify benchmark policies that represent the extreme ends of the tradeoff between exploration and exploitation to compare with the policies we propose and assess the performance enhancements brought about by judicious combinations of exploration and exploitation. Through our experiments, we aim to answer two main questions for various synthetic and real-data networks: (i) Can exploration policies do better than exploitation policies, and if so, when would that be the case? (ii) What parameters would affect the performance of exploration and exploitation policies? These are

important questions to shed light on the role of exploration. These questions are particularly raised by Theorem 23 in which we prove that exploration can significantly outperform exploitation in some (stylized) networks. We design the experiments in order to shed light on the above questions and to understand the extent of the necessity of exploration in different network models and scenarios.

Network parameters We consider the following parameters: (i) The unregulated delay ℓ which is the time from the initial start of the spread to the first time testing and intervention starts; (ii) The *(global) clustering coefficient* [230, Chapter 3], denoted by γ_c, which is defined as a measure of the degree to which nodes in a graph tend to cluster together; (iii) The *path-length*, denoted by L_p, which measures the average shortest distance between every possible pair of nodes. We consider attributes such as the initialization of the process, and the lack of knowledge about $\{\underline{v}_i(t)\}_i$.

Performance metrics We consider the expected number of infected nodes in a time horizon $[0, T]$ as the performance measure for various policies. Let $C^0(T)$ be the number of infected nodes if there is no testing and isolation, $C^{RbEx}(T), C^{REEr}(T)$ be the corresponding numbers respectively for the *RbEx* policy (Algorithm 6) and the *REEr* policy (Algorithm 7). We consider a ratio between the expectations of these:

$$\texttt{Ratio} = \frac{\mathbb{E}[C^{RbEx}(T)] - \mathbb{E}[C^{REEr}(T)]}{\mathbb{E}[C^0(T)]}. \tag{6.18}$$

We define the *estimation error* $\texttt{Err}_\pi(t)$ towards capturing the impact of the lack of knowledge about $\{\underline{v}_i(t)\}_i$.

$$\texttt{Err}_\pi(t) = \frac{1}{N(t)} \sum_{j \in \mathcal{G}(t)} ||\underline{v}(t) - \underline{u}(t)||_2^2. \tag{6.19}$$

We consider the difference between the estimation errors of *RbEx* and *REEr* policies: $\Delta_{\texttt{Err}} = \texttt{Err}_{RbEx}(T) - \texttt{Err}_{REEr}(T)$.

Benchmark policies We will compare the proposed policies with 3 benchmark policies. (i) (Forward) Contact Tracing: we tested every day the nodes who have infectious neighbors (in a forward manner), denoted by *candidate nodes*. Only some candidate nodes are selected randomly

due to testing resources being limited. Note that only exploitation is utilized under this benchmark. (ii) Random Testing: Every day, we randomly select nodes to test. Typical testing policies that could come out of SIR optimal control formulations for our problem would naturally reduce to random testing as they treat all nodes to be statistically identical and ignore the impact of network topology. One can interpret that random testing implements exploration to its full extent. (iii) Contact Tracing with Active Case Finding: A small portion of (for example, 5%) testing budget is utilized for active case finding [135]. This portion of the testing budget is used to test nodes by Random Testing. The remaining budget is utilized for forward contact tracing. (iv) Logistic Regression: We use ideas presented in [148], where simple classifiers were proposed based on the features of real data. In our setting, we choose the classifier to be based on logistic regression, and we define the feature of node i as $X_i(t) = [1, n_i(t) + \epsilon]^T$. Here, $n_i(t)$ is the number of quarantined neighbors node i has contacted before and including day t, and $\epsilon \neq 0$ is a superparameter aiming to avoid the case where $n_i(t) = 0$. In simulations, we set $\epsilon = 0.1$. Let the observation $Y_i(t)$ be the testing result of node i. In particular, if node i is not tested on day t, then we do *not* collect the data $(X_i(t), Y_i(t))$. Thus, the probability of node i being infectious is defined as the *Sigmoid* function

$$\frac{1}{1 + \exp(-X_i(t) \cdot w^T)},$$

where w is the parameter that should be learned.

Simulation Setting We consider a process as described in Section 6.2 with n_0 randomly located initial infected nodes. The process evolved without any testing/intervention for ℓ days and we refer to ℓ as the *unregulated delay*. After that, one of the (initial) infectious nodes, denoted by node i_0, is (randomly) provided to the policies. Subsequently, the initial estimated probability vector is set to $\underline{u}_{i_0}(\ell) = (1, 0, 0, 0)$, and $\underline{u}_i(\ell) = (0, 0, 0, 1)$ when $i \neq i_0$. We consider the budget to be equal to the expected number of infected nodes at time t, i.e., $B(t) = \sum_{j=1}^{N(t)} v_I^{(j)}(t)$.

We choose model parameters considering the particular application of COVID-19 spread. In particular, 1) the mean latency period is $1/\lambda = 1$ or 2 days [225]; 2) the mean duration in the infectious

state (I) is $1/\gamma = 7 \sim 14$ days [225, 226, 227]; 3) we choose the transmission rate β in a specific network such that after a long time horizon, if no testing and isolation policies were applied, then around $60 \sim 90$ percent individuals are infected. We did not consider the case where 100 percent of individuals are infected because, given the recovery rate (and the topology), the spread may not reach every node.

We consider both synthetic networks such as Watts-Strogatz (WS) networks [231], Scale-free (SF) networks [232], Stochastic Block Models (SBM) [233] and a variant of it (V-SBM), as well as real-data networks. Descriptions and further results for the synthetic networks and real networks are presented in Appendix E.13.

Watts-Strogatz Networks. We consider a network $\mathrm{WS}(N, d, \delta)$ with N nodes, degree d, and rewiring probability δ. The transmission probability of the spread is set to $\beta = 0.4$ and the number of initial seeds is $n_0 = 3$.

Scale-free Networks. We consider a network $\mathrm{SF}(N, \alpha)$ with N nodes, and the fraction of nodes with degree k follows a power law $k^{-\alpha}$, where $\alpha = 2.1, 2.3, 2.5, 2.7, 2.9$. The transmission probability of the spread is set to $\beta = 0.5$ and the number of initial seeds is $n_0 = 3$.

Stochastic Block Models. The SBM is a generative model for random graphs. The graph is divided into several communities, and subsets of nodes are characterized by being connected with particular edge densities. The intra-connection probability is p_1, and the inter-connection probability is p_2. We denote the SBM as $\mathrm{SBM}(N, M, p_1, p_2)$[25]. The transmission probability of the spread is set to $\beta = 0.04$ and the number of initial seeds is $n_0 = 3$. The construction of SBM is given in Appendix E.13.1.

A Variant of Stochastic Block Models. Different from SBM, we only allow nodes in cluster i to connect to nodes in successive clusters (the neighbor clusters). Denote a variant of SBM as $\text{V-SBM}(N, M, p_1, p_2)$. The transmission probability of the spread is set to $\beta = 0.04$ and the number of initial seeds is $n_0 = 3$. The construction of V-SBM is given in Appendix E.13.1.

[25]Here, we assume that M is an exact divisor of N.

Real-data Network I. We consider a contact network of university students in the Copenhagen Networks Study [234]. The network is built based on the proximity between participating students recorded by smartphones, at 5 minute resolution. According to the definition of close contact by [227], we only used proximity events between individuals that lasted more than 15 minutes to construct the daily contact network. The contact network has 672 individuals spanning 28 days. To guarantee a long time horizon, we replicate the contact network 4 times so that the time horizon is 112 days. We set $\beta = 0.05$ and $n_0 = 5$ to have a realistic simulation of the Covid-19 spread. Note that the network is relatively dense, so we choose a relatively small value of β to avoid the unrealistic case in which the disease spreads very fast (see Figure 6.11 (left)).

Real-data Network II. We consider a publicly available dataset on human social interactions collected specifically for modeling infectious disease dynamics [235, 236, 237]. The data set consists of pairwise distances between users of the BBC Pandemic Haslemere app over time. The contact network has 469 individuals spanning 576 days. Since the network is very sparse, then we compress contacts among individuals during 4 successive days to one day. Then, we have 469 individuals spanning 144 days. We set $\beta = 0.95$ and $n_0 = 30$ to have a realistic simulation of the Covid-19 spread. Note that the network is relatively sparse, so we choose a relatively large value of β to avoid the unrealistic case in which the disease spreads very slowly (see Figure 6.11 (left)).

6.5.2. Simulation Results in Synthetic Networks

In this section, we compare the performances of our proposed policies and the benchmarks (defined in Section 6.5.1) in synthetic networks. We start with some specific networks and parameters for this purpose (see Figure 6.5, Figure 6.6, Figure 6.7, and Figure 6.8). The figures reveal that our proposed policies, i.e., the RbEx and REEr policies, outperform the benchmarks. In particular, in Figure 6.5 and Figure 6.6 (i.e., the WS and SF networks), the REEr policy outperforms the RbEx policy, and the REEr policy provides a more accurate estimation for $\{\underline{v}_i(t)\}_i$. In Figure 6.7 and Figure 6.8 (i.e., the SBM and V-SBM networks), the RbEx policy outperforms the REEr policy, and the RbEx policy provides a more accurate estimation for $\{\underline{v}_i(t)\}_i$. In addition, in Figure 6.5, we show that Algorithm 5 outperforms the RbEx policy but performs worse than the REEr policy (recall that the computation time of Algorithm 5 is high, we therefore only plot the performance of

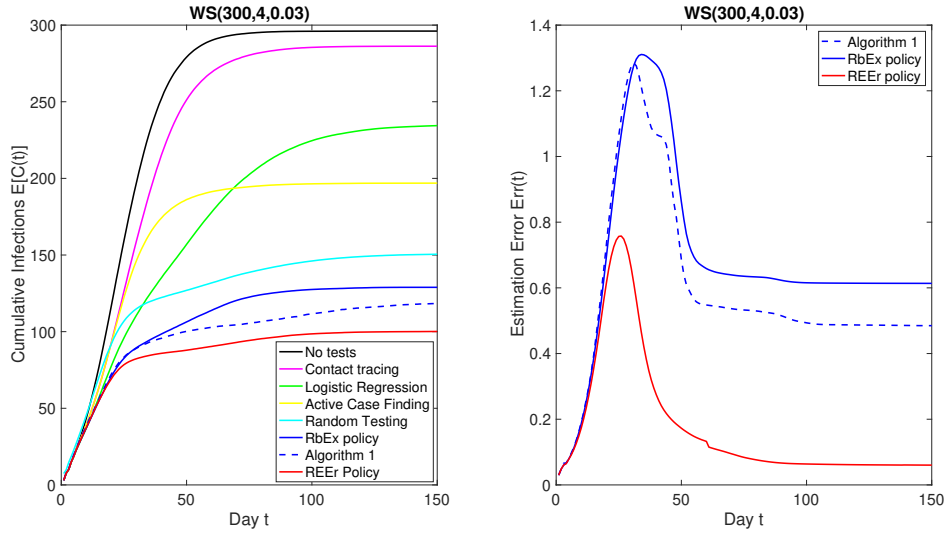


Figure 6.5: Performances and estimation errors of different policies in WS(300, 4, 0.03) when $\ell = 3$.

Algorithm 5 in Figure 6.5 as an example). This implies that without exploration, exploitation in a greedy manner can not perform well in WS networks.

From the discussions above, the advantages of exploration in distinct settings (different network topologies with variant parameters) are different. To investigate the advantages of exploration in distinct settings, it suffices to show how the main parameters affect the exploration. In this work, we consider three main parameters which are defined in Section 6.5.1, i.e., the unregulated delay ℓ, the global clustering coefficient γ_c, and the path-length L_p. Detailed discussions are later given in Section 6.5.2.

Impact of Network Parameters

In this subsection, we consider the impact of network parameters on the tradeoff between exploration and exploitation.

Impact of ℓ. We first investigate the impact of the unregulated delay, ℓ. Specifically, from Table 6.2, Table 6.3, Table 6.4, and Table 6.5, as ℓ increases, so does `Ratio` and $\Delta_{\texttt{Err}}$, implying that exploration becomes more effective. With an increase in ℓ, the infection continues in the network for longer, there is a greater number of infectious nodes in the network and they are scattered

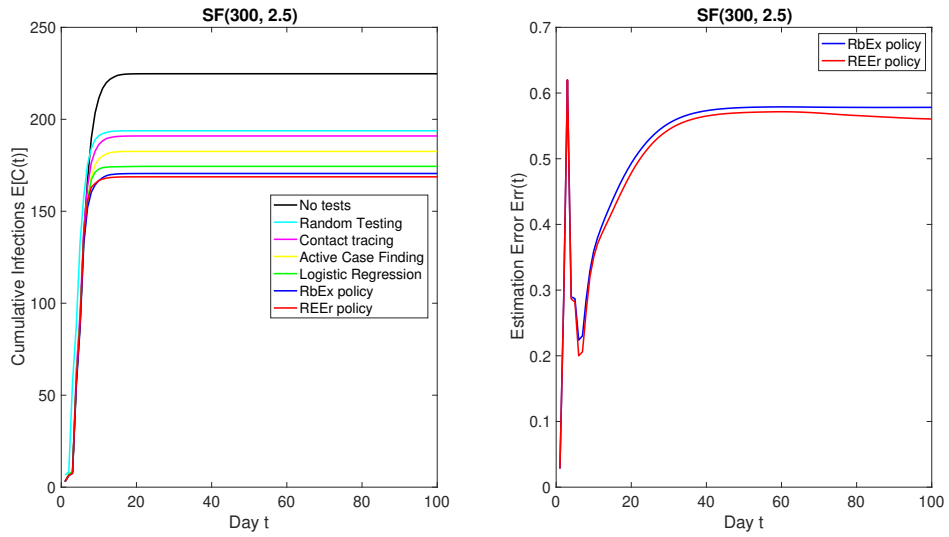


Figure 6.6: Performances and estimation errors of different policies in SF(300, 2.5) when $\ell = 3$.

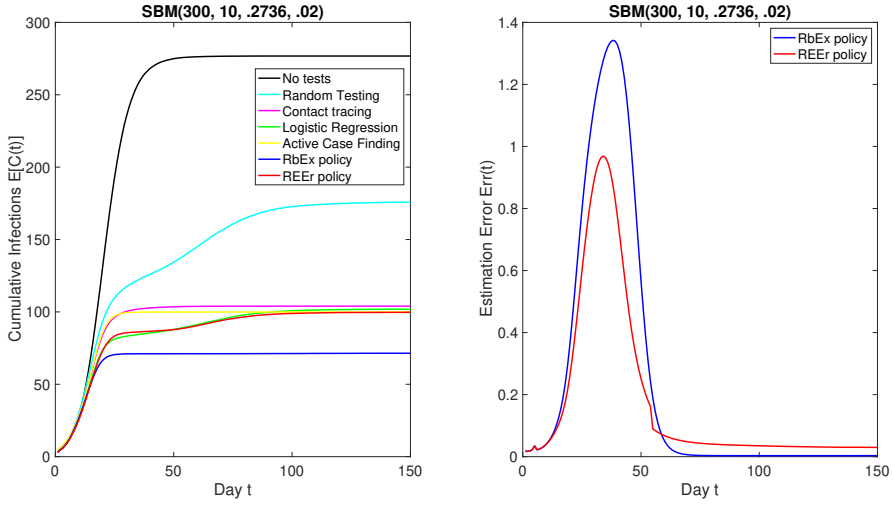


Figure 6.7: Performances and estimation errors of different policies in SBM(300, 10, .2736, .02) when $\ell = 5$.

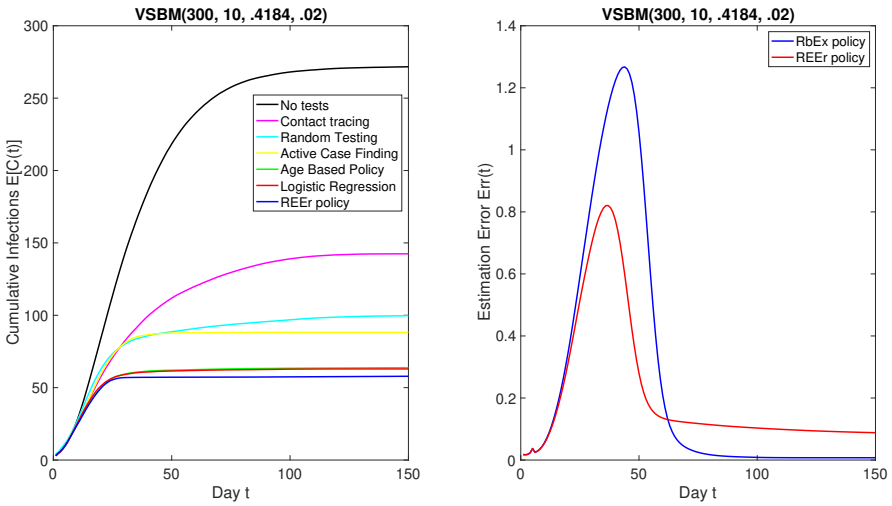


Figure 6.8: Performances and estimation errors of different policies in VSBM(300, 10, .4184, .02) when $\ell = 5$.

throughout the network, thus exploration is better suited to locate them. Thus, the REEr policy can contain the spread of the disease faster.

In particular, the REEr policy is always better in WS networks. This is because exploitation may confine the tests in neighborhoods of some infected nodes. While in the SBM networks, the RbEx policy always outperforms the REEr policy. In both the SF and V-SBM networks, the RbEx policy is better when ℓ is small, and the REEr policy is better when ℓ is large. One interesting observation is that in the V-SBM networks, the REEr policy performs better when ℓ is large ($= 11, 13$), but the corresponding estimation errors are larger than those in the RbEx policy. In this specific network topology, it appears that smaller estimation error does not always correspond to better cumulative infections. One potential reason is that the REEr policy is sensitive to ℓ in this topology, i.e., we can achieve smaller cumulative infections under the REEr policy even if the estimation error is larger.

WS, ℓ	3	5	7	9	11
`Ratio`	0.097	0.128	0.177	0.207	0.297
$\Delta_{\texttt{Err}}$	0.553	0.814	1.092	1.197	1.449

Table 6.2: Role of the unregulated delay ℓ in WS networks when $\delta = 0.03$.

SF, ℓ	3	5	7	9	11
`Ratio`	-0.0009	0.0026	0.0033	0.0042	0.0059
$\Delta_{\texttt{Err}}$	-0.0014	0.0237	0.0334	0.0434	0.1212

Table 6.3: Role of the unregulated delay ℓ in SF networks when $\alpha = 2.1$.

SBM, ℓ	5	7	9	11	13
`Ratio`	-0.092	-0.079	-0.042	-0.035	-0.025
$\Delta_{\texttt{Err}}$	-0.026	-0.015	-0.010	-0.009	-0.009

Table 6.4: Role of the unregulated delay ℓ in SBMs when $(p_1, p_2) = (.274, .02)$.

V-SBM, ℓ	5	7	9	11	13
`Ratio`	-0.022	-0.016	-0.007	0.011	0.019
$\Delta_{\texttt{Err}}$	-0.081	-0.066	-0.046	-0.033	-0.025

Table 6.5: Role of the unregulated delay ℓ in V-SBMs when $(p_1, p_2) = (.418, .02)$.

Impact of γ_c and L_p. Then, we investigate the impact of the global clustering coefficient, i.e., γ_c, and the average shortest path length, i.e., L_p. In Table 6.6, both γ_c and L_p decrease as δ increases. In Table 6.7, γ_c decreases as α increases. For the SF networks, the graphs are often disconnected, so we only calculate γ_c in Table 6.7. In Table 6.8 and Table 6.9, both γ_c and L_p decrease as p_2 increases.

From these tables, as L_p or γ_c decreases, the benefits of exploration compared to exploitation decrease as well. This confirms the intuition that exploration is particularly helpful in clustered networks with larger path lengths where undetected infection can spread without any intervention as exploitation largely confines the tests in neighborhoods of the infections that were previously detected. This is also supported by the fact that exploration lowers estimation error in such scenarios, as shown in Table 6.6, Table 6.7, Table 6.8, and Table 6.9. Furthermore, we investigate the role of γ_c and L_p individually in Appendix E.13.2.

6.5.3. Simulation Results in Real-data Networks

In this section, we verify our proposed policies in real data networks (Real-data Network I and Real-data Network II). In Figure 6.9, our proposed policies outperform the baselines, and the RbEx policy outperforms the REEr policy. In Figure 6.10, the REEr policy can contain the spread and

WS, δ	γ_c	L_p	Ratio	$\Delta_{\texttt{Err}}$
0	.5	62.876	0.191	1.153
.0075	.489	21.264	0.182	1.423
.015	.473	14.253	0.174	0.991
.0225	.467	12.171	0.126	0.779
.03	.456	10.81	0.097	0.554

Table 6.6: Role of the clustering coefficient and the path length in WS networks when $\ell = 3$.

SF, α	γ_c	Ratio	$\Delta_{\texttt{Err}}$
2.1	.5017	0.0080	0.0334
2.3	.3374	0.0057	0.0253
2.5	.2348	0.0032	0.0177
2.7	.1496	−0.0019	0.0124
2.9	.0219	−0.0064	0.0081

Table 6.7: Role of the clustering coefficient and the path length in SF networks when $\ell = 3$.

SBM, (p_1, p_2)	γ_c	L_p	Ratio	$\Delta_{\texttt{Err}}$
$(0.274, 0.02)$	0.111	2.573	−0.092	−0.026
$(0.214, 0.026)$	0.075	2.518	−0.103	−0.023
$(0.159, 0.032)$	0.056	2.492	−0.113	−0.026
$(0.102, 0.039)$	0.048	2.480	−0.118	−0.023
$(0.045, 0.045)$	0.043	2.455	−0.124	−0.027

Table 6.8: Role of the clustering coefficient and the path length in SBMs when $\ell = 5$.

V-SBM, (p_1, p_2)	γ_c	L_p	Ratio	$\Delta_{\texttt{Err}}$
$(0.418, 0.020)$	0.3557	4.4264	−0.022	−0.081
$(0.351, 0.052)$	0.2365	3.6584	−0.091	−0.045
$(0.284, 0.085)$	0.1769	3.307	−0.104	−0.055
$(0.217, 0.085)$	0.1385	3.1562	−0.112	−0.041
$(0.150, 0.0150)$	0.1170	3.0563	−0.123	−0.042

Table 6.9: Role of the clustering coefficient and the path length in V-SBMs when $\ell = 5$.

Real-data Network I, ℓ	5	8	11
Ratio	−0.0559	−0.0255	0.009
$\Delta_{\texttt{Err}}$	−0.061	−0.030	0.035

Table 6.10: Role of the unregulated delay ℓ in the real-data network I.

Real-data Network II, ℓ	5	8	11
Ratio	0.0808	0.1039	0.1208
$\Delta_{\texttt{Err}}$	0.0317	0.0535	0.0615

Table 6.11: Role of the unregulated delay ℓ in the real-data network II.

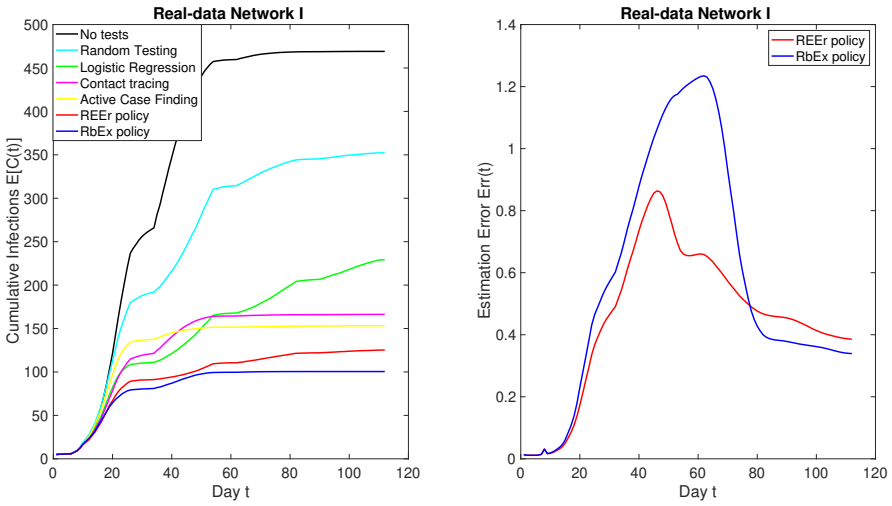


Figure 6.9: Performances and estimation errors of different policies in the real-data network I when $\ell = 8$.

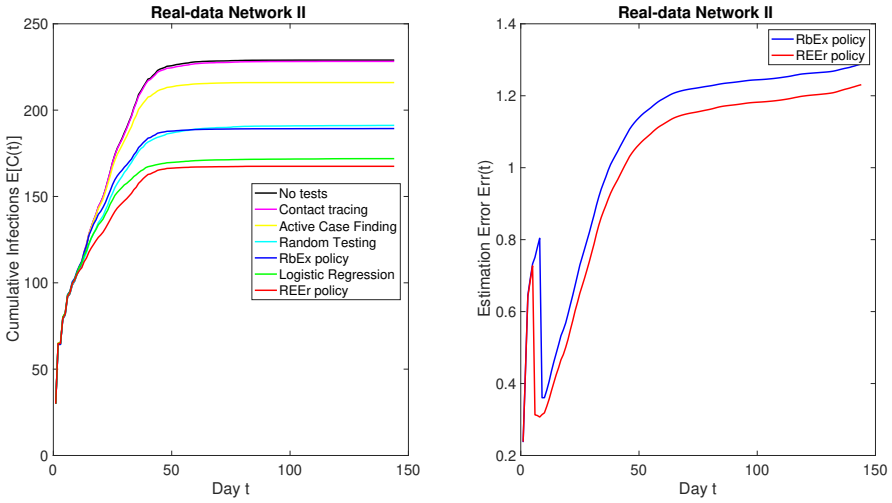


Figure 6.10: Performances and estimation errors of different policies in the real-data network II when $\ell = 8$.

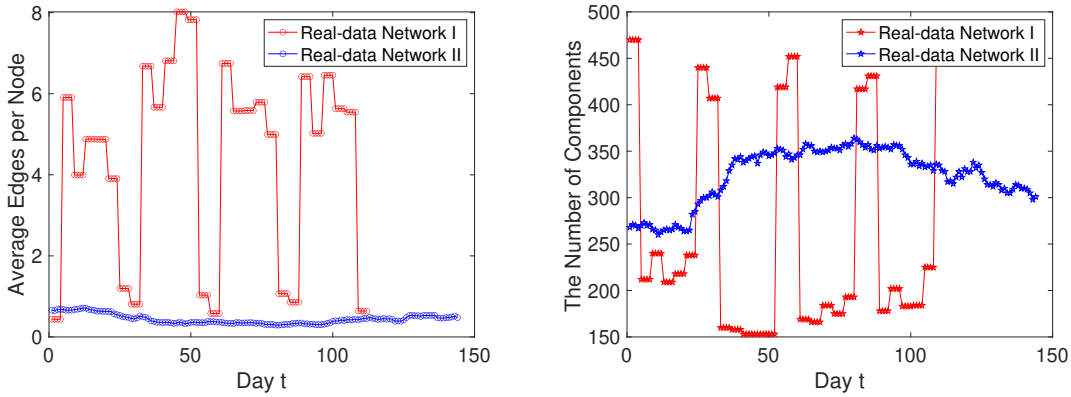


Figure 6.11: The average number of edges per node on each day (left). The number of components on each day (right).

outperform other baselines and RbEx, while the Logistic Regression policies outperform RbEx. Comparing Figure 6.9 and Figure 6.10, we find that the RbEx policy performs well in Real-data Network I (better than the REEr policy), but performs not well in Real-data Network II (much worse than the REEr policy). In Figure 6.11 (left), we calculate the average edges per node every day, and in Figure 6.11 (right), we calculate the number of components every day. From Figure 6.11 (left), the Real-data Network I is denser than the Real-data Network II. However, from Figure 6.11 (right), the Real-data Network II often has more components (subgraphs) than the Real-data Network I. Thus, exploitation may become confined within some components (subgraphs), and fail to locate infectious nodes elsewhere, and exploration becomes more effective in presence of a large number of components. This explains the relative performances of REEr and RBEx in these. Contact tracing policy employs only exploitation, while active case finding policy uses most of its test budget for exploitation (and a small amount of the residual test budget for exploration). From Figure 6.9 and Figure 6.10, the contact tracing and the active case finding policies perform relatively poorly in the Real-data Network II compared to that in the Real-data Network I; this may again be attributed to the presence of a large number of components in the former.

As ℓ increases, as we show in Table 6.10 and Table 6.11 that the benefit of exploitation decreases. In Table 6.11, because of a large number of components, exploration always outperforms exploitation. However, in Table 6.10, we observe that exploration outperforms exploitation only for larger values

of ℓ. Our results are thus consistent with synthetic networks.

6.6. Conclusions and Future Research

In this chapter, we studied the problem of containing a spreading process (e.g. an infectious disease such as COVID-19) through sequential testing and isolation. We modeled the spread process by a compartmental model that evolves in time and stochastically spreads over a given contact network. Given a daily test budget, we aimed to minimize the cumulative infections. Under mild conditions, we proved that the problem can be cast as minimizing a supermodular function expressed in terms of nodes' probabilities of infection and proposed a greedy testing policy that attains a constant factor approximation ratio. We subsequently designed a computationally tractable reward-based policy that preferentially tests nodes that have higher rewards, where the reward of a node is defined as the expected number of new infections it induces in the next time slot. We showed that this policy effectively minimizes an upper bound on cumulative infections.

These policies, however, need knowledge about nodes' infection probabilities which are unknown and evolving. Thus, they have to be actively learned by testing. We discussed how testing has a dual role in this problem: (i) identifying the infected nodes and isolating them in order to contain the spread, and (ii) providing better estimates for the nodes' infection probabilities. We proved that this dual role of testing makes decision-making more challenging. In particular, we showed that reward-based policies that make decisions based on nodes' estimated infection probabilities can be arbitrarily sub-optimal while incorporating simple forms of exploration can boost their performance by a constant factor. Motivated by this finding, we devised exploration policies that probabilistically test nodes according to their rewards and numerically showed that when (i) the unregulated delay, (ii) the global clustering coefficient, or (iii) the average shortest path length increase, exploration becomes more beneficial as it provides better estimates of the nodes' probabilities of infection.

Given the history of observations, computing nodes' estimated probabilities of infection is itself a core challenge in our problem. We developed a message-passing framework to estimate these probabilities utilizing the observations in the form of test results. This framework passes messages back and forth in time to iteratively predict the probabilities in the future and correct the errors in

the estimates in prior time instants. This framework can also be of independent interest.

We showed novel tradeoffs between exploration and exploitation, different from the ones commonly observed in multi-armed bandit settings: (i) in our setting, the number of arms is time-variant and actions cannot be repeated; (ii) the tradeoffs in our setting are not due to lack of knowledge about the network or the process model, but rather due to lack of knowledge about the time-evolving unknown set of infected nodes.

We now describe directions for future research.

Our framework can be extended to incorporate delay and/or error in test results in a relatively straightforward manner (an outline of the extension incorporating a delay is given in Appendix E.9), but generalizing the performance guarantees for the proposed policies in these cases forms a direction of future research. This includes establishing fundamental lower bounds using genie-aided myopic policies.

6.6.1. Impact Statements

We have made several assumptions for the purpose of analytical and computational tractability which do not hold in practice: (1) the infections from different nodes are independent (2) given the entire history of testing results the states of nodes on the truncation day are independent (Assumption 9), (3) the symptoms need not be considered in deciding who should be tested and (4) the public health authority knows the entire network topology and uses it to determine who should be tested (5) independence of states of nodes (in one step). The first two assumptions were used to derive the message-passing framework and to prove that the objective function is super-modular which in turn led to a myopic testing strategy which is also optimal. The first assumption is reasonable as specific actions of infected individuals, eg, coughing, touching, and spreading the infection, which is undertaken independently.

We now consider the second assumption, ie, Assumption 9, in which we assume that the nodes' states $\zeta(t-g)$ (in the posterior probability space on the day $t-g$) are independent. Note that g is the truncation time for each backward step, that is, once we get the observations $\underline{Y}(t)$, we do

the backward step and truncate at time $t-g$. This assumption does not impose independence on the state of the nodes, but only in the posterior space at a specific time. That is, in the process of propagating information back to time $t-g$, we are assuming that there is no further correlation between time $t-g-1$ and time $t-g$ worthwhile to exploit given observations at time t. Naturally, as g gets larger and larger, our framework and calculations become more precise, as the impact of the testing results at time t in inferring about the nodes' probabilities at time $t-g$ vanishes as g gets large. But an increase in g significantly increases the computation time. Therefore, for computational tractability, of the backward update equations, we use $g=1$. In principle, the derivations of the backward update equations can be generalized in a straightforward manner to $g>1$. But designing approximation strategies that ensure computational tractability for larger g constitutes a direction of future research.

Consider the third assumption. We have not considered symptoms in determining who to test. But for some infectious diseases, symptoms are a reliable manifestation of the disease (e.g., Ebola). In principle, our testing framework can be generalized in a straightforward manner to consider symptoms by introducing additional states in the compartmental model for the evolution of the disease. But the introduction of additional states significantly increases the computation time, for example of the forward and backward updates of the probabilities that individuals have the disease, which renders implementation of our framework challenging. Considering symptoms while retaining computational tractability constitutes a direction for future research.

Next, consider the fourth assumption. In practice, public health authorities will not typically know contact networks in their entirety particularly when they are large, for example, as in large cities. However, small network topologies, for example, contact networks within a community, may be observed by the public health authority. As a specific example, the Government of China fully detected contact networks in many communities in Wuhan and tracked paths traversed by every individual [238]. This tracking may also generate concerns about privacy which is beyond the scope of this paper. Nonetheless, the technology for learning contact networks in their entirety for small communities exists and our framework can be utilized for those. Generalizing our framework to

obtain approximation guarantees when contact networks can only be partially observed constitutes a direction of future research.

Finally, consider the last assumption. Note that it is a strong assumption and clearly does not hold in general but it has been resorted to for only one step in the entire framework. Specifically to obtain Equation (6.5) we have assumed that the state of the nodes are independent. This allows us to obtain a simple expression in (6.5) in terms of the infection probabilities. We do not use this independence assumption in the rest of the chapter.

APPENDIX A

Proofs in Chapter 2

A.1. Sufficiency of Unit Buffer Size

Consider two types of policies: policies with buffer size 1, denoted by π_1, and policies with larger buffer sizes, denoted by π_2. To differentiate the two policies and their corresponding queues, we label the packets inside the queues by *new* and *old*. A new packet in a queue refers to the latest arrival. A packet in a queue is considered old if there is a newer packet in the same queue or if the packet (or a fresher packet) from that source is already delivered at the receiver. In the following, we refer to the freshest old packet as the old packet. At a given time slot, denote the new packet and the old packet of source i by $p_{new}^{(i)}$ and $p_{old}^{(i)}$, respectively. Denote the arrival times of the new and old packets as $t_i^{(\mathrm{n})}$ and $t_i^{(\mathrm{o})}$. It is clear that $t_i^{(\mathrm{n})} > t_i^{(\mathrm{o})}$. We will show that no matter what policy π_2 does, there is always a policy of type π_1 whose resulting age is at least as low as π_2 with respect to every source node.

At time slot t', suppose policy π_2 chooses certain action, then we design policy π_1 to follow the same action with the new packet. In this time slot, under π_2 a subset of sources transmit packets. Denote the index of these sources by $\mathcal{I}$. For the sources which do not transmit packets, the AoI under both policies will increase by 1. For the sources in $\mathcal{I}$, we have the following two cases:

Case 1. Suppose collision happens in time slot t'. Then, no packet is delivered, and the AoI of these sources under both policies will increase by 1.

Case 2. If a packet is delivered, which implies the cardinal of $\mathcal{I}$, $|\mathcal{I}| = 1$. Denote the index of this source by i. Then at the next time slot, the AoI under π_1 drops to $h_i^{\pi_1}(t'+1) = t' - t_i^{(\mathrm{n})} + 1$, and the AoI under π_2 drops to $h_i^{\pi_2}(t'+1) = t' - t_i^{(\mathrm{o})} + 1 > h_i^{\pi_1}(t'+1)$. This means that from t' onward $h_i^{\pi_2}(t)$ will be point-wise larger or equal to $h_i^{\pi_1}(t)$, $t > t'$.

A.2. Proof of Proposition 1

Consider any transmission policy and a large time horizon K. Let L_i be the number of remaining time slots after the last packet delivery in source i. The NAAoI defined in (2.3) can be re-written as follows:

$$J_K^{\pi} = \frac{1}{M^2}\sum_{i=1}^{M}\frac{1}{K}\sum_{k=1}^{K}h_i(k) = \frac{1}{M^2}\sum_{i=1}^{M}\frac{1}{K}\Big(\sum_{m=1}^{N_i(K)}\Gamma_i(m) + \frac{1}{2}L_i^2 + D_i(N_i(K))L_i - \frac{1}{2}L_i\Big),$$

where $\Gamma_i(m)$ was expressed in (2.4). Since $D_i(m) \geq 1$ for all $1 \leq m \leq N_i(K)$, we can lower bound $\Gamma_i(m)$ by substituting $D_i(m-1) = 1$ in (2.4). Using similar steps as [13, Eqns. (9) - (14)], we find

$$J^{\pi}(M) \geq \lim_{K\to\infty}\mathbb{E}[\frac{1}{2M^2}\sum_{i=1}^{M}\frac{K}{N_i(K)} + \frac{1}{2M}]. \tag{A.1}$$

Recall that $N_i(K)$ is the total number of packets delivered by source i. In the limit of $K \to \infty$, $\frac{N_i(K)}{K}$ is the throughput of source i. By the model assumption, in every time slot, at most one packet is delivered in the system. Therefore,

$$\lim_{K\to\infty}\mathbb{E}[\sum_{i=1}^{M}\frac{N_i(K)}{K}] \leq C_{RA}. \tag{A.2}$$

Now note that by the Cauchy-Schwarz inequality, we have

$$\lim_{K\to\infty}\mathbb{E}[\sum_{i=1}^{M}\frac{N_i(K)}{K}]\mathbb{E}[\sum_{i=1}^{M}\frac{K}{N_i(K)}] \geq M^2. \tag{A.3}$$

Thus using (A.2) and (A.3), we find

$$\lim_{K\to\infty}\mathbb{E}[\sum_{i=1}^{M}\frac{K}{N_i(K)}] \geq \frac{M^2}{C_{RA}}. \tag{A.4}$$

Inserting (A.4) back into (A.1), we obtain

$$J_M^{\pi} \geq \frac{1}{2C_{RA}} + \frac{1}{2M}. \tag{A.5}$$

A.3. Proof of Proposition 2

Suppose all packets are delivered instantaneously with one time-unit delay and without experiencing collisions. A lower bound to NAAoI in this scenario constitutes a lower bound to NAAoI in our setup. Let $X_i(m)$ denote the inter-arrival time between the mth and $(m+1)$st packets. $\{X_i(m)\}_m$ is a geometric iid sequence. Under the assumption of instantaneous delivery, $I_i(m) = X_i(m)$. It hence follows from (2.4) that

$$\Gamma_i(m) = \sum_{k=T_i(m)}^{T_i(m)+X_i(m)-1} h_i(k) = \frac{1}{2}X_i(m)^2 + \frac{1}{2}X_i(m). \tag{A.6}$$

Thus, similar with [13], the time-average AoI of source i, denoted by H_i, is

$$\mathbb{E}[H_i] = \lim_{K\to\infty} \frac{1}{K}\sum_{k=1}^{K} h_i(k) = \frac{\mathbb{E}[X^2]}{2\mathbb{E}[X]} + \frac{1}{2}. \tag{A.7}$$

Since X in (A.7) has a geometric distribution with parameter θ, we find $\mathbb{E}[H_i] = \frac{2-\theta}{2\theta} + \frac{1}{2}$. Note that NAAoI can be captured by $\frac{1}{M^2}\sum_{i=1}^{M} H_i$ and one can hence conclude that

$$J^{\pi}(M) \geq \frac{1}{M\theta}. \tag{A.8}$$

A.4. Proof of Proposition 3

First, consider a source node i whose queue is empty. This means that no new packet has arrived at that transmitter since the last delivery (from that source node) at the receiver; i.e., $h_i(k) = w_i(k)$ and hence $\delta_i(k) = 0$. Such nodes i are thus irrelevant because $\delta_j(k) \geq 0$ for all source nodes j. Now consider nodes with non-empty queues. Among these nodes, $d_i(k)$ is non-zero if and only if $(\lambda_1, \ldots, \lambda_M)$ is a vector consisting of 0's except for $\lambda_i = 1$. Hence at most one $d_i(k)$ can be equal to 1. Call the corresponding source node $\ell(k)$. Expression (2.11) is minimized when $d_{\ell(k)}(k)$ picks the largest $\delta_j(k)$.

A.5. Probabilities of idle, deliveries and collisions under Slotted ALOHA

Lemma 15. *Consider any stabilized slotted ALOHA scheme. Define G as the expected number of attempted transmissions in a slot. Then, for M large, the probability of delivering a packet is (asymptotically) Ge^{-G}, the probability of an idle system is (asymptotically) e^{-G}, and the probability of collisions is (asymptotically) $1-e^{-G}-Ge^{-G}$. In particular, when $G=1$, the maximum probability of delivery is $1/e$, the corresponding probabilities of collisions and idle systems are $1-2/e$ and $1/e$, respectively.*

Proof. The idea of the proof is very similar to [152, Chapter 4]. However, the settings are different: [152, Chapter 4] considered that packets arrive as a Poisson process (in a continuous-time system) and the buffer size is infinite, while this proof considers that packets arrive as a Bernoulli process (in a discrete-time system) and the buffer size is 1. Define the nodes that are not backlogged as *fresh* nodes. Each fresh node transmits a packet directly in a slot if it is not empty, and it generates/receives a packet with probability θ, thus a fresh node transmits a packet with probability θ. Let $P_a\big(i,n(k)\big)$ be the probability that i fresh nodes transmit a packet in a time slot and let $P_s\big(j,n(k)\big)$ be the probability that j backlogged nodes transmit. We have:

$$P_a\big(i,n(k)\big) = \binom{M-n(k)}{i}(1-\theta)^{M-n(k)-i}\theta^i \tag{A.9}$$

$$P_s\big(j,n(k)\big) = \binom{n(k)}{j}(1-p_b(k))^{n(k)-i}p_b(k)^i. \tag{A.10}$$

Thus, in slot k, when a packet is delivered, i.e., $\sum_{i=1}^{M} d_i(k)=1$, the probability is

$$\Pr(\sum_{i=1}^{M} d_i(k)=1) = P_a\big(1,n(k)\big)P_s\big(0,n(k)\big) + P_a\big(0,n(k)\big)P_s\big(1,n(k)\big). \tag{A.11}$$

If the channel does not transmit a packet in a slot, i.e., we have an idle channel, $\sum_{i=1}^{M} d_i(k)=0$, $c(k)=0$. The probability of an idle system in slot k is

$$\Pr(\sum_{i=1}^{M} d_i(k)=0, c(k)=0) = P_a\big(0,n(k)\big)P_s\big(0,n(k)\big). \tag{A.12}$$

Define the attempt rate $G = (M - n(k))\theta + n(k)p_b(k)$ as the expected number of attempted transmissions in a slot. From (A.9) and (A.10), the probability of delivery is

$$\Pr(\sum_{i=1}^{M} d_i(k) = 1) = \big(M - n(k)\big)(1-\theta)^{M-n(k)-1}\theta(1-p_b)^{n(k)} + (1-\theta)^{M-n(k)}n(k)(1-p_b)^{n(k)-1}p_b$$

and the probability of an idle channel is

$$\Pr(\sum_{i=1}^{M} d_i(k) = 0, c(k) = 0) = (1-\theta)^{M-n(k)}(1-p_b)^{n(k)}.$$

Note that the valid regime of θ is $\theta M < \frac{1}{e}$, and thus θ, p_b are small. Using the approximation $(1-x)^{-y} \approx \exp(-xy)$ for small x, we find

$$\begin{aligned}
&\Pr(\sum_{i=1}^{M} d_i(k) = 1) \approx Ge^{-G} \\
&\Pr(\sum_{i=1}^{M} d_i(k) = 0, c(k) = 0) \approx e^{-G} \\
&\Pr(c(k) = 1) \approx 1 - Ge^{-G} - e^{-G}.
\end{aligned}$$

Taking the first derivative of the function Ge^{-G}, we can find the maximum point is 1 for $0 < G \leq 1$. So the maximum probability of delivery is

$$\Pr(\sum_{i=1}^{M} d_i(k) = 1) \approx 1/e, \tag{A.13}$$

correspondingly, we have

$$\Pr(\sum_{i=1}^{M} d_i(k) = 0, c(k) = 0) \approx 1/e, \tag{A.14}$$

$$\Pr(c(k) = 1) = 1 - 2/e. \tag{A.15}$$

□

A.6. Proof of Theorem 1

The proof is organized into three parts:

Part 1: Preliminaries. In time slot k, denote the time just before the arrival of new packets by k^- and the time just after the arrival of new packets by k^+. We hence write $\delta_i(k^-) = h_i(k^-) - w_i(k^-)$ and $\delta_i(k^+) = h_i(k^+) - w_i(k^+)$. Suppose a packet is delivered from the i^{th} source at the end of time slot $k-1$. We then have $\delta_i\big(k^-\big) = 0$. From (2.14), since all nodes have the same arrival rate and transmission policy, the sequences $\{h_i(k^-)\}_{k=1}^{\infty}$, $\{h_i(k^+)\}_{k=1}^{\infty}$, $\{w_i(k^-)\}_{k=1}^{\infty}$, $\{w_i(k^+)\}_{k=1}^{\infty}$, $\{\delta_i(k^-)\}_{k=1}^{\infty}$, $\{\delta_i(k^+)\}_{k=1}^{\infty}$ are identical random variables across $i = 1, 2, \cdots, M$, respectively. Recall that source nodes with $\delta_i(k^-) = 0$ are 0-order nodes and define $n_0(k^-)$ as the number of 0-order nodes at time k^-.

At the beginning of time slot k, on average, θM new packets arrive at the sources, and $\theta n_0(k^-)$ 0-order nodes receive new packets. Suppose source i is a 0-order node and $h_i(k^-) - w_i(k^-) = 0$. If source i receives new packets, then the source's AoI changes from $w_i(k^-)$ to $w_i(k^+) = 0$ and the destination's AoI $h_i(k^-)$ remains the same as $h_i(k^+)$. Thus,

$$\delta_i(k^+) = h_i(k^+) - w_i(k^+) = h_i(h^+) = h_i(k^-) > h_i(k^-) - w_i(k^-) = \delta_i(k^-) = 0,$$

which implies that if a 0-order source receives a new packet, then it is not a 0-order source at k^+.

Fix any large M and denote the maximum throughput of Slotted ALOHA with $C_{SA}(M)$. We know that $\lim_{M\to\infty} C_{SA}(M) = e^{-1}$. The recursion of the expected number of 0-order nodes is:

$$\mathbb{E}[n_0\big((k+1)^-\big)] = (1-\theta)\mathbb{E}[n_0(k^-)] + \min(M\theta, C_{SA}(M)) \tag{A.16}$$

where the second term on the right-hand side is the average number of delivered packets per time slot. Since we consider a stabilized slotted ALOHA, $\lim_{k\to\infty} \mathbb{E}[n_0(k^-)]$ exists. Denote

$$n_0^* = \lim_{k\to\infty} \mathbb{E}[n_0(k^-)].$$

Letting $k \to \infty$ on both sides of (A.16), we have

$$n_0^* = (1-\theta)n_0^* + \min(M\theta, C_{SA}(M)). \tag{A.17}$$

Note that

$$\lim_{M\to\infty} \min(M\theta, C_{SA}(M)) = \lim_{M\to\infty} M\theta = \eta. \tag{A.18}$$

From (A.17) and (A.18), we have

$$\lim_{M\to\infty} \frac{n_0^*}{M} = 1. \tag{A.19}$$

Part 2: Find the expression of NAAoI. Using (2.3), we have

$$J^{SA}(M) = \lim_{K\to\infty} \mathbb{E}\left[\frac{1}{M^2}\sum_{i=1}^{M}\frac{1}{K}\sum_{k=1}^{K} h_i(k^-)\right] \triangleq J_1 + J_2$$

where

$$J_1 = \lim_{K\to\infty} \mathbb{E}\left[\frac{1}{M^2}\sum_{i=1}^{M}\frac{1}{K}\sum_{k=1}^{K} w_i(k^-)\right]$$
$$J_2 = \lim_{K\to\infty} \mathbb{E}\left[\frac{1}{M^2}\sum_{i=1}^{M}\frac{1}{K}\sum_{k=1}^{K} \delta_i(k^-)\right].$$

Part 3: Find the limit of NAAoI. First, we consider J_1. $w_i(k^-)$ has a geometric distribution starting from 1 with parameter θ for all i. Employing the law of large number, we find

$$J_1 = \frac{1}{M\theta}. \tag{A.20}$$

Next, we consider J_2 and prove that its limit in large M approaches zero. Note from (2.12) that $\delta_i(k) = 0$ if source i is empty in time slot k and $\delta_i(k) > 0$ if a packet remains in source i in time slot

k. We first note that $\delta_i(k)$ is upper bounded by $h_i(k)$. Let us consider a worse case in which buffer sizes are infinite. In this case, assuming stationarity[26], denote the inter-arrival time and delay of packets with respect to source i by X_i and D_i. Since the Bernoulli arrival process has parameter θ, we have $\mathbb{E}[X_i] = \frac{1}{\theta} = \frac{M}{\eta}$. Moreover, $\mathbb{E}[D_i]$ is approximately bounded by some constant independent of the number of sources M [239]. Now we observe that for each packet delivery, the expected peak age at the destination is upper bounded by $\mathbb{E}[X_i] + \mathbb{E}[D_i]$. We can hence write

$$\mathbb{E}[\delta_i(k)|\delta_i(k) > 0] \leq \mathbb{E}[X_i] + \mathbb{E}[D_i] \tag{A.21}$$

which implies that $\mathbb{E}[\delta_i(k)]$ is $O(M)$.

Now expand J_2:

$$\begin{aligned}
J_2 &= \lim_{K\to\infty} \mathbb{E}[\frac{1}{M^2}\sum_{i=1}^{M}\frac{1}{K}\sum_{k=1}^{K}\delta_i(k^-)] = \lim_{K\to\infty} \mathbb{E}[\frac{1}{M^2}\frac{1}{K}\sum_{k=1}^{K}\sum_{i=1}^{M}\delta_i(k^-)1_{\delta_i(k^-)>0}] \\
&= \lim_{K\to\infty} \frac{1}{M^2}\frac{1}{K}\sum_{k=1}^{K}\sum_{i=1}^{M}\mathbb{E}[\delta_i(k^-)1_{\delta_i(k^-)>0}] \leq \limsup_{k\to\infty} \frac{1}{M^2}\sum_{i=1}^{M}\mathbb{E}[\delta_i(k^-)1_{\delta_i(k^-)>0}] \\
&= \limsup_{k\to\infty} \frac{1}{M^2}\sum_{i=1}^{M}\Big(\Pr(\delta_i(k^-) > 0)\mathbb{E}[\delta_i(k^-)|\delta_i(k^-) > 0]\Big).
\end{aligned}$$

Since for k large enough the conditional expectation $\mathbb{E}[\delta_i(k^-)|\delta_i(k^-) > 0]$ is $O(M)$, it remains to prove that in the limit of large M, $\lim_{k\to\infty}\frac{1}{M}\sum_{i=1}^{M}\Pr(\delta_i(k^-) > 0)$ vanishes. But this holds because we can write

$$\begin{aligned}
\lim_{k\to\infty}\frac{1}{M}\sum_{i=1}^{M}\Pr(\delta_i(k^-) > 0) &= \lim_{k\to\infty}\mathbb{E}[\frac{1}{M}\sum_{i=1}^{M}1_{\delta_i(k^-)>0}] \\
= \lim_{k\to\infty}\mathbb{E}[\frac{1}{M}(M - n_0(k^-))] &= \frac{M - n_0^*}{M}
\end{aligned}$$

which goes to zero by (A.19).

Finally, we prove that for any scheme, $J^{SA}(M)$ is lower bounded by $1/\eta$. From Proposition 2 (and

[26]This assumption approximately holds for infinite time horizon T

letting $M \to \infty$), we have

$$\lim_{M\to\infty} J^{SA}(M) \geq \lim_{M\to\infty} \frac{1}{M\theta} = \frac{1}{\eta}.$$

Therefore, slotted ALOHA can reach the lower bound when $\theta \in (0, \frac{1}{eM}]$ and is hence optimal.

A.7. Proof of Lemma 1

Before presenting the proof, we state the following straightforward lemma (whose proof is omitted).

Lemma 16. *At the beginning of time slot k, before new packets arrive at source i, $w_i(k^-) > 0$ and its probability distribution is*

$$\Pr\left(w_i(k^-) = j\right) = \theta(1-\theta)^{j-1},\ j = 1, 2, 3, \cdots . \tag{A.22}$$

First, consider $m = 0$ and suppose source i is a 0-order node. From Lemma 16, we know that $w_i(k^-) > 0$. Moreover, since $\delta_i(k^-) = 0$, we conclude $h_i(k^-) = w_i(k^-) > 0$. Once the 0-order node has a new arrival, $w_i(k^+) = 0$ and $h_i(k^+) = h_i(k^-)$, resulting in $\delta_i(k^+) = h_i(k^+) > 0$; i.e., the order of the node increases. In other words, the order of a 0-order node increases once it receives a new packet. In total, the fraction of 0-order nodes that become of higher order is on average $\theta\ell_0(k-1)$. Thus,

$$\ell_0(k^+) = (1-\theta)\ell_0(k-1).$$

Similarly, we consider $m \geq 1$. The fraction of m-order nodes that have new arrivals is $\theta\ell_m(k-1)$. These nodes will have larger orders. Suppose source i is of order m, $m \geq 1$, i.e., $\delta_i(k^-) = h_i(k^-) - w_i(k^-) = m$, once a new packet arrives, then $w_i(k^+) = 0$, $h_i(k^+) = h_i(k^-)$, and $\delta_i(k^+) = h_i(k^+) = m + w_i(k^-)$. From Lemma 16, $w_i(k^-) > 0$, then $\delta_i(k^+) > \delta_i(k^-) = m$. The order of a m-order node increases once it receives a new packet. In total, the fraction of m-order nodes that have larger orders is $\theta\ell_m(k-1)$.

More precisely, consider a j-order node, $j < m$. This node becomes an m-order node if it receives a new packet and $w_i(k^-) = m - j$. Using Lemma 16, we can write

$$\begin{aligned}\ell_m(k^+) =&(1-\theta)\ell_m(k-1) + \sum_{j=0}^{m-1} \theta\ell_j(k-1)\Pr\left(w_i(k^-) = m-j\right)\\ =&(1-\theta)\ell_m(k-1) + \sum_{j=0}^{m-1} \theta\ell_j(k-1)\theta(1-\theta)^{m-j-1}\\ =&(1-\theta)\ell_m(k-1) + \theta^2\sum_{j=0}^{m-1} \ell_j(k-1)(1-\theta)^{m-j-1}. \end{aligned} \tag{A.23}$$

where the second term in (A.23) on the left-hand side is the average fraction of nodes that have just become of order m. Denoting it by a_m, we have

$$a_m(k) = \theta^2\sum_{j=0}^{m-1} \ell_j(k-1)(1-\theta)^{m-j-1}. \tag{A.24}$$

A.8. Proof of Lemma 2

From the expression of ℓ_m^* in (2.29), $0 \le m \le T^* - 1$, we obtain

$$\ell_0^* = \frac{1}{eM\theta} \tag{A.25}$$

$$\ell_m^* = \frac{a_m^*}{\theta}, \quad 0 \le m \le T^* - 1. \tag{A.26}$$

From (2.26), a_m^* depends on $\{\ell_j^*\}_{j\le m-1}$ and from (A.26), ℓ_m^* depends on a_m^* for $1 \le m \le T^* - 1$. So they can be recursively found and in particular, it is not difficult to prove for all $1 \le m \le T^* - 1$:

$$a_m^* = \frac{\theta}{eM} \tag{A.27}$$

$$\ell_m^* = \frac{1}{eM}. \tag{A.28}$$

We prove this by mathematical induction on $T^* \geq 2$. For $T^* - 1 = 1$, the statement holds because

$$a_1^* = \theta^2 \ell_0^* = \frac{\theta}{eM} \tag{A.29}$$

$$\ell_1^* = \frac{a_1^*}{\theta} = \frac{1}{eM}. \tag{A.30}$$

Now suppose the statements (A.27) - (A.28) hold for $m \leq \mathtt{T}^* - 1 = k$. We prove the statement for $\mathtt{T}^* - 1 = k + 1$ and in particular we find a_{k+1}^* and ℓ_{k+1}^* below:

$$\begin{aligned} a_{k+1}^* =& \theta^2 \sum_{j=0}^{k} \ell_j^* (1-\theta)^{k-j} = \theta^2 \frac{1}{eM} \sum_{j=1}^{k} (1-\theta)^{k-j} + \theta^2 (1-\theta)^k \frac{1}{eM\theta} \\ =& \theta^2 \frac{1}{eM} \frac{1-(1-\theta)^k}{\theta} + \theta (1-\theta)^k \frac{1}{eM} = \frac{\theta}{eM}. \end{aligned}$$

Next, using (A.26), we find

$$\ell_{k+1} = \frac{1}{eM}. \tag{A.31}$$

Moreover, using the derivation in (A.31), we also find

$$a_{T^*}^* = \frac{\theta}{eM}. \tag{A.32}$$

Finally, from (2.25), we obtain

$$\ell_m^{+*} = \frac{1}{eM}, \quad 1 \leq m \leq \mathtt{T}^* - 1. \tag{A.33}$$

A.9. Proof of Theorem 2

Summing (2.29) on both sides, we have $\sum_{m \geq 1} a_m^* = \theta$. Moreover, $\mathtt{T}^*$ satisfies

$$\mathtt{T}^* = \max\{t | \sum_{m \geq t} a_m^* \geq \frac{1}{eM}\} \tag{A.34}$$

by its definition in (2.19). The term $\sum_{m\geq \mathtt{T}^*} a_m^*$ can be re-written as follows:

$$\sum_{m\geq \mathtt{T}^*} a_m^* = \sum_{m\geq 1} a_m^* - \sum_{m<\mathtt{T}^*} a_m^* \overset{(a)}{=} \theta - (\mathtt{T}^*-1)\frac{\theta}{eM} \tag{A.35}$$

where (a) follows by (2.32) in Lemma 2. On the other hand, $\sum_{m\geq \mathtt{T}^*} a_m^*$ satisfies the following inequality by (A.34):

$$\sum_{m\geq \mathtt{T}^*} a_m^* \geq \frac{1}{eM}. \tag{A.36}$$

Putting (A.35) and (A.36) together, we find

$$\mathtt{T}^* = \lfloor eM - \frac{1}{\theta} + 1 \rfloor \tag{A.37}$$

since $\mathtt{T}^*$ is an integer.

A.10. Proof of Theorem 3

The proof is organized into three parts:

Part 1: Preliminaries. In this part, we discuss some notations and preliminaries which will be used in the proof. Denote the time just before the arrival of new packets by k^- and the time just after the arrival of new packets by k^+. Since we have assumed that all nodes are identical, the sequence $\{h_i(k^+)\}_{k=1}^{\infty}$ is identical (but not independent) across all $i = 1, 2, \cdots, M$. From (2.2), $\{w_i(k^+)\}_{k=1}^{\infty}$ are i.i.d with respect to i. Therefore, the sequence $\{\delta_i(k^+)\}_{k=1}^{\infty}$ is identical but not independent for all $i = 1, 2, \cdots, M$.

Since $\theta = \frac{1}{o(M)}$ and in particular $\theta > \frac{1}{eM}$, from Lemma 2, $\ell_m^{+*} = \frac{1}{eM}$ for $m = 1, 2, \cdots, \mathtt{T}^*$ and $\ell_0^{+*} = \frac{o(M)}{eM}$. From Theorem 2, $\mathtt{T}^* = \lfloor eM - 1/\theta + 1 \rfloor = \lfloor eM - o(M) + 1 \rfloor$. Denote

$$s_{\mathtt{T}^*} = \sum_{m=0}^{\mathtt{T}^*-1} \ell_m^{+*}.$$

In the limit of large M, we have

$$\lim_{M\to\infty} s_{\mathtt{T}^*} = \lim_{M\to\infty} \frac{o(M) + \lfloor eM - o(M) + 1 \rfloor - 1}{eM} = 1. \tag{A.38}$$

The expected number of inactive nodes is $Ms_{\mathtt{T}^*}$ and the expected number of active nodes is $M(1 - s_{\mathtt{T}^*})$.

Part 2: Find the expression of NAAoI. Let $\alpha_i = \frac{1}{M}$ for $i = 1, 2, \cdots, M$ in (2.3):

$$J^{SAT}(M) = \lim_{K\to\infty} \mathbb{E}[\frac{1}{M^2} \sum_{i=1}^{M} \frac{1}{K} \sum_{k=1}^{K} h_i(k^+)] \triangleq J_1 + J_2$$

where

$$J_1 = \lim_{K\to\infty} \mathbb{E}[\frac{1}{M^2} \sum_{i=1}^{M} \frac{1}{K} \sum_{k=1}^{K} w_i(k^+)]$$
$$J_2 = \lim_{K\to\infty} \mathbb{E}[\frac{1}{M^2} \sum_{i=1}^{M} \frac{1}{K} \sum_{k=1}^{K} \delta_i(k^+)].$$

In addition, $J_2 = J_{21} + J_{22}$, where

$$J_{21} = \lim_{K\to\infty} \mathbb{E}[\frac{1}{K} \sum_{k=1}^{K} \frac{1}{M^2} \sum_{i:\delta_i(k^+)<\mathtt{T}^*} \delta_i(k^+)]$$
$$J_{22} = \lim_{K\to\infty} \mathbb{E}[\frac{1}{K} \sum_{k=1}^{K} \frac{1}{M^2} \sum_{i:\delta_i(k^+)\geq\mathtt{T}^*} \delta_i(k^+)].$$

Part 3: Find the limit of NAAoI with respect to M. First, we consider J_1. From (2.2), $w_i(k^+)$ has a geometric distribution with parameter θ $\big($with $w_i(k^+) = 0, 1, 2, \cdots\big)$ for all i. Let w have the same distribution as $w_i(k^+)$. We thus have

$$J_1 = \frac{1}{M}\mathbb{E}[w] = \frac{1-\theta}{M\theta}. \tag{A.39}$$

Next, we consider J_{21}:

$$\begin{aligned}
\lim_{M\to\infty} J_{21} &= \lim_{M\to\infty}\lim_{K\to\infty}\mathbb{E}[\frac{1}{K}\sum_{k=1}^{K}\frac{1}{M^2}\sum_{i:\delta_i(k^+)<\mathtt{T}^*}\delta_i(k^+)] \\
&= \lim_{M\to\infty}\lim_{K\to\infty}\mathbb{E}[\frac{1}{K}\sum_{k=1}^{K}\frac{1}{M^2}\sum_{i=1}^{M}\delta_i(k^+)1_{(\delta_i(k^+)<\mathtt{T}^*)}] \\
&= \lim_{M\to\infty}\lim_{K\to\infty}\mathbb{E}[\frac{1}{K}\sum_{k=1}^{K}\frac{1}{M}\sum_{j=1}^{\mathtt{T}^*-1}\frac{\sum_{i=1}^{M}\delta_i(k^+)1_{(\delta_i(k^+)=j)}}{M}] \\
&= \lim_{M\to\infty}\lim_{K\to\infty}\mathbb{E}[\frac{1}{K}\sum_{k=1}^{K}\frac{1}{M}\sum_{j=1}^{\mathtt{T}^*-1}\frac{j\sum_{i=1}^{M}1_{(\delta_i(k^+)=j)}}{M}] \\
&= \lim_{M\to\infty}\lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K}\frac{1}{M}\sum_{j=1}^{\mathtt{T}^*-1}j\frac{\mathbb{E}[\sum_{i=1}^{M}1_{(\delta_i(k^+)=j)}]}{M}. \qquad \text{(A.40)}
\end{aligned}$$

Substituting $\ell_j(k^+)$ for the term $\frac{\mathbb{E}[\sum_{i=1}^{M}1_{(\delta_i(k^+)=j)}]}{M}$, we find

$$\lim_{M\to\infty} J_{21} = \lim_{M\to\infty}\lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K}\frac{1}{M}\sum_{j=1}^{\mathtt{T}^*-1}j\ell_j(k^+). \qquad \text{(A.41)}$$

By stationarity, note that

$$\ell_j^{*+} = \lim_{k\to\infty}\ell_j(k^+).$$

By the Cesaro Mean Lemma,

$$\lim_{K\to\infty}\frac{\sum_{k=1}^{K}\ell_j(k^+)}{K} = \ell_j^{*+}.$$

Therefore,

$$\lim_{M\to\infty} J_{21} = \lim_{M\to\infty}\frac{1}{M}\sum_{j=1}^{\mathtt{T}^*-1}j\ell_j^{*+} = \lim_{M\to\infty}\frac{1}{M}\frac{\mathtt{T}^*(\mathtt{T}^*-1)}{2}\frac{1}{eM} = \frac{e}{2} \qquad \text{(A.42)}$$

where in the last step we have substituted $\ell_j^{*+} = \frac{1}{eM}$ for $j = 1, \ldots, \mathtt{T}^* - 1$ (see Lemma 2).

Finally, we consider J_{22}:

$$\begin{aligned}\lim_{M\to\infty} J_{22} &= \lim_{M\to\infty}\lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K}\frac{1}{M^2}\sum_{i=1}^{M}\mathbb{E}[\delta_i(k^+)1_{\delta_i(k^+)\geq \mathtt{T}^*}] \\ &= \lim_{M\to\infty}\lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K}\frac{1}{M^2}\sum_{i=1}^{M}\Big(\mathbb{E}[\delta_i(k^+)|\delta_i(k^+)\geq \mathtt{T}^*]\Pr(\delta_i(k^+)\geq \mathtt{T}^*)\Big) \\ &\overset{(a)}{\leq} \lim_{M\to\infty}\lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K}\frac{1}{M^2}\sum_{i=1}^{M} cM\Pr(\delta_i(k^+)\geq \mathtt{T}^*). \end{aligned} \tag{A.43}$$

In the above chain of inequalities, step (a) holds because $\mathbb{E}[\delta_i(k^+)|\delta_i(k^+)\geq \mathtt{T}^*] = O(M)$. To show this, we first observe that $\delta_i(k)$ is increasing in k until a delivery occurs. Now, note that $\delta_i(k^+)$ is upper bounded by $\mathtt{T}^*$ plus the peak age at the first delivery after time slot k. The peak age is bounded by X_i (the inter-arrival time), which is $o(M)$ on average, plus delay D_i, which is constant on average (similar to (A.21)). The threshold $\mathtt{T}^*$ is also $O(M)$. So overall, we have

$$\mathbb{E}[\delta_i(k^+)|\delta_i(k^+)\geq \mathtt{T}^*] \leq cM$$

for some constant c. Note that

$$\Pr(\delta_i(k^+)=j) = \mathbb{E}[1_{\{\delta_i(k^+)=j\}}]$$

therefore

$$\begin{aligned}\frac{1}{M}\sum_{i=1}^{M}\Pr(\delta_i(k^+)\geq \mathtt{T}^*) &= \frac{1}{M}\sum_{i=1}^{M}\sum_{j\geq \mathtt{T}^*}\Pr(\delta_i(k^+)=j) = \sum_{j\geq \mathtt{T}^*}\frac{1}{M}\sum_{i=1}^{M}\Pr(\delta_i(k^+)=j) \\ &= \sum_{j\geq \mathtt{T}^*}\frac{1}{M}\sum_{i=1}^{M}\mathbb{E}[1_{\{\delta_i(k^+)=j\}}] = \sum_{j\geq \mathtt{T}^*}\ell_j(k^+).\end{aligned}$$

Again, by the Cesaro Mean Lemma,

$$\lim_{M\to\infty} J_{22} \le \lim_{M\to\infty}\lim_{K\to\infty} \frac{1}{K}\sum_{k=1}^{K} \frac{1}{M} cM \left(\sum_{j\ge T^*} \ell_j(k^+)\right)$$
$$= \lim_{M\to\infty} \frac{1}{M} cM \left(\sum_{j\ge T^*} \ell_j^*\right) = \lim_{M\to\infty} \frac{1}{M} cM(1-s_T^*) = 0.$$

The last equality follows from (A.38) ($\lim_{M\to\infty} s_{T^*} = 1$). Finally, summing J_1, J_{21} and J_{22}, we find

$$\lim_{M\to\infty} \mathbb{E}[J^{SAT}(M)] = \frac{e}{2}.$$

A.11. Proof of Theorem 4

Summing (2.29) on both sides, we have

$$\sum_{m\ge 1} a_m^* = \theta. \tag{A.44}$$

From the definition of the threshold in (2.19), $\mathtt{T}^*$ satisfies

$$\mathtt{T}^* = \max\left\{ t | \sum_{m\ge t} a_m^* \ge \min\left(\theta, \frac{C^{\pi^{(1)}}}{M}\right)\right\}. \tag{A.45}$$

If $\theta \le \frac{C^{\pi^{(1)}}}{M}$, we have $\mathtt{T}^* = 1$ by (A.44). If $\theta > \frac{C^{\pi^{(1)}}}{M}$, however, we have

$$\frac{C^{\pi^{(1)}}}{M} \le \sum_{m\ge \mathtt{T}^*} a_m^* = \sum_{m\ge 1} a_m^* - \sum_{m<\mathtt{T}^*} a_m^* \overset{(a)}{=} \theta - (\mathtt{T}^*-1)\frac{\theta C^{\pi^{(1)}}}{M} \tag{A.46}$$

where (a) follows from (A.44) and (2.37). Using (A.46) and noting that $\mathtt{T}^*$ is integer, we find

$$\mathtt{T}^* = \left\lfloor \frac{M}{C^{\pi^{(1)}}} - \frac{1}{\theta} + 1 \right\rfloor. \tag{A.47}$$

A.12. Proof of Theorem 5

The proof of Theorem 5 is almost exactly the same as that of Theorem 3. After replacing the sum arrival rate of the channel, e^{-1}, by $C^{\pi^{(1)}}$, from Part 1, Part 2 and Part 3 in the proof of Theorem 3, we have

$$J_1 = \frac{1}{M}\mathbb{E}[w] = \frac{1-\theta}{M\theta} \tag{A.48}$$

$$\lim_{M\to\infty} J_{21} = \lim_{M\to\infty} \frac{1}{M}\sum_{j=1}^{\mathtt{T}^*-1} j\ell_j^{*+} = \lim_{M\to\infty} \frac{1}{M}\frac{\mathtt{T}^*(\mathtt{T}^*-\mathtt{1})}{2}\frac{C^{\pi^{(1)}}}{M} = \frac{1}{2C^{\pi^{(1)}}} \tag{A.49}$$

and

$$\lim_{M\to\infty} J_{22} \leq \lim_{M\to\infty}\lim_{K\to\infty} \frac{1}{K}\sum_{k=1}^{K}\frac{1}{M^2}\sum_{i=1}^{M} cM\Pr(\delta_i(k^+) > \mathtt{T}^*). \tag{A.50}$$

From Part 3 in the proof of Theorem 3, we knew that the last inequality holds because $\mathbb{E}[\delta_i(k^+)|\delta_i(k^+) > \mathtt{T}^*] = O(M)$. This, however, is not as oblivious here. To show this, we first observe that $\delta_i(k)$ is increasing in k until a delivery occurs. Now, note that $\delta_i(k^+)$ is upper bounded by $\mathtt{T}^*$ plus the peak age at the first delivery after time slot k. The peak age is bounded by X_i (the inter-arrival time), which is $o(M)$ on average, plus delay D_i, whose expectation is upper bounded by a constant times M as formulated in Lemma 17 below. Therefore, by the counterpart of the proof of Theorem 3, we have $\lim_{M\to\infty} J_{22} = 0$ and summing J_1, J_{21} and J_{22},

$$\lim_{M\to\infty} \mathbb{E}[J^{GSAT}(M)] = \frac{1}{2C^{\pi^{(1)}}}.$$

Lemma 17. *The expectation of delay, $\mathbb{E}[D_i]$, satisfies*

$$\mathbb{E}[D_i] \leq c'M$$

where c' is a constant that depends on the employed transmission policy.

Proof. Recall that $\lim_{M\to\infty} C^{\pi^{(1)}}(M) = C^{\pi^{(1)}}$. Denote the inter-delivery time for source i by I_i. Thus the expected number of received packets from source i from time slot 0 to K is $\frac{K}{\mathbb{E}[I_i]}$. Since $C^{\pi^{(1)}}(M)$ is the sum throughput, we have

$$C^{\pi^{(1)}}(M) = \lim_{K\to\infty} \frac{\sum_{i=1}^{M} \frac{K}{\mathbb{E}[I_i]}}{K}.$$

Moreover, all nodes are statistically identical. Therefore, $C^{\pi^{(1)}}(M) = \frac{M}{\mathbb{E}[I_i]}$ and $\mathbb{E}[I_i] = \frac{M}{C^{\pi^{(1)}}(M)}$. Note that $\mathbb{E}[D_i] \leq \mathbb{E}[I_i]$ and for any $\epsilon > 0$, there exists a $N_0 > 0$ such that $C^{\pi^{(1)}}(M) \geq C^{\pi^{(1)}} - \epsilon$ for all $M \geq N_0$. Therefore,

$$\mathbb{E}[D_i] \leq \frac{M}{C^{\pi^{(1)}} - \epsilon} \triangleq c'M.$$

□

APPENDIX B

Proofs in Chapter 3

B.1. The Strong Law of Large Numbers holds for $\{I_\ell^{(i)}\}_\ell$

From Definition 3 and Definition 4, $I_\ell^{(i)} = J_\ell^{(i)} + U_\ell^{(i)} - 1$, and $J_\ell^{(i)}$ is measurable and independent of $U_\ell^{(i)}$. Consider $I_1^{(i)}$ and $I_m^{(i)}$, $m \geq 1$. $J_1^{(i)}$ is independent of $J_m^{(i)}$, $U_1^{(i)}$ and $U_m^{(i)}$. Then

$$\mathbb{E}[I_1^{(i)} I_m^{(i)}] - \mathbb{E}[I_1^{(i)}]\mathbb{E}[I_m^{(i)}] = \mathbb{E}[U_1^{(i)} U_m^{(i)}] - \mathbb{E}[U_1^{(i)}]\mathbb{E}[U_m^{(i)}]$$

which implies the correlation between $I_1^{(i)}$ and $I_m^{(i)}$ is the same as the correlation between $U_1^{(i)}$ and $U_m^{(i)}$.

Now we consider the correlation between $U_1^{(i)}$ and $U_m^{(i)}$. We first claim that the Markov process $S(k) = \big(N(k), \hat{N}(k)\big)$ is geometrically ergodic [240]. In fact, by Assumption 2, the system is stabilized. Note that we set $\hat{\lambda}(k) = e^{-1}$ in (3.8) and $\lambda(k) < e^{-1}$ for all k. Since $\lambda_m = \limsup_k \lambda(k) < e^{-1} = \hat{\lambda}(k)$. From [240, Theorem 3.1 and Section IV], Markov process $S(k)$ is geometrically ergodic. Define the state space of $S(k)$ as $\mathcal{P}$. For any $i, j \in \mathcal{P}$, define $p_{ij}(k) = [P(k)]_{ij}$ and $\Pi = [\pi_i]_i$ as the transition probability in time slot k and the stationary distribution, respectively. A Markov chain is geometrically ergodic [241] if there are $\rho < 1$ and $C < \infty$ such that for all i, j, k

$$|p_{ij}(k) - \pi_j| \leq C\rho^k. \tag{B.1}$$

From (B.1), in the limit of k, the transition probability equals to the stationary distribution, i.e., $\lim_{k\to\infty} p_{ij}(k) = \pi_j$ for any $i, j \in \mathcal{P}$.

Now, we consider $U_1^{(i)} = n$ and $U_m^{(i)} = l$.

$$\Pr(U_1^{(i)} = n, U_m^{(i)} = l) = \Pr(U_1^{(i)} = n) \Pr(U_m^{(i)} = l | U_1^{(i)} = n) \tag{B.2}$$

Define number of the time slots between $U_1^{(i)}$ and $U_m^{(i)}$ as m', $m' \geq m$. Define the states of $S(k)$ just

before and after $U_1^{(i)}$ as s_1 and s_2. Define the state of $S(k)$ just before $U_m^{(i)}$ as s_m. In the following steps, we use π_{s_i} and $\Pr(s_i)$ interchangeably. Then, due to the Markovity of $S(k)$,

$$\begin{aligned}
&\Pr\left(U_1^{(i)} = n, U_m^{(i)} = l\right) = \Pr(U_1^{(i)} = n) \Pr(U_m^{(i)} = l | U_1^{(i)} = n) \\
&= \sum_{s_1, s_2, s_m \in \mathcal{P}} \Pr(s_1) \Pr(U_1^{(i)} = n | s_1) \Pr(s_2 | U_1^{(i)} = n, s_1) \times \Pr(s_m | s_2) \Pr(U_m^{(i)} = l | s_m).
\end{aligned}$$

From (B.1),

$$p_{ij}(k) = \pi_j + \epsilon_{ij}(k) \tag{B.3}$$

where $|\epsilon_{ij}(k)| \leq C\rho^k$ for all $i, j \in \mathcal{P}$. Note that the number of time slot between $U_1^{(i)}$ and $U_m^{(i)}$ is m'. By the definition of transition probabilities,

$$\Pr(s_m | s_2) = \sum_{s_{m-1}} p_{s_{m-1} s_m} \Pr(s_{m-1} | s_2).$$

Let $\epsilon(m') = \max_{s_{m-1}, s_m \in \mathcal{P}} |\epsilon_{s_{m-1} s_m}(m')|$. Then,

$$\Pr(s_m | s_2) \leq \left(\pi_{s_m} + \epsilon(m')\right) \sum_{s_{m-1}} \Pr(s_{m-1} | s_2) \leq \pi_{s_m} + \epsilon(m').$$

Thus,

$$\Pr(U_m^{(i)} = l | U_1^{(i)} = n) \leq \sum_{s_m \in \mathcal{P}} \left(\pi_{s_m} + \epsilon(m')\right) \Pr(U_m^{(i)} = l | s_m).$$

Consider the stationary distribution Π, define

$$\delta = \min_i \{\pi_i > 0\},$$

δ is a constant depending on the stationary distribution, hence the number of nodes M. Then,

$$\begin{aligned}
&\Pr(U_m^{(i)} = l | U_1^{(i)} = n) \leq \sum_{s_m \in \mathcal{P}} \big(\pi_{s_m} + \epsilon(m')\big) \Pr(U_m^{(i)} = l | s_m) \\
&\leq \Pr(U_m^{(i)} = l) + \epsilon(m') \sum_{s_m \in \mathcal{P}} \frac{\pi_{s_m}}{\delta} \Pr(U_m^{(i)} = l | s_m) \\
&= \Pr(U_m^{(i)} = l) + \frac{\epsilon(m')}{\delta} \sum_{s_m \in \mathcal{P}} \Pr(s_m) \Pr(U_m^{(i)} = l | s_m) \\
&= \Pr(U_m^{(i)} = l) \Big(1 + \frac{\epsilon(m')}{\delta}\Big).
\end{aligned}$$

Therefore,

$$\mathbb{E}[U_1^{(i)} U_m^{(i)}] \leq \big(1 + \frac{\epsilon(m')}{\delta}\big) \sum_n n \Pr(U_1^{(i)} = n) \times \sum_l l \Pr(U_m^{(i)} = l) \leq \big(1 + \frac{\epsilon(m')}{\delta}\big) \mathbb{E}[U_1^{(i)}] \mathbb{E}[U_m^{(i)}]$$

Note that $\epsilon(m') \leq C\rho^{m'}$, so

$$\mathbb{E}[U_1^{(i)} U_m^{(i)}] - \mathbb{E}[U_1^{(i)}] \mathbb{E}[U_m^{(i)}] \leq \mathbb{E}[U_1^{(i)}] \mathbb{E}[U_m^{(i)}] \frac{C}{\delta} \rho^{m'} \leq C' \rho^m.$$

The last equality holds because $m' \geq m$ and $\rho < 1$.

B.2. Proof of Lemma 3

Continuing from (3.13), we have

$$\alpha_\beta = \mathbb{E}\left[\lim_{K \to \infty} \frac{1}{MK} \sum_{k=1}^{K} \sum_{i=1}^{M} \mathbf{1}(\text{node } i \text{ is active at time } k)\right] = \mathbb{E}\left[\frac{1}{M} \sum_{i=1}^{M} p_a^{(i)}\right] \tag{B.4}$$

where $p_a^{(i)}$ is the fraction of time that node i is active in the limit of $K \to \infty$,

$$p_a^{(i)} = \lim_{n \to \infty} \frac{\sum_{\ell=1}^{n} U_\ell^{(i)}}{\sum_{\ell=1}^{n} I_\ell^{(i)}}. \tag{B.5}$$

Furthermore,

$$\mathbb{E}[p_a^{(i)}] = \mathbb{E}[\lim_{n\to\infty} \frac{\sum_{\ell=1}^n U_\ell^{(i)}}{\sum_{\ell=1}^n I_\ell^{(i)}}] = \mathbb{E}[\lim_{n\to\infty} \frac{\sum_{\ell=1}^n U_\ell^{(i)}/n}{\sum_{\ell=1}^n I_\ell^{(i)}/n}]$$
$$\overset{(b)}{=} \frac{\mathbb{E}[\lim_{n\to\infty} \frac{\sum_{\ell=1}^n U_\ell^{(i)}}{n}]}{\mathbb{E}[I_\ell^{(i)}]} \overset{(c)}{=} \frac{1}{\mathbb{E}[I_\beta]} \lim_{n\to\infty} \mathbb{E}[\frac{\sum_{\ell=1}^n U_\ell^{(i)}}{n}] = \frac{\mathbb{E}[U_\ell^{(i)}]}{\mathbb{E}[I_\ell^{(i)}]}.$$

(b) holds due to $\lim_{n\to\infty} \frac{\sum_{\ell=1}^n I_\ell^{(i)}}{n} = \mathbb{E}[I_\ell^{(i)}]$ in Appendix B.1. (c) holds by the dominated convergence theorem because $U_\ell^{(i)}$ is measurable. Therefore,

$$\alpha_\beta = \mathbb{E}\left[\frac{1}{M}\sum_{i=1}^M p_a^{(i)}\right] = \frac{\mathbb{E}[U_\ell^{(i)}]}{\mathbb{E}[I_\ell^{(i)}]}.$$

B.3. Proof of Lemma 4

(1) Note that the channel throughput is $c(M)$. Define n_i as the total delivered number of packets delivered from node i up to and including time slot K. Note that the transmission policy is stationary, so $n_i \to \infty$ implies $K \to \infty$. By Appendix B.1, the Law of Large Number holds for $\{I_\ell^{(i)}\}$, so the throughput is

$$\sum_{i=1}^M \lim_{n_i\to\infty} \frac{n_i}{\sum_{\ell=1}^{n_i} I_\ell^{(i)}} = \frac{M}{\mathbb{E}[I_\beta]} = c(M),$$

which implies

$$\mathbb{E}[I_\beta] = \frac{M}{c(M)}.$$

(2) Using Lemma 3, we then obtain $\mathbb{E}[U_\ell^{(i)}] = \frac{M}{c(M)}\alpha_\beta$, hence $\mathbb{E}[U_\beta] = \frac{M}{c(M)}\alpha_\beta$. From (3.14), noting that $\alpha_\beta \le 1$, we find

$$\alpha_\beta = \frac{1-\sqrt{1-4\lambda_m/M}}{2} = \frac{2\lambda_m/M}{1+\sqrt{1-4\lambda_m/M}} \le \frac{2\lambda_m}{M}.$$

Therefore, we have

$$M\alpha_\beta \leq 2\lambda_m < 2e^{-1}.$$

From Assumption 2, under an optimal β, if M is sufficient large, then $c(M) \approx e^{-1}$, so there exists $\epsilon > 0$, such that $c(M) > e^{-1} - \epsilon$. Therefore, from (3.16), we find

$$\mathbb{E}[U_\beta] = o(M).$$

B.4. Proof of Lemma 5

At the beginning of time slot k, the estimation error is

$$X_i(k) - \hat{X}_i(k) = X_i(k) - X_i(k_{\ell-1}^{(i)}) = \sum_{l=1}^{k-k_{\ell-1}^{(i)}} W_i\big(l + k_{\ell-1}^{(i)}\big).$$

By the stationarity of $\{W_i(k)\}_{k=1}^{\infty}$ and using (3.5), we conclude

$$X_i(k) - \hat{X}_i(k) \sim \sum_{l=1}^{h_i(k)} W_i(l).$$

Now note that $h_i(k)$ is independent of $\{W_i(k)\}_{k=1}^{\infty}$ under oblivious policies. Therefore, using Wald's equality, we find

$$\mathbb{E}[X_i(k) - \hat{X}_i(k)] = 0$$
$$\mathbb{E}[\big(X_i(k) - \hat{X}_i(k)\big)^2] = \mathbb{E}[h_i(k)]\sigma^2.$$

B.5. Proof of Proposition 5

Denote $\underline{T}^{(i)}(k) = \{k_j^{(i)}\}_{j=0}^{\ell}$ with $k_\ell^{(i)} \leq k$. We devise the MW policy using techniques from Lyapunov Optimization. Define the Lyapunov function

$$L(k) = \frac{1}{M}\sum_{i=1}^{M}\big(X_i(k) - \hat{X}_i(k)\big)^2 \tag{B.6}$$

and the one-slot Lyapunov Drift

$$LD(k) = \mathbb{E}[L(k+1) - L(k)|\underline{T}^{(i)}(k)]. \tag{B.7}$$

Recall that the proposed policy is oblivious to the monitored process. So $W_i(j)$'s are independent of $h_i(k)$. Using (3.18), (B.6), and (B.7), we write

$$LD(k) = \mathbb{E}[L(k+1) - L(k)\big|\underline{T}^{(i)}(k)] = \frac{\sigma^2}{M}\sum_{i=1}^{M}\mathbb{E}\big[h_i(k+1) - h_i(k)\big]. \tag{B.8}$$

Moreover, the age functions have the following recursion:

$$h_i(k+1) = d_i(k) + (1 - d_i(k))(h_i(k) + 1). \tag{B.9}$$

where $d_i(k) \in \{0, 1\}$ indicates a successful delivery from source i at time k. Note $\sum_{i=1}^{M} d_i(k) = 1$. Under the MW policy, no collisions occur in every time slot, so $h_i(k)$ is a scalar (not a random variable) for all i, k. Substituting $h_i(k+1)$ from (B.9) into (B.8), we obtain

$$LD(k) = \frac{\sigma^2}{M}\sum_{i=1}^{M}\Big(1 - h_i(k)d_i(k)\Big).$$

Thus, minimizing $LD(k)$ is equivalent to choosing i^* such that $h_{i^*}(k) = \max_i h_i(k)$.

Since we assumed $h_i(0) = 1$ for all nodes, from Lemma 2 in [13, Section III], the above MW policy is equivalent to a Round-Robin policy. Consequently, for all $i = 1, \ldots, M$, and $k \geq i$, we get

$h_i(k) = 1, 2, \cdots, M$ successively and periodically, and

$$\lim_{K\to\infty} \frac{1}{K} \sum_{i=1}^{M} h_i(k) = \frac{M(M+1)}{2}.$$

Therefore,

$$\lim_{M\to\infty} L^{MW}(M) = \lim_{M\to\infty} \frac{\sigma^2}{M^2} \frac{M(M+1)}{2} = \frac{\sigma^2}{2}.$$

B.6. Proof of Lemma 6

From (3.30), (3.31) can be written as

$$L^{EbT}(M) = \lim_{K\to\infty} \mathbb{E}[\frac{1}{M^2 K} \sum_{i=1}^{M} \sum_{\ell=1}^{n_i} \Delta_\ell^{(i)}] \tag{B.10}$$

where n_i is the total number of packets delivered from source i up to and including time slot K. From the proof of Appendix B.1, $\{I_\ell^{(i)}\}$ is measurable. Then, from (3.31), $\Delta_\ell^{(i)}$ is measurable. By the dominated convergence theorem, we can exchange the order of $\lim_{K\to\infty}$ and $\mathbb{E}$ in (B.10).

Note that $\{K \to \infty\}$ is equivalent to $\{n_i \to \infty\}$ for all i. It follows that in the limit of large time horizon K (equivalently, large n_i for all i), we have

$$\begin{aligned} L^{EbT}(M) =& \mathbb{E}[\frac{1}{M^2} \sum_{i=1}^{M} \lim_{n_i\to\infty} \sum_{\ell=1}^{n_i} \frac{\Delta_\ell^{(i)}}{I_\ell^{(i)}}] = \frac{1}{M^2} \sum_{i=1}^{M} \mathbb{E}\left[\lim_{n_i\to\infty} \frac{\sum_{\ell=1}^{n_i} \Delta_\ell^{(i)}/n_i}{\sum_{\ell=1}^{n_i} I_\ell^{(i)}/n_i} \right] \\ =& \frac{1}{M^2} \sum_{i=1}^{M} \frac{1}{\mathbb{E}[I_\ell^{(i)}]} \lim_{n_i\to\infty} \mathbb{E}\left[\frac{\sum_{\ell=1}^{n_i} \Delta_\ell^{(i)}}{n_i} \right] = \frac{1}{M} \frac{\mathbb{E}[\Delta_\ell^{(i)}]}{\mathbb{E}[I_\ell^{(i)}]}. \end{aligned}$$

The last equality holds because $\Delta_\ell^{(i)}$ is identical over ℓ. Recall that Δ_β and I_β have the same distribution as $\Delta_\ell^{(i)}$ and $I_\ell^{(i)}$, respectively. Therefore,

$$L^{EbT}(M) = \frac{1}{M} \frac{\mathbb{E}[\Delta_\beta]}{\mathbb{E}[I_\beta]}.$$

B.7. Proof of (3.36)

For any $J_\beta + 1 \leq j \leq J_\beta + U_\beta - 1$, W_j is independent of S_{J_β}, hence J_β. Therefore,

$$\mathbb{E}[S_j^2] = \mathbb{E}[(S_{J_\beta} + W_{J_\beta+1} + \cdots + W_j)^2] = \mathbb{E}[S_{J_\beta}^2] + \mathbb{E}[j - J_\beta]\sigma^2.$$

Note that given β, J_β and U_β are independent, This helps to further simplify the numerator of $L_2^{EbT}(M)$ in (3.36),

$$\begin{aligned}
\mathbb{E}\big[\sum_{j=J_\beta+1}^{J_\beta+U_\beta-1} S_j^2\big] =& \mathbb{E}_{U_\beta}\Big\{\mathbb{E}\big[\sum_{j=J_\beta+1}^{J_\beta+U_\beta-1} (S_{J_\beta} + W_{J_\beta+1} + \cdots + W_j)^2 | U_\beta\big]\Big\} \\
=& \mathbb{E}_{U_\beta}\Big\{(U_\beta - 1)\mathbb{E}[S_{J_\beta}^2] + \mathbb{E}[\sum_{j=J_\beta+1}^{J_\beta+U_\beta-1} (j - J_\beta)]\sigma^2 | U_\beta\Big\} \\
=& \mathbb{E}_{U_\beta}\Big\{(U_\beta - 1)\mathbb{E}[S_{J_\beta}^2] + \frac{U_\beta(U_\beta-1)}{2}\sigma^2 | U_\beta\Big\} = (\mathbb{E}[U_\beta] - 1)\mathbb{E}[S_{J_\beta}^2] + \mathbb{E}\big[\frac{U_\beta(U_\beta-1)}{2}\big]\sigma^2.
\end{aligned}$$

Substituting (3.29) into the equation above,

$$L_2^{EbT}(M) = \frac{1}{M} \cdot \frac{2\mathbb{E}[J_\beta](\mathbb{E}[U_\beta] - 1) + \mathbb{E}[U_\beta^2] - \mathbb{E}[U_\beta]}{2\mathbb{E}[I_\beta]}\sigma^2.$$

B.8. Proof of Lemma 7

The proof of the first part is the same as that of Theorem 7.5.5 and Theorem 7.5.9 in [194, Chapter 7]. Here, we prove the second part. Using [32, Lemma 4], we have

$$\mathbb{E}[\int_0^J B_t^2 dt] = \frac{1}{6}\mathbb{E}[B_J^4].$$

From the definition of J, $B_J^4 = a^4$, then $\mathbb{E}[B_J^4] = a^4$, hence

$$\mathbb{E}[\int_0^J B_t^2 dt] = \frac{1}{6}a^4.$$

From Theorem 7.5.9 in [194, Chapter 7], $\mathbb{E}[J^2] = \frac{5}{3}a^4$, so $\mathbb{E}[\int_0^J B_t^2 dt] = \frac{1}{10}\mathbb{E}[J^2]$.

B.9. Proof of Theorem 6

We start with the expression of $\hat{L}^{EbT}(M)$ in (3.40). Using (3.10), (3.40) can be re-written as

$$\frac{1}{\sigma^2}\hat{L}^{EbT}(M) = \frac{\frac{1}{5}\mathbb{E}[J_\beta^2] + \mathbb{E}[U_\beta^2]}{2M\mathbb{E}[I_\beta]} = \frac{\frac{1}{5}\mathbb{E}[J_\beta^2] + \mathbb{E}[(I_\beta - J_\beta - 1)^2]}{2M\mathbb{E}[I_\beta]}$$
$$= \frac{\frac{1}{5}\mathbb{E}[J_\beta^2] + \mathbb{E}[I_\beta^2] + \mathbb{E}[J_\beta^2] + 1 - 2\mathbb{E}[I_\beta] + 2\mathbb{E}[J_\beta] - 2\mathbb{E}[I_\beta J_\beta]}{2M\mathbb{E}[I_\beta]}$$

Now replace for I_β in $\mathbb{E}[I_\beta J_\beta]$ using (3.10). Consider M sufficiently large, and note that $J_\beta \leq I_\beta$. We can approximately write the equation above as follows

$$\begin{aligned}
\hat{L}^{EbT}(M) \approx & \frac{\frac{1}{5}\mathbb{E}[J_\beta^2] + \mathbb{E}[I_\beta^2] - 2\mathbb{E}[(U_\beta + 1)J_\beta] - \mathbb{E}[J_\beta^2]}{2M\mathbb{E}[I_\beta]}\sigma^2 \\
\overset{(a)}{=} & \frac{\frac{1}{5}\mathbb{E}[J_\beta^2] + \mathbb{E}[I_\beta^2] - 2\mathbb{E}[U_\beta + 1]\mathbb{E}[J_\beta] - \mathbb{E}[J_\beta^2]}{2M\mathbb{E}[I_\beta]}\sigma^2 \\
\overset{(b)}{\approx} & \frac{\frac{1}{5}\mathbb{E}[J_\beta^2] + \mathbb{E}[I_\beta^2] - \mathbb{E}[J_\beta^2]}{2M\mathbb{E}[I_\beta]}\sigma^2 \qquad \text{(B.11)} \\
\overset{(c)}{=} & \frac{\frac{1}{5}\mathbb{E}[J_\beta^2] + \big(\mathbb{E}[I_\beta]\big)^2 + Var(U_\beta) - \big(\mathbb{E}[J_\beta]\big)^2}{2M\mathbb{E}[I_\beta]}\sigma^2 \qquad \text{(B.12)}
\end{aligned}$$

where (a) holds because U_β and J_β are independent given β, (b) holds because $\mathbb{E}[U_\beta] = o(M)$ (see (3.16)) and $J_\beta < I_\beta$, and (c) holds by (3.10) and the independence of U_β and J_β which leads to $Var(U_\beta) + Var(J_\beta) = Var(I_\beta)$.

Substituting (3.15) and (3.38) into (B.12), we obtain

$$\hat{L}^{EbT}(M) \approx \frac{\frac{M^2}{c(M)^2} - \frac{2\beta^4}{3\sigma^4} + Var(U_\beta)}{2\frac{M^2}{c(M)}}\sigma^2. \qquad \text{(B.13)}$$

Note that J_β, as defined before, is a stopping time of the discretization of the considered Wiener process $B(t)$, and therefore $J_\beta > J$, almost everywhere. We thus conclude that

$$\mathbb{E}[J_\beta] > \mathbb{E}[J]. \qquad \text{(B.14)}$$

Delay per transmission is 1 time slot, so $U_\beta \geq 1$. Using (3.15) and (B.14), we can write

$$0 \leq \mathbb{E}[U_\beta] - 1 = \frac{M}{c(M)} - \mathbb{E}[J_\beta] < \frac{M}{c(M)} - \mathbb{E}[J]. \tag{B.15}$$

Substituting $\mathbb{E}[J] = \frac{\beta^2}{\sigma^2}$ (see (3.38)) into (B.15), we find $\beta \leq \sigma\sqrt{\frac{M}{c(M)}}$. It is now easy to see that $\hat{\beta} = \sigma\sqrt{\frac{M}{c(M)}}$ is the minimum point of $\frac{-\frac{2\beta^4}{3\sigma^4}}{2\frac{M^2}{c(M)}}$. We will next show that the term $\frac{Var(U_{\beta^*})}{2\frac{M^2}{c(M)}}$ (in (B.13)) is negligible and therefore $\beta^* \approx \hat{\beta}$ is approximately optimal. This will lead to $\hat{L}^{EbT} = \frac{1}{6c(M)}\sigma^2$.

Recall from [240, Theorem 3.1 and Section IV] (see also Appendix B.6) that $S(k) = \big(N(k), \hat{N}(k)\big)$ (depending on β^*) is geometrically ergodic. Let

$$S(k) \in \{(e_1, s_1), (e_2, s_2), \cdots, (e_m, s_m)\},$$

and m be finite. Define the state space of $S(k)$ as $\mathcal{P}$, $|\mathcal{P}| = m$. Let $p_{ij}(k)$ be the transition probability matrix in time slot k, $i, j \in \mathcal{P}$. Let π_i be the stationary distribution of $S(k)$. For $i = 1, 2, \cdots, m$, the transmitting probability ν_i is obtained by (3.8), and the corresponding probability of a successful delivery of *each* active node, denoted by r_i, is

$$r_i = \begin{cases} \nu_i(1-\nu_i)^{e_i - 1} & e_i \geq 1 \\ 0 & e_i = 0 \end{cases}$$

for $i = 1, 2, \cdots, m$. Denote by $\Pr(i, k)$ the probability that the system is in state i in time slot k. Let $H(j, i, k) \in \{0, 1\}$ be an indicator. $H(j, i, k) = 1$ represents that node j becomes newly active in time slot k when the system is in state i. Consider any *node* j. We have

$$\Pr(U_{\beta^*} = 1) = \lim_{k \to \infty} \sum_{i=1}^{m} \Pr(i, k)\mathbb{E}[H(j, i, k)]r_i$$

Note that the system is stationary, and $\lim_{k\to\infty} \mathbb{E}[H(j, i, k)]$ exists for all j, i. Since all nodes are

identical, when the system is stationary, we have

$$\lim_{k\to\infty} \mathbb{E}[H(1,i,k)] = \cdots = \lim_{k\to\infty} \mathbb{E}[H(M,i,k)] \triangleq \tilde{h}_i.$$

In addition, when the system is stationary, $\lim_{k\to\infty} \Pr(i,k) = \pi_i$. Denote the dominant term of $\Pr(U_{\beta^*} = 1)$ as Δ_1, thus $\Delta_1 = \sum_{i=1}^m \pi_i \tilde{h}_i r_i$. Similarly,

$$\Pr(U_{\beta^*} = 2) = \lim_{k\to\infty} \sum_{i=1}^m \Pr(i,k)\mathbb{E}[H(j,i,k)](1-r_i) \times \sum_{l=1}^m p_{il}(k+1)r_l.$$

Since $S(k)$ is geometrically ergodic, we have $|p_{ij}(k) - \pi_j| \leq C\rho^k$, where $C < \infty$, $0 < \rho < 1$. Thus, the dominant term of $\Pr(U_{\beta^*} = 2)$ when $k \to \infty$, denoted by Δ_2, is

$$\Delta_2 = \sum_{i=1}^m \pi_i \tilde{h}_i (1-r_i) \sum_{j=1}^m \pi_j r_j = (y - \Delta_1)\mu$$

where $y = \sum_{i=1}^m \pi_i \tilde{h}_i$ and $\mu = \sum_{j=1}^m \pi_j r_j$. By a similar process, denote the dominant term of $\Pr(U_{\beta^*} = l)$ as Δ_l:

$$\Delta_l = (\mu - \Delta_1)(1-\mu)^{l-2}\mu, \quad l \geq 3.$$

So the dominant term of $\mathbb{E}[U_{\beta^*}]$, denoted by Λ_1, is

$$\Lambda_1 = \Delta_1 + (y - \Delta_1)\mu \sum_{l=2}^{\infty} (1-\mu)^{l-2} l = \Delta_1 + 2(y - \Delta_1) + \frac{1-\mu}{\mu}(y - \Delta_1) = y + (y - \Delta_1)\frac{1}{\mu}.$$

By Lemma 4, $\mathbb{E}[U_{\beta^*}] = o(M)$, so $\Lambda_1 = o(M)$. Note that $\frac{1}{\mu} = \frac{\Lambda_1 - y}{y - \Delta_1}$, then $\frac{1}{\mu} = o(M)$ since y and Δ_1 are scalars.

The dominant term of $\mathbb{E}[U_{\beta^*}^2]$, denoted by Λ_2, is similarly

$$\begin{aligned}\Lambda_2 =&\Delta_1 + (y-\Delta_1)\mu\sum_{l=2}^{\infty}(1-\mu)^{l-2}l^2 = \Delta_1 + (y-\Delta_1)\mu\sum_{l=2}^{\infty}(1-\mu)^{l-2}\big((l-2)^2+4(l-2)-4\big)\\ =&\Delta_1 + (y-\Delta_1)\big(\frac{(1-\mu)(2-\mu)}{\mu^2}+4\frac{1-\mu}{\mu}-1\big).\end{aligned}$$

Note that $\frac{1}{\mu} = o(M)$, thus $\Lambda_2 = o(M^2)$, $\mathbb{E}[U_{\beta^*}^2] = o(M^2)$, and $Var(U_{\beta^*}) = o(M^2)$ which implies $\frac{Var(U_{\beta^*})}{M^2} \approx 0$.

So $\beta^* \approx \hat{\beta} = \sigma\sqrt{\frac{M}{c(M)}}$ is approximately optimal and $\hat{L}^{EbT} \approx \frac{1}{6c(M)}\sigma^2$. From Assumption 2, when M is sufficiently large, $c(M) \approx e^{-1}$, then $\hat{L}^{EbT} \approx \frac{e}{6}\sigma^2$, and the corresponding $\beta^* \approx \hat{\beta} = \sigma\sqrt{eM}$.

B.10. Assumptions 1, 2 are (approximately) satisfied

We first verify Assumption 2. Recall that $N(k)$ is the number of active nodes at the end of time slot k. Denote $D(k)$ as the number of nodes that become inactive (from the active state) in time slot k. Then, $\lim_{k\to\infty}\mathbb{E}[D(k)] = c(M)$. Thus,

$$N(k+1) = \min\{N(k) + \lambda(k+1), M\} - D(k+1). \tag{B.16}$$

From (3.12) and Lemma 3, when $k \to \infty$, the limit $\lim_{k\to\infty}\mathbb{E}[N(k)]$ exists. Therefore, letting $k \to \infty$, (B.16) is reduced to

$$c(M) = \lambda_m. \tag{B.17}$$

Since $c(M) < e^{-1}$, then $\lambda_m < 1/e$, which implies the system is stabilized.

Then, we investigate $\mathbb{E}[U_{\beta^*}]$. From Lemma 3 and Lemma 4, when M is sufficiently large,

$$\alpha_\beta = \frac{o(M)}{M} \approx 0, \tag{B.18}$$

then from (3.14) and (B.18),

$$M\alpha_\beta \approx \lambda_m. \tag{B.19}$$

From (B.17), (B.19) and Lemma 4,

$$\mathbb{E}[U_{\beta^*}] \approx \frac{\lambda_m}{c(M)} = 1. \tag{B.20}$$

Note that I_β is the inter-delivery time, and from (3.7), our goal (3.2) can be re-written as

$$\frac{1}{M}\mathbb{E}\big[\sum_{k=0}^{I_{\beta^*}} \psi_i^2(k)\big],$$

which increases with $\mathbb{E}[I_{\beta^*}]$. From (3.15), $\mathbb{E}[I_{\beta^*}]$ decreases with $c(M)$. To obtain the minimum $\frac{1}{M}\mathbb{E}\big[\sum_{k=0}^{I_{\beta^*}} \psi_i^2(k)\big]$ or (3.2), we need the minimum $\mathbb{E}[I_{\beta^*}]$, thus we let $c(M) \approx e^{-1}$. From (B.17), $\lambda_m \approx e^{-1}$. Thus, Assumption 2 is (approximately) satisfied.

Next, we verify Assumption 1. Denote $q(n, L)$ as the probability that n of L inactive nodes become newly active in one slot. Since the system is stationary, $q(n, L)$ will not change over time and only depends on the error process profile. Recall that $a(k)$ is the number of newly active nodes in time k. We need to show $\{a(k) = l_1\}$ and $\{a(k+1) = l_2\}$ are independent where l_1, l_2 are non-negative integers. In fact,

$$\Pr\{a(k) = l_1, a(k+1) = l_2\} = \Pr\{a(k) = l_1\} \Pr\{a(k) = l_2 | a(k) = l_1\}$$

Denote $d_i(k) = 1$ as an indicator such that node i delivers a packet successfully in time k, otherwise

$d_i(k) = 0$. Then, $\sum_{i=1}^{M} d_i(k) = 1$ represents that a packet is delivered in time slot k. Thus,

$$\begin{aligned}\Pr\{a(k) = l_2 | a(k) = l_1\} = &\Pr\{a(k) = l_2 | a(k) = l_1, \sum_{i=1}^{M} d_i(k) = 0\} \Pr\{\sum_{i=1}^{M} d_i(k) = 0\} \\ &+ \Pr\{a(k) = l_2 | a(k) = l_1, \sum_{i=1}^{M} d_i(k) = 1\} \Pr\{\sum_{i=1}^{M} d_i(k) = 1\}\end{aligned}$$

Note that $c(M) \approx e^{-1}$, then $\Pr\{\sum_{i=1}^{M} d_i(k) = 1\} \approx e^{-1}$, thus

$$\Pr\{a(k) = l_2 | a(k) = l_1\} \approx q(l_2, M - l_1 - 1)1/e + q(l_2, M - l_1)(1 - 1/e).$$

Recall that $\mathbb{E}[a(k)] = \lambda_m$ when $k \to \infty$, i.e., the system is stationary. Note that $a(k)$ is non-negative, so by Markov's Inequality, we have

$$\Pr\{a(k) \geq O(M)\} \leq \frac{\mathbb{E}[a(k)]}{O(M)}.$$

Let $k, M \to \infty$, we have $\Pr\{a(k) \geq O(M)\} \to 0$, which implies $a(k) = o(M)$ with probability 1 when M is sufficiently large. $q(l_2, M - l_1 - 1) \approx q(l_2, M - l_1) \approx q(l_2, M) \approx \Pr(a(k + 1) = l_2)$, thus

$$\Pr\{a(k) = l_1, a(k + 1) = l_2\} \approx \Pr\{a(k) = l_1\} \Pr\{a(k) = l_2\}$$

when M is sufficiently large. Assumption 1 is thus (approximately) satisfied.

All the proofs in this Appendix are valid for the cases when $\gamma \neq 1$.

APPENDIX C

Proofs of Chapter 4

C.1. Proof of Theorem 10

First, we utilize the term $T_{B_1,W_{\Xi_n}}\rho(T_{B_2,W}X)$ to modify $\|Y_n - Y\|$, by triangle inequality, we have

$$\begin{aligned}
&\|Y_n - Y\| \\
=&\|T_{B_1}(W_{\Xi_n})\rho\big(T_{B_2,W_{\Xi_n}}X_n\big) - T_{B_1,W}\rho\big(T_{B_2,W}X\big)\| \\
\leq&\|T_{B_1,W_{\Xi_n}}\rho\big(T_{B_2,W_{\Xi_n}}X_n\big) - T_{B_1,W_{\Xi_n}}\rho\big(T_{B_2,W}X\big)\| \\
+&\|T_{B_1,W_{\Xi_n}}\rho\big(T_{B_2,W}X\big) - T_{B_1,W}\rho\big(T_{B_2,W}X\big)\|.
\end{aligned}$$

For the first term, note that $T_{B_1,W_{\Xi_n}}$ satisfy Assumption 3, the norm of the operator $T_{B_1,W_{\Xi_n}}$ is bounded by 1, hence

$$\begin{aligned}
&\|T_{B_1,W_{\Xi_n}}\rho\big(T_{B_2,W_{\Xi_n}}X_n\big) - T_{B_1,W_{\Xi_n}}\rho\big(T_{B_2,W}X\big)\| \\
=&\|T_{B_1,W_{\Xi_n}}\big(\rho(T_{B_2,W_{\Xi_n}}X_n) - \rho(T_{B_2,W}X)\big)\| \\
\leq&\|\rho(T_{B_2,W_{\Xi_n}}X_n) - \rho(T_{B_2,W}X)\| \leq \|T_{B_2,W_{\Xi_n}}X_n - T_{B_2,W}X\|.
\end{aligned}$$

The last inequality holds due to Assumption 4. By Lemma 10,

$$\|T_{B_2,W_{\Xi_n}}X_n - T_{B_2,W}X\| \leq \Theta(\Omega_2,\omega_2)\|X\| + (\Omega_2\epsilon + 2)\|X - X_n\|. \tag{C.1}$$

Note that the activation function ρ is pointwise non-linear. Then, for the second term, again, by Lemma 10,

$$\|T_{B_1,W_{\Xi_n}}\rho\big(T_{B_2,W}X\big) - T_{B_1,W}\rho\big(T_{B_2,W}X\big)\| \leq \Theta(\Omega_1,\omega_1)\|\rho\big(T_{B_2,W}X\big)\|.$$

Note that $T_{B_2,W}$ satisfies Assumption 3 and ρ satisfies Assumption 4, so $\|\rho\big(T_{B_2,W}X\big)\| \leq \|T_{B_2,W}X\| \leq$

$\|X\|$. Thus,

$$\|T_{B_1,W_{\Xi_n}}\rho\big(T_{B_2,W}X\big) - T_{B_1,W}\rho\big(T_{B_2,W}X\big)\| \leq \Theta(\Omega_1,\omega_1)\|X\| \tag{C.2}$$

From (C.1) and (C.2), we obtain the desired results.

C.2. Proof of Theorem 11

First, we utilize the term $T_{\mathcal{B}_1,W_{\Xi_n}}\rho\big(T_{\mathcal{B}_2,W}X\big)$ to modify $\|Y_n - Y\|$, by triangle inequality, we have

$$\begin{aligned}\|Y_n - Y\| =&\|T_{\mathcal{B}_1,W_{\Xi_n}}\rho\big(T_{\mathcal{B}_2,W_{\Xi_n}}X_n\big) - T_{\mathcal{B}_1,W}\rho\big(T_{\mathcal{B}_2,W}X\big)\|\\ \leq&\|T_{\mathcal{B}_1,W_{\Xi_n}}\rho\big(T_{\mathcal{B}_2,W_{\Xi_n}}X_n\big) - T_{\mathcal{B}_1,W_{\Xi_n}}\rho\big(T_{\mathcal{B}_2,W}X\big)\|\\ &+\|T_{\mathcal{B}_1,W_{\Xi_n}}\rho\big(T_{\mathcal{B}_2,W}X\big) - T_{\mathcal{B}_1,W}\rho\big(T_{\mathcal{B}_2,W}X\big)\|.\end{aligned}$$

For the first term, note that the convolutional filters that make up its layers all satisfy Assumption 3, hence

$$\begin{aligned}&\|T_{\mathcal{B}_1,W_{\Xi_n}}\rho\big(T_{\mathcal{B}_2,W_{\Xi_n}}X_n\big) - T_{\mathcal{B}_1,W_{\Xi_n}}\rho\big(T_{\mathcal{B}_2,W}X\big)\|\\ =&\|T_{\mathcal{B}_1,W_{\Xi_n}}\Big(\rho\big(T_{\mathcal{B}_2,W_{\Xi_n}}X_n\big) - \rho\big(T_{\mathcal{B}_2,W}X\big)\Big)\|\\ \leq&\|\rho\big(T_{\mathcal{B}_2,W_{\Xi_n}}X_n\big) - \rho\big(T_{\mathcal{B}_2,W}X\big)\|\\ \leq&\|T_{\mathcal{B}_2,W_{\Xi_n}}X_n - T_{\mathcal{B}_2,W}X\|.\end{aligned}$$

The last inequality holds because of Assumption 4. By Lemma 11,

$$\|T_{\mathcal{B}_2,W_{\Xi_n}}X_n - T_{\mathcal{B}_2,W}X\| \leq LF^{L-1}\Theta(\Omega_2,\omega_2)\|X\| + (\Omega_2\epsilon + 2)\|X - X_n\|. \tag{C.3}$$

For the second term, again, by Lemma 11,

$$\|T_{\mathcal{B}_1,W_{\Xi_n}}\rho\big(T_{\mathcal{B}_2,W}X\big) - T_{\mathcal{B}_1,W}\rho\big(T_{\mathcal{B}_2,W}X\big)\| \leq LF^{L-1}\Theta(\Omega_1,\omega_1)\|\rho\big(T_{\mathcal{B}_2}(W)X\big)\|.$$

Note that $T_{\mathcal{B}_2,W}$ satisfies Assumption 3 and ρ satisfies Assumption 4, so $\|\rho\big(T_{\mathcal{B}_2,W}X\big)\| \leq \|T_{\mathcal{B}_2,W}X\| \leq$

$\|X\|$. Thus,

$$\|T_{\mathcal{B}_1,W_{\Xi_n}}\rho\big(T_{\mathcal{B}_2,W}X\big) - T_{\mathcal{B}_1,W}\rho\big(T_{\mathcal{B}_2,W}X\big)\| \leq LF^{L-1}\Theta(\Omega_1,\omega_1)\|X\|. \tag{C.4}$$

From (C.3) and (C.4), we obtain the desired results.

C.3. Proof of Theorem 13

For any given Δ, from the construction of W_Δ, we know that T_{W_Δ} is continous, then T_{W_Δ} is bounded [242], i.e., for any $x \in L^2([0,1])$, we have $\|T_{W_\Delta}x\| \leq \|T_{W_\Delta}\|\|x\|$. Then, by Cauchy–Schwarz inequality,

$$\begin{aligned}\|\mathcal{X}(Y_1,Y_2) - \mathcal{X}_n(Y_{n,1},Y_{n,2})\| &= \|\langle Y_1 - Y_{n,1}, T_{W_\Delta}(Y_2 - Y_{n,2})\rangle\| \\ \leq &\|Y_1 - Y_{n,1}\| \cdot \|T_{W_\Delta}(Y_2 - Y_{n,2})\| \leq \|T_{W_\Delta}\|\|Y_1 - Y_{n,1}\|\|Y_2 - Y_{n,2}\|.\end{aligned}$$

Note that $\|Y_1 - Y_{n,1}\|$ and $\|Y_2 - Y_{n,2}\|$ have been bounded by (4.46). Substituting (4.46) into the above equation, we get the desired results.

C.4. Proof of Theorem 14

Note that $\|A - A_n\| = \|\tilde{F}_{\text{softmax}}\mathcal{X} - \tilde{F}_{\text{softmax}}\mathcal{X}_n\|$. The softmax function F_{softmax} is Lipschit [243], and $\tilde{F}_{\text{softmax}}$ is the continous version of F_{softmax}, $\tilde{F}_{\text{softmax}}$ is in Besov class and is Lipschit [244]. Then, there exists a constant Γ, which is independent of the WRNN $\Psi(\mathcal{B}_1,\mathcal{B}_2,\mathcal{B}_3,W_{\Xi_n},\cdot,T)$, such that

$$\|A - A_n\| = \Gamma\|\mathcal{X} - \mathcal{X}_n\|. \tag{C.5}$$

The rest task is to find the upper bound of $\|\mathcal{X} - \mathcal{X}_n\|$.

Since $X_1, X_2 \in L^2([0,1])$, then $\|X_1\|, \|X_2\| \leq N$, where N is a constant. When $n \to \infty$, we have $X_{n,1} \to X_1$, $X_{n,2} \to X_2$, and $W_{\Xi_n} \to W$. For any small $\eta > 0$, we can choose n and ϵ, such that $C_N^2\|X_1\|\|X_2\| < \frac{\eta}{3\|T_{W_\Delta}\|}$, $(\Omega_3+2)^2\|X_1 - X_{n,1}\|\|X_2 - X_{n,2}\| < \frac{\eta}{3\|T_{W_\Delta}\|}$, and $C_N(\Omega_3+2)(\|X_1\|\|X_1 -$

$X_{n,1}\| + \|X_2\|\|X_2 - X_{n,2}\|) < \frac{\eta}{3\|T_{W_\Delta}\|}$, where C_N, Ω_3 are given in Theorem 13, which implies

$$\|\mathcal{X}(Y_1, Y_2) - \mathcal{X}_n(Y_{n,1}, Y_{n,2})\| \leq \eta$$

for any Y_1, Y_2, and $\mathcal{G}_n$. Since given Y_1, Y_2, and $\mathcal{G}_n$, $Y_{n,1}$ and $Y_{n,2}$ are given, then

$$\begin{aligned}
&\|\mathcal{X} - \mathcal{X}_n\| \\
&\leq \int_0^1 \int_0^1 \|\mathcal{X}(Y_1, Y_2) - \mathcal{X}_n(Y_{n,1}, Y_{n,2})\| dY_1 dY_2 \\
&\leq \eta,
\end{aligned}$$

which gives the desired results.

C.5. Transferability on Stochastic Graphs

First of all, we provide the following useful assumptions.

Assumption 6. *([213, AS2]) The graphon W is Ω_w-Lipschitz, i.e., $|W(u_2, v_2) - W(u_1, v_1)| \leq \Omega_w(|u_2 - u_1| + |v_2 - v_1|)$.*

Assumption 7. *([213, AS3]) The graphon signal X is Ω_x-Lipschitz.*

Assumption 8. *([213, AS4]) Given $\chi_3 \in (0, 1)$, n is such that*

$$n - \frac{\log(2n/\chi_3)}{d_W} > 2\frac{A_w}{d_W}$$

where $d_W = \max_u \int_0^1 W(u, v)dv$.

Define

$$\Theta_2(\Omega, \omega) = g_1(\Omega, \chi_1, \chi_3, n) + g_3(\omega, \epsilon) \tag{C.6}$$

$$\Theta_3(\Omega, \omega) = g_1'(\Omega, \chi_1, \chi_3, n_1, n_2) + 2g_3(\omega, \epsilon) \tag{C.7}$$

where $g_1(\cdot)$ decreases with n at rate $O(\frac{\log n}{n})$ with the limit 0, $g_3(\cdot)$ linealy increases with ω and ϵ, and $g_1'(\cdot)$ decreases with n at rate $O(\max_{i\in\{1,2\}} \frac{\log n_i}{n_i})$ with the limit 0.

Lemma 18. *([213, Proposition 5]) Let $T_{\mathcal{B}_1,W}$ be a WNN satisfies Assumptions 4, 6, and 7, and that the convolutional filters that make up its layers all satisfy Assumption 3. Given a stochastic graph $\mathcal{G}$ with GSO Ξ_n, let $\chi_1, \chi_2, \chi_3 \in (0, 0.3]$, and for any $0 < \epsilon \leq 1$ and $n \geq 4/\chi_2$ satisfying Assumption 8, with probability at least $[1 - 2\chi_1] \times [1 - \chi_2] \times [1 - \chi_3]$ it holds that*

$$\|Y - Y_n\| \leq \Theta_2(\Omega_1, \omega_1)\|X\| + g_2(\omega_1, \chi_1, n) \tag{C.8}$$

where $g_2(\cdot)$ decreases with n at rate $O(\frac{\log n}{n})$ with the limit 0, and Ω_1 and ω_1 are Lipschitz constants in Assumption 3.

Lemma 19. *([213, Proposition 6]) Let $T_{\mathcal{B}_1,W}$ be the WNN in Lemma 18. Given a stochastic graph $\mathcal{G}_{n_1}$ and $\mathcal{G}_{n_2}$ with GSOs Ξ_{n_1} and Ξ_{n_2}, respectively, let $\chi_1, \chi_2, \chi_3 \in (0, 0.3]$, and for any $0 < \epsilon \leq 1$ and $n \geq 4/\chi_2$ satisfying Assumption 8, with probability at least $[1 - 2\chi_1]^2 \times [1 - \chi_2]^2 \times [1 - \chi_3]^2$ it holds that*

$$\|Y_n - Y\| \leq \Theta_3(\Omega_1, \omega_1)\|X\| + g_2'(\omega_1, \chi_1, n_1, n_2) \tag{C.9}$$

where $g_2'(\cdot)$ decreases with n at the rate $O(\max_{i\in\{1,2\}} \frac{\log n_i}{n_i})$ with the limit 0, and Ω_1 and ω_1 are Lipschitz constants in Assumption 3.

Note that Lemma 19 follows directly from Lemma 18 and the trangle inequality. Straighforwardly, results in Lemma 19 can be extended to GRNN.

Theorem 24. *Let $T_{\mathcal{B}_1,W}$, $T_{\mathcal{B}_2,W}$, and $T_{\mathcal{B}_3,W}$ be WRNNs satisfy Assumptions 4, 6, and 7, and that the convolutional filters that make up their layers all satisfy Assumption 3. Given a stochastic graph $\mathcal{G}_{n_1}$ and $\mathcal{G}_{n_2}$ with GSOs Ξ_{n_1} and Ξ_{n_2}, respectively, let $\chi_1, \chi_2, \chi_3 \in (0, 0.3]$, and for any $0 < \epsilon \leq 1$ and $n_1, n_2 \geq 4/\chi_2$ satisfying Assumption 8, with probability at least $[1-2\chi_1]^2 \times [1-\chi_2]^2 \times [1-\chi_3]^2$ it holds that*

$$\|Y - Y_n\| \leq \sum_{i=1}^{3} N_i \Theta_3(\Omega_i, \omega_i)\|X\| + g_2'(\omega_3, \chi_1, n_1, n_2), \tag{C.10}$$

where $g_2'(\cdot)$ is given in Lemma 19, $\{\Omega_i, \omega_i\}_{i=1}^3$ are Lipschitz constants in Assumption 3, and N_1, N_2, and N_3 are in Theorem 12.

Proof. From the definition of WRNN in (4.36) and (4.37), to prove Theorem 24, we only need to apply Lemma 19 repeatedly. And the number of repetitions only depends on T, i.e., the number of recurrences, which are N_1, N_2, and N_3 given in Theorem 12. We have the desired results. □

Finally, we show that the transferability holds in action distributions. For $i \in \{1, 2\}$, a generic graph $\mathcal{G}_{n_i}$ with n_i nodes and corresponding node labels $\{u_i\}_{i=1}^{n_i}$ has the GSO Ξ_{n_i} sampled from W. Let x_{n_i} be the graph signal induced from X based on $\{u_i\}_{i=1}^{n_i}$, and let y_{n_i} be the outputs of (4.23), which is a GRNN instantiated from (4.39) on this graph. Then, we obtain a $W_{\Xi_{n_i}}$ and a graphon signal X_{n_i} induced by Ξ_{n_i} and x_{n_i}. Denote Y_{n_i} as the outputs of (4.39). Let $\mathcal{X}_{n_i}$ be defined in Definition 15. Denote the corresonding action distributions as $A_{n_i} \sim \tilde{F}_{\text{softmax}}\mathcal{X}_{n_i}$.

Theorem 25. *Let $T_{\mathcal{B}_1,W}$, $T_{\mathcal{B}_2,W}$, and $T_{\mathcal{B}_3,W}$ be in Theorem 24. Given a stochastic graph $\mathcal{G}_{n_1}$ and $\mathcal{G}_{n_2}$ with GSOs Ξ_{n_1} and Ξ_{n_2}, respectively, let $\chi_1, \chi_2, \chi_3 \in (0, 0.3]$, and for any $0 < \epsilon \leq 1$ and $n_1, n_2 \geq 4/\chi_2$ satisfying Assumption 8, with probability at least $[1-2\chi_1]^2 \times [1-\chi_2]^2 \times [1-\chi_3]^2$ it holds that*

$$\begin{aligned}
&\|\mathcal{X}_{n_1}(Y_{n_1,1}, Y_{n_1,2}) - \mathcal{X}_{n_2}(Y_{n_2,1}, Y_{n_2,2})\| \\
&\leq \|T(W_F)\|\Big(D_N^2\|X_1\| \cdot \|X_2\| + \big(g_2'(\omega_3, \chi_1, n_1, n_2)\big)^2 \\
&+ D_N g_2'(\omega_3, \chi_1, n_1, n_2)(\|X_1\| + \|X_2\|)\Big),
\end{aligned} \tag{C.11}$$

where $D_N = \sum_{i=1}^3 N_i \Theta_3(\Omega_i, \omega_i)$ and N_1, N_2, and N_3 are given in Theorem 12, $g_2'(\cdot)$ is given in Lemma 19, and $\{\Omega_i, \omega_i\}_{i=1}^3$ are Lipschitz constants in Assumption 3.

Remark 29. *Similar to Remark 22, for $\epsilon_0 > 0$, we can choose large n_1, n_2, and small ϵ, such that $g_1'(\cdot) < \epsilon_0$, $g_2'(\cdot) < \epsilon_0$, and $g_3'(\cdot) < \epsilon_0$. This implies $\|\mathcal{X}_{n_1}(Y_{n_1,1}, Y_{n_1,2}) - \mathcal{X}_{n_2}(Y_{n_2,1}, Y_{n_2,2})\|$ is bounded by a small scalar for any $Y_{n_1,1}$, $Y_{n_1,2}$, $Y_{n_2,1}$ and $Y_{n_2,2}$.*

Proof. The proof is very similar to that of Theorem 13, so we omit the proof here. □

Theorem 26. *Let $T_{\mathcal{B}_1,W}$, $T_{\mathcal{B}_2,W}$, and $T_{\mathcal{B}_3,W}$ be in Theorem 24. Given a stochastic graph $\mathcal{G}_{n_1}$ and $\mathcal{G}_{n_2}$ with GSOs Ξ_{n_1} and Ξ_{n_2}, respectively, let $\chi_1, \chi_2, \chi_3 \in (0, 0.3]$, and for any small $\eta > 0$, there exists a $0 < \epsilon \leq 1$ and $n_1, n_2 \geq 4/\chi_2$ satisfying Assumption 8, with probability at least $[1 - 2\chi_1]^2 \times [1 - \chi_2]^2 \times [1 - \chi_3]^2$ it holds that*

$$\|A_{n_1} - A_{n_2}\| \leq \Gamma \cdot \eta, \tag{C.12}$$

where Γ is given in Theorem14.

Proof. The proof is very similar to that of Theorem 14, so we omit the proof here. □

APPENDIX D

Proofs in Chapter 5

D.1. Time-sharing Policies

By very similar ideas and proofs in [13, 14, 20, 222], we devise Max-Weight scheduling policies (without coding) using techniques from Lyapunov Optimization.

Definition 27. *Let $f_i(k)$ be defined in* (5.19)*. In each slot k, the MW policy chooses the action $Q_{i,\varnothing}$ that has the maximum weight as shown in Table D.1:*

$A(k)$	Weights
$Q_{i,\varnothing}$	$\beta_i\delta_{i,\varnothing}(k)(1-\epsilon_i)+\lambda(1-\epsilon_i)f_i(k)$

Table D.1: Actions and their weights.

D.2. Proof of Lemma 12

For any $\mathcal{S}\subset[M]\backslash i$, we first consider the case where $\mathcal{S}=\varnothing$ (Step 1), then consider the case where $\mathcal{S}\neq\varnothing$ (Step 2).

Step 1. Consider $\mathcal{S}=\varnothing$. If p_j is delivered to user i in time slot k, then from Definition 21, the AoI of user i is $\min\{k-k_j, h_i(k-1)+1\}$. Note that $k-k_1>k-k_2$, hence

$$\min\{k-k_1, h_i(k-1)+1\}\geq\min\{k-k_2, h_i(k-1)+1\},$$

which implies p_1 provides a larger AoI reduction (for user i) than that of p_2. Then, transmitting p_2 can not be worse than transmitting p_1 in terms of AoI.

Step 2. Consider $\mathcal{S}\neq\varnothing$. Without loss of generality, suppose that suppose that $p_1,p_2\in Q_{i,\mathcal{S}}$, and $i,j_1,j_2,\cdots,j_l$ form a maximal clique. Denote the corresponding coded packet as x_τ, where x_τ contains uncoded packet p_τ, $\tau\in\{1,2\}$. Note that x_1 and x_2 are fully decodable for users $i,j_1,j_2,\cdots,j_l$, and provide the same AoI reduction to users $j_1,j_2,\cdots,j_l$, respectively. From Definition 21, the AoI of user i is $\min\{k-k_\tau, h_i(k-1)+1\}$ when recovering x_τ. Note that $k-k_1>k-k_2$,

so $\min\{k-k_1, h_i(k-1)+1\} \geq \min\{k-k_2, h_i(k-1)+1\}$, and x_2 provides smaller AoI for user i. As a result, encoding p_2 can not be worse than encoding p_1 in terms of AoI. In this case, transmitting p_2 is not worse than transmitting p_1 in the current time slot.

Now, consider the case where p_1 and p_2 have been encoded in two different coded packets, and p_1 *has been encoded first*. Suppose that users $\{i, l_1, l_2, \cdots, l_m\}$ and $\{i, n_1, n_2, \cdots, n_t\}$ forms two maximal cliques. Denote the coded packet containing p_1 (respectively, p_2) as x' (respectively, x''), and the age reduction for user i as Δ_1 (respectively, Δ_2). The total age reduction for user i once decoding p_1 and p_2 is $\Delta_1 + \Delta_2$. Note that from the condition defined in (5.3), p_2 has been cached by users $n_1, n_2, \cdots, n_t$. Now, we consider another encoding action: the encoder first encodes p_2, i.e., replacing p_1 by p_2 in x', denoted by x'_{p_2}. So, p_1 is useless for user i after decoding p_2, and the encoder can get another coded packet $\tilde{x}$, where only $l_1, l_2, \cdots, l_m$ forms a maximal clique. In this case, The age reduction for users $n_1, n_2, \cdots, n_t$ and $l_1, l_2, \cdots, l_m$ when transmitting x'_{p_2} and $\tilde{x}$ are the same as those when transmitting x' and x'', which implies encoding p_2 is not worse than encoding p_1 in this case.

Thus, from **Step 1** and **Step 2**, encoding p_2 can not be worse than encoding p_1.

D.3. Proof of Theorem 16

Recall that $t^{\pi}_{i,\mathcal{S}_i}(k) \in \{0,1\}$ and more specifically, if $A(k) = Q_{i,\mathcal{S}_i} \oplus_{j\in[l]} Q_{\tau_j,\mathcal{S}_{\tau_j}}$, then $t^{\pi}_{i,\mathcal{S}_i}(k) = 1$ and $t^{\pi}_{i,\tilde{\mathcal{S}}_i}(k) = 0$ where $\tilde{\mathcal{S}}_i \neq \mathcal{S}_i$. So we have

$$\sum_{\mathcal{S}_i \subset [M]\setminus i} t^{\pi}_{i,\mathcal{S}_i}(k) \leq 1. \tag{D.1}$$

Let $\Theta(k) = \Theta_1(k) + \lambda\Theta_2(k)$, where

$$\begin{aligned}
\Theta_1(k) =& \mathbb{E}\Big[\sum_{i=1}^{M} \beta_i h_i(k+1) - \sum_{i=1}^{M} \beta_i h_i(k) | \vec{s}(k)\Big] \\
\Theta_2(k) =& \mathbb{E}\Big[\sum_{i=1}^{M} \big(y_i^+(k+1)\big)^2 - \sum_{i=1}^{M} \big(y_i^+(k)\big)^2 | \vec{s}(k)\Big].
\end{aligned}$$

Recall that the channel erasure probability is ϵ_i, then

$$\mathbb{E}[d_i^{\pi}(k)] = (1-\epsilon_i) \sum_{\mathcal{S} \subset [M] \setminus i} \mathbb{E}[t_{i,\mathcal{S}}^{\pi}(k)]. \tag{D.2}$$

We first consider $\Theta_1(k)$. Using (D.1),

$$h_i(k+1) = d_i(k) \sum_{\mathcal{S} \subset [M] \setminus i} t_{i,\mathcal{S}}(k)\big(w_{i,\mathcal{S}}(k)+1\big) + \big(1 - d_i(k) \sum_{\mathcal{S} \subset [M] \setminus i} t_{i,\mathcal{S}}(k)\big)\big(h_i(k)+1\big). \tag{D.3}$$

From (5.12) and (D.3), we can re-write $\Theta_1(k)$ as follows:

$$\begin{aligned}
&\sum_{i=1}^{M} \beta_i \mathbb{E}\left[d_i(k) \sum_{\mathcal{S}_i \subset [M] \setminus i} t_{i,\mathcal{S}_i}(k)\big(w_{i,\mathcal{S}_i}(k) - h_i(k)\big) + 1 \Big| \vec{s}(k) \right] \\
&= -\sum_{i=1}^{M} \beta_i \mathbb{E}\Big[d_i(k) \sum_{\mathcal{S}_i \subset [M] \setminus i} t_{i,\mathcal{S}_i}(k) \delta_{i,\mathcal{S}_i}(k) \Big| \vec{s}(k) \Big] + \sum \beta_i \\
&= -\sum_{i=1}^{M} \beta_i (1-\epsilon_i) \sum_{\mathcal{S}_i \subset [M] \setminus i} \mathbb{E}\big[t_{i,\mathcal{S}_i}(k) | \vec{s}(k) \big] \delta_{i,\mathcal{S}_i}(k) + \sum \beta_i.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\Theta_1(k) = &\sum_{i=1}^{M} \beta_i - \sum_{i=1}^{M} 1_{\{A(k) = Q_{i,\varnothing}\}} \beta_i \delta_{i,\varnothing}(k)(1-\epsilon_i) \\
&- \sum_{l=2}^{M} \sum_{\tau_1, \tau_2, \cdots, \tau_l} 1_{\{A(k) = \oplus_{u \in [l]} Q_{\tau_j, \mathcal{S}_{\tau_u}}\}} \times \sum_{u=1}^{l} \beta_{\tau_u} \delta_{i,\mathcal{S}_{\tau_u}}(k)(1-\epsilon_{\tau_u}).
\end{aligned} \tag{D.4}$$

Next, we consider $\Theta_2(k)$. Given $\vec{s}(k)$, the term $\sum_{i=1}^{M} \big(y_i^+(k)\big)^2$ is fixed and it is, therefore, sufficient to consider

$$\tilde{\Theta}_2(k) = \mathbb{E}\Big[\sum_{i=1}^{M} \big(y_i^+(k+1)\big)^2 | \vec{s}(k) \Big].$$

Note that

$$\mathbb{E}[\big(y_i^+(k+1)\big)^2 | \vec{s}(k)] = \mathbb{E}\Big[\Big(\big(y_i(k) + q_i - d_i(k)\big)^+ \Big)^2 | \vec{s}(k) \Big], \tag{D.5}$$

then,

$$\mathbb{E}[\big(y_i^+(k+1)\big)^2|\vec{s}(k)] = \begin{cases} \big((y_i(k)+q_i)^+\big)^2 & \text{if } \sum_{\mathcal{S}_i\subset[M]\setminus i} t_{i,\mathcal{S}_i}(k) = 0 \\ \big((y_i(k)+q_i-1)^+\big)^2(1-\epsilon_i) & \\ +\big((y_i(k)+q_i)^+\big)^2\big(1-(1-\epsilon_i)\big) & \text{if } \sum_{\mathcal{S}_i\subset[M]\setminus i} t_{i,\mathcal{S}_i}(k) = 1. \end{cases} \tag{D.6}$$

Now, we consider the following two cases:

Case 1. If $A(k) = Q_{i,\varnothing}$, then

$$\tilde{\Theta}_2(k) = (1-\epsilon_i)\bigg(\Big(\big(y_i(k)+q_i-1\big)^+\Big)^2 - \Big(\big(y_i(k)+q_i\big)^+\Big)^2\bigg) + \sum_{j=1}^{M}\Big(\big(y_j(k)+q_j\big)^+\Big)^2.$$

Case 2. If $A(k) = \oplus_{u\in[l]} Q_{\tau_u,\mathcal{S}_{\tau_u}}$, then

$$\tilde{\Theta}_2(k) = \sum_{u=1}^{l}(1-\epsilon_{\tau_u})\bigg(\Big(\big(y_{\tau_u}(k)+q_{\tau_u}-1\big)^+\Big)^2 - \Big(\big(y_{\tau_u}(k)+q_{\tau_u}\big)^+\Big)^2\bigg) + \sum_{j=1}^{M}\Big(\big(y_j(k)+q_j\big)^+\Big)^2.$$

Putting the two cases together, we have

$$\tilde{\Theta}_2(k) = \sum_{j=1}^{M}\Big(\big(y_j(k)+q_j\big)^+\Big)^2 - \sum_{j=1}^{M} 1_{\{A(k)=Q_{j,\varnothing}\}} f_j(k) - \sum_{l=2}^{M}\sum_{\tau_1,\tau_2,\cdots,\tau_l} 1_{\{A(k)=\oplus_{u\in[l]}Q_{\tau_u,\mathcal{S}_{\tau_u}}\}} \sum_{u=1}^{l} f_{\tau_u}(k). \tag{D.7}$$

Using (D.4) and (D.7), one can see that minimizing $\Theta(k)$ is equivalent to maximizing

$$\begin{aligned} &\sum_{i=1}^{M} 1_{\{A(k)=Q_{i,\varnothing}\}}(1-\epsilon_i)\Big(\beta_i\delta_{i,\varnothing}(k) + \lambda f_i(k)\Big) \\ &+ \sum_{l=2}^{M}\sum_{\tau_u,u\in[l]} 1_{\{A(k)=\oplus_{u\in[l]}Q_{\tau_u,\mathcal{S}_{\tau_u}}\}} \times \sum_{u=1}^{l}(1-\epsilon_{\tau_u})\Big(\beta_{\tau_u}\delta_{i,\mathcal{S}_{\tau_u}}(k) + \lambda f_{\tau_u}(k)\Big). \end{aligned}$$

D.4. Proof of Theorem 17

In the proof, we consider $M = 3$ users. The approach can be generalized to M users straightforwardly.

Step 1: Consider a stationary randomized policy satisfying the minimum requirements of rate

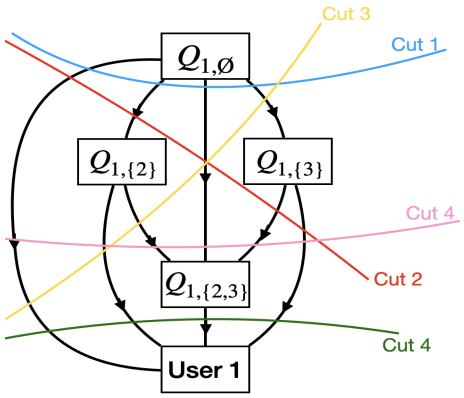


Figure D.1: The flow of user 1 in the virtual network under uncoded caching.

(q_1, q_2, q_3), and obtain the feasible region of $(\mu, \zeta_1, \zeta_2, \zeta_3, \zeta_4)$.

Recall that $\{y_i^+(k)\}_{i=1}^3$ is strongly stabilized under (q_1, q_2, q_3) using the ARM policy. By [155, 245, 246], there exists a stationary randomized policy that can reach the optimal throughput/rate, hence it satisfies the rate constraints. Consider a proper stationary randomized policy defined in (5.20) and (5.21). We first need to find the feasible region of $\{\mu_{i,\varnothing}\}_i$, ζ_1, ζ_2 and ζ_3.

We define Figure D.1 as the equivalent information flow graph for user 1. An uncoded or coded packet moves through the direction of an arrow. For example, a uncoded packet from $Q_{1,\varnothing}$ might be cached in $Q_{1,\{2\}}$, $Q_{1,\{3\}}$, $Q_{1,\{2,3\}}$ or decoded by user 1; a coded packet from $Q_{1,\{3\}}$ and another virtual queue $\big(Q_{3,\{1\}}$ or $Q_{3,\{1,2\}}\big)$ might be cached by $Q_{1,\{2,3\}}$ or decoded by user 1. When a packet leaves $Q_{1,\{3\}}$, there must be another packet leaves $Q_{3,\{1\}}$ or $Q_{3,\{1,2\}}$ in the equivalent information flow graph for user 3 simultaneously. Define a *cut* as a line which separates $Q_{1,\varnothing}$ and user 1 in Figure D.1. The expected amount of packets transmitting in the equivalent information flow graph for user 1 can be calculated by cuts. We have 5 cuts in Figure D.1.

Then, we need to find the (expected) throughput inequalities based on the cuts. We describe Cut 1 and Cut 2 as examples and display the inequalities associated with Cuts $3, 4, 5$ directly. Cut 1 straightforwardly implies that

$$\mu_{1,\varnothing}(1-\epsilon_1) + \sum_{j\in\{2,3\}} \mu_{1,\varnothing}\sigma([3]\backslash j) + \mu_{1,\varnothing}\sigma(\{1\}) \geq q_1, \tag{D.8}$$

In (D.8), we do not consider the probability of empty $Q_{1,\varnothing}$. Inequality (D.8) still holds (but results in a smaller feasible region) without considering the probability of empty $Q_{1,\varnothing}$ because the stationary randomized policy satisfies the rate constraints. Then, to simplify the analysis, we ignore the probability of empty queues. Note that we consider an independent and symmetric channel, (D.8) is equivalent to $\mu(1-\epsilon^3) \geq q$.

In Cut 2, the expected amount of packets transmitting through 3 arrows oriented $Q_{1,\varnothing}$ (see Figure D.1) is

$$\mu_{1,\varnothing}(1-\epsilon_1) + \mu_{1,\varnothing}\sigma(\{1,3\}) + \mu_{1,\varnothing}\sigma(\{1\}).$$

The expected amount of packets transmitting through the arrow from $Q_{1,\{3\}}$ to user 1 (see Figure D.1) is

$$\big(\mu_{1,\{3\},3,\{1\}} + \mu_{1,\{3\},3,\{1,2\}}\big)\big(1-\epsilon_1\big).$$

The expected amount of packets transmitting through the arrow from $Q_{1,\{3\}}$ to $Q_{1,\{2,3\}}$ (see Figure D.1) is

$$\mu_{1,\{3\},3,\{1,2\}}\big(\sigma(\{1\}) + \sigma(\{1,3\})\big),$$

because the encoded packet from $Q_{1,\{3\}}$ and $Q_{3,\{1\}}$ can not be cached by user 2 (note that we consider coding policies with uncoded caching). In total, Cut 2 implies that

$$\begin{aligned} &\mu_{1,\varnothing}(1-\epsilon_1) + \mu_{1,\varnothing}\sigma(\{1,3\}) + \mu_{1,\varnothing}\sigma(\{1\}) + \mu_{1,\{3\},3,\{1\}}(1-\epsilon_1) \\ &+ \mu_{1,\{3\},3,\{1,2\}}\big(1-\epsilon_1+\sigma(\{1\})+\sigma(\{1,3\})\big) \geq q_1, \end{aligned} \tag{D.9}$$

which is equivalent to $(\mu+\zeta_2)(1-\epsilon^2) + \zeta_1(1-\epsilon) \geq q$.

By similar analysis, Cut 3 implies that $(\mu+\zeta_2)(1-\epsilon^2) + \zeta_1(1-\epsilon) \geq q$. Cut 4 implies that $\mu(1-2\epsilon^2+\epsilon^3) + 2\zeta_1(1-\epsilon) + 2\zeta_2(1-\epsilon^2) \geq q$. Cut 5 implies that $\big(\mu+2\zeta_1+4\zeta_2+2\zeta_3+\zeta_4\big)(1-\epsilon) \geq q$.

Similar process can be done for users 2 and 3, the feasible region of μ, ζ_1, ζ_2, ζ_3 and ζ_4 is

$$\begin{aligned}
&3\mu + 3\zeta_1 + 6\zeta_2 + 3\zeta_3 + \zeta_4 = 1\\
&\mu(1-\epsilon^3) \geq q,\ (\mu+\zeta_2)(1-\epsilon^2) + \zeta_1(1-\epsilon) \geq q\\
&\mu(1-2\epsilon^2+\epsilon^3) + 2\zeta_1(1-\epsilon) + 2\zeta_2(1-\epsilon^2) \geq q\\
&\big(\mu + 2\zeta_1 + 4\zeta_2 + 2\zeta_3 + \zeta_4\big)(1-\epsilon) \geq q.
\end{aligned} \tag{D.10}$$

Step 2: Find an upper bound on AoI under the stationary randomized policy. Using (D.4) from the proof of Theorem 16, we write

$$\Theta_1(k) = \sum_{i=1}^{3}\beta_i - \sum_{i=1}^{3}\mu_{i,\varnothing}\beta_i\delta_{i,\varnothing}(k)(1-\epsilon_i) - \sum_{l=2}^{3}\sum_{\tau_1,\mathcal{S}_{\tau_1},\cdots,\tau_l,\mathcal{S}_{\tau_l}} \mu_{(\tau_1,\mathcal{S}_{\tau_1},\cdots,\tau_l,\mathcal{S}_{\tau_l})} \times \sum_{j=1}^{l}\beta_{\tau_j}\delta_{i,\mathcal{S}_{\tau_j}}(k)(1-\epsilon_{\tau_j}).$$

Note that $w_{i,\mathcal{S}}(k) \leq h_i(k)$ for all i and $\mathcal{S}$, and hence, $\mathbb{E}[\delta_{i,\mathcal{S}}(k)] \geq 0$. Dropping the terms related to $\mathbb{E}[\delta_{i,\mathcal{S}}(k)]$ when $\mathcal{S} \neq \varnothing$, we find

$$\Theta_1(k) \leq \sum_{i=1}^{3}\beta_i - \sum_{i=1}^{3}\mu_{i,\varnothing}\beta_i\delta_{i,\varnothing}(k)(1-\epsilon_i). \tag{D.11}$$

To deal with $\Theta_2(k)$, we first note that (see [14, Eqn. (60) - (62)])

$$\big(y_i^+(k+1)\big)^2 - \big(y_i^+(k)\big)^2 \leq -2y_i^+(k)\big(d_i(k) - q_i\big) + 1.$$

Take the expectation on both sides, we have

$$\Theta_2(k) \leq \sum_{i=1}^{3} -2y_i^+(k)\big(\mathbb{E}[d_i(k)] - q_i\big) + 3. \tag{D.12}$$

Under a stationary, $\mathbb{E}[d_i(k)]$ keeps a constant over k, and $\mathbb{E}[d_i(k)] \geq q_i$. Using the bounds in (D.11) and (D.12), we now upper bound the K step drift:

$$\sum_{k=1}^{K}\mathbb{E}[\Theta(k)] \leq \sum_{k=1}^{K}\sum_{i=1}^{3}\beta_i(1-\epsilon_i)\mu_{i,\varnothing}\mathbb{E}[w_{i,\varnothing}(k) - h_i(k)] - \sum_{i=1}^{3}2\lambda\mathbb{E}[y_i^+(k)](\mathbb{E}[d_i(k)] - q_i) + K\sum_{i=1}^{3}\beta_i + 3K\lambda.$$

The above inequality can also be re-written as follows:

$$\begin{aligned}&\sum_{k=1}^{K}\sum_{i=1}^{3}\beta_i(1-\epsilon_i)\mu_{i,\varnothing}\mathbb{E}[h_i(k)]+\sum_{k=1}^{K}\sum_{i=1}^{3}2\lambda\mathbb{E}[y_i^+(k)](\mathbb{E}[d_i(k)]-q_i)\\&\leq-\sum_{k=1}^{K}\mathbb{E}[\Theta(k)]+\sum_{k=1}^{K}\sum_{i=1}^{3}\beta_i(1-\epsilon_i)\mu_{i,\varnothing}\mathbb{E}[w_{i,\varnothing}(k)]+K\sum_{i=1}^{3}\beta_i+3K\lambda,\end{aligned}\tag{D.13}$$

which implies

$$L_1+L_2\leq L_3+\sum_{k=1}^{K}\sum_{i=1}^{3}\beta_i(1-\epsilon_i)\mu_{i,\varnothing}\mathbb{E}[w_{i,\varnothing}(k)]+C\tag{D.14}$$

where

$$\begin{aligned}&L_1=\sum_{k=1}^{K}\sum_{i=1}^{3}\beta_i(1-\epsilon_i)\mu_{i,\varnothing}\mathbb{E}[h_i(k)],\ L_2=\sum_{k=1}^{K}\sum_{i=1}^{3}2\lambda\mathbb{E}[y_i^+(k)](\mathbb{E}[d_i(k)]-q_i)\\&L_3=-\sum_{k=1}^{K}\Theta(k),\ C=K\sum_{i=1}^{3}\beta_i+3K\lambda.\end{aligned}$$

Letting $K\to\infty$, we find $\lim_{K\to\infty}\frac{L_3}{K}\leq\lim_{K\to\infty}\frac{L(\vec{s}(1))}{K}=0$. Dividing by $3K$, we obtain

$$\lim_{K\to\infty}\frac{L_1}{3K}+\lim_{K\to\infty}\frac{L_2}{3K}\leq\frac{\sum_{i=1}^{3}\beta_i(1-\epsilon_i)\mu_{i,\varnothing}\sum_{k=1}^{K}\mathbb{E}[w_{i,\varnothing}(k)]}{3K}+\frac{\sum_{i=1}^{3}\beta_i}{3}+\lambda.\tag{D.15}$$

Now note that the recursion of $w_{i,\varnothing}(k)$ in (5.10) has a geometric distribution with parameter θ_i. By Wald's equation, $\mathbb{E}[w_{i,\varnothing}(k)]=\frac{1}{\theta_i}$, (D.15) simplifies to

$$\lim_{K\to\infty}\frac{L_1}{K}+\lim_{K\to\infty}\frac{L_2}{K}\leq\frac{1}{3}\sum_{i=1}^{3}\beta_i\big(\frac{(1-\epsilon_i)\mu_{i,\varnothing}}{\theta_i}+1\big)+\lambda.\tag{D.16}$$

Furthermore, substituting for L_1, L_2 and noting that $L_1>0$, we find

$$\sum_{i=1}^{3}2\lambda(\mathbb{E}[d_i(k)]-q_i)\lim_{K\to\infty}\frac{\sum_{k=1}^{K}\mathbb{E}[y_i^+(k)]}{K}\leq\frac{1}{3}\sum_{i=1}^{3}\beta_i\big(\frac{(1-\epsilon_i)\mu_{i,\varnothing}}{\theta_i}+1\big)+\lambda.\tag{D.17}$$

We conclude that

$$\lim_{K\to\infty} \frac{\sum_{k=1}^{K} \mathbb{E}[y_i^+(k)]}{K} < \infty, \quad i = 1, 2, \cdots, M,$$

which implies the strong stability of the process $y_i^+(k)$ is satisfied [**?**], [223, Theorem 2.8].

Finally, since $L_2 > 0$, we conclude that

$$\lim_{K\to\infty} \frac{L_1}{3K} \leq \frac{1}{3} \sum_{i=1}^{3} \beta_i \big(\frac{(1-\epsilon_i)\mu_{i,\varnothing}}{\theta_i} + 1 \big) + \lambda. \tag{D.18}$$

From (D.18), set $\beta_i = \frac{\alpha_i}{\mu_{i,\varnothing}(1-\epsilon_i)}$, we have

$$J(\underline{q}) \leq \frac{1}{3} \sum_{i=1}^{3} \big(\frac{\alpha_i}{\theta_i} + \frac{\alpha_i}{\mu_{i,\varnothing}(1-\epsilon_i)} \big) + \lambda.$$

Combining **Step 1** and **Step 2**, we obtain the desired results (5.23).

D.5. Transmitting x_2 can not be worse than transmitting x_1 in terms of AoI

We first consider the case where x_1 and x_2 are transmitted directly. If x_1 is decoded by users $1, 2$, then from Definition 21, the AoI reduction of the users $1, 2$ is

$$\alpha_1 \min\{k - e(t_1), h_1(k-1) + 1\} + \alpha_2 \min\{k - v(t_2), h_2(k-1) + 1\},$$

which is larger than the AoI reduction when x_2 is decoded,

$$\alpha_1 \min\{k - e(t_3), h_1(k-1) + 1\} + \alpha_2 \min\{k - v(t_4), h_2(k-1) + 1\}.$$

This is because $\alpha_1 e(t_1) + \alpha_2 v(t_2) > \alpha_1 e(t_3) + \alpha_2 v(t_4)$. Then, transmitting x_2 can not be worse than transmitting x_1.

Then, if the encoder transmits a coded packet which is formed by $c_1 = x_1 \oplus p$ or $c_2 = x_2 \oplus p$, where $p \in Q_{3,\mathcal{S}_3}$. Note that c_1 and c_2 provide the same AoI reduction to user 3. By similar analysis above,

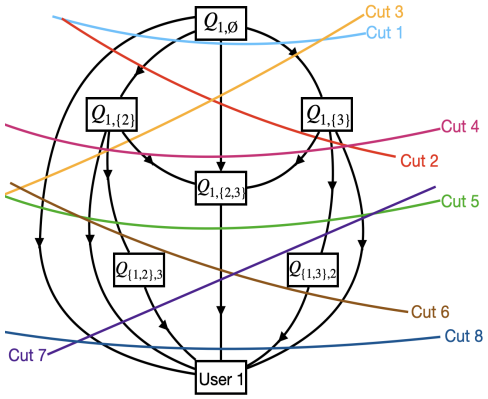


Figure D.2: The flow of user 1 in the virtual network under coded caching.

c_2 provides a larger (weighted sum of) AoI reduction for users $1, 2$. Then, encoding x_2 can not be worse than encoding x_1 in terms of AoI.

Finally, we consider the case where x_1 is decoded first. Suppose that x_1 has been decoded within $c_1' = x_1 \oplus p_1$ and $p_1 \in Q_{3,\mathcal{S}_3'}$. Denote the (weighted sum of) age reduction for users $1, 2$ as Δ'. After that, x_2 has been decoded by $c_2' = x_2 \oplus p_2$ and $p_2 \in Q_{3,\mathcal{S}_3''}$. Denote the age reduction for users $1, 2$ as Δ''. The total (weighted sum of) age reduction for users $1, 2$ once decoding x_1 and x_2 is $\Delta' + \Delta''$. Now, consider another coded packet formed by replacing x_1 with x_2 in c_1', denoted by c'''. In this case, the (weighted sum of) age reductions for users $1, 2$ is $\Delta' + \Delta''$ after decoding c'''. Therefore, x_1 is useless for users $1, 2$ after decoding x_2, in the next time slot, user 3 can have the same AoI reduction by decoding the fresher packet in p_1 and p_2. We can conclude that encoding x_2 is not worse than encoding x_1.

D.6. Proof of Theorem 19.

The proof is similar to the proof of Theorem 17 in Appendix D.4. The main difference is that we define a new equivalent information flow graph (for user 1) in Figure D.2, where we have 8 cuts in total. Then, by a similar process in the proof of Theorem 17 in Appendix D.4, we get the desired results.

D.7. Proof of Corollary 5

The proof consists of minimizing (5.34) over all rate tuples $(r_1^\pi, \ldots, r_M^\pi) \in \hat{\mathcal{C}}$ where $\hat{\mathcal{C}}$ is an outer bound on the capacity region [153, Section III].

From [153, Section III], denote $\hat{C}$ as the channel capacity outer bound. Let π is a permutation of $[M]$ such that $\pi(M-i+1) = \hat{\pi}(i)$, where $\hat{\pi}$ is the permutation defined in [153, Definition 1]. Recall that $\sigma(\mathcal{I})$ is the probability that an erasure occurs for all users in $\mathcal{I}$. Denote $\hat{\epsilon}_{\pi(i)} = \sigma(\cup_{j=1}^{i}\{\pi(j)\})$. From [153, Lemma 3, Lemma 4], we can obtain the outer bound $\hat{C} = \cap_\pi \hat{C}_\pi$ where

$$\hat{C}_\pi = \Big\{(R_1, R_2, \cdots, R_M) | \sum_{i\in[M]} \frac{R_{\pi(i)}}{1-\hat{\epsilon}_{\pi(i)}} \leq 1\Big\}. \tag{D.19}$$

Consider symmetric and independent channels, i.e., $\epsilon_i = \epsilon$ for all $i \in [M]$. From (D.19), the capacity outer bound $\hat{\mathcal{C}}$ is given by

$$\hat{\mathcal{C}} = \Big\{(R_1, \cdots, R_M) | 0 \leq R_i \leq \frac{1}{\sum_{j=1}^{M} 1/(1-\epsilon^j)}\Big\}. \tag{D.20}$$

From (D.20), LB_1^π can be re-written as

$$LB_1^\pi = \frac{M}{2\epsilon(M)\sum_{i=1}^{M} 1/\alpha_i} + \sum_{i=1}^{M} \frac{\alpha_i}{2M}.$$

where $\epsilon(M) = \frac{1}{\sum_{j=1}^{M} 1/(1-\epsilon^j)}$.

APPENDIX E

Proofs in Chapter 6

E.1. Proof of Lemma 13

Note that a node is counted in $C^\pi(t)$ once it has been infected. Then, on the day $t+1$, $C^\pi(t+1)$ increases (compared to $C^\pi(t)$) only because some susceptible nodes are infected by infectious nodes and are in the latent state for the first time.

After testing, positive nodes in $\mathcal{K}^\pi(t)$ would not infect others because they are quarantined, and negative nodes would not infect others due to the model assumptions. Hence

$$C^\pi(t+1) = C^\pi(t) + \sum_{i \in \mathcal{V}(t)} F_i(\mathcal{V}(t) \backslash \mathcal{K}^\pi(t); t).$$

Taking the expectation on both sides, we obtain the desired result.

E.2. Proof of Theorem 21

To show $S\big(\mathcal{K}^\pi(t); t\big)$ defined in (6.6) is a supermodular function. It suffices to show that for any $\mathcal{A} \subset \mathcal{B} \subset \mathcal{V}(t)$, and for $x \in \mathcal{V}(t) \backslash \mathcal{B}$, we have

$$S\big(\mathcal{A} \cup \{x\}; t\big) - S\big(\mathcal{A}; t\big) \leq S\big(\mathcal{B} \cup \{x\}; t\big) - S\big(\mathcal{B}; t\big). \tag{E.1}$$

Then, it suffices to show for any $i \in \mathcal{V}(t)$,

$$\mathbb{E}[F_i\big(\mathcal{A} \cup \{x\}; t\big)] - \mathbb{E}[F_i\big(\mathcal{A}; t\big)] \leq \mathbb{E}[F_i\big(\mathcal{B} \cup \{x\}; t\big)] - \mathbb{E}[F_i\big(\mathcal{B}; t\big)]. \tag{E.2}$$

Now, we consider three cases.

Case 1. If $\mathcal{A} \cap \partial_i(t) = \mathcal{B} \cap \partial_i(t)$, then from (6.5), the LHS and RHS in (E.2) are exactly the same. Hence, (E.2) holds.

Case 2. If $\mathcal{A} \cap \partial_i(t) \subset \mathcal{B} \cap \partial_i(t)$, and $x \notin \partial_i$, then from (6.5), $f_i(\mathcal{A} \cup \{x\}) = f_i(\mathcal{A})$ and $f_i(\mathcal{B} \cup \{x\}) =$

$f_i(\mathcal{B})$. Hence (E.2) holds.

Case 3. If $\mathcal{A}\cap\partial_i(t)\subset\mathcal{B}\cap\partial_i(t)$, and $x\in\partial_i(t)$, let $\mathcal{Y}=\big(\mathcal{B}\cap\partial_i(t)\big)\backslash\big(\mathcal{A}\cap\partial_i(t)\big)$. Here $x\notin\mathcal{Y}$. From (6.5), we can compute

$$\begin{aligned}
&\mathbb{E}[F_i(\mathcal{A}\cup\{x\};t)]-\mathbb{E}[F_i(\mathcal{A};t)]\\
&=v_S^{(i)}(t)\prod_{j\in\partial_i(t)\backslash(\mathcal{A}\cup\{x\})}\big(1-\beta v_I^{(j)}(t)\big)\\
&\times\Big(1-\prod_{j\in\partial_i(t)\cap(\mathcal{A}\cup\{x\})}(1-\beta v_I^{(j)}(t))-(1-\beta v_I^{(x)}(t))\big(1-\prod_{j\in\partial_i(t)\cap\mathcal{A}}(1-\beta v_I^{(j)}(t))\big)\Big),
\end{aligned}$$

which implies

$$\begin{aligned}
&\mathbb{E}[F_i(\mathcal{A}\cup\{x\};t)]-\mathbb{E}[F_i(\mathcal{A};t)]\\
&=v_S^{(i)}(t)\prod_{j\in\partial_i(t)\backslash(\mathcal{A}\cup\{x\})}\big(1-\beta v_I^{(j)}(t)\big)\\
&\times\Big(\beta v_I^{(x)}(t)-\prod_{j\in\partial_i(t)\cap(\mathcal{A}\cup\{x\})}(1-\beta v_I^{(j)}(t))+\prod_{j\in\partial_i(t)\cap(\mathcal{A}\cup\{x\})}(1-\beta v_I^{(j)}(t))\Big)\\
&=v_S^{(i)}(t)\prod_{j\in\partial_i(t)\backslash(\mathcal{A}\cup\{x\})}\big(1-\beta v_I^{(j)}(t)\big)\beta v_I^{(x)}(t).
\end{aligned}$$

Similarly, note that $\big(\mathcal{B}\cap\partial_i(t)\big)=\big(\mathcal{A}\cap\partial_i(t)\big)\cup\mathcal{Y}$. We have

$$\mathbb{E}[F_i(\mathcal{B}\cup\{x\};t)]-\mathbb{E}[F_i(\mathcal{B};t)]=v_S^{(i)}(t)\prod_{j\in\partial_i(t)\backslash(\mathcal{A}\cup(\{x\}\cup\mathcal{Y}))}\big(1-\beta v_I^{(j)}(t)\big)\beta v_I^{(x)}(t).$$

Thus,

$$\frac{\mathbb{E}[F_i(\mathcal{A}\cup\{x\};t)]-\mathbb{E}[F_i(\mathcal{A};t)]}{\mathbb{E}[F_i(\mathcal{B}\cup\{x\};t)]-\mathbb{E}[F_i(\mathcal{B};t)]}=\prod_{y\in\mathcal{Y}}\big(1-\beta v_I^{(y)}(t)\big)\le 1,$$

which implies $S\big(\mathcal{TP}^\pi(t)\big)$ is supmodular.

To show $S\big(\mathcal{K}^\pi(t);t\big)$ is an increasing monotone function on $\mathcal{K}^\pi(t)$, it suffices to show $\mathbb{E}[F_i\big(\mathcal{K}^\pi(t);t\big)]$ is an increasing monotone function on $\mathcal{K}^\pi(t)$ for any i.

For $\mathcal{A} \subset \mathcal{B}$, we have $\partial_i(t) \backslash \mathcal{B} \subset \partial_i(t) \backslash \mathcal{A}$, and $\mathcal{A} \cap \partial_i(t) \subset \mathcal{B} \cap \partial_i(t)$. Then

$$\prod_{j \in \partial_i(t) \backslash \mathcal{B}} \big(1 - \beta v_I^{(j)}(t)\big) \geq \prod_{j \in \partial_i(t) \backslash \mathcal{A}} \big(1 - \beta v_I^{(j)}(t)\big)$$
$$\prod_{j \in \mathcal{B} \cap \partial_i(t)} (1 - \beta v_I^{(j)}(t)) \leq \prod_{j \in \mathcal{A} \cap \partial_i(t)} (1 - \beta v_I^{(j)}(t)),$$

and thus, from (6.5), we have $\mathbb{E}[F_i\big(\mathcal{A}; t\big)] \leq \mathbb{E}[F_i\big(\mathcal{B}; t\big)]$.

E.3. Complexity of Algorithm 5

First of all, we consider the complexity of (6.5). Suppose $\{\underline{v}_i(t)\}_{i \in \mathcal{V}(t)}$ is given for every day t. For any $\mathcal{K}^\pi(t)$, the complexity of computing (6.5) is

$$1 + |\partial_i(t) \backslash \mathcal{K}^\pi(t)| - 1 + 1 + |\partial_i(t) \backslash \mathcal{K}^\pi(t)| + |\partial_i(t) \cap \mathcal{K}^\pi(t)| - 1 + |\partial_i(t) \cap \mathcal{K}^\pi(t)| = 2|\partial_i(t)|.$$

Then, for any $\mathcal{K}^\pi(t)$, the complexity of computing $S\big(\mathcal{K}^\pi(t); t\big)$ is $2 \sum_{j \in \mathcal{V}(t)} |\partial_j(t)|$. From Algorithm 5, in step i, the complexity is $\big(N(t) - i + 1\big) \times 2 \sum_{j \in \mathcal{V}(t)} |\partial_j(t)|$. And in total we have $\big(N(t) - |\mathcal{K}^\pi(t)|\big)$ steps, therefore, on day t the complexity of Algorithm 5 is

$$\sum_{i=0}^{N(t) - |\mathcal{K}^\pi(t)|} 2\big(N(t) - i + 1\big) \sum_{j \in \mathcal{V}(t)} |\partial_j(t)|.$$

Recall that the time horizon is T, then the total complexity of Algorithm 5 is

$$\sum_{t=0}^{T-1} \sum_{i=0}^{N(t) - |\mathcal{K}^\pi(t)|} 2\big(N(t) - i + 1\big) \sum_{j \in \mathcal{V}(t)} |\partial_j(t)|.$$

Note that

$$\sum_{i=1}^{N(t) - |\mathcal{K}^\pi(t)|} 2\big(N(t) - i + 1\big) \leq O\big(N^2(t)\big)$$
$$\sum_{i \in \mathcal{V}(t)} |\partial_i(t)| \leq O\big(N^2(t)\big).$$

Then, the total complexity is bounded by $O\Big(\sum_{t=0}^{T-1} N^4(t)\Big)$.

E.4. Proof of Lemma 14

As defined in [229] (an equivalent definition of footnote 3), consider a finite set I, $f : 2^I \to \mathbb{R}$ is a supermodular function if for all $X, Y \subset I$,

$$f(X \cup Y) + f(X \cap Y) \geq f(X) + f(Y). \tag{E.3}$$

Following the supermodularity of function $S(\cdot)$ as shown in Theorem 21, set $X = \mathcal{V}(t)\backslash\mathcal{K}^\pi(t)$ and $Y = \mathcal{K}^\pi(t)$ in (E.3), we have

$$S\big(\mathcal{V}(t)\backslash\mathcal{K}^\pi(t); t\big) \leq S\big(\mathcal{V}(t); t\big) - S\big(\mathcal{K}^\pi(t); t\big). \tag{E.4}$$

Again, set $X = \mathcal{K}^\pi(t)\backslash\{i\}$ and $Y = \{i\}$ in (E.3), and use (E.3) repeatedly to obtain:

$$S\big(\mathcal{K}^\pi(t); t\big) \geq \sum_{i\in\mathcal{K}^\pi(t)} S\big(\{i\}; t\big) = \sum_{i\in\mathcal{K}^\pi(t)} r_i(t). \tag{E.5}$$

Substituting (E.5) in (E.4), we obtain

$$S\big(\mathcal{V}(t)\backslash\mathcal{K}^\pi(t); t\big) \leq S\big(\mathcal{V}(t); t\big) - \sum_{i\in\mathcal{K}^\pi(t)} r_i(t).$$

E.5. Local Transition Equations

In this section, we will describe the local transition matrix $\mathtt{P}_i\big(\{\underline{v}_j(t)\}_{j\in\partial_i^+(t)}\big)$ used in (6.12). The state of each node evolves as follows: (i) if node i is susceptible on day t, then it might be infected by its neighbors in $\partial_i(t)$; (ii) an infectious node remains in the latent state with probability $1-\lambda$, and changes state to the infectious state (I) with probability λ; (iv) if node i is in state I, it will recover after a geometric distribution with parameter γ. Let $\xi_i(t) = 1 - \prod_{m\in\partial_i(t)} \big(1 - v_I^{(m)}(t)\beta\big)$. In particular, define $\xi_i(t) = 0$ if $\partial_i(t) = \varnothing$. Then, the probabilities of nodes being in different states

evolve in time as follows:

$$v_I^{(i)}(t+1) = v_I^{(i)}(t)(1-\gamma) + v_L^{(i)}(t)\lambda \tag{E.6}$$

$$v_L^{(i)}(t+1) = v_L^{(i)}(t)(1-\lambda) + v_S^{(i)}(t)\xi_i(t) \tag{E.7}$$

$$v_R^{(i)}(t+1) = v_R^{(i)}(t) + v_I^{(i)}(t)\gamma \tag{E.8}$$

$$v_S^{(i)}(t+1) = v_S^{(i)}(t)\big(1-\xi_i(t)\big). \tag{E.9}$$

Note that row vector $\underline{v}_i(t)$ is defined in (6.3). Collecting (E.6) - (E.9), we define the local transition probability matrix as given below:

$$\mathtt{P}_i\big(\{\underline{v}_j(t)\}_{j\in\partial_i^+(t)}\big) = \begin{bmatrix} (1-\gamma) & 0 & \gamma & 0 \\ \lambda & 1-\lambda & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & \xi_i(t) & 0 & 1-\xi_i(t) \end{bmatrix}. \tag{E.10}$$

and we obtain (6.12).

E.6. Proofs of (6.13) and (6.14)

First of all, we give the following definition.

Definition 28. *Let X be a random variable and $\mathcal{B}$ be an event. Define $X|_{\mathcal{B}}$ as the random variable X given $\mathcal{B}$; i.e.,*

$$\Pr\big(X|_{\mathcal{B}} = x\big) = \Pr\big(X = x|\mathcal{B}\big). \tag{E.11}$$

For brevity, let us define $\theta_i(t) = \sigma_i(t)|_{\{\underline{Y}(\tau)\}_{\tau=1}^{t-1}}$, $\zeta_i(t) = \sigma_i(t)|_{\{\underline{Y}(\tau)\}_{\tau=1}^{t}}$. We thus have

$$u_x^{(i)}(t) = \Pr\big(\theta_i(t) = x\big), \quad w_x^{(i)}(t) = \Pr\big(\zeta_i(t) = x\big).$$

Recall that $\underline{v}_i(t) = \left[v_x^{(i)}(t)\right]_{x\in\mathcal{X}}$, $v_x^{(i)}(t) = \Pr\left(\sigma_i(t) = x\right)$. Then, (6.12) can be re-written as

$$\Pr\left(\sigma_i(t+1) = x_i'\right) = \Pr\left(\sigma_i(t) = x_i\right)\mathtt{P}_i\Big(\left\{\sigma_j(t)\right\}_{j\in\partial_i^+(t)} = \left\{x_j\right\}_{j\in\partial_i^+(t)}\Big), \tag{E.12}$$

where $x_i', \{x_j\}_{j\in\partial_i^+(t)} \in \mathcal{X}$. Conditioning both sides of (E.12) on $\{\underline{Y}(\tau)\}_{\tau=1}^{t-1}$, state variables $\sigma_i(t)$ and $\sigma_i(t-1)$ in (E.12) can be replaced by $\theta_i(t)$ and $\zeta_i(t-1)$, respectively, to obtain

$$\underline{u}_i(t) = \underline{w}_i(t-1) \times \mathtt{P}_i\big(\{\underline{w}_j(t-1)\}_{j\in\partial_i^+(t-1)}\big), \tag{E.13}$$

which gives (6.13). In addition, define $\phi_i(t) = \sigma_i(t)|_{\{\underline{Y}(\tau)\}_{\tau=1}^{t+1}}$, and

$$\begin{aligned} \underline{e}_i(t-1) &= (e_x^{(i)}(t-1), x\in\mathcal{X}), \\ e_x^{(i)}(t-1) &= \Pr\left(\phi_i(t-1) = x\right). \end{aligned} \tag{E.14}$$

This notation implies

$$\phi_i(t-1) = \theta_i(t-1)|_{\underline{Y}(t)}. \tag{E.15}$$

Similarly, conditioning both sides of (E.13) on $\underline{Y}(t)$, we find

$$\underline{w}_i(t) = \underline{e}_i(t-1) \times \tilde{\mathtt{P}}_i\big(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)}\big), \tag{E.16}$$

which gives (6.14). $\tilde{P}_i(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)})$ is obtained in the following subsection.

E.6.1. Computing the transition probability matrix $\tilde{P}_i(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)})$

Note that $\tilde{P}_i(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)})$ is not the same as $P_i(\{\underline{w}_j(t)\}_{j\in\partial_i^+(t)})$. *This is because "future" observations were available in* $\tilde{P}_i(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)})$. To get $\tilde{\mathtt{P}}_i\big(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)}\big)$, we split the nodes $\mathcal{V}(t)$ into two classes of nodes: (i) nodes that do not get new observations and (ii) nodes that get new observations. $\tilde{P}_i(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)})$ is obtained by the following rules. For the first class of nodes, the local transition matrix in (E.16), i.e., $\tilde{P}_i(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)})$, is the same as that in

(E.13). However, for the second class of nodes, the local transition matrices are changed accordingly because of the new observations. Let $[A]_{\{i,:\}}$ be the i^{th} row of matrix A, and q_i be a 1×4 vector with the i^{th} element being one and the rest zero. For brevity, denote the local transition matrices in (E.13) and (E.16) by $\mathtt{P}_i(t-1)$ and $\tilde{\mathtt{P}}_i(t-1)$, respectively. We have the following three cases:

(i) If node i is not observed, then node i does not have new observation and we have

$$\tilde{P}_i(t-1) = P_i(t-1). \tag{E.17}$$

(ii) If $Y_i(t) = 0$, then node i is not infectious in day t with probability 1. The local transition matrix is changed to

$$[\tilde{\mathtt{P}}_i(t-1)]_{\{j,:\}} = \begin{cases} q_3 & j = 1 \\ q_2 & j = 2 \\ [\mathtt{P}_i(t-1)]_{\{j,:\}} & \text{otherwise} \end{cases}. \tag{E.18}$$

(iii) If $Y_i(t) = 1$, then node i is infectious in day t with probability 1. The local transition matrix is changed to

$$[\tilde{\mathtt{P}}_i(t-1)]_{\{j,:\}} = \begin{cases} q_1 & j = 1 \\ q_1 & j = 2 \\ [\mathtt{P}_i(t-1)]_{\{j,:\}} & \text{otherwise} \end{cases}. \tag{E.19}$$

E.7. Proofs of (6.16) and (6.17)

Using new observations, we aim to move backward in time and update our belief (posterior probability) in previous time slots. Define a *truncation number* g and suppose that $\{\underline{Y}(t)\}$ affects the posterior probabilities from day t to day $t-g$. We call day $t-g$ the *truncation day* associated with day t. To get accurate posterior probabilities every day, we need to set $g = t$ on every day t

and track back to the initial time. However, the influence weakens as time elapses backward, and for computation tractability, we continue under the following assumption where $g = 1$. Recall that $\zeta_i(t) = \sigma_i(t)|_{\{\underline{Y}(\tau)\}_{\tau=1}^{t}}$.

Assumption 9. *On the truncation day $(t-g)$, $\{\zeta_i(t-g)\}_i$ are independent over i. In the following, the truncation number is assumed to be $g = 1$.*

Remark 30. *In Assumption 9, we assume that the nodes' states $\zeta(t-g)$ (in the posterior probability space on the day $t-g$) are independent. This assumption is only used at time t of our probability update in a moving window kind of way. It provides us with a truncation time for each backward step. In particular, under Assumption 9, once we get the observations $\underline{Y}(t)$, we do the backward step and truncate at time $t-g$. For example, in the trivial case of $g = t$, the assumption holds. This assumption does not impose independence on the state of the nodes, but only in the posterior space at a specific time. In a sense, in the process of propagating information back to time $t-g$, we are assuming that there is no further correlation between time $t-g-1$ and time $t-g$ worthwhile to exploit given observations at time t. Naturally, as g gets larger and larger, our framework and calculations become more precise but this comes at a huge computational cost. The idea behind truncating the backward step lies in the observation that the impact of the testing results at time t in inferring about the nodes' probabilities at time $t-g$ vanishes as g gets large. For simplicity of derivations and to have manageable complexity, we set $g = 1$. The idea and the derivations can be generalized in a straightforward manner to larger g.*

Note that the posterior probabilities on the day $t-1$, $\underline{w}_i(t-1)$, $i \in \mathcal{V}(t-1)$, are assumed known (and are conditioned on the history of observations $\{\underline{Y}(\tau)\}_{\tau=1}^{t-1}$). The probability vector $\underline{e}_i(t-1)$ is the new posterior probability at time $t-1$ which is updated (from $\underline{w}_i(t-1)$) based on new observations $\underline{Y}(t)$. In other words, we infer the previous state of the nodes given new observations at present.

To obtain $\{\underline{w}_i(t)\}_{i\in\mathcal{V}(t)}$, it suffices to obtain $\underline{e}_i(t-1)$ and the corresponding local transition matrix $\tilde{\mathsf{P}}_i\big(\{\underline{e}_j(t-1)\}_{j\in\partial_i^+(t-1)}\big)$, see (6.14). Note that the posterior probabilities $\underline{w}_i(t-1)$, $i \in \mathcal{V}(t-1)$, which

are calculated based on $\underline{Y}(t-1)$, are known. The vector $\underline{e}_i(t-1)$ is the new posterior probability which is updated based on $\underline{Y}(t)$ and $\underline{w}_i(t-1)$.

Equation (6.16), which we aim to prove, simply follows from Definition 28, (E.14)-(E.15), and Bayes rule:

$$e_x^{(i)}(t-1) = \Pr\big(\zeta_i(t-1)=x|\underline{Y}(t)\big) = \frac{\Pr\big(\underline{Y}(t)|\zeta_i(t-1)=x\big)w_x^{(i)}(t-1)}{\Pr\big(\underline{Y}(t)\big)}. \tag{E.20}$$

To find $\Pr\big(\underline{Y}(t)|\zeta_i(t-1)=x\big)$, and establish (6.17), we now proceed as follows. We introduce $\{\theta_j(t)\}_{j\in\mathcal{O}(t)}$ into (E.20). In particular, we have

$$\Pr\big(\underline{Y}(t)|\zeta_i(t-1)=x\big) = \sum_{\theta_j(t),\ j\in\mathcal{O}(t)} \Pr\big(\{\theta_j(t)\}_{j\in\mathcal{O}(t)}, \underline{Y}(t)|\zeta_i(t-1)=x\big)$$

By the chain rule of conditional probability,

$$\begin{aligned}&\Pr\big(\underline{Y}(t)|\zeta_i(t-1)=x\big)\\ =&\sum_{\theta_j(t),\ j\in\mathcal{O}(t)} \Pr\big(\underline{Y}(t)|\{\theta_j(t)\}_{j\in\mathcal{O}(t)},\zeta_i(t-1)=x\big)\times\Pr\big(\{\theta_j(t)\}_{j\in\mathcal{O}(t)}|\zeta_i(t-1)=x\big).\end{aligned}$$

From (6.15), $\{\zeta_j(t)\}_{j\in\mathcal{V}(t)}$ and $\{\theta_j(t)\}_{j\in\mathcal{V}(t)}$ are variables defined by $\{\sigma_j(t)\}_{j\in\mathcal{V}(t)}$ in posterior spaces of $\{\underline{Y}(\tau)\}_{\tau=1}^{t}$ and $\{\underline{Y}(\tau)\}_{\tau=1}^{t-1}$, respectively. Since $\underline{Y}(t)$ is a deterministic function of $\{\sigma_j(t)\}_{j\in\mathcal{O}(t)}$, and hence $\{\theta_j(t)\}_{j\in\mathcal{O}(t)}$, then $\underline{Y}(t)$ is independent of $\zeta_i(t-1)$ given $\{\theta_j(t)\}_{j\in\mathcal{O}(t)}$. In addition, the testing result $Y_j(t)$ (on day t) of node j only depends on its state, i.e., given $\theta_j(t)$, the testing results are determined. Therefore, we have

$$\Pr\big(\underline{Y}(t)|\{\theta_j(t)\}_{j\in\mathcal{O}(t)},\zeta_i(t-1)=x\big) = \Pr\big(\underline{Y}(t)|\{\theta_j(t)\}_{j\in\mathcal{O}(t)}\big) = \prod_{j\in\mathcal{O}(t)}\Pr\big(Y_j(t)|\theta_j(t)\big).$$

The product above is an indicator that takes values on $\{0,1\}$. We can thus re-write it as follows:

$$\Pr\big(\underline{Y}(t)|\{\theta_j(t)\}_{j\in\mathcal{O}(t)},\zeta_i(t-1)=x\big) \triangleq \delta(\{Y_j(t),\theta_j(t)\}_{j\in\mathcal{O}(t)}).$$

where $\delta(\{Y_j(t),\theta_j(t)\}_{j\in\mathcal{O}(t)})=1$ if the pairs $\{Y_j(t),\theta_j(t)\}_{j\in\mathcal{O}(t)}$ are consistent, and $\delta(\{Y_j(t),\theta_j(t)\}_{j\in\mathcal{O}(t)})=$

0 otherwise.

Next, define

$$\Theta_i(t) = \{j | j \in \partial_k^+(t-1), k \in \mathcal{O}(t)\} \backslash \{i\}$$

to represent the neighbors (in day $t-1$) of nodes in $\mathcal{O}(t)$ excluding node i. Then,

$$\Pr\left(\{\theta_j(t)\}_{j\in\mathcal{O}(t)} | \zeta_i(t-1) = x\right) = \sum_{\zeta_l(t-1),\ l\in\Theta_i(t)} \Pr\left(\{\theta_j(t)\}_{j\in\mathcal{O}(t)}, \{\zeta_l(t-1)\}_{l\in\Theta_i(t)} | \zeta_i(t-1) = x\right). \tag{E.21}$$

By the chain rule of conditional probability,

$$\begin{aligned}
&\Pr\left(\{\theta_j(t)\}_{j\in\mathcal{O}(t)} | \zeta_i(t-1) = x\right) \\
&= \sum_{\zeta_l(t-1),\ l\in\Theta_i(t)} \Pr\left(\{\theta_j(t)\}_{j\in\mathcal{O}(t)} | \{\zeta_l(t-1)\}_{l\in\Theta_i(t)}, \zeta_i(t-1) = x\right) \times \Pr\left(\{\zeta_l(t-1)\}_{l\in\Theta_i(t)} | \zeta_i(t-1) = x\right).
\end{aligned}$$

Given $\{\zeta_l(t-1)\}_{l\in\Theta_i(t)} \cup \{\zeta_i(t-1)\}$, $\{\theta_j(t)\}_{j\in\mathcal{O}(t)}$ are independent. We thus have

$$\begin{aligned}
&\Pr\left(\{\theta_j(t)\}_{j\in\mathcal{O}(t)} | \{\zeta_l(t-1)\}_{l\in\Theta_i(t)}, \zeta_i(t-1) = x\right) \\
&= \prod_{j\in\mathcal{O}(t)} \Pr\left(\theta_j(t) | \{\zeta_l(t-1)\}_{l\in\Theta_i(t)}, \zeta_i(t-1) = x\right) \\
&= \prod_{j\in\mathcal{O}(t)} \Pr\left(\theta_j(t) | \{\zeta_l(t-1)\}_{l\in\partial_j^+(t-1)\backslash\{i\}}, \zeta_i(t-1) = x\right).
\end{aligned}$$

Based on Assumption 9,

$$\Pr\left(\{\zeta_l(t-1)\}_{l\in\Theta_i} | \zeta_i(t-1) = x\right) = \prod_{l\in\{\Theta_i(t)\}} \Pr\left(\zeta_l(t-1)\right).$$

Therefore,

$$\begin{aligned}\Pr\big(\underline{Y}(t)|\zeta_i(t-1)=x\big) &= \sum_{\theta_j(t),\ j\in\mathcal{O}(t)} \delta(\{Y_j(t),\theta_j(t)\}_{j\in\mathcal{O}(t)}) \\ \times \sum_{\zeta_l(t-1)} \prod_{j\in\mathcal{O}(t)} \Pr\big(\theta_j(t)|\{\zeta_l(t-1)\}_{l\in\partial_j^+(t-1)}\backslash\{i\},\zeta_i(t-1)=x\big) &\times \prod_{l\in\{\Theta_i(t)\}} \Pr\big(\zeta_l(t-1)\big).\end{aligned} \tag{E.22}$$

Denote $\{x_j\}_{j\in\mathcal{O}(t)}$ as a realization of $\{\theta_j(t)\}_{j\in\mathcal{O}(t)}$ and $\{y_l\}_{l\in\Theta_i(t)}$ as a realization of $\{\zeta_l(t-1)\}_{l\in\Theta_i(t)}$. Then,

$$\begin{aligned}\Pr\big(\underline{Y}(t)|\zeta_i(t-1)=x\big) &= \sum_{\{x_j\}_{j\in\mathcal{O}(t)}} \delta(\{Y_j(t),x_j\}_{j\in\mathcal{O}(t)}) \\ \times \sum_{\{y_l\}_{l\in\Theta_i(t)}} \prod_{j\in\mathcal{O}(t)} \Pr\big(x_j|\{y_l\}_{l\in\partial_j^+(t-1)\backslash\{i\}},\zeta_i(t-1)=x\big) &\times \prod_{l\in\{\Theta_i(t)\}} \Pr\big(\zeta_l(t-1)=y_l\big).\end{aligned}$$

Denote

$$\begin{aligned}&\rho\big(\{x_j\}_{j\in\mathcal{O}(t)},x\big) \\ &= \sum_{\{y_l\}_{l\in\Theta_i(t)}} \prod_{j\in\mathcal{O}(t)} \Pr\big(x_j|\{y_l\}_{l\in\partial_j^+(t-1)\backslash\{i\}},\zeta_i(t-1)=x\big) \times \prod_{l\in\Theta_i(t)} \Pr\big(\zeta_l(t-1)=y_l\big).\end{aligned} \tag{E.23}$$

Then,

$$\Pr\big(\underline{Y}(t)|\zeta_i(t-1)=x\big) = \sum_{x_j\in\mathcal{X},j\in\mathcal{O}(t)} \delta(\{Y_j(t),x_j\}_{j\in\mathcal{O}(t)})\rho\big(\{x_j\}_{j\in\mathcal{O}(t)},x\big). \tag{E.24}$$

Based on Assumption 9, we can further simplify (E.24). Consider node i, $\underline{Y}(t)$ can be split into $\underline{Y}_{i,1}(t)$ and $\underline{Y}_{i,2}(t)$, where $\underline{Y}_{i,1}(t)$ is the observations of the set $\mathcal{O}(t)\cap\partial_i^+(t-1)$, and $\underline{Y}_{i,2}(t)$ is the observations of the rest of the nodes. Note that $\underline{Y}_{i,1}(t)\cup\underline{Y}_{i,2}(t)=\underline{Y}(t)$ and $\underline{Y}_{i,1}(t)\cap\underline{Y}_{i,2}(t)=\varnothing$.

Lemma 20. *Conditioned on $\underline{Y}_{i,1}(t)$, $\zeta_i(t-1)$ is independent of $\underline{Y}_{i,2}(t)$.*

Proof. To show Lemma 20, we use the structured belief network as defined in [247]. $\zeta_j(t)$ is the random variable associated with node j. Note that $Y_j(t)$ is the test result of $\zeta_j(t)$ on day t. Now,

we consider $j \in \big(\mathcal{O}(t)\backslash(\mathcal{O}(t) \cap \partial_i^+(t-1))\big)$. By [247, Theorems 1] and Bayes ball algorithm defined in [248, Section 2], we investigate the following two cases.

(i) For any $j \in \big(\mathcal{O}(t)\backslash(\mathcal{O}(t) \cap \partial_i^+(t-1))\big)$ with $Y_j(t) = 1$, the corresponding state $\zeta_j(t)$ is determined (which is I). Then, probabilities conditioning on $Y_j(t)$ is equivalent to (equal to) probabilities conditioning on $\zeta_j(t)$. By Bayes ball algorithm [247, 248], the information (the ball) is blocked at $\zeta_j(t)$ when the information (the ball) reaches $\zeta_j(t)$, which implies the information (the ball) can not reach $\zeta_i(t-1)$.

(ii) For any $j \in \big(\mathcal{O}(t)\backslash(\mathcal{O}(t) \cap \partial_i^+(t-1))\big)$ with $Y_j(t) = 0$, $\zeta_j(t)$ is not determined. By Bayes ball algorithm [247, 248], when the information (the ball) reaches $\zeta_j(t)$, it can traverse $Y_j(t)$ when blocking $Y_j(t)$ (conditioning on $Y_j(t)$). However, by Assumption 9, $\zeta_i(t-1)$ and $\zeta_j(t-1)$ are independent, so any path between $\zeta_i(t-1)$ and $\zeta_j(t-1)$ is blocked, including the path $\zeta_j(t-1) \leftrightarrow \zeta_j(t) \leftrightarrow Y_j(t) \leftrightarrow \zeta_j(t) \leftrightarrow \zeta_i(t-1)$. Thus, the information (the ball) can not reach $\zeta_i(t-1)$.

A simple example is given in Figure E.1: Let $Y_1(t) = 0$ and $Y_2(t) = 1$. Given $Y_1(t)$ and $Y_2(t)$, $Y_3(t)$ is independent of $\zeta_1(t-1)$. □

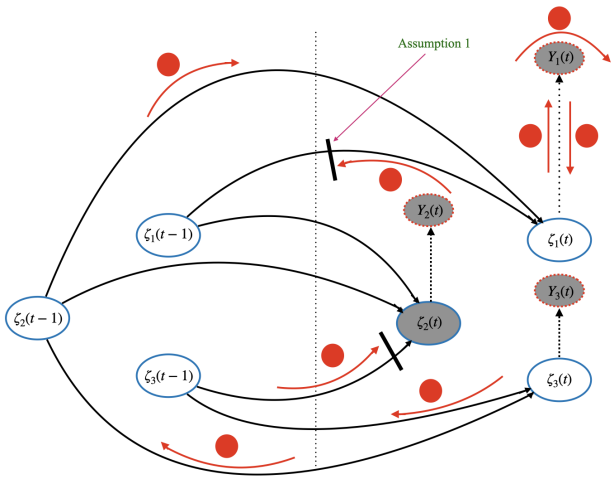


Figure E.1: Bayes ball algorithm in the network of 3 nodes. The terms on which we have conditioning are shaded gray and are equivalently blocked.

From Lemma 20,

$$e_x^{(i)}(t-1) = \Pr\big(\zeta_i(t-1) = x|\underline{Y}(t)\big) = \Pr\big(\zeta_i(t-1) = x|\underline{Y}_{i,1}(t)\big). \tag{E.25}$$

We simplify (E.24) based on Lemma 20 or (E.25). From (E.25), denote the observations of nodes in $\partial_i^+(t-1)$ as $\underline{Y}_{\partial_i^+}(t)$, $\underline{Y}_{\partial_i^+}(t)$ is independent of $\zeta_i(t-1)$. Denote $\Psi_i(t) = \mathcal{O}(t) \cap \partial_i^+(t-1)$. Then, We can replace $\mathcal{O}(t)$ by $\Psi_i(t)$ in (6.16). Subsequently, denote $\Phi_i(t) = \{j|j \in \partial_k^+(t-1), k \in \Psi_i(t)\}\backslash\{i\}$, and we can replace $\Theta_i(t)$ by $\Phi_i(t)$ in (E.21). Thus, from (E.23) and (E.24), we respectively have

$$\rho\big(\{x_j\}_{j\in\Psi_i(t)}, x\big) = \sum_{\{y_l\}_{l\in\Phi_i(t)}} \prod_{j\in\Psi_i(t)} \Pr\big(x_j|\{y_l\}_{l\in\partial_j^+(t-1)\backslash\{i\}}, \zeta_i(t-1) = x\big) \times \prod_{l\in\Phi_i(t)} \Pr\big(\zeta_l(t-1) = y_l\big) \tag{E.26}$$

and

$$\Pr\big(\{Y_j(t)\}_{j\in\Psi_i(t)}|\zeta_i(t-1) = x\big) = \sum_{x_j\in\mathcal{X}, j\in\Psi_i(t)} \delta(\{Y_j(t), x_j\}_{j\in\Psi_i(t)})\rho\big(\{x_j\}_{j\in\Psi_i(t)}, x\big) \tag{E.27}$$

which give the desired result (6.17).

E.8. A Simple Example for Algorithm 8

In this section, we give a simple example to illustrate the ideas and steps of Algorithm 8. Besides, we compare our proposed algorithm (Algorithm 8) with the Naive approach discussed in Remark 28. Consider a simple network with three nodes. Node 1 has an edge with node 2, and node 2 has an edge with node 3 (see Fig E.2). Nodes 1 and 3 are symmetric and statistically identical, and node 2 has a higher degree.

Consider the following situation: on the initial day (day 0), assume that nodes 1 and 3 are susceptible, and node 2 is infectious. On day 1, let node 2 be tested. Recall that we define the posterior probability vectors at the end of every day, and the prior probability vectors at the beginning of every day. Nodes' states change at the beginning of every day and testing is also done at the beginning of every day. Let the initial belief, i.e., the posterior probability $\underline{w}_i(0)$ and the prior probability

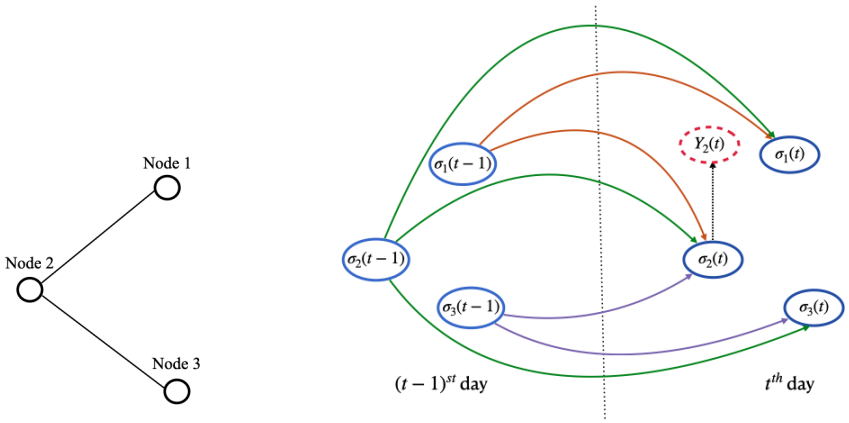


Figure E.2: The original graph (left). The graphical model of states and observations (right)

$\underline{u}_i(0)$ on day 0, of the nodes be

$$\underline{w}_i(0) = [1/3, 0, 0, 2/3], \quad i = 1, 2, 3$$
$$\underline{u}_i(0) = [1/3, 0, 0, 2/3], \quad i = 1, 2, 3.$$

Day 0: No tests on day 0, the prior probabilities are updated by the forward step (**Step 2** in Algorithm 8).

Day 1: By **Step 2** in Algorithm 8, we have

$$\underline{u}_1(1) = [0.3144, 0.0373, 0.0633, 0.5849]$$
$$\underline{u}_2(1) = [0.3559, 0.0676, 0.0633, 0.5131]$$
$$\underline{u}_3(1) = [0.3144, 0.0373, 0.0633, 0.5849].$$

After testing node 2, we know that node 2 is positive. We use the test result to infer the state of the nodes in prior times. In particular, we update the posterior probability on the day 0 ($\underline{w}_i(0)$).

Denoting the updated posterior probability as $\underline{e}_i(0)$, by Step 1 in Algorithm 8, we find

$$\underline{e}_1(0) = [0.3144, 0.0373, 0.0633, 0.5849]$$
$$\underline{e}_2(0) = [0.9615, 0.0385, 0.0, 0.0]$$
$$\underline{e}_3(0) = [0.3144, 0.0373, 0.0633, 0.5849].$$

We can now say that at the end of the day 0, node 2 was infectious with probability 0.9615 and it was in the latent state with probability 0.0385. Moreover, we see that the (posterior) infection probabilities of nodes 1 and 3 on day 0 have increased since they may have infected node 2 on day 0, i.e., $0.3517 = e_I^{(i)}(0) + e_L^{(i)}(0) > 1/3 = w_I^{(i)}(0) + w_L^{(i)}(0)$ with $i = 1, 3$. Next, we obtain the posterior probability on day 1. Recall that $\underline{w}_i(t)$ describes the posterior probability vector of node i at the end of day t. By Step 1 in Algorithm 8,

$$\underline{w}_1(1) = [0.4008, 0.0, 0.1268, 0.4724]$$
$$\underline{w}_2(1) = [0.90, 0.0, 0.10, 0.0]$$
$$\underline{w}_3(1) = [0.4008, 0.0, 0.1268, 0.4724].$$

One may wonder why the posterior probability is $[0.9, 0, 0.1, 0]$ rather than $[1, 0, 0, 0]$. This is because testing is done at the beginning of time t and the posterior probabilities are defined at the end of time slots t. The infected node may have recovered by the end of time $t = 1$ and this is reflected in the posterior probabilities computed.

Day 2: We can get the prior probability vectors on the day 2 by our forward update (making predictions):

$$\underline{u}_1(2) = [0.2883, 0.0, 0.1268, 0.5849]$$
$$\underline{u}_2(2) = [0.8135, 0.0, 0.1865, 0.0]$$
$$\underline{u}_3(2) = [0.2883, 0.0, 0.1268, 0.5849].$$

On the other hand, if we apply the naive updating rule defined in Remark 28, on day 2, we find

$$\underline{u}_1'(2) = [0.4074, 0.0896, 0.0948, 0.4082]$$
$$\underline{u}_2'(2) = [0.81, 0.0, 0.19, 0.0]$$
$$\underline{u}_3'(2) = [0.4074, 0.0896, 0.0948, 0.4082].$$

Recall that we use Assumption 9 in the proposed algorithm (Algorithm 8), and the Naive approach in Remark 28 does not have the backward step, so both approaches do not capture the correlations among nodes. By Monte Carlo simulations, the correlations among nodes are captured, and the nodes' probability vectors are approximated on day 2 as follows:

$$\underline{v}_1(2) = [0.3235, 0.0976, 0.0196, 0.5593]$$
$$\underline{v}_2(2) = [0.7244, 0, 0.2756, 0]$$
$$\underline{v}_3(2) = [0.3158, 0.1072, 0.019, 0.558]$$

which yields the following comparison of the incurred estimation errors:

$$0.4342 = \sum_{i=1}^{3} ||\underline{u}_i(2) - \underline{v}_i(2)|| < \sum_{i=1}^{3} ||\underline{u}_i'(2) - \underline{v}_i(2)|| = 0.5018.$$

The lef- hand side shows the estimation error under our proposed backward-forward update and the right-hand side shows the estimation error under the naive approach.

E.9. Delay of Testing Results

One can extend the framework to a more realistic case where testing results are not able to be obtained on the same day but will be obtained after a delay a. In other words, if nodes are tested on day $t - a$, the test results are provided on day t. The extended framework is summarized as follows.

On day t, before getting the test results of day $t - a$, the algorithm knows the following in-

formation: (i) the network topology from day $t-a-1$ to day t, i.e., $\mathcal{G}(t-a-1),\cdots,\mathcal{G}(t)$ (it is affected by the past actions); (ii) the posterior probability of nodes on day $t-a-1$, $\left\{\underline{w}_i(t-a-1)\right\}_{i\in\mathcal{G}(t-a-1)}$; and (iii) and the prior probability vectors of nodes from day $t-a$ to day t, i.e., $\left\{\underline{u}_i(t-a)\right\}_{i\in\mathcal{G}(t-a)},\cdots,\left\{\underline{u}_i(t)\right\}_{i\in\mathcal{G}(t)}$.

After getting the test results on day $t-a$, we can obtain the updated posterior probability vectors on day $t-a-1$, and the posterior probability vectors on day $t-a$, i.e., $\left\{\underline{e}_i(t-a-1)\right\}_{i\in\mathcal{G}(t-a-1)}$, and $\left\{\underline{w}_i(t-a)\right\}_{i\in\mathcal{G}(t-a)}$, by **Step 1** in Algorithm 8.

Based on $\{\underline{w}(t-a)\}_{i\in\mathcal{G}(t-a)}$, by **Step 2** in Algorithm 8, we update the prior probability from day $t-a+1$ to day t, and obtain the prior probability on day $t+1$, i.e., $\{\underline{u}_i(t+1)\}_{i\in\mathcal{G}(t+1)}$.

Repeating the process, we can compute the estimated probability vectors of nodes and apply the exploration and exploitation policies.

E.10. Proof of Theorem 22

Step 1: Preliminaries.

We divide the distributions of initial infectious nodes into two complementary events:

$$\begin{aligned}\mathcal{I}_1 =&\{\text{No node is infectious}\}\\ \mathcal{I}_2 =&\mathcal{I}_1^c.\end{aligned}$$

Let N be sufficiently large,

$$\begin{aligned}&\Pr\{\mathcal{I}_1\}=(1-1/N)^N\approx 1/e\\ &\Pr\{\mathcal{I}_2\}\approx 1-1/e.\end{aligned}$$

In event $\mathcal{I}_1$, since there is no infection on the initial day, then no node is infectious in the future, i.e., the true probability of nodes $v_I^{(i)}(t)=0$ for all $i\in\mathcal{V}(t)$ and $t\geq 1$.

Note that in Example 3, each node can be in one of two states, S and I. The transmission probability $\beta = 1$. So, on day t, the probability of node i being in state I includes the infection of node i on day $t-1$, and the infection from its neighbors. Then, based on (6.13), we have

$$\begin{aligned} u_I^{(i)}(t) =& w_I^{(i)}(t-1) + \{1 - w_I^{(i)}(t-1)\}\Big\{1 - \big(1 - w_I^{(i-1)}(t-1)\big)\big(1 - w_I^{(i+1)}(t-1)\big)\Big\} \\ =& 1 - \{1 - w_I^{(i-1)}(t-1)\}\{1 - w_I^{(i)}(t-1)\}\{1 - w_I^{(i+1)}(t-1)\}. \end{aligned} \tag{E.28}$$

For convention, we assume that nodes 0 and $N+1$ are two virtual nodes with no probability of infection, i.e., $u_I^{(0)}(t) = u_I^{(N+1)}(t) = 0$ for all t, and no tests are applied to these two nodes all the time.

Since $w_I^{(i)}(t+1), w_I^{(i-1)}(t+1)$, $w_I^{(i+1)}(t+1) \in [0,1]$, then from (E.28),

$$u_I^{(i)}(t) \geq 1 - 1 \times (1 - w_I^{(i)}(t-1)) \times 1 = w_I^{(i)}(t-1). \tag{E.29}$$

Thus, by symmetry over $w_I^{(i-1)}(t-1)$, $w_I^{(i)}(t-1)$, and $w_I^{(i+1)}(t-1)$ we get the inequality

$$u_I^{(i)}(t) \geq \max\{w_I^{(i-1)}(t-1), w_I^{(i)}(t-1), w_I^{(i+1)}(t-1)\}. \tag{E.30}$$

Step 2: Consider the computation of $\{\underline{u}_i(t)\}_i$ based on (6.13) $\big($equivalently (E.28)$\big)$ under event $\mathcal{I}_1$.

Recall that $B(t) = 1$ for all t. On any day t, if node i_0 is tested, then the result is negative, and $w_I^{(i_0)}(t) = 0$, and

$$w_I^{(i)}(t) = u_I^{(i)}(t) \text{ for all } i \neq i_0. \tag{E.31}$$

In (E.30), at most one of $w_I^{(i-1)}(t-1)$, $w_I^{(i)}(t-1)$, and $w_I^{(i+1)}(t-1)$ is updated to 0. We first prove the following facts.

Fact 1. $u_I^{(i)}(t) \geq \frac{1}{N}$ for all t. On any day t, $w_I^{(i)}(t) \geq \frac{1}{N}$ with $i \neq i_0$, where i_0 is the index of node tested on day t.

Proof. We prove **Fact 1** by mathematical induction. On the initial day, by model assumption in Example 3, $u_I^{(i)}(0) = \frac{1}{N}$ for all i. Then, if node i_0 is tested, then as mentioned above, $w_I^{(i_0)}(0) = 0$, and by (E.31) $w_I^{(i)}(0) = u_I^{(i)}(0) = 1/N$ for all $i \neq i_0$.

Suppose **Fact 1** holds for all $\tau \leq t-1$. Now, we consider $\tau = t$. From (E.30), we have $u_I^{(i)}(t) \geq \max\{w_I^{(i-1)}(t-1), w_I^{(i)}(t-1), w_I^{(i+1)}(t-1)\} \geq 1/N$. Then, if node i_0 is tested, we have $w_I^{(i_0)}(t) = 0$, and then by (E.31), $w_I^{(i)}(t) = u_I^{(i)}(t) \geq 1/N$ for all $i \neq i_0$. □

Fact 2. If node i has not been tested up to day t, then $u_I^{(i)}(t)$ tends to 1 as $t \to \infty$.

Proof. Since node i is not tested from the initial day to day t, then

$$w_I^{(i)}(\tau) = u_I^{(i)}(\tau),\ \tau \leq t. \tag{E.32}$$

Note that at most one of its neighbors is tested on the day t. By (E.28) and **Fact 1**,

$$u_I^{(i)}(t) \geq 1 - (1-1/N)(1 - w_I^{(i)}(t-1)) = 1 - (1-1/N)(1 - u_I^{(i)}(t-1)),,$$

which implies

$$(1-1/N)(1 - u_I^{(i)}(t-1)) \geq 1 - u_I^{(i)}(t),$$

which implies

$$1 - u_I^{(i)}(t) \leq (1-1/N)^t(1 - u_I^{(i)}(0)) = (1-1/N)^{t+1}.$$

Letting $t \to \infty$ completes the proof. □

Fact 3. If node i is not tested on day $t-1$, then

$$u_I^{(i)}(t) \geq w_I^{(i)}(t-1) + \frac{1}{N}(1 - w_I^{(i)}(t-1))w_I^{(i)}(t-1).$$

Proof. By **Fact 3**, if node i is not tested on day $t-1$, then $w_I^{(i)}(t-1) > 0$. From (E.28), by some algebra,

$$\begin{aligned} u_I^{(i)}(t) =& w_I^{(i)}(t-1) + (1 - w_I^{(i)}(t-1))(w_I^{(i-1)}(t-1) + w_I^{(i+1)}(t-1) - w_I^{(i-1)}(t-1)w_I^{(i+1)}(t-1)) \\ =& (1+\epsilon) w_I^{(i)}(t-1) \end{aligned}$$

where

$$\epsilon = \frac{1 - w_I^{(i)}(t-1)}{w_I^{(i)}(t-1)} \times (w_I^{(i-1)}(t-1) + w_I^{(i+1)}(t-1) - w_I^{(i-1)}(t-1)w_I^{(i+1)}(t-1)).$$

Note that at most one of the neighbors of the node i is tested on the day $t-1$, then

$$\begin{aligned} \frac{1 - w_I^{(i)}(t-1)}{w_I^{(i)}(t-1)} \geq& 1 - w_I^{(i)}(t-1) \\ w_I^{(i-1)}(t-1) + w_I^{(i+1)}(t-1) - w_I^{(i-1)}(t-1)w_I^{(i+1)}(t-1) \geq& \max\{w_I^{(i-1)}(t-1), w_I^{(i+1)}(t-1)\}. \end{aligned}$$

From **Fact 1**, $\max\{w_I^{(i-1)}(t-1), w_I^{(i+1)}(t-1)\} \geq 1/N$. Thus, $\epsilon \geq (1 - w_I^{(i)}(t-1)) \times 1/N$. Hence, $u_I^{(i)}(t) \geq w_I^{(i)}(t-1) + \frac{1}{N}(1 - w_I^{(i)}(t-1))w_I^{(i)}(t-1)$. □

Since we consider all possible sequential testing policies, then we divide all nodes into two sets

$$\begin{aligned} \mathcal{S}_1(t) =& \{\text{nodes that have not been tested up to day } t\} \\ \mathcal{S}_2(t) =& \mathcal{S}_1^c(t). \end{aligned}$$

In the following proof, let $t \to \infty$. By **Fact 2**, $u_I^{(i)}(t) \to 1$ if $i \in \mathcal{S}_1(t)$. Next, we focus on the set $\mathcal{S}_2(t)$. Denote the index of node which is tested on the day $t-1$ as $i_0(t)$. By **Fact 1**, $w_I^{(i)}(t-1) \geq 1/N$ for all $i \neq i_0(t)$. Then, we define

$$\begin{aligned} \mathcal{S}_{21}(t) =& \{i | 1/N \leq w_I^{(i)}(t-1) < 1 - 1/N\} \\ \mathcal{S}_{22}(t) =& \{i | 1 - 1/N \leq w_I^{(i)}(t-1)\}. \end{aligned}$$

Thus, we have $\mathcal{S}_2(t) = \mathcal{S}_{21}(t) \cup \mathcal{S}_{22}(t) \cup \{i_0(t)\}$. Due to the equivalence of norms, without loss of generality, we consider L_1 norm in the rest of the proof.

(i) If $i \in \mathcal{S}_1(t)$, then $u_I^{(i)}(t) \to 1$. Thus $\|\underline{u}_i(t) - \underline{v}_i(t)\|_1 \to 2$.

(ii) If $i \in \mathcal{S}_{21}(t)$, then $\|\underline{u}_i(t) - \underline{v}_i(t)\|_1 \geq \|\underline{u}_i(t-1) - \underline{v}_i(t-1)\|_1 + \frac{2(N-1)}{N^3}$. In fact, since $i \in \mathcal{S}_{21}(t)$, then $i \neq i_0(t)$, thus by (E.31) and **Fact 3**,

$$u_I^{(i)}(t) \geq u_I^{(i)}(t-1) + \frac{1}{N}(1 - w_I^{(i)}(t-1))w_I^{(i)}(t-1).$$

Note that N is sufficiently large, so $1/N < 1/2 < 1 - 1/N$. If $x \in [1/N, 1-1/N)$, then the fuction $f(x) = x(1-x)$ has the minimum value $\frac{N-1}{N^2}$ when $x = 1/N$. Thus,

$$u_I^{(i)}(t) \geq u_I^{(i)}(t-1) + \frac{N-1}{N^3}. \tag{E.33}$$

Recall that $v_I^{(i)}(t) = 0$ and $v_S^{(i)}(t) = 1$ for all t, and $u_I^{(i)}(t) + u_S^{(i)}(t) = 1$, then

$$\|\underline{u}_i(t) - \underline{v}_i(t)\|_1 = |u_I^{(i)}(t) - v_I^{(i)}(t)| + |u_S^{(i)}(t) - v_S^{(i)}(t)| = 2|u_I^{(i)}(t) - v_I^{(i)}(t)|. \tag{E.34}$$

From (E.33),

$$\begin{aligned}\|\underline{u}_i(t) - \underline{v}_i(t)\|_1 &= 2|u_I^{(i)}(t) - v_I^{(i)}(t)| \geq 2|u_I^{(i)}(t-1) + \frac{N-1}{N^3} - v_I^{(i)}(t-1)| \\ &\geq 2|u_I^{(i)}(t-1) - v_I^{(i)}(t-1)| + \frac{2(N-1)}{N^3} = \|\underline{u}_i(t-1) - \underline{v}_i(t-1)\|_1 + \frac{2(N-1)}{N^3}.\end{aligned}$$

(iii) If $i \in \mathcal{S}_{22}(t)$, then node i is not tested on day t, thus from (E.30), $u_I^{(i)}(t) \geq w_I^{(i)}(t-1) = 1 - 1/N$. Thus, by (E.34), $\|\underline{u}_i(t) - \underline{v}_i(t)\|_1 \geq \frac{2(N-1)}{N}$.

Since we consider N sufficiently large, then we can prove the following lemma.

Lemma 21. $\lim_{t\to\infty} \mathcal{S}_{21}(t) = \varnothing$.

Proof. We first prove the following Claims.

Claim 1. If (i) $u_I^{(i-1)}(t) \geq 1-1/N$ and node $i-1$ is not tested on day t, or (ii) $u_I^{(i+1)}(t) \geq 1-1/N$ and node $i+1$ is not tested on day t, or (iii) $u_I^{(i-1)}(t) \geq 1-1/N$ and $u_I^{(i+1)}(t) \geq 1-1/N$, then $u_I^{(i)}(t+1) \geq 1-1/N$.

Proof. By (E.28) and (E.31), we can derive $u_I^{(i)}(t+1) \geq 1-1/N$ directly. □

Claim 2. No node can stay in $\mathcal{S}_{21}(t)$ for successive $\lceil N^3/(N-1) \rceil$ days.

Proof. if node i stays in $\mathcal{S}_{21}(t)$ for successive $\lceil N^3/(N-1) \rceil$ days, i.e., from day τ to day $\tau + \lceil N^3/(N-1) \rceil$, then by (E.33), $u_I^{(i)}(\tau + \lceil N^3/(N-1) \rceil) > 1$, which contradicts with $u_I^{(i)}(t) \leq 1$ for all t. □

Now, we prove the lemma by contradiction. Based on **Claim 2**, assume there exists at least one j and an increasing sequence $\{t_i\}_{i=0}^{\infty}$ with $\lim_{n\to\infty} t_n = \infty$, such that $j \in \mathcal{S}_{21}(t_i)$ for all $\{t_i\}_{i=0}^{\infty}$.

For some i, node j is in $\mathcal{S}_{22}(t_i-1)$ on day t_i-1, and node j is in $\mathcal{S}_{21}(t_i)$ on day t_i. In other words, $u_I^{(j)}(t_i) < 1-1/N \leq u_I^{(j)}(t_i-1)$. From (E.28) and **Calim 1**, $u_I^{(j)}(t_i) < 1-1/N \leq u_I^{(j)}(t_i-1)$ holds only because node j is tested on day t_i-1, and all of its neighbors (i.e., nodes $j-1$, $j+1$) have $u_I^{(j-1)}(t_i-1) < 1-1/N$ and $u_I^{(j+1)}(t_i-1) < 1-1/N$. However, since $u_I^{(j)}(t_i-1) \geq 1-1/N$ and node j is tested on day t_i-1, then by **Claim 1**, $u_I^{(j-1)}(t_i) \geq 1-1/N$ and $u_I^{(j+1)}(t_i) \geq 1-1/N$. Subsequently, by **Claim 1**, we have $u_I^{(j)}(t_i+1) \geq 1-1/N$. Thus, on day t_i+1, at least one of its neighbors, say $j-1$, has $u_I^{(j-1)}(t_i+1) \geq 1-1/N$. By **Claim 1**, node j never fall into $\mathcal{S}_{21}(t)$ for $t \in \{t_{i+1}, t_{i+2}, \cdots\}$, which contradicts with the assumption. □

From Lemma 21, when $t \to \infty$, we have $|\mathcal{S}_1(t)| = \Theta(N)$ or $|\mathcal{S}_{22}(t)| = \Theta(N)$. Thus, $\sum_{i=1}^{N} \|\underline{u}_i(t) - \underline{v}_i(t)\|_1 = \Theta(N)$.

Step 3: Consider the computation of $\{\underline{u}_i(t)\}_i$ based on Algorithm 8.

In this step, we consider a specific testing policy: We test node i on day k, where $k \equiv i-1 (mod\ M)$ for all $1 \leq i \leq M$.

In event $\mathcal{I}_2$, since the transmission probability $\beta = 1$, then all nodes are infected at most N days because there is no recovery. Thus, no node with a positive testing result is repeatedly tested. So in at most $2N$ days, all nodes are infectious, and the algorithm finds all infected nodes, so $\underline{u}_i(t) = \underline{v}_i(t)$, $t \geq 2N$.

In event $\mathcal{I}_1$, whenever a node is tested, it is negative. Node 1 is tested on day 0, the result is negative. On day 1, node 2 is tested and the result is negative. By backward updating, since $\beta = 1$ and no recovery, then nodes 1&3 are inferred to be in state S on day 0. Since node 2 is in state S on day 1. Then, node 1 is inferred in state S on days 0 and 1.

Assume that nodes $1, 2, \cdots, k-2$ are inferred to be in state S by day $k-1$. Now, we day k, where $k \leq N$. On day k, node $k-1$ is tested negative, hence by backward updating, nodes $k-2$ and k are inferred to be in state S on day $k-1$. By the testing result of node $k-1$ on day k, nodes $1, 2, \cdots, k-1$ are inferred in state S by day k. By induction, after N days, it clears every node, so $\underline{u}_i(t) = \underline{v}_i(t)$, $t \geq N$.

From **Steps 1∼3**, we complete the proof.

E.11. α-linking Backward Updating

E.11.1. Complexity Reduction

Let $\{x_j\}_{j\in\mathcal{O}(t)}$ be a realization of $\{\theta_j(t)\}_{j\in\mathcal{O}(t)}$ and $\{y_l\}_{l\in\Theta_i(t)}$ be a realization of $\{\zeta_l(t-1)\}_{l\in\Theta_i(t)}$. Let node i have state x in day $t-1$. Consider one node $k \in \partial_j^+(t-1)\backslash\{i\}$ and the probability

$$\Pr\left(x_j|\{y_l\}_{l\in\partial_j^+(t-1)\backslash\{i\}}, x\right), j \in \Psi_i(t).$$

Since node k is not infectious if $y_k = L$, $y_k = R$ or $y_k = S$, then the probability above remains the same no matter whether $y_k = L$, $y_k = R$ or $y_k = S$.

Thus, we introduce a new state, denoted by E, to be a replacement of $\{L, R, S\}$, and

$$\Pr\big(y_k = E\big) = \sum_{x\in\{L,R,S\}} \Pr\big(y_k = x\big).$$

Next, denote $\mathcal{X}' = \{I, E\}$. Equation (6.17) can be re-written as follows:

$$\begin{aligned}
&\Pr\big(\{Y_j(t)\}_{j\in\Psi_i(t)}|\zeta_i(t-1)=x\big)\\
&= \sum_{\{x_j\}_{j\in\Psi_i(t)}} \prod_{j\in\Psi_i(t)} \Pr\big(Y_j(t)|\theta_j(t)\big) \times \sum_{\{y_l\}_{l\in\Theta_i(t)}} \prod_{j\in\Psi_i(t)} \mathtt{P}_j\big(x_j|\{y_l\}_{l\in\partial_j^+(t-1)\setminus\{i\}}, x\big)\\
&\times \prod_{z_l\in\mathcal{X}', l\in\Theta_i(t)} \Pr\big(\zeta_l(t-1)=z_l\big),
\end{aligned} \tag{E.35}$$

with reducing the computation complexity. Subsequently, $\underline{e}_i(t-1)$ in (6.16) can be calculated by (E.35) directly.

E.11.2. α-linking Backward Updating

In the backward step, the computation complexity is large even in (E.35). To further reduce the complexity in (E.35), one way is to update the posterior probability $\underline{e}_i(t)$ in a sparser network. Now, we define *α-linking Backward Updating* as follows:

(i) We generate a subgraph $\mathcal{G}_\alpha(t)$ based on the pre-determined graph $\mathcal{G}(t)$: Suppose that each edge (in $\mathcal{G}(t)$) exists with probability α, $0 \le \alpha \le 1$. If $\alpha = 1$, then $\mathcal{G}_\alpha(t) = \mathcal{G}(t)$; if $\alpha = 0$, then $\mathcal{G}_\alpha(t)$ is a graph with no edges.

(ii) Backward updating in $\mathcal{G}_\alpha(t)$: Similar with $\partial_i(t)$, $\Psi_i(t)$, $\Phi_i(t)$ and $\Theta_i(t)$, we define $\partial_{i,\alpha}(t)$, $\Psi_{i,\alpha}(t)$, $\Phi_{i,\alpha}(t)$ and $\Theta_{i,\alpha}(t)$ on graph $\mathcal{G}_\alpha(t)$, respectively. Subsequently, replace $\partial_i(t)$, Ψ_i, $\Phi_i(t)$ and $\Theta_i(t)$ by $\partial_{i,\alpha}(t)$, $\Psi_{i,\alpha}(t)$, $\Phi_{i,\alpha}(t)$ and $\Theta_{i,\alpha}(t)$ in (E.35), respectively.

E.12. Proof of Theorem 23

Step 1. Preliminaries.

In Example 4, $\beta = 1$, $\lambda = 0$, and $\gamma = 0$, there is no recovery and we assume no latent state.

Based on (6.9), the expression of rewards $\hat{r}_i(t)$ for every node is given as follows. If node i has two neighbors (without quarantine)

$$\hat{r}_i(t) = u_S^{(i-1)}(t)(1 - u_I^{(i-2)}(t))u_I^{(i)}(t) + u_S^{(i+1)}(t)(1 - u_I^{(i+2)}(t))u_I^{(i)}(t). \tag{E.36}$$

If node i only has one neighbor, then

$$\hat{r}_i(t) = u_S^{(i+d)}(t)\big(1 - u_I^{(i+2d)}(t)\big)u_I^{(i)}(t),\ d \in \{-1, 1\}. \tag{E.37}$$

For simplicity, we introduce artificial nodes $-1, 0, N+1, N+2$ with $u_I^{(-1)}(t) = u_I^{(0)}(t) = u_I^{(N+1)}(t) = u_I^{(N+2)}(t) = 0$ for all t, and these 4 nodes are never tested.

Step 2. The RbEx policy.

Under the RbEx policy, the algorithm always tests the nodes with maximum rewards. Let an infectious node be found, for the first time, on day aN, where a is a positive real number. Note that until the first infected node is found, in any application of the RbEx policy, $u_I^{(i)}(t)$ is the same for any given i, and hence $\hat{r}_i(t)$ is also the same. So, a is the same for any application of the RbEx policy. Recall that in Example 4, nodes that are tested positive will be isolated. The cumulative infections are at least $\min\{aN, N\}$ in the end.

Step 3. Consider the exploration process of the specific exploration policy.

Recall that from **Step 2**, an infectious node is found, for the first time, by the RbEx policy with budget 10 tests on day aN. Under the specific defined exploration policy, we can choose a specific b' with $b' < a$, such that no infectious node is tested by the RbEx policy with budget 9 tests before and including day $t = b'N$.

We know that on day τ, nodes $1, 2, \cdots, \tau$ are infectious since $\beta = 1$. Note that one test is applied

to exploration (randomly choice) every day, so with probability

$$\prod_{\tau'=1}^{\tau}(1-\frac{\tau'}{N}), \tag{E.38}$$

no infectious node is explored from the initial day to day τ. Then, with probability

$$\prod_{\tau'=1}^{\tau-1}(1-\frac{\tau'}{N})\cdot\frac{\tau}{N},$$

one infectious node is detected on day τ. Thus, with probability

$$\sum_{\tau=1}^{t}\prod_{\tau'=1}^{\tau-1}(1-\frac{\tau'}{N})\cdot\frac{\tau-1}{N}, \tag{E.39}$$

one infectious node is tested by the exploration process on day τ ($\tau \leq t$), and this node is not the new infectious one on day τ, i.e., has index τ. The probability defined in (E.39) increases with t when N is fixed, and it can be close to 1 when t is close to N. Therefore, We can choose proper parameters b' and N such that the probability defined in (E.39) is larger than or equal to p_0. In particular, if N is large, we can choose a relatively small b'. In Theorem 23, we set $p_0 \geq 99/100$.

Let the infectious node detected (for the first time) by the exploration process have index j on day t', where $t' \leq t$. As discussed above, node j is not the new infectious node on the day t', so we have $j < t'$. In other words, node $j+1$ must be infectious on day t' with a positive test result, i.e., $Y_j(t') = 1$. By Step 1 in Algorithm 8, the updated posterior probability of node j

$$e_I^{(j)}(t'-1) = 1, \quad e_S^{(j)}(t'-1) = 0. \tag{E.40}$$

Again, by Step 1 in Algorithm 8,

$$w_I^{(j-1)}(t') = w_I^{(j+1)}(t') = 1. \tag{E.41}$$

Then, by Step 2 in Algorithm 8,

$$u_I^{(j-2)}(t'+1) = u_I^{(j-1)}(t'+1) = u_I^{(j)}(t'+1) = u_I^{(j+1)}(t'+1) = u_I^{(j+2)}(t'+1) = 1. \tag{E.42}$$

Since j is detected and isolated on day t', then,

$$\hat{r}_j(t'+1) = 0. \tag{E.43}$$

By (E.37) and (E.42),

$$\begin{aligned} \hat{r}_{j-1}(t'+1) &= u_S^{(j-2)}(t'+1)\big(1 - u_I^{(j-3)}(t'+1)\big) = 0 \\ \hat{r}_{j+1}(t'+1) &= u_S^{(j+2)}(t'+1)\big(1 - u_I^{(j+3)}(t'+1)\big) = 0. \end{aligned} \tag{E.44}$$

By (E.36) and (E.42),

$$\begin{aligned} \hat{r}_{j-2}(t'+1) &= u_S^{(j-3)}(t'+1)\big(1 - u_I^{(j-4)}(t'+1)\big) \\ \hat{r}_{j+2}(t'+1) &= u_S^{(j+3)}(t'+1)\big(1 - u_I^{(j+4)}(t'+1)\big). \end{aligned} \tag{E.45}$$

Step 4.The exploitation process of the specific exploration policy.

We first study an extreme case where no tests are applied. In this case, denote the prior probability of node i on day τ as $U_I^{(i)}(\tau)$, which can be calculated by the following recursion:

$$U_I^{(i)}(\tau+1) = U_I^{(i)}(\tau) + U_S^{(i)}(\tau)\big(1 - (1 - U_I^{(i-1)}(\tau))(1 - U_I^{(i+1)}(\tau))\big). \tag{E.46}$$

Based on (E.46), recall that $U_I^{(i)}(0) = 0$ if $i \leq \frac{9N}{10}$, and $U_I^{(i)}(0) = \frac{10\epsilon}{N}$ if $\frac{9N}{10} < i \leq N$, then $U_I^{(i)}(\tau)$ increases over τ and is a function of ϵ. Then, given b', N and $t = b'N$, we can choose a small enough ϵ, denoted by $\epsilon(b', N)$, such that $U_I^{(i)}(2t) < \frac{1}{2}$ for all i. Since $U_I^{(i)}(\tau)$ increases over τ, then $U_I^{(i)}(\tau) < \frac{1}{2}$, $\tau \leq 2t$.

Now, we introduce the exploitation process. Let $t = b'N < \min\{\frac{9}{40}, a\}N$. There are at most $2t$

infectious nodes on day $2t$, i.e., nodes $1, 2, \cdots, 2t$. Since $t < \min\{\frac{9}{40}, a\}N$, then nodes with index from $9N/10-2t$ to N are in state S, which implies nodes with index from $9N/10-2t$ to N can never be tested positive before day $2t$. Thus, on any day $\tau \leq 2t$, for $9N/10-2t \leq i \leq N$, if node i is tested, and the testing result is negative. Recall that $U_I^{(i)}(\tau)$ in (E.46) is calculated without any negative testing results. Hence, $u_I^{(i)}(\tau) \leq U_I^{(i)}(\tau)$. Furthermore, with the condition $t = b'N < \min\{\frac{9}{40}, a\}N$, we can find a small enough $\epsilon(b', N)$, such that under the specific exploration policy,

$$u_I^{(i)}(\tau) < \frac{1}{2}, \quad \tau \leq 2t,\ 9N/10 - 2t \leq i \leq N. \tag{E.47}$$

In the rest, we divide the nodes in to 3 sets: $\mathcal{Q}_1 = \{i | i \leq 2t\}$, $\mathcal{Q}_2 = \{i | 2t < i < 9N/10 - 2t\}$, and $\mathcal{Q}_3 = \{i | 9N/10 - 2t \leq i \leq N\}$.

Fact 1. For $i \in \mathcal{Q}_1$ and $\tau \leq 2t$, $u_I^{(i)}(\tau) = 1$ or $u_I^{(i)}(\tau) = 0$.

Proof. If no test is applied to $\mathcal{Q}_1$, then $u_I^{(i)}(\tau) = 0$ for all $i \in \mathcal{Q}_1$.

On some day $\tau \leq 2t$, if one node with index $j \in \mathcal{Q}_1$ is tested positive on day $\tau - 1$, then by (E.42), $u_I^{(j-2)}(\tau) = u_I^{(j-1)}(\tau) = u_I^{(j)}(\tau) = u_I^{(j+1)}(\tau) = u_I^{(j+2)}(\tau) = 1$. In other words, if node j is tested positive on day $\tau - 1$, then node j, its neighbors, and neighbors of neighbors have probabilities of infections equal to 1 on day τ.

On some day τ, if node j is not tested positive on day $\tau - 1$, and neither of its neighbors and neighbors of neighbors are is not tested positive, then $u_I^{(j)}(\tau) = 1$ only when $u_I^{(j)}(\tau - 1) = 1$, or $u_I^{(j-1)}(\tau - 1) = 1$ or $u_I^{(j+1)}(\tau - 1) = 1$ since $\beta = 1$. Otherwise $u_I^{(j)}(\tau) = 0$. □

Fact 2. For $i \in \mathcal{Q}_1$ and $\tau \leq 2t$, $\hat{r}_i(\tau) = 1$ or $\hat{r}_i(\tau) = 0$.

Proof. If $u_I^{(i)}(\tau) = 0$, then $\hat{r}_i(\tau) = 0$ by (E.36) and (E.37).

Now, we consider $u_I^{(i)}(\tau) = 1$ in the following cases: (i) If both neighbors of node i are isolated, then $\hat{r}_i(\tau) = 0$. (ii) If one of neighbors of node i (for example, node $i - 1$) is isolated, then by (E.37), $\hat{r}_i(\tau) = 0$ when $u_I^{(i+1)}(\tau) = 1$, and $\hat{r}_i(\tau) = 1$ when $u_I^{(i+1)}(\tau) = 0$. (iii) If both neighbors

are not isolated, then $u_I^{(i-1)}(\tau-1)=1$ or $u_I^{(i+1)}(\tau-1)=1$, otherwise, $u_I^{(i)}(\tau)=0$. Since there is no recovery, then $u_I^{(i-1)}(\tau)=1$ or $u_I^{(i+1)}(\tau)=1$. By **Fact 1**, $u_I^{(j)}(\tau)=1$ or $u_I^{(j)}(\tau)=0$ when $j\in\mathcal{Q}_1$. If $u_I^{(i+1)}(\tau)=u_I^{(i-1)}(\tau)=1$, then $\hat{r}_i(\tau)=0$. If $u_I^{(i+1)}(\tau)=1$, then by (E.36), $\hat{r}_i(\tau)=0$ when $u_I^{(i-2)}(\tau)=1$, $\hat{r}_i(\tau)=1$ when $u_I^{(i-2)}(\tau)=0$. If $u_I^{(i-1)}(\tau)=1$, then by (E.36), $\hat{r}_i(\tau)=0$ when $u_I^{(i+2)}(\tau)=1$, $\hat{r}_i(\tau)=1$ when $u_I^{(i+2)}(\tau)=0$. □

From (E.47), for all $\tau\le 2t$ and $i\in\mathcal{Q}_3$, we have

$$\hat{r}_i(\tau)\le 2u_I^{(i)}(\tau)<1. \tag{E.48}$$

Note that only nodes in $\mathcal{Q}_1$ and $\mathcal{Q}_3$ may have positive probabilities of infections. For all $\tau\le 2t$ and $i\in\mathcal{Q}_2$, since $u_I^{(i)}(\tau)=0$, then $\hat{r}_i(\tau)=0$. Therefore, a node with a reward equal to 1 has the largest reward.

Recall that on day t', node j tested positive. From (E.45), nodes $j-2$ and $j+2$ have largest rewards $(=1)$ on day $t'+1$, which are exploited on day $t'+1$, and all other nodes in $\mathcal{Q}_1$ have rewards 0. This is because t' is the first day when a positive node is found. Since node j is tested positive and isolated on day t', then all infectious nodes with indices less than j can no longer infect other nodes in the line network. Now, we consider the nodes with indices larger than j. Recall that $j<t'$, so node $j+1$ must be infectious on the day t', and node $j+2$ must be infectious on day $t'+1$ since $\beta=1$. Thus, node $j+2$ is tested positive and is isolated. Since the network is a line, both nodes $j+1$ and $j+2$ can no longer infect other nodes once node $j+2$ is isolated. Note that nodes in $\mathcal{Q}_3$ have positive rewards. When N is sufficiently large, in the rest of the exploitation process, nodes in $\mathcal{Q}_3$ are tested. Recall that we have one test for exploration, and we can isolate at least 2 infectious nodes with indices larger than j.

Repeat the process, we exploit nodes $j+4, j+6, \cdots$ on day $t'+2, t'+3, \cdots$, respectively. Consider the direction from node 1 to node N. On every day, there is at most one new infectious node, but at least two infectious nodes can be isolated. On some day, denote as day $t'+x$, the exploitation process can progress beyond the infections (exceeding by one node) for the first time. In other

words, node $j+2x$ is tested negative on day $t'+x$. By Step 2 in Algorithm 8, $e_I^{(j+2x)}(t'+x-1)=0$. However, since $w_I^{(j+2x-1)}(t'+x)=1$ becuase node $j+2x-2$ is tested positive on day $t'+x-1$. By Step 1 in Algorithm 8, $u_I^{(j+2x)}(t'+x+1)=1$, hence by (E.45), $\hat{r}_{j+2x}(t'+x+1)=1$, which implies node $j+2x$ has the largest reward and is exploited on day $t'+x+1$, and it will be tested positive. On day $t'+x+1$, all infectious nodes are isolated.

Finally, we can calculate the total number of infections to be

$$j+2(t'-j)=2t'-j\leq 2t'\leq 2t=2b'N.$$

Let $b=2b'$. This is an improvement by a factor of at least $\frac{a}{b}$ in comparison to the RbEx strategy, where $\frac{a}{b}$ can be as large as desired by increasing the value of N or decreasing p_0.

E.13. Construction of Networks and Further Results

E.13.1. Constructions of SBM and V-SBM

In this section, we construct SBMs and their variants.

SBM The SBM is a generative model for random graphs. The graph is divided into several communities, and subsets of nodes are characterized by being connected with one another with particular edge densities.[27] The intra-connection probability is p_1, and inter-connection probability is p_2. We denote the SBM as SBM(N,M,p_1,p_2). Note that the (expected) number of edges, denoted by $|\mathcal{E}|$, is

$$|\mathcal{E}|=\frac{p_1}{2}N(\frac{N}{M}-1)+\frac{p_2}{2}\frac{N^2}{M}(M-1). \tag{E.49}$$

Now, we fix $|\mathcal{E}|$, and choose the pair (p_1,p_2) under a fixed $|\mathcal{E}|$ in (E.49). The aim of fixing $|\mathcal{E}|$ is to guarantee that the transmission of the disease would not be affected by edges.

V-SBM Now, we consider a variant of SBM, denoted by V-SBM. Different from SBM, we only allow nodes in cluster i to connect to nodes in successive clusters (the neighbor clusters). Denote

[27] Here, we assume that M is an exact divisor of N.

WS, (d, δ)	γ_c	L_p	Ratio
$(6, 0.05)$	**0.504**	4.952	.0003
$(4, 0)$	**0.500**	62.876	0.191
$(6, 0.1)$	**0.456**	5.718	-0.027
$(4, 0.03)$	**0.456**	10.810	0.097

Table E.1: Clustering coefficients of WS networks.

WS, (d, δ)	γ_c	L_p	Ratio
$(6, .001)$	0.599	**21.188**	0.209
$(4, .0075)$	0.489	**21.264**	0.182
$(6, .005)$	0.592	**14.310**	0.211
$(4, .015)$	0.473	**14.253**	0.174
$(6, .009)$	0.585	**12.081**	0.137
$(4, .0225)$	0.467	**12.171**	0.125

Table E.2: Shortest path lengths of WS networks.

the V-SBM as V-SBM(N, M, p_1, p_2). Similarly, the expected number of edges, denoted by $|\mathcal{E}|$, is

$$|\mathcal{E}| = \frac{p_1}{2} N(\frac{N}{M} - 1) + p_2 \frac{N^2}{M}. \tag{E.50}$$

Now, we fix $|\mathcal{E}|$, and choose the pair (p_1, p_2) under a fixed $|\mathcal{E}|$ in (E.50). The aim of fixing $|\mathcal{E}|$ is to guarantee that the transmission of the disease would not be affected by edges.

E.13.2. The impact of γ_c and L_p individually

In this subsection, we investigate the role of γ_c and L_p individually, not through the common factor δ. We consider different WS networks with degrees $d = 4, 6$ and then adjust the rewiring probability δ, such that one of (γ_c, L_p) is almost constant, and the other is varying. We can see that the trend is similar to what we observed by varying δ in Table 6.6.

BIBLIOGRAPHY

[1] S. Kaul, M. Gruteser, V. Rai, and J. Kenny, "Minimizing age of information in vehicular networks," in *2011 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2011.

[2] S. Kaul, R. Yates, and M. Gruteser, "On piggybacking in vehicular networks," in *2011 IEEE Global Telecommunications Conference - GLOBECOM 2011*, 2011.

[3] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?," in *2012 Proceedings IEEE INFOCOM*, 2012.

[4] R. Yates and S. Kaul, "The age of information: Real-time status updating by multiple sources," *IEEE Transactions on Information Theory*, vol. 65, no. 3, pp. 1807 – 1827, 2019.

[5] M. Costa, M. Codreanu, and A. Ephremides, "On the age of information in status update systems with packet management," *IEEE Transactions on Information Theory*, vol. 62, no. 4, pp. 1897 – 1910, 2016.

[6] R. Yates, E. Najm, E. Soljanin and J. Zhong, "Timely updates over an erasure channel," in *2017 IEEE International Symposium on Information Theory (ISIT)*, 2017.

[7] P. Mayekar, P. Parag, and H. Tyagi, "Optimal lossless source codes for timely updates," in *2018 IEEE International Symposium on Information Theory (ISIT)*, 2018.

[8] R. Devassy, G. Durisi, G. Ferrante, O. Simeone and E. Uysal-Biyikoglu, "Delay and peak-age violation probability in short-packet transmissions," in *2018 IEEE International Symposium on Information Theory (ISIT)*, 2018.

[9] E. Najm, E. Telatar and R. Nasser, "Optimal age over erasure channels," in *2019 IEEE International Symposium on Information Theory (ISIT)*, 2019.

[10] B. Buyukates, A. Soysal and S. Ulukus, "Age of information in multihop multicast networks," *Journal of Communications and Networks*, vol. 21, pp. 256 – 267, 2019.

[11] S. Farazi, A. Klein, J. McNeill and D. Brown, "On the Age of Information in Multi-Source Multi-Hop Wireless Status Update Networks," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2018.

[12] A. Bedewy, Y. Sun and N. Shroff, "The age of information in multihop networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 3, pp. 1248 – 1257, 2019.

[13] I. Kadota, A. Sinha, E. Uysal-Biyikoglu, R. Singh, and E. Modiano, "Scheduling policies for minimizing age of information in broadcast wireless networks," *IEEE/ACM Transactions on*

Networking, vol. 26, no. 6, pp. 2637 – 2650, 2018.

[14] I. Kadota, A. Sinha and E. Modiano, "Scheduling algorithms for optimizing age of information in wireless networks with throughput constraints," *IEEE/ACM Transactions on Networking*, vol. 27, no. 4, pp. 1359 – 1372, 2019.

[15] A. Bedewy, Y. Sun and N. Shroff, "Minimizing the age of information through queues," *IEEE Transactions on Information Theory*, vol. 65, no. 8, pp. 5215 – 5232, 2019.

[16] Q. He, D. Yuan, and A. Ephremides, "Optimal link scheduling for age minimization in wireless systems," *IEEE Transactions on Information Theory*, vol. 64, no. 7, pp. 5381 – 5394, 2018.

[17] Y. Hsu, "Age of information: Whittle index for scheduling stochastic arrivals," in *2018 IEEE International Symposium on Information Theory (ISIT)*, 2018.

[18] Y. Hsu, E. Modiano and L. Duan, "Scheduling algorithms for minimizing age of information in wireless broadcast networks with random arrivals," *IEEE Transactions on Mobile Computing*, vol. 19, no. 12, pp. 2903 – 2915, 2019.

[19] C. Joo and A. Eryilmaz, "Wireless Scheduling for Information Freshness and Synchrony: Drift-Based Design and Heavy-Traffic Analysis," *IEEE/ACM Transactions on Networking*, vol. 26, no. 6, pp. 2556 – 2568, 2018.

[20] I. Kadota and E. Modiano, "Minimizing the age of information in wireless networks with stochastic arrivals," *IEEE Transactions on Mobile Computing*, vol. 20, no. 3, pp. 1173 – 1185, 2021.

[21] S. Kaul and R. Yates, "Status Updates Over Unreliable Multiaccess Channels," in *2017 IEEE International Symposium on Information Theory (ISIT)*, 2017.

[22] R. Talak, S. Karaman, and E. Modiano, "Distributed scheduling algorithms for optimizing information freshness in wireless networks," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2018.

[23] A. Kosta, N. Pappas, A. Ephremides and V. Angelakis, "Age of information performance of multiaccess strategies with packet management," *Journal of Communications and Networks*, vol. 21, no. 3, pp. 244 – 255, 2019.

[24] Z. Jiang, B. Krishnamachari, X. Zheng, S. Zhou, and Z. Niu, "Timely status update in massive IoT systems: Decentralized scheduling for wireless uplinks." arXiv:1801.03975, Jan 2018.

[25] Z. Jiang, B. Krishnamachari, S. Zhou, Z. Niu, "Can decentralized status update achieve universally near-optimal age-of-information in wireless multiaccess channels?," in *2018 30th International Teletraffic Congress (ITC 30)*, 2018.

[26] A. M. Bedewy, Y. Sun, R. Singh, and N. B. Shroff, "Optimizing information freshness using low-power status updates via sleep-wake scheduling," in *Twenty-First International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing(Mobihoc)*, 2020.

[27] A. M. Bedewy, Y. Sun, R. Singh, and N. B. Shroff, "Low-Power Status Updates via Sleep-Wake Scheduling," *IEEE/ACM Transactions on Networking*, vol. 29, no. 5, pp. 2129 – 2141, 2021.

[28] X. Chen, K. Gatsis, H. Hassani and S. Saeedi-Bidokhti, "Age of information in random access channels," in *IEEE International Symposium on Information Theory (ISIT)*, 2020.

[29] R. Yates and S. Kaul, "Age of Information in Uncoordinated Unslotted Updating," in *2020 IEEE International Symposium on Information Theory (ISIT)*, 2020.

[30] D. C. Atabay, E. Uysal, and O. Kaya, "Improving Age of Information in Random Access Channels," in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2020.

[31] T. Shreedhar, S. Kual and R. Yates, "ACP: Age Control Protocol for Minimizing Age of Information over the Internet," in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, 2018.

[32] Y. Sun, Y. Polyanskiy, and E. Uysal-Biyikoglu, "Remote Estimation of the Wiener Process over a Channel with Random Delay," *IEEE Transactions on Information Theory*, vol. 66, no. 2, pp. 1118 – 1135, 2020.

[33] X. Zhang, M. M. Vasconcelos, W. Cui and U. Mitra, "Distributed remote estimation over the collision channel with and without local communication," *IEEE Transactions on Control of Network Systems*, vol. 9, no. 1, pp. 282 – 294, 2022.

[34] X. Chen, X. Liao and S. Saeedi-Bidokhti, "Real-time Sampling and Estimation on Random Access Channels: Age of Information and Beyond," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, 2021.

[35] R. Talak and E. Modiano, "Age-Delay Tradeoffs in Queueing Systems," *IEEE Transactions on Information Theory*, vol. 67, no. 3, pp. 1743 – 1758, 2021.

[36] X. Chen and S. Saeedi Bidokhti, "Benefits of Coding on Age of Information in Broadcast Networks," in *2019 IEEE Information Theory Workshop (ITW)*, 2019.

[37] X. Chen, K. Gatsis, H. Hassani, and S. Saeedi Bidokhti, "Age of Information in Random Access Channels," *IEEE Transactions on Information Theory*, vol. 68, no. 10, pp. 6548 – 6568, 2022.

[38] O. Imer and T. Basar, "Optimal estimation with limited measurements," *International Journal of Systems, Control and Communications*, vol. 2, no. 1, pp. 5 – 29, 2010.

[39] M. Rabi, G. V. Moustakides, and J. S. Baras, "Adaptive Sampling for Linear State Estimation," *SIAM Journal on Control and Optimization*, vol. 50, no. 2, pp. 672–702, 2012.

[40] G. Lipsa and N. Martins, "Remote State Estimation with Communication Costs for First-Order LTI Systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 9, pp. 2013 – 2025, 2011.

[41] A. Molin and S. Hirche, "Event-Triggered State Estimation: An Iterative Algorithm and Optimality Properties," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5939 – 5946, 2017.

[42] A. Nayyar, T. Basar, D. Teneketzis, V. V. Veeravalli, "Communication scheduling and remote estimation with energy harvesting sensor," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, 2012.

[43] J. Chakravorty and A. Mahajan, "Remote Estimation Over a Packet-Drop Channel With Markovian State," *IEEE Transactions on Automatic Control*, vol. 65, no. 5, pp. 2016 – 2031, 2020.

[44] X. Gao, E. Akyol, and T. Basar, "Optimal communication scheduling and remote estimation over an additive noise channel," *Automatica*, vol. 88, pp. 57 – 69, 2018.

[45] K. Huang, W. Liu, Y. Li, and B. Vucetic, "To Retransmit or Not: Real-Time Remote Esimation in Wireless Networked Control," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019.

[46] H. Sac, T. Bacinoglu, E. Uysal-Biyikoglu and G. Durisi, "Age-Optimal Channel Coding Blocklength for an M/G/1 Queue with HARQ," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2018.

[47] L. Meier, J. Peschon, and R.M. Dressler, "Optimal control of measurement subsystems," *IEEE Transactions on Automatic Control*, vol. 12, no. 5, pp. 528 – 536, 1967.

[48] Y. Oshman, "Optimal sensor selection strategy for discrete-time state estimators," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 30, no. 2, pp. 307 – 314, 1994.

[49] A. Logothetis and A. Isaksson, "On sensor scheduling via information theoretic criteria," in *1999 American Control Conference*, 1999.

[50] V. Gupta, T. H. Chung, B. Hassibi, and R. M. Murray, "On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage," *Automatica*, vol. 42, no. 2, pp. 251 – 260, 2006.

[51] M. P. Vitus, W. Zhang, A. Abate, J. Hu, and C. J. Tomlin, "On efficient sensor scheduling for linear dynamical systems," in *Proceedings of the 2010 American Control Conference*, 2010.

[52] L. Zhao, W. Zhang, J. Hu, A. Abate, and C. J. Tomlin, "On the Optimal Solutions of the Infinite-Horizon Linear Sensor Scheduling Problem," *IEEE Transactions on Automatic Control*, vol. 59, no. 10, pp. 2825 – 2830, 2014.

[53] S. T. Jawaid and S. L. Smith, "Submodularity and greedy algorithms in sensor scheduling for linear dynamical systems," *Automatica*, vol. 61, pp. 282 – 288, 2015.

[54] K. Gatsis, M. Pajic, A. Ribeiro and G. Pappas, "Opportunistic Control Over Shared Wireless Channels," *IEEE Transactions on Automatic Control*, vol. 60, no. 12, pp. 3140 – 3155, 2015.

[55] K. Gatsis, A. Ribeiro, and G. Pappas, "Control with Random Access Wireless Sensors," in *2015 IEEE 54th Annual Conference on Decision and Control (CDC)*, 2015.

[56] G. Taricco, "Joint Channel and Data Estimation for Wireless Sensor Networks," *IEEE Wireless Communications Letters*, vol. 1, no. 5, pp. 532 – 535, 2012.

[57] O. T. Yavascan and E. Uysal, "Analysis of slotted ALOHA with an age threshold," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1456 – 1470, 2021.

[58] G. Papoudakis, F. Christinos, L. Schafer, and etc, "Benchmarking multi-agent deep reinforcement learning algorithms in cooperative tasks," in *Advances in Neural Information Processing Systems*, 2021.

[59] G. Papoudakis, F. Christinos, A. Rahman, and etc, "Dealing with Non-Stationarity in Multi-Agent Deep Reinforcement Learning." arXiv, Jun 2019.

[60] P. Hernandez-Leal, B. Kartal and M. E. Taylor, "A survey and critique of multiagent deep reinforcement learning," in *International Conference on Autonomous Agents and Multi-Agent Systems*, 2019.

[61] M. Tan, "Multi-agent reinforcement learning: Independent vs. cooperative agents," in *International Conference on Machine Learning*, 1993.

[62] R. Lowe, Y. Wu, A. Tamar, and etc, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Advances in Neural Information Processing Systems*, 2017.

[63] C. Yu, A. Velu, E. Vinitsky, and etc, "The Surprising Effectiveness of MAPPO in Cooperative, Multi-Agent Games." arXiv, 2021.

[64] P. Sunehag, G. Lever, A. Gruslys, and etc, "Value-decomposition networks for cooperative multi-agent learning," in *International Conference on Autonomous Agents and Multi-Agent Systems*, 2018.

[65] T. Rashid, M. Samvelyan, C. Schroeder-De-Witt, and etc, "QMIX: monotonic value function factorisation for deep multi-agent reinforcement learning," in *International Conference on Machine Learning*, 2018.

[66] A. Feriani and E. Hossain, "Single and Multi-Agent Deep Reinforcement Learning for AI-Enabled Wireless Networks: A Tutorial," *IEEE Communications Survey & Tutorials*, vol. 33, no. 2, pp. 1226 – 1252, 2021.

[67] S. Bhattacharya, A. Banerjee, S. Peruru, and etc, "Fast Link Scheduling in Wireless Networks Using Regularized Off-Policy Reinforcement Learning," *IEEE Networking Letters*, vol. Early Access, 2023.

[68] K. Yang, D. Li, C. Shen, and etc, "Multi-Agent Reinforcement Learning for Wireless User Scheduling: Performance, Scalablility, and Generalization," in *56th Asilomar Conference on Signals, Systems, and Computers*, 2022.

[69] N. Naderializadeh, J. Sydir, M. Simsek , and etc, "Resource Management in Wireless Networks via Multi-Agent Deep Reinforcement Learning," *IEEE Transactions on Wireless Communications*, vol. 20, no. 6, pp. 3507 – 3523, 2021.

[70] Y. Nasir and D. Guo, "Multi-Agent Deep Reinforcement Learning for Dynamic Power Allocation in Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2239 – 2250, 2019.

[71] Y. Zhang, B. Di, Z. Zheng, J. Lin, and etc, "Distributed Multi-Cloud Multi-Access Edge Computing by Multi-Agent Reinforcement Learning," *IEEE Transactions on Wireless Communications*, vol. 20, no. 4, pp. 2565 – 2578, 2021.

[72] H. Li, Y. Wu, and M. Chen, "Adaptive Fault-Tolerant Tracking Control for Discrete-Time Multiagent Systems via Reinforcement Learning Algorithm," *IEEE Transactions on Cybernetics*, vol. 51, no. 3, pp. 1163 – 1174, 2021.

[73] F. Yao and L. Jia, "A Collaborative Multi-Agent Reinforcement Learning Anti-Jamming Algorithm in Wireless Networks," *IEEE Wireless Communications Letters*, vol. 8, no. 4, pp. 1024 – 1027, 2019.

[74] N. Garg and T. Ratnarajah, "Cooperative Scenarios for Multi-Agent Reinforcement Learning in Wireless Edge Caching," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

[75] R. Zhong, X. Liu, Y. Liu, and etc, "Multi-Agent Reinforcement Learning in NOMA-Aided UAV Networks for Cellular Offloading," *IEEE Transactions on Wireless Communications*, vol. 21, no. 3, pp. 1498 – 1512, 2022.

[76] K. Menda, Y. Chen, J. Grana, and etc, "Deep Reinforcement Learning for Event-Driven

Multi-Agent Decision Processes," *IEEE Transactions on Intelligent Transaportation Systems*, vol. 20, no. 4, pp. 1259 – 1268, 2019.

[77] I. Liu, R. A. Yeh, and A. G. Schwing, "PIC: Permutation Invariant Critic for Multi-Agent Deep Reinforcement Learning." arXiv, 2019.

[78] F. Gama, A. G. Marques, G. Leus, and A. Ribeiro, "Convolutional Graph Neural Networks," in *The 53rd Asilomar Conference on Circuits, Systems and Computers (ACSSC)*, 2019.

[79] F. Gama, J. Bruna, and A. Ribeiro, "Stability properties of graph neural networks," *IEEE Transactions on Signal Processing*, vol. 68, pp. 5680 – 5695, 2020.

[80] J. Du, J. Shi, S. Kar, and etc, "On graph convolution for graph CNNs," in *IEEE Data Science Workshop (DSW)*, 2018.

[81] L. Ruiz, F. Gama, and A. Ribeiro, "Gated Graph Recurrent Neural Networks," *IEEE Transactions on Signal Processing*, vol. 68, pp. 6303 – 6318, 2020.

[82] M. Gori, G. Mondardini, and F. Scarselli, "A new model for learning in graph domains," in *IEEE International Joint Conference on Neural Networks*, 2005.

[83] F. Scarselli, M. Gori, A. C. Tsoi, and etc, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61 – 80, 2009.

[84] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations*, 2017.

[85] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *30th Conference on Neural Information Processing Systems*, 2016.

[86] F. Gamma, A. G. Marques, G. Leus, and etc, "Convolutional neural network architectures for signals supported on graphs," *IEEE Transactions on Signal Processing*, vol. 67, no. 4, pp. 1034 – 1049, 2019.

[87] R. Pascanu, C. Gulcehre, K. Cho, and etc, "How to construct deep recurrent neural networks," in *International Conference on Learning Representations*, 2014.

[88] A. Graves, "Generating sequences with recurrent neural networks." arXiv, Jun. 2014.

[89] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673 – 2681, 1997.

[90] Y. Seo, M. Defferrard, P. Vandergheynst, and etc, "Structured sequence modeling with graph convolutional recurrent networks," in *International Conference on Neural Information Pro-*

cessing, 2018.

[91] Y. Li, R. Yu, C. Shahabi, and etc, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *International Conference on Learning Representations*, 2018.

[92] G. Shen, W. Zhou, W. Zhang, and etc, "Bidirectional spatial–temporal traffic data imputation via graph attention recurrent neural network," *Neurocomputing*, vol. 531, pp. 151 – 162, 2023.

[93] C. Capanema, G. Oliveira, F. Silva, and etc, "Combining recurrent and Graph Neural Networks to predict the next place's category," *Ad Hoc Networks*, vol. 138, no. 103016, 2023.

[94] K. Chen and L. Huang, "Age-of-Information in the Presence of Error," in *2016 IEEE International Symposium on Information Theory (ISIT)*, 2016.

[95] P. Parag, A. Taghavi and J. Chamberland, "On Real-Time Status Updates over Symbol Erasure Channels," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, 2017.

[96] S. Feng and J. Yang, "Procoding and Scheduling for AoI Minimization in MIMO Broadcast Channels," *IEEE Transactions on Information Theory*, vol. 68, no. 8, pp. 5185 – 5202, 2022.

[97] S. Feng and J. Yang, "Adaptive Coding for Information Freshness in a Two-user Broadcast Erasure Channel," in *2019 IEEE Global Communications Conference (GLOBECOM)*, 2019.

[98] M. Costa and Y. Sagduyu, "Age of information with network coding," *Ad Hoc Networks*, vol. 86, pp. 15 – 22, 2019.

[99] B. Shulgin, L. Stone and Z. Agur, "Pulse vaccination strategy in the SIR epidemic model," *Bulletin of Mathematical Biology*, vol. 60, no. 6, pp. 1123 – 1148, 1998.

[100] P. Tapaswi and J. Chattopadhyay, "Global stability results of a "susceptible-infective-immune-susceptible" (SIRS) epidemic model," *Ecological Modelling*, vol. 87, no. 1, pp. 223–226, 1996.

[101] L.Stone, B.Shulgin and Z.Agur, "Theoretical examination of the pulse vaccination policy in the SIR epidemic model," *Mathematical and Computer Modelling*, vol. 31, no. 4, pp. 207 – 215, 2000.

[102] Y. Takeuchi, W. Ma and E. Beretta, "Global asymptotic properties of a delay SIR epidemic model with finite incubation times," *Nonlinear Analysis: Theory, Methods & Applications*, vol. 42, pp. 931 – 947, 2000.

[103] J. Aron, "Acquired immunity dependent upon exposure in an SIRS epidemic model," *Mathematical Biosciences*, vol. 88, pp. 37 – 47, 1988.

[104] L. Allen, "Some discrete-time SI, SIR, and SIS epidemic models," *Mathematical Biosciences*,

vol. 124, pp. 83 – 105, 1994.

[105] A. M. Ramos, M. R. Ferrandez, M. Vela-Perez and et al, "A simple but complex enough θ-SIR type model to be used with COVID-19 real data. Application to the case of Italy," *Physica D*, vol. 421, no. 132839, 2021.

[106] A. G. M. Neves and G. Guerrero, "Predicting the evolution of the COVID-19 epidemic with the A-SIR model: Lombardy, Italy and Sao Paulo state, Brazil," *Physica D*, vol. 413, no. 132693, 2020.

[107] A. Simha, R. Prasad and S. Narayana, "A simple Stochastic SIR model for COVID-19 Infection Dynamics for Karnataka after interventions – Learning from European Trends." arXiv: 2003.11920, 2020.

[108] B. Ndiaye, L. Tendeng and D. Seck, "Analysis of the COVID-19 pandemic by SIR model and machine learning technics for forecasting." arXiv: 2004.01574, 2020.

[109] J. Zhu, P. Ge, C. Jiang and et al, "Deep-learning artificial intelligence analysis of clinical variables predicts mortality in COVID-19 patients," *Journal of the American College of Emergency Physicians Open*, vol. 1, no. 6, pp. 1364–1373, 2020.

[110] C. Mahanty, R. Kumar, B. K. Mishra, and et al, "Prediction of COVID-19 active cases using exponential and non-linear growth models," *Expert Systems*, vol. 39, no. 3, pp. 1 – 22, 2020.

[111] E. B. Postnikov, "Estimation of COVID-19 dynamics "on a back-of-envelope": Does the simplest SIR model provide quantitative parameters and predictions?," *Chaos, Solitons and Fractals*, vol. 135, no. 109841, 2020.

[112] B. Ndiaye, L. Tendeng and D. Seck, "Comparative prediction of confirmed cases with COVID-19 pandemic by machine learning, deterministic and stochastic SIR models." arXiv: 2004.13489, 2020.

[113] I. Rahimi, A. H. Gandomi, P. G. Asteris and et al, "Analysis and Prediction of COVID-19 Using SIR, SEIQR, and Machine Learning Models: Australia, Italy, and UK Cases," *Information*, vol. 12, no. 109, 2021.

[114] G. Hu and J. Geng, "Heterogeneity learning for SIRS model: an application to the COVID-19," *Statistics and Its Interface*, vol. 14, no. 1, pp. 73 – 81, 2021.

[115] R. Vega, L. Flores and R. Greiner, "SIMLR: Machine Learning inside the SIR Model for COVID-19 Forecasting," *Forecasting*, vol. 4, no. 1, pp. 72 – 94, 2022.

[116] H. Bastani, K. Drakopoulos, V. Gupta and et al, "Efficient and targeted COVID-19 border testing via reinforcement learning," *Nature*, vol. 599, no. 7883, pp. 108 – 113, 2021.

[117] S. A. Alanazi, M. M. Kamruzzaman, M. Alruwaili and et al, "Measuring and Preventing COVID-19 Using the SIR Model and Machine Learning in Smart Health Care," *Journal of Healthcare Engineering*, vol. 2020, no. 8857346, 2020.

[118] G. Perakis, D. Singhvi, O. S. Lami, and et al, "COVID-19: A multiwave SIR-based model for learning waves," *Production and Operations Management*, no. 13681, pp. 1 – 9, 2022.

[119] S. Chowdhury, S. Roychowdhury and I. Chaudhuri, "Universality and herd immunity threshold : Revisiting the SIR model for COVID-19," *International Journal of Modern Physics C*, vol. 32, no. 10, p. 2150128, 2021.

[120] W. Choi and E. Shim, "Optimal strategies for social distancing and testing to control COVID-19," *Journal of Theoretical Biology*, vol. 512, no. 110568, 2021.

[121] D. Acemoglu, A. Fallah, A. Giometto and et al, "Optimal adaptive testing for epidemic control: combining molecular and serology tests." arXiv:2101.00773, 2021.

[122] L. Abraham, G. Becigneul and B. Scholkopf, "Crackovid: Optimizing Group Testing." arXiv:2005.06413, 2020.

[123] C. Tsay, F. Lejarza, M. Stadtherr and et al, "Modeling, state estimation, and optimal control for the US COVID-19 outbreak," *Scientific reports*, vol. 10, no. 10711, 2020.

[124] F. Piguillem and L. Shi, "Optimal COVID-19 quarantine and testing policies," *Nature Communications*, vol. 12, no. 356, 2021.

[125] K. Kuga, M. Tanaka and J. Tanimoto, "Pair approximation model for the vaccination game: predicting the dynamic process of epidemic spread and individual actions against contagion," *Proceedings of the Royal Society A*, vol. 477, no. 2246, 2021.

[126] K. K. K. Kabir and J. Tanimoto, "The impact of information spreading on epidemic vaccination game dynamics in a heterogeneous complex network-a theoretical approach," *Chaos, Solitons & Fractals*, vol. 132, p. 109548, 2020.

[127] K. Kabir and J. Tanimoto, "Evolutionary vaccination game approach in metapopulation migration model with information spreading on different graphs," *Chaos, Solitons & Fractals*, vol. 120, pp. 41 – 55, 2019.

[128] L. Willem, S. Abrams, P. J. K. Libin and et al, "The impact of contact tracing and household bubbles on deconfinement strategies for COVID-19," *Nature Communications*, vol. 12, no. 1524, 2021.

[129] J. Kim, X. Chen, H. Nikpey and et al, "Tracing and testing multiple generations of contacts to COVID-19 cases: cost-benefit tradeoffs," *Royal Society Open Science*, vol. 9, no. 10, pp. 1 – 20, 2022.

[130] A. Aleta, D. Martin-Corral, A. Piontti and et al, "Modelling the impact of testing, contact tracing and household quarantine on second waves of COVID-19," *Nature Human Behaviour*, vol. 4, no. 9, pp. 964–971, 2020.

[131] J. Hellewell, S. Abbott, A. Gimma and et al, "Feasibility of controlling COVID-19 outbreaks by isolation of cases and contacts," *The Lancet Global Health*, vol. 8, no. 4, 2020.

[132] A. J. Kucharski, A. J. K. Conlan, S. M. Kissler, etc, "Effectiveness of isolation, testing, contact tracing, and physical distancing on reducing transmission of SARS-CoV-2 in different settings: a mathematical modelling study," *The Lancet Infectious Diseases*, vol. 20, no. 10, pp. 1151 – 1160, 2020.

[133] S. Kojaku, L. Hebert-Dufresne, E. Mones, and et al, "The effectiveness of backward contact tracing in networks," *Nature Physics*, vol. 17, pp. 652 – 658, 2021.

[134] A. Perrault, M. Charpignon, J. Gruber, etc, "Designing Efficient Contact Tracing Through Risk-Based Quarantining," Working Paper, National Bureau of Economic Research, Nov. 2020.

[135] H. Ou, A. Sinha, S. Suen, etc, "Who and when to screen: Multi-round active screening for network recurrent infectious diseases under uncertainty," in *Proceedings of 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2020.

[136] P. Auer, N. Cesa-Bianchi and P. Fischer, "Finite-time Analysis of the Multiarmed Bandit Problem," *Machine Learning*, vol. 47, pp. 235–256, 2002.

[137] S. Agrawal and N. Goyal, "Analysis of Thompson sampling for the multi-armed bandit problem," vol. 23, pp. 1–26, 2012.

[138] S. Agrawal and N. Goyal, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *Foundations and Trends in Machine Learning*, vol. 5, no. 1, pp. 1–122, 2012.

[139] K. Madhama and T. Murata, "A multi-armed bandit approch for exploring partially observed networks," *Applied Network Science*, vol. 4, no. 26, pp. 1–18, 2019.

[140] M. Bilgic, L. Mihalkova and L. Getoor, "Active learning for networked data," *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pp. 79–86, 2010.

[141] X. Wang and R. Garnett and J. Schneider, "Active search on graphs," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013.

[142] Y. Ma and T. K. Huang and J. Schneider, "Active search and bandits on graphs using sigma-optimality," in *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, 2015.

[143] R. Garnett, Y. Krishnamurthy, D. Wang and et al, "Bayesian optimal active search on graphs," in *Proceedings of the 29th International Coference on International Conference on Machine Learning*, 2011.

[144] D. Zhao, J. Liu, R. Wu and et al, "Data-Efficient Reinforcement Learning Using Active Exploration Method," in *International Conference on Neural Information Processing*, 2018.

[145] Y. Burda, H. Edwards, A. Storkey and et al, "Exploration by random network distillation," in *International Conference on Learning Representations*, 2019.

[146] M. Bellemare, S.Srinivasan, G. Ostrovski and et al, "Unifying count-based exploration and intrinsic motivation," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 2016.

[147] R. Singh, F. Liu and N. B. Shroff, "A Partially Observable MDP Approach for Sequential Testing for Infectious Diseases such as COVID-19." arXiv:2007.13023, 2020.

[148] H. Grushka-Cohen, R. Cohen, B. Shapira and et al, "A framework for optimizing COVID-19 testing policy using a Multi Armed Bandit approach." arXiv:2007.14805, 2020.

[149] E. Meirom, H. Maron, S. Mannor, and G. Chechik, "Controlling Graph Dynamics with Reinforcement Learning and Graph Neural Networks," in *Proceedings of the 38th International Conference on Machine Learning*, no. 139, pp. 7565 – 7577, 2021.

[150] L. Kaelbling and M. Littman and A. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[151] G. Monahan, "State of the Art — A Survey of Partially Observable Markov Decision Processes: Theory, Models, and Algorithms," *Management Science*, vol. 28, no. 1, pp. 1–16, 1982.

[152] D. Bertsekas and R. Gallager, *Data Networks (2Nd Ed.)*. Prentice-Hall, Inc., 1992.

[153] M. Gatzianas, L. Georgiadis, L. Tassiulas, "Multiuser Broadcast Erasure Channel with Feedback—Capacity and Algorithms," *IEEE Transactions on Information Theory*, vol. 59, no. 9, pp. 5779 – 5804, 2013.

[154] S. Saeedi Bidokhti, M. Gatzianas and C. Fragouli, "A class of feedback-based coding algorithms for broadcast erasure channels with degraded message sets." http://infoscience.epfl.ch/record/175823, 2012.

[155] M. Heindlmaier and S. Saeedi-Bidokhti, "Capacity Regions of Two-Receiver Broadcast Erasure Channels With Feedback and Memory," *IEEE Transactions on Information Theory*, vol. 64, no. 7, pp. 5042 – 5069, 2018.

[156] S. C. Lin, I. H. Wang and A. Vahid, "No feedback, no problem: Capacity of erasure broadcast

channels with single-user delayed csi," in *2019 IEEE International Symposium on Information Theory (ISIT)*, 2019.

[157] B. Tasybakov and N. Likhanov, "Upper bound on the capacity of a random multiple-access system," *Problems of Information Transmission*, vol. 23, no. 3, pp. 224–236, 1987.

[158] P. Huang and X. Lin, "Achieving Optimal Throughput Utility and Low Delay With CSMA-Like Algorithms: A Virtual Multichannel Approach," *IEEE/ACM Transactions on Networking*, vol. 23, no. 2, pp. 505 – 518, 2014.

[159] L. Jiang, M. Leconte, J. Ni, R. Srikant and J. Walrand, "Fast Mixing of Parallel Glauber Dynamics and Low-Delay CSMA Scheduling," *IEEE Transactions on Information Theory*, vol. 58, no. 10, pp. 6541 – 6555, 2012.

[160] Y. Yang and T. Yum, "Delay distributions of slotted ALOHA and CSMA," *IEEE Transactions on Communications*, vol. 51, no. 11, pp. 1846 – 1857, 2003.

[161] A. Maatouk, M. Assaad and A. Ephremides, "On the age of information in a CSMA environment," *IEEE/ACM Transactions on Networking*, vol. 28, no. 2, pp. 818–831, 2020.

[162] M. Wang and Y. Dong, "Broadcast Age of Information in CSMA/CA Based Wireless Networks," in *2019 15th International Wireless Communications & Mobile Computing Conference*, 2019.

[163] L. Kleinrock, "Packet Switching in Radio Channels: Part I-Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics," *IEEE Transactions on Communications*, vol. 23, no. 12, pp. 1400 – 1416, 1975.

[164] N. Abramson, "The ALOHA System-Another Alternative for Computer Communications," in *AFIPS Fall Joint Computing Conference*, 1970.

[165] L. Kleinrock and S. Lam, "Packet Switching in Multiaccess Broadcast Channel: Performance Evaluation," *IEEE Transactions on Communications*, vol. 23, no. 4, pp. 410 – 423, 1975.

[166] R. Gallager, "Conflict resolution in random access broadcast networks," *Computer Science*, pp. 74–76, 1978.

[167] J. Mosely, "An efficient contention resolution algorithm for multiple access channels," Master's thesis, Massachusetts Institute of Technology, May 1979.

[168] Y. Ouyang and D. Teneketzis, "A common information-based multiple access protocol achieving full throughput," in *2015 IEEE International Symposium on Information Theory (ISIT)*, 2015.

[169] J. Ni, B. Tan, and R. Srikant, "Q-CSMA: Queue-Length-Based CSMA/CA Algorithms for

Achieving Maximum Throughput and Low Delay in Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 20, no. 3, pp. 825 – 836, 2012.

[170] R. Gallager, "A Perspective on Multiaccess Channels," *IEEE Transactions on Information Theory*, vol. 32, no. 2, pp. 124 – 142, 1985.

[171] Y. Gai, S. Ganesan, and B. Krishnamachari, "The saturation throughput region of p-persistent CSMA," in *Information Theory and Applications Workshop*, 2011.

[172] G. I. Papadimitriou and A. S. Pomportsis, "Adaptive MAC protocols for broadcast networks with bursty traffic," *IEEE Transactions on Communications*, vol. 51, no. 4, pp. 553 – 557, 2003.

[173] M. Fang, D. Malone, K. R. Duffy and D. J. Leith, "Decentralised Learning MACs for Collision-free Access in WLANs," *Wireless Networks*, vol. 19, no. 1, pp. 83 – 98, 2013.

[174] D. Shah and J. Shin, "Randomized scheduling algorithm for queueing networks," *The Annals of Applied Probability*, vol. 22, no. 1, pp. 128 – 171, 2012.

[175] L. Jiang, D. Shah, J. Shin, and J. Walrand, "Distributed Random Access Algorithm: Scheduling and Congestion Control," *IEEE Transactions on Information Theory*, vol. 56, no. 12, pp. 6182 – 6207, 2010.

[176] S. A. Ahmad, "A Waterfilling Algorithm for Multiple Access Point Connectivity With Constrained Backhaul Network," *IEEE Wireless Communications Letters*, vol. 4, pp. 517 – 520, 2015.

[177] Y. Liu, Z. Qin and Z. Ding, *Non-Orthogonal Multiple Access for Massive Connectivity.* Springer in Computer Science, 2020.

[178] P. Pan and L. Yang, "Spatially Modulated Code-Division Multiple-Access for High-Connectivity Multiple Access," *IEEE Transactions on Wireless Communications*, vol. 18, pp. 4031 – 4046, 2019.

[179] L. Chen, N. Zhao, Y. Chen, F. R. Yu and G. Wei, "Toward Optimal Rate-Delay Tradeoff for Computation Over Multiple Access Channel," *IEEE Transactions on Communications*, vol. 69, pp. 4335 – 4346, 2021.

[180] P. Li and J. Xu, "Fundamental Rate Limits of UAV-Enabled Multiple Access Channel With Trajectory Optimization," *IEEE Transactions on Wireless Communications*, vol. 19, pp. 458 – 474, 2019.

[181] X. Zhao and W. Chen, "Non-Orthogonal Multiple Access for Delay-Sensitive Communications: A Cross-Layer Approach," *IEEE Transactions on Communications*, vol. 67, pp. 5053 – 5068, 2019.

[182] J. Choi, "On the Power Allocation for MIMO-NOMA Systems With Layered Transmissions," *IEEE Transactions on Wireless Communications*, vol. 15, pp. 3226 – 3237, 2016.

[183] T. Wang, A. Seyedi, A. Vosoughi and W. Heinzelman, "Optimal Rate Allocation for Distributed Source Coding over Gaussian Multiple Access Channels," *IEEE Transactions on Wireless Communications*, vol. 12, pp. 2002 – 2013, 2013.

[184] D. Calabuig, R. H. Gohary and H. Yanikomeroglu, "Optimum Transmission Through the Multiple-Antenna Gaussian Multiple Access Channel," *IEEE Transactions on Information Theory*, vol. 62, no. 1, pp. 230 – 243, 2015.

[185] J. Dai, Z. Ye and X. Xu, "Power Allocation for Maximizing the Minimum Rate With QoS Constraints," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 9, pp. 4989 – 4996, 2009.

[186] S. Riaz, F. A. Khan, S. Saleem and Q. Z. Ahmed, "Reducing the Mutual Outage Probability of Cooperative Non-Orthogonal Multiple Access," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 16207 – 16212, 2020.

[187] K. Wang, Z. Ding, D. K. C. So and G. K. Karagiannidis, "Stackelberg Game of Energy Consumption and Latency in MEC Systems With NOMA," *IEEE Transactions on Wireless Communications*, vol. 69, pp. 2191 – 2206, 2021.

[188] B. Di, L. Song, Y. Li and G. Y. Li, "Non-Orthogonal Multiple Access for High-Reliable and Low-Latency V2X Communications in 5G Systems," *IEEE Journal on Selected Areas in Communications*, vol. 35, pp. 2383 – 2397, 2017.

[189] J. Luo, L. Jiang and C. He, "Cross-Layer Optimization for Energy-Timeliness Tradeoff in TDMA Based Sensor Networks," in *IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference*, 2008.

[190] C. Xiao, J. Zeng, W. Ni, X. Su, R. Liu, T. Lv and J. Wang, "Downlink MIMO-NOMA for Ultra-Reliable Low-Latency Communications," *IEEE Journal on Selected Areas in Communications*, vol. 37, pp. 780 – 794, 2019.

[191] R. Gallager, *Discrete Stochastic Process*. Springer, Boston, MA, 1996.

[192] L. Russell, "Strong laws of large numbers for weakly correslated random variables," *Michigan Mathematical Journal*, vol. 35, no. 3, pp. 353–359, 1988.

[193] T. Z. Ornee and Y. Sun, "Sampling for Remote Estimation through Queues: Age of Information and Beyond," in *International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT)*, 2019.

[194] R. Durrett, *Probability: Theorey and Examples*. New York, NY: Cambridge University Press,

2019.

[195] A. M. Horowitz, "The second order Langevin equation and numerical simulations," *Nuclear Physics B*, vol. 280, no. 1, pp. 510 – 522, 1987.

[196] A. Novikov and N. Kordzakhia, "Martingales and first passage times of AR(1) sequences," *Stochastics*, vol. 80, no. 2-3, pp. 197 – 210, 2008.

[197] A. A. Novikov, "On distributions of first passage times and optimal stopping of AR(1) sequences," *Theory Probability & Its Applications*, vol. 53, no. 3, pp. 419–429, 2009.

[198] S. Y. Hwang, "Arbitrary initial values and random norm for explosice AR(1) processes generated by stationary errors," *Statistics and Probabilities Letters*, vol. 83, pp. 127 – 134, 2013.

[199] E. T. Ceran, D. Gunduz, and A. Gyorgy, "Learning to Minimize Age of Information over an Unreliable Channel with Energy Harvesting." arXiv, Jun 2021.

[200] Z. Huang, W. Wu, C. Fu, V. Chau, X. Liu, J. Wang, and J. Luo, "AoI-Constrained Bandit: Information Gathering over Unreliable Channels with Age Guarantees." arXiv, Dec 2021.

[201] S. Cui, J. Xiao, A. Goldsmith, Z. Luo, and V. Poor, "Estimation Diversity and Energy Efficiency in Distributed Sensing," *IEEE Transactions on Signal Processing*, vol. 55, no. 9, pp. 4683 – 4695, 2007.

[202] D. T. Nguyen, A. Kumar, and H. C. Lau, "Credit assignment for collective multiagent RL with global rewards," in *Advances in Neural Information Processing Systems*, 2018.

[203] J. Foerster, N. Nardelli, G. Farquhar, and etc, "Stabilising experience replay for deep multi-agent reinforcement learning," in *International Conference on Machine Learning*, 2017.

[204] J. Foerster, G. Farquhar, T. Afouras, and etc, "Counterfactual multi-agent policy gradients," in *Association for the Advancement of Artificial Intelligence*, 2018.

[205] S. Iqbal and F. Sha, "Actor-attention-critic for multi-agent reinforcement learning," in *International Conference on Machine Learning*, 2019.

[206] J. Jiang and Z. Lu, "Learning attentional communication for multi-agent cooperation," in *Advances in Neural Information Processing Systems*, 2018.

[207] A. Das, T. Gervet, J. Romoff, and etc, "Tarmac: Targeted multi-agent communication," in *International Conference on Machine Learning*, 2019.

[208] J. N. Foerster, Y. M. Assael, N. Freitas, and etc, "Learning to communicate with deep multi-agent reinforcement learning," in *International Conference on Machine Learning*, 2016.

[209] D. Kim, S. Moon, D. Hostallero, and etc, "Learning to schedule communication in multi-agent reinforcement learning," in *International Conference on Learning Representations*, 2019.

[210] T. Shu and Y. Tian, "Mind-aware multi-agent management reinforcement learning," in *International Conference on Learning Representations*, 2019.

[211] L. Han, P. Sun, Y. Du, and etc, "Grid-wise control for multi-agent reinforcement learning in video game AI," in *International Conference on Machine Learning*, 2019.

[212] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning (Adaptive Computation and Machine Learning series)*. The MIT Press, 2016.

[213] L. Ruiz, L. F. O. Chamon, and A. Ribeiro, "Transferability Properties of Graph Neural Networks." arXiv, Dec 2021.

[214] V. Mnih, A. Badia, M. Mirza, and etc, "Asynchronous methods for deep reinforcement learning," in *International Conference on Machine Learning*, 2016.

[215] P. Dhariwal, C. Hesse, O. Klimov, and etc, "Openai baselines," tech. rep., OpenAI, Jan 2020.

[216] J. Schulman, F. Wolski, P. Dhariwal, and etc, "Proximal policy optimization algorithms." arXiv, 2017.

[217] S. Athanasiadou, M. Gatzianas, L. Georgiadis, and L. Tassiulas, "XOR-based coding for the 3-user broadcast erasure channel with feedback," in *2012 10th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, 2012.

[218] H. Maleki, V. R. Cadambe and S. A. Jafar, "Index Coding – An Interference Alignment Perspective," *IEEE Transactions on Information Theory*, vol. 60, no. 9, pp. 5402 – 5432, 2014.

[219] M. Effros, S. E. Rouayheb and M. Langberg, "An Equivalence Between Network Coding and Index Coding," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2478 – 2487, 2015.

[220] J. M. Moon and L. Moser, "On cliques in graphs," *Israel Journal of Mathematics*, vol. 3, pp. 23 – 28, 1965.

[221] A. Kosta, N. Pappas and V. Angelakis, "Age of information: A new concept, metric, and tool," *Foundations and Trends in Networking*, vol. 12, no. 3, 2017.

[222] I. Kadota, A. Sinha, E. Modiano, "Optimizing Age of Information in Wireless Networks with Throughput Constraints," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, 2018.

[223] M. J. Neely, *Stochastic Network Optimization With Application to Communication and Queueing Systems*. Morgan and Claypool Publishers, 2010.

[224] G. Walter and M. Contreras, *Compartmental Modeling with Networks*. Birkhauser, Boston, MA, 1999.

[225] S. Ma, J. Zhang, M. Zeng and et al., "Epidemiological parameters of coronavirus disease 2019: a pooled analysis of publicly reported individual data of 1155 cases from seven countries." medRxiv: https://doi.org/10.1101/2020.03.21.20040329, Feb 2020.

[226] A. Byrne, D. McEvoy, A. Collins and et al, "Inferred duration of infectious period of SARS-CoV-2: rapid scoping review and analysis of available evidence for asymptomatic and symptomatic COVID-19 cases," *BMJ Open*, vol. 10, 2020.

[227] CDC, "The U.S. Centers for Disease Control and Prevention (CDC)." https://www.cdc.gov/coronavirus/2019-ncov/php/contact579tracing/contact-tracing-plan/contact-tracing.html, Sept 2020.

[228] D. Topkis, *Supermodularity and Complementarity*. Princeton University Press, 1998.

[229] V. Ilev, "An approximation guarantee of the greedy descent algorithm for minimizing a supermodular set function," *Discrete Applied Mathematics*, vol. 114, pp. 131–146, 2001.

[230] D. L. Hansen, B. Shneiderman, M. A. Smith and et al, *Analyzing Social Media Networks with NodeXL (Second Edition)*. Morgan Kaufmann, 2020.

[231] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-world networks," *Nature*, vol. 393, no. 4, pp. 440–442, 1998.

[232] A. D. Broido and A. Clauset, "Scale-free networks are rare," *Nature Communications*, vol. 10, no. 1017, pp. 1 – 10, 2019.

[233] C. Lee and D. J. Wilkinson, "A review of stochastic block models and extensions for graph clustering," *Applied Network Science*, vol. 4, no. 122, 2019.

[234] P. Sapiezynski, A. Stopczynski, D. D. Lassen and et al, "Interaction data from the Copenhagen Networks Study," *Scientific Data*, vol. 6, no. 1, pp. 1–10, 2019.

[235] S. M. Kissler, P. Klepac, M. Tang, etc, "Sparking "The BBC Four Pandemic": Leveraging citizen science and mobile phones to model the spread of disease." bioRxiv https://doi.org/10.1101/479154., 2018.

[236] P. Klepac, S. Kissler and J. Gog, "Contagion! The BBC Four Pandemic – the model behind the documentary.," *Epidemics*, vol. 24, pp. 49 – 59, 2018.

[237] J. A. Firth, J. Hellewell, P. Klepac, etc, "Using a real-world network to model localized COVID-19 control strategies," *Nature Medicine*, vol. 26, pp. 1616 – 1622, 2020.

[238] X. Yu and N. Li, "How Did Chinese Government Implement Unconventional Measures Against COVID-19 Pneumonia.," *Risk Manag Healthc Policy*, vol. 13, pp. 491 – 499, 2020.

[239] E. Modiano, *Packet Multiple Access and the ALOHA Protocol.* Massachusetts Institute of Technology Department of Aeronautics and Astronautics, 2009.

[240] J. Tsitsiklis, "Analysis of a multiaccess control scheme," *IEEE Transactions on Automatic Control*, vol. 32, no. 11, pp. 1017 – 1020, 1987.

[241] F. M. Spieksma, *Geometrically ergodic Markov chains and the optimal control of queues.* Rijksuniversiteit te Leiden, 1990.

[242] W. Rudin, *Functional Analysis (Second Edition).* McGraw-Hill, 1991.

[243] B. Gao and L. Pavel, "On the Properties of the Softmax Function with Application in Game Theory and Reinforcement Learning." arXiv, Apr 2017.

[244] A. B. Aleksandrov and V. V. Peller, "Operator Lipschitz functions," *Russian Mathematical Surveys*, vol. 71, no. 4, pp. 605 – 702, 2016.

[245] R. G. Wood, T. Linder, and S. Yuksel, "Optimal Zero Delay Coding of Markov Sources: Stationary and Finite Memory Codes," *IEEE Transactions on Information Theory*, vol. 63, no. 9, pp. 5968 – 5980, 2017.

[246] L. Yang and S. Shakkottai, "On Throughput Optimality With Delayed Network-State Information," *IEEE Transactions on Information Theory*, vol. 57, no. 8, pp. 5116 – 5132, 2011.

[247] R. D. Shachter, "Bayes-Ball: The Rational Pastime (for Determining Irrelevance and Requisite Information in Belief Networks and Influence Diagrams)." arXiv: 1301.7412, 2013.

[248] M. Jordan, "An Introduction to Probabilistic Graphical Models." https://people.eecs.berkeley.edu/ jordan/prelims, 2003.