An Analytical Framework for Resource Allocation Between Data and Delayed Network State Information

Jie Chuai and Victor O. K. Li, *Fellow, IEEE*

***Abstract*—The data transmission performance of a network protocol is closely related to the amount of available information about the network state. In general, more network state information results in better data transmission performance. However, acquiring such state information expends network bandwidth resource. Thus a trade-off exists between the amount of network state information collected, and the improved protocol performance due to this information. A framework has been developed in previous efforts to study the optimal trade-off between the amount of collected information and network performance. However, the effect of information delay is not considered in the previous analysis. In this paper, we extend the framework to study the relationship between the amount of collected state information and the achievable network performance under the assumption that information is subject to delay. Based on the relationship we could then obtain the optimal resource allocation between data transmission and network state information acquisition in a time-varying network. We have considered both memoryless and memory-exploited scenarios in our framework. Structures of the Pareto optimal information collection and decision-making strategies are discussed. Examples of multiuser scheduling and multi-hop routing are used to demonstrate the framework's application to practical network protocols.**

***Index Terms*—Protocol design, Resource allocation, Network state information, Delayed information**

I. Introduction

Communication protocols require network state information to achieve good performance. For example, in cellular communications, base stations need to collect various kinds of state information (e.g., channel quality information, buffer state information) from mobile terminals to achieve good scheduling performance. Intuitively, network performance improves as more correct state information is collected. However, the information collection process occupies valuable bandwidth resource, which results in less bandwidth for data transmissions. Therefore, there is a trade-off between the resource used for state information collection and the performance improvement due to such information. In [1], we have developed a general framework to analyze the relationship between the quantity of collected state information and network performance, based on which we could design the optimal scheme for allocating resource between data and network state information. The work in [1] assumes instantaneous utilization of network state information, i.e., there is no delay between the time of information collection and the time of information utilization. However, in many application scenarios, this assumption of zero delay does not hold: first, it takes time for the collected information to propagate from the place where it is generated to the place where it is to be utilized; second, there is always some processing time before the information is transmitted and after it has been received; in addition, the state information is generally not updated instantaneously as the underlying network changes. As a result, this time delay makes the collected state information outdated and inaccurate in a time-varying network, rendering the information less valuable and probably resulting in degradation of network performance. Thus, it is worth investigating how the delay will affect the value of collected network state information and the optimal bandwidth resource allocation scheme. More specifically, we ask the following two questions: Q1: Assume the information collection is subject to delay, what is the minimum amount of state information required to achieve a given network performance? Q2: In a time-varying network, what is the optimal scheme to allocate resource between data and network state information? More precisely, how often should the information be updated, and how much information should be collected each time to optimize the overall bandwidth efficiency?

In this paper, we extend the work in [1] to provide a general framework to analyze the above two questions. Our framework can give the minimum amount of delayed state information required to achieve a given network performance. Based on this relationship, the optimal bandwidth resource allocation scheme could then be derived. We start by considering the utilization of a single piece of delayed state information; then we consider a series of collected state information and analyze both memoryless and memory-exploited periodic information collection strategies. The memoryless information collection strategy has been discussed in our previous paper [2]. In the present work, we further investigate how the exploitation of memory affects the optimal resource allocation scheme. Structures of Pareto optimal information collection and decision-making strategies are analyzed in both cases. The application of our framework is illustrated by examples of multiuser scheduling and multi-hop routing.

We would like to clarify some terms we shall use throughout this paper. The term *information collection* is defined from the viewpoint of the protocol decision maker. Since the decision maker does not have access to the true network states, it needs an *information collection strategy* to extract useful network information to make protocol decisions. The term *network performance* in our paper is a measure of the goodness of

The authors are with the Department of Electrical and Electronic Engineering, The University of Hong Kong.

protocol decisions. In general, a good network performance indicates a good match between the true network states and the protocol decisions. This measure does not count the overhead of network state information collection. A *resource allocation scheme* is a method to distribute bandwidth between network state information collection and data transmission. We call a resource allocation scheme *optimal* if it maximizes the overall bandwidth efficiency by considering the overhead of information collection. We use *performance-rate relationship* as an abbreviation of "the relationship between the network performance and the amount of collected state information (i.e., information rate)" in our paper. An *optimal performance-rate relationship* therefore refers to "the performance-rate relationship that achieves the best network performance given the same amount of collected state information". These terms will be illustrated with more details in our formulation and examples.

The paper is organized as follows. Section II reviews the related work. Section III introduces the framework we have developed to solve the above problems. Sections IV, V and VI present examples to illustrate our framework. Section VII concludes the paper with suggestions for future research.

II. Literature Review

It has been well acknowledged in the research community that the available network state information can influence the performance of network protocols. For example, Seada *et al.* studied the impact of localization errors on geographical routing protocols [3]; Hong and Li [4] provided theoretical analysis on the relationship between the available routing information and the performance of distance vector based routing protocols.

Considering that the network state information is generally subject to delay, many researchers have investigated the effect of outdated information on network performance. The influence of delayed feedback Channel State Information (CSI) on channel capacity has been studied for various channel models, including finite-state Markov channels [5], multiuser MIMO systems [6] and finite-state Multiple Access Channels [7]. Besides the evaluations of channel capacities, there are also discussions on how delayed network state information affects the performance of network-wide protocols. For example, Murugesan *et al.* [8] studied the maximum throughput of opportunistic multiuser scheduling with randomly delayed Automatic Repeat reQuest (ARQ) feedback; the authors of [9] showed how link state information update frequency affects the bandwidth blocking probability of multi-path routing protocols; Ying and Shakkottai analyzed the relationship between the delay of channel/queue information and the throughput region of routing and scheduling in wireless networks [10].

The above work illustrates how the available (erroneous, incomplete or outdated) network state information influences network performance. However, the bandwidth costs of obtaining the network state information are not quantified and discussed in these papers. Therefore, no bandwidth resource allocation schemes are provided to balance between network state information and data transmissions.

As real communication protocols have to allocate part of the bandwidth resources to collect network state information, some researchers have studied the optimal resource allocation schemes under different scenarios. For example, the authors of [11] and [12] discussed the quantity of resource that should be allocated for training and feedback to provide channel state information to the receiver and transmitter in a multiple-antenna system. Chaporkar *et al.* studied the optimal probing and scheduling strategies in downlink opportunistic scheduling for the purpose of queue stablization [13]. Aiming at optimizing bandwidth efficiency, the authors of [14] and [15] analyzed the resource allocation between channel probing and data transmissions in opportunistic scheduling. Hong and Li showed the relationship between the collection range of scheduling information and the network throughput in multi-hop wireless networks in [16] and discussed the optimal resource allocation schemes. As seen from the above, this problem has been addressed under different scenarios; however, a unified framework to tackle this issue is not provided in the above papers.

As this trade-off between the state information collection overhead and the network performance improvement exists in many network communication protocols, it is certainly desirable to establish a unified framework which could be applied to analyze this trade-off and provide guidelines on the design of network protocols. Hong and Li established a general framework based on rate distortion theory to analyze the minimum amount of information required to achieve a target network performance [17]. Their paper provides a systematic way to resolve this problem. However, rate distortion theory can only be used to derive an asymptotic performance-rate relationship, which is only achievable by assuming an infinite delay in information collection. Thus, the application of the framework to practical network protocols is very limited. In our previous work [1], we have provided a general formulation to model the non-asymptotic relationship between the quantity of state information and network performance, which could be applied to practical network protocols to derive the optimal resource allocation between state information collection and data transmission. In this work, we extend the formulation by considering delayed information, and study the non-asymptotic performance-rate relationship and the optimal resource allocation scheme under information delay. Compared to an early version of this work [2], the current version further studies the exploitation of memory in state information collection and utilization, and its impact on the optimal resource allocation of communication protocols.

Finally, we would like to clarify the difference between our work and the literature on multiuser channel allocation problems, e.g., [18]–[20]. The term *resource allocation* used in our paper refers to the bandwidth allocation between *network state information* and *data transmissions* rather than allocation among multiple users. Furthermore, we aim at providing a general framework to analyze the trade-off between state information collection and data transmissions. The examples of multiuser scheduling in our paper are only for illustration, and we do not intend to propose new multiuser scheduling strategies. Therefore, the literature on multiuser channel allocation problems is not directly related to the present work.

III. The Framework

In this paper, we aim at analyzing the relationship between the amount of collected network state information and network performance when there is information delay. We start the analysis in Section III-A by considering a single protocol decision and evaluating the performance-rate relationship with a random information delay. Then in Section III-B, we study the memoryless periodic information collection scheme where network state information are updated periodically and several protocol decisions are made within a time period until new state information is updated. Section III-C then extends the analysis to the case where memory is exploited in the periodic information collection scheme to utilize the past state information. For all three scenarios, our framework provides formulations to derive the optimal performance-rate relationships. Structures of the solutions that achieve the optimal performance-rate relationship have also been analyzed.

A. Network State Information Required to Achieve a Given Performance with a Random Delay

We first investigate the relationship between network performance and the amount of collected state information with a random information delay. We will also define the terms and notations we use throughout the paper.

Scenario: Consider time instant t, and assume the network state (e.g., a channel's state, a node's buffer state) at time t is represented by a random variable X^t, which is unknown to the decision maker. The collected state information at time t is Y^t. Let us assume that the information collection could be probabilistic, i.e., given the true network state, the collected information may vary. We use $P_{Y^t|X^t}$ to denote the transition probability matrix from X^t to Y^t, and $p(y^t|x^t)$ denotes the conditional probability that $Y^t = y^t$ given $X^t = x^t$. We call $P_{Y^t|X^t}$ an *information collection strategy* at time t. Note that $P_{Y^t|X^t}$ defines a probabilistic mapping from X^t to Y^t, and Y^t could be regarded as a coded version of X^t. In this paper, we do not consider the noise in information collection, i.e., the probabilistic mapping from X^t to Y^t is completely determined by the protocol designer.

The collected state information Y^t is transmitted to a network decision maker (e.g., a base station or a mobile terminal), and a protocol decision (e.g., a scheduling/routing decision) is made at time $t + D$ ($D > 0$) based on Y^t. The time difference D between the decision-making and the information collection is modeled as a random variable and is called *information delay*. Here the information delay could be a sum of several factors: propagation delay, processing delay, the delay in making decisions, etc. We use $p(d)$ to denote the probability (or probability density if D is a continuous random variable) that the information delay $D = d$, and we use $\mathcal{D}$ to represent the alphabet of D.

We first consider the case where D has a particular value d. The protocol decision at time $t + d$ is denoted by Z^{t+d}. The decision rule is assumed to be probabilistic. Given the delay is d, we use $P_{Z^{t+d}|Y^t}$ to denote the transition probability matrix from Y^t to Z^{t+d}, and $p(z^{t+d}|y^t)$ is the conditional probability that $Z^{t+d} = z^{t+d}$ given $Y^t = y^t$. We refer to $P_{Z^{t+d}|Y^t}$ as a *decision-making strategy* at time $t + d$.

We use $P_{X^{t+d}|X^t}$ to denote the network state transition matrix at time t with a transition time step d, and $p(x^{t+d}|x^t)$ is the probability for network state to transit from x^t to x^{t+d} in the time step d.

Expected Network Performance: Given the information delay d, the network performance at time $t + d$ given the true network state X^{t+d} and the protocol decision Z^{t+d} is denoted by $g(X^{t+d}, Z^{t+d})$. The network performance can be an arbitrary metric of interest about the network, e.g., the network throughput, the packet delivery ratio. The exact expression of $g(\cdot, \cdot)$ could be defined given a particular network scenario. The expected network performance at time $t+d$ therefore can be calculated as

$$\begin{aligned} G_d =& E\left[g\left(X^{t+d}, Z^{t+d}\right)\right] \\ =& \sum_{x^{t+d}} \sum_{z^{t+d}} p(x^{t+d}, z^{t+d}) g(x^{t+d}, z^{t+d}) \end{aligned} \tag{1}$$

where $p(x^{t+d}, z^{t+d})$ and $g(x^{t+d}, z^{t+d})$ represent the joint probability and network performance, respectively, with random variable $X^{t+d} = x^{t+d}$ and random variable $Z^{t+d} = z^{t+d}$, given the delay d.

Note that the expected network performance G_d is totally determined by the information collection and decision-making strategies given a particular network scenario. The joint probability $p(x^{t+d}, z^{t+d})$ can be computed as

$$\begin{aligned} & p(x^{t+d}, z^{t+d}) \\ =& \sum_{x^t} \sum_{y^t} p(x^t) p(x^{t+d}|x^t) p(y^t|x^t) p(z^{t+d}|y^t) \end{aligned} \tag{2}$$

where we have used the fact that $p(y^t|x^t, x^{t+d}) = p(y^t|x^t)$ and $p(z^{t+d}|x^t, x^{t+d}, y^t) = p(z^{t+d}|y^t)$, i.e., the collected information y^t only depends on the network state x^t and the protocol decision z^{t+d} only depends on the received information y^t. Given a particular network scenario, $p(x^t)$ and $p(x^{t+d}|x^t)$ are determined by the underlying network state evolution characteristics, and $g(\cdot, \cdot)$ is determined by the network scenario considered and the performance metric used; therefore, the expected performance G_d in this particular case is a function of the information collection strategy $P_{Y^t|X^t}$ and decision-making strategy $P_{Z^{t+d}|Y^t}$.

Following the above calculations, it is straightforward to obtain the expected network performance with a random information delay D. The expected network performance at time $t + D$, denoted by G_D, is calculated as

$$G_D = \sum_{d \in \mathcal{D}} p(d) G_d \tag{3}$$

if D is a discrete random variable, or

$$G_D = \int_{d \in \mathcal{D}} p(d) G_d \, \mathrm{d}d \tag{4}$$

if D is a continuous random variable.

Resource for Information Collection: Next, we determine the amount of resource used for transmitting the network state information. As we are concerned with the bandwidth resource in this paper, we calculate the expected number of bits required to transmit the collected state information. Let $l(y^t)$ be the number of information bits required, or the codeword length used, when $Y^t = y^t$. Then the expected resource used for Y^t is

$$R(Y^t) = \sum_{y^t} p(y^t) l(y^t) \tag{5}$$

where $p(y^t) = \sum_{x^t} p(x^t) p(y^t|x^t)$.

According to information theory, if we use the optimal variable-length coding (e.g., Huffman coding), the codeword length $l(y^t)$ is

$$l(y^t) = \lceil \log_2 \frac{1}{p(y^t)} \rceil \tag{6}$$

and thus $R(Y^t)$ has the following upper and lower bounds:

$$H(Y^t) \leq R(Y^t) < H(Y^t) + 1 \tag{7}$$

where $H(Y^t)$ is the entropy of Y^t and is defined as

$$H(Y^t) = -\sum_{y^t} p(y^t) \log_2 p(y^t) \tag{8}$$

In information theory, $H(Y^t)$ measures the amount of uncertainty contained in the random variable Y^t.

In practice, the codewords of Y^t may require additional encoding (e.g., channel coding), and the value of $R(Y^t)$ depends on the specific implementation scheme.

Formulation: The problem of finding the optimal performance-rate relationship can be formulated into the following form:

$$\min_{P_{Y^t|X^t}, \mathbf{P}_{Z^{t+d}|Y^t}} \lambda R(Y^t) - G_D \tag{9}$$

where we use $\mathbf{P}_{Z^{t+d}|Y^t}$ to denote the set $\{P_{Z^{t+d}|Y^t} : d \in \mathcal{D}\}$. Note that in some cases $R(Y^t)$ can affect the probability distribution of D and therefore also affect G_D according to Equations (3) and (4). To minimize the objective, we need to optimize the information collection strategy $P_{Y^t|X^t}$ and the decision-making strategy $P_{Z^{t+d}|Y^t}$ for all $d \in \mathcal{D}$.

The optimization objective is a weighted sum of the amount of collected network state information and the network performance, with the weight coefficient $\lambda \geq 0$. When $\lambda = 0$, we obtain the strategies that collect the maximum amount of network state information and achieve the best network performance; when λ increases, the optimal solution to Problem (9) achieves worse network performance but collects less network state information. By changing the value of λ, we can obtain different trade-offs between information collection rate and network performance for a given delay D.

Given the value of λ, the obtained optimal solution $(P^*_{Y^t|X^t}, \mathbf{P}^*_{Z^{t+d}|Y^t})$ of Problem (9) is Pareto optimal: there are no other strategies that can achieve a higher network performance (or lower information rate) without sacrificing the information rate (or the network performance).

Structure of Pareto Optimal Strategies: Next, we will show that it is sufficient for us to search within deterministic information collection and decision-making strategies in order to obtain Pareto optimal solutions. We prove this by showing that: 1) given the information collection strategy, the decision-making strategy that minimizes the objective function in Problem (9) should be deterministic; 2) given the decision-making strategy, the information collection strategy that minimizes the objective function in Problem (9) should be deterministic.

Deterministic decision-making strategies: When the information collection strategy at time t is already given, we need to choose a decision-making strategy that maximizes the network performance at time $t + d$ for all $d \in \mathcal{D}$ to minimize the objective function of Problem (9).

Theorem 1. *For any delay $d \in \mathcal{D}$, given the information collection strategy $P_{Y^t|X^t}$ and the codeword length $l(y^t)$ for each value of y^t, the decision-making strategy $P_{Z^{t+d}|Y^t}$ that maximizes the network performance at time $t + d$ can always be represented in a deterministic form, i.e., the probability distribution of z^{t+d} given y^t should have the form*

$$p(z^{t+d}|y^t) = \begin{cases} 1 & \text{if } z^{t+d} = z_o^{t+d} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where z_o^{t+d} is a particular protocol decision.

Proof. For any particular value d, from Equation (1), we have

$$\begin{aligned} G_d &= \sum_{x^{t+d}} \sum_{z^{t+d}} p(x^{t+d}, z^{t+d}) g(x^{t+d}, z^{t+d}) \\ &= \sum_{x^{t+d}} \sum_{y^t} \sum_{z^{t+d}} p(x^{t+d}, y^t, z^{t+d}) g(x^{t+d}, z^{t+d}) \\ &= \sum_{x^{t+d}} \sum_{y^t} \sum_{z^{t+d}} p(y^t) p(x^{t+d}, z^{t+d}|y^t) g(x^{t+d}, z^{t+d}) \end{aligned} \tag{11}$$

Consider the part contributed by a particular state information y^t, it could be written as

$$p(y^t) \sum_{z^{t+d}} p(z^{t+d}|y^t) \sum_{x^{t+d}} p(x^{t+d}|y^t) g(x^{t+d}, z^{t+d}) \tag{12}$$

where we have used the fact that $p(z^{t+d}|y^t) = p(z^{t+d}|x^{t+d}, y^t)$, i.e., the protocol decision is independent of the true network state given the collected information.

It is easy to see that, to maximize the total contribution from y^t, we should make $p(z^{t+d}|y^t) = 1$ if z^{t+d} maximizes the value of $\sum_{x^{t+d}} p(x^{t+d}|y^t) g(x^{t+d}, z^{t+d})$. We denote this particular protocol decision as z_o^{t+d}, i.e.,

$$z_o^{t+d} = \arg\max_{z^{t+d}} \sum_{x^{t+d}} p(x^{t+d}|y^t) g(x^{t+d}, z^{t+d}) \tag{13}$$

In fact, z_o^{t+d} is the decision that maximizes the posteriori expected network performance at time $t+d$ given the delayed information y^t. When there are multiple protocol decisions z^{t+d} giving the same posteriori expected network performance, we can just randomly choose one of them to be z_o^{t+d}.

Combining Equations (11) (12) (13), we have

$$G_d = \sum_{y^t} p(y^t) \sum_{x^{t+d}} p(x^{t+d}|y^t) g(x^{t+d}, z_o^{t+d}) \tag{14}$$

□

Deterministic information collection strategies: Next, we show that the information collection strategy should also be deterministic when the decision-making strategy is fixed.

Theorem 2. *Given the set of decision-making strategies* $\mathbf{P}_{Z^{t+d}|Y^t}$*, the codeword length* $l(y^t)$ *for each value of* y^t *and the value of* λ*, the information collection strategy* $P_{Y^t|X^t}$ *that minimizes the objective function of Problem (9) can always be written in a deterministic form, i.e., the probability distribution of* y^t *given* x^t *should have the form*

$$p(y^t|x^t) = \begin{cases} 1 & \text{if } y^t = y_o^t \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

where y_o^t *is a particular value of transmitted information.*

Proof. In the following proof, we will discuss the case where D is a discrete random variable. Proof for a continuous random variable D will be similar to this thus is omitted.

Combining Equations (1), (3) and (5), the objective function of Problem (9) is written as Equation (16).

We only consider the part contributed by a particular true network state x^t, and it can be transformed into the form as in Equation (17).

Therefore, to minimize the part contributed by this particular true state x^t, $p(y^t|x^t)$ should be 1 if $y^t = y_o^t$, where y_o^t is given in Equation (18). The first term is the weighted codeword length of y^t, and the second sum term is the expected network performance at time $t + D$ by selecting y^t. Equation (18) means that we need to select the codeword that minimizes the weighted sum of the codeword length and the expected network performance after a delay D, which indicates that the information collection needs to consider the future network performance when the collected network state information is subject to delay. □

As we have shown that both the information collection and decision-making strategies should be deterministic, Problem (9) has been converted to the optimal entropy-coded quantizer design problem [21], which in general could be solved by the generalized Lloyd algorithm [21].

B. Memoryless Periodic Information Update

In the previous part, we have considered a single piece of collected information with a random delay. In this section, we consider the case where network state information is updated periodically, and the updated information is used to make multiple protocol decisions within a time period. Again, we will discuss the optimal performance-rate relationship and the structures of Pareto optimal solutions.

Problem Formulation: Assume the series of the system states over time is a stationary ergodic process. The information is updated every T time units, and the information Y^t collected at time t is used during the period $[t, t+T]$. Assume M protocol decisions are made at a series of distinct time instants within this duration, where M is a discrete random variable and $p(m)$ denotes the probability that $M = m$. The time instants of the M protocol decisions are also random and are denoted by $t+D_1, t+D_2, \cdots, t+D_M$, respectively, with $D_1 < D_2 < \cdots < D_M$. In the later analysis, we will use $\vec{D}$ to denote the M random delays, and $\vec{d}$ to denote a particular realization of $\vec{D}$. The probability (or probability density) that $\vec{D} = \vec{d}$ is represented as $p(\vec{d})$. The alphabet of D_i is denoted by $\mathcal{D}$ and that of $\vec{D}$ is denoted by $\vec{\mathcal{D}}$.

We measure the *expected average network performance* during the period. Given a particular realization $\vec{d} = (d_1, d_2, \cdots, d_m)$, the average network performance over a period is defined as

$$G_{\vec{d}} = \frac{1}{m}\sum_{i=1}^{m} G_{d_i} \tag{19}$$

where G_{d_i} is defined in Equation (1).

In some cases, different protocol decisions may not be weighted equally. For example, the outcomes of some decisions may last for a longer period, thus they can have higher impacts on the overall network performance. We can assign different weights to the performance functions to reflect those unequal impacts, i.e.,

$$G_{\vec{d}} = \sum_{i=1}^{m} w_i G_{d_i} \tag{20}$$

where w_i is the weight of the i^{th} protocol decision and we assume $w_1 + w_2 + ... + w_m = 1$.

Therefore, it is easy to obtain the expected average network performance with random delays $\vec{D}$:

$$G_{\vec{D}} = \sum_{\vec{d}} p(\vec{d}) G_{\vec{d}} \tag{21}$$

if D_i is a discrete random variable, or

$$G_{\vec{D}} = \int_{\vec{d}\in\vec{\mathcal{D}}} p(\vec{d}) G_{\vec{d}}\, \mathrm{d}\vec{d} \tag{22}$$

if D_i is a continuous random variable.

To calculate $G_{\vec{D}}$ based on Equations (21) and (22), we need to work out $G_{\vec{d}}$ for all possible $\vec{d}$. This might be inconvenient sometimes. In fact, we may determine $G_{\vec{D}}$ without calculating the values of $G_{\vec{d}}$ explicitly. Consider Equations (19) and (21), we have

$$\begin{aligned} G_{\vec{D}} &= \sum_{\vec{d}} p(\vec{d}) \left(\frac{1}{m}\sum_{i=1}^{m} G_{d_i} \right) \\ &= \sum_{\vec{d}} \sum_{i=1}^{m} \frac{1}{m} p(\vec{d}) G_{d_i} \end{aligned} \tag{23}$$

As $d_i \in \mathcal{D}$, Equation (23) can be written as the following

$$\begin{aligned} G_{\vec{D}} &= \sum_{d\in\mathcal{D}} \sum_{\vec{d}} \sum_{i=1}^{m} \frac{1}{m} p(\vec{d}) G_{d_i} \mathbb{1}_{d_i = d} \\ &= \sum_{d\in\mathcal{D}} \sum_{\vec{d}:d_i=d,\exists i} \frac{1}{m} p(\vec{d}) G_d \\ &= \sum_{d\in\mathcal{D}} \alpha_d G_d \end{aligned} \tag{24}$$

where $\mathbb{1}_{d_i=d}$ is the indicator function which equals 1 if $d_i = d$ and 0 otherwise, and the coefficient α_d equals $\sum_{\vec{d}:d_i=d,\exists i} \frac{1}{m} p(\vec{d})$. That is, the coefficient α_d is a sum of the

$$
\begin{aligned}
&\lambda R(Y^t) - G_D \\
=&\lambda \sum_{y^t} p(y^t) l(y^t) - \sum_d p(d) G_d \\
=&\lambda \sum_{y^t} p(y^t) l(y^t) - \sum_d p(d) \sum_{x^{t+d}} \sum_{z^{t+d}} p(x^{t+d}, z^{t+d}) g(x^{t+d}, z^{t+d}) \\
=&\lambda \sum_{x^t} \sum_{y^t} p(x^t, y^t) l(y^t) - \sum_d p(d) \sum_{x^t} \sum_{y^t} \sum_{x^{t+d}} \sum_{z^{t+d}} p(x^t, y^t, x^{t+d}, z^{t+d}) g(x^{t+d}, z^{t+d})
\end{aligned}
\tag{16}
$$

$$
\begin{aligned}
&\lambda \sum_{y^t} p(x^t, y^t) l(y^t) - \sum_d p(d) \sum_{y^t} \sum_{x^{t+d}} \sum_{z^{t+d}} p(x^t, y^t, x^{t+d}, z^{t+d}) g(x^{t+d}, z^{t+d}) \\
=&\lambda \sum_{y^t} p(x^t) p(y^t|x^t) l(y^t) - \sum_d p(d) \sum_{y^t} \sum_{x^{t+d}} \sum_{z^{t+d}} p(x^t) p(y^t|x^t) p(x^{t+d}|x^t) p(z^{t+d}|y^t) g(x^{t+d}, z^{t+d}) \\
=&\sum_{y^t} p(x^t) p(y^t|x^t) \left(\lambda l(y^t) - \sum_d p(d) \sum_{x^{t+d}} \sum_{z^{t+d}} p(x^{t+d}|x^t) p(z^{t+d}|y^t) g(x^{t+d}, z^{t+d}) \right)
\end{aligned}
\tag{17}
$$

$$
y_o^t = \arg\min_{y^t} \left(\lambda l(y^t) - \sum_d p(d) \sum_{x^{t+d}} \sum_{z^{t+d}} p(x^{t+d}|x^t) p(z^{t+d}|y^t) g(x^{t+d}, z^{t+d}) \right) \tag{18}
$$

scaled probability $p(\vec{d})/m$ where $\vec{d}$ satisfies the condition that d is one of its components (i.e., one of the protocol decision is made after delay d).[1]

Therefore, $G_{\vec{D}}$ is expressed as a linear summation of G_d for $d \in \mathcal{D}$. By determining the values of G_d and α_d for the given network scenario, we can obtain the expected average network performance $G_{\vec{D}}$.

The calculations of network performance function G_d and $R(Y^t)$ are the same as those in Section III-A. In this part, we only consider time-invariant strategies, i.e.,

$$
\begin{aligned}
p(Y^t = y|X^t = x) &= p(Y^{t-nT} = y|X^{t-nT} = x) \\
p(Z^{t+d} = z|Y^t = y) &= p(Z^{t-nT+d} = z|Y^{t-nT} = y)
\end{aligned}
\tag{25}
$$

for all values of x, y, z, n, t. Therefore, we will drop t in the following analysis, and use $P_{Y|X}$ and $P_{Z^d|Y}$ to denote the information collection and the decision-making strategies, respectively. Moreover, $R(Y^t)$ should have the same value as $R(Y^{t+nT})$ for any n given the time-invariant $P_{Y|X}$. Therefore, we will just use $R(Y)$ to denote the amount of state information collected each time. Instead of using $R(Y)$ as a measure of the information collection overhead directly, we need to use the average amount of information collected per time unit $R(Y)/T$ when the information is collected periodically.

Therefore, the optimal relationship between the average amount of information collected per time unit and the average expected network performance can be obtained by solving the following problem ($\lambda \geq 0$)

$$
\min_{P_{Y|X}, \mathbf{P}_{Z^d|Y}, T} \lambda \frac{R(Y)}{T} - G_{\vec{D}} \tag{26}
$$

where we use $\mathbf{P}_{Z^d|Y}$ to denote the set $\{P_{Z^d|Y} : d \in \mathcal{D}\}$.

[1]Similarly, we could determine the value of α_d for the case where D_i is a continuous random variable, or $G_{\vec{d}}$ is a weighted sum of G_{d_i}, and then express $G_{\vec{D}}$ in terms of G_d and α_d.

When $\lambda = 0$, we will get the solution with the maximum amount of information collected per time unit and the best network performance; when λ increases, we will obtain a solution with worse network performance and less information collection overhead per time unit. When $\lambda \to \infty$, the amount of collected information per time unit tends to zero.

Moreover, when $\lambda \to 0$, more information will be collected each time and the optimal information collection period T^* in Problem (26) will become shorter. This is easy to understand: as λ decreases, the optimal solution to Problem (26) results in better network performance and more information collection overhead per time unit. As the amount of information collected each time increases, it is possible to improve the average network performance within a period. As the information collection period decreases, we are forcing the information delay to a smaller value, which certainly increases the relevance of the collected information in decision-making.

Structure of Pareto Optimal Strategies: Similar to Section III-A, the Pareto optimal information collection and decision-making strategies obtained from Problem (26) should be deterministic. As the proof is similar, we just present the conclusions in the following.

Given the value of T and the information collection strategy $P_{Y|X}$, the decision-making strategy $P_{Z^d|Y}$ that maximizes the network performance at time $t + d$ could be calculated using Equation (13). Similar to Equation (18), given the value of T, the codeword length $l(y^t)$ and the decision-making strategies at time $t+d$ for discrete delay $d \in \mathcal{D}$, the selected information y_o^t given true network state x^t is given in Equation (27).

C. Periodic Update with Memory Exploited

In the above sections, the updated information Y^t at time t only depends on the system state X^t, which means that the information collection strategy does not consider the past collected state information. However, as the system states

$$y_o^t = \arg\min_{y^t} \left(\lambda \frac{l(y^t)}{T} - \sum_d \alpha_d \sum_{x^{t+d}} \sum_{z^{t+d}} p(x^{t+d}|x^t) p(z^{t+d}|y^t) g(x^{t+d}, z^{t+d}) \right) \quad (27)$$

are temporally correlated in general, utilizing the past state information can generally improve the system performance. Therefore, in this part, we discuss the periodic information update scheme which exploits the past collected state information in both information collection and decision-making.

A General Scenario: Assume the series of the system states over time is a stationary ergodic process. Information is updated periodically, and the update period is T time units. Assume information is collected at time instances $..., t-T, t, t+T, ...$ The collected information Y^t at time t depends on all the past and current true system states, and therefore the information collection strategy could be denoted by $P_{Y^t|X^t_\infty}$, where X^t_∞ represents all the past true network states up to time t (i.e., $X^t, X^{t-1}, X^{t-2}, ...$).

The collected information Y^t is then transmitted to a decision maker, and a series of protocol decisions are made at time instants $t+D_1$, $t+D_2$, $\cdots$, $t+D_M$ during the period $[t, t+T]$ as in Section III-B. The protocol decisions are made based on all the received state information up to time t (i.e., all the past collected information $Y^t, Y^{t-T}, Y^{t-2T}, ...$). The decision-making strategy at a particular delay d is denoted by $P_{Z^{t+d}|Y^t_\infty}$, where Y^t_∞ represents all the received state information up to time t.

Formulation as a Tracking System: In the above scenario, the information collector and the decision maker have different sets of "past information" or "memory": at time t, the past memory at the information collector includes $X^{t-1}, X^{t-2}, ...$, and the past memory at the decision maker is $Y^{t-T}, Y^{t-2T}, ...$ Obviously, the information collector has a larger memory than the decision maker in this case.

To simplify the problem, in the following, we would like to consider a specific class of systems where the information collector and the decision maker have the same set of past information – the set of all the collected information before time t (i.e., $Y^{t-T}, Y^{t-2T}, Y^{t-3T}, ...$). This type of system has been mentioned previously in Gaarder and Slepian's paper [22] and is referred to as a "tracking system", where the decision maker can keep track of the memory state of the information collector. This assumption of memory tracking is reasonable in many communication protocols. For example, the information collector usually is also the decision maker (e.g., a base station), which does not have access to all the true network state information in the past, and therefore can only use the past collected information to decide its next information collection strategy.

Next, we show that the information collector and the decision maker can be represented as state machines. We introduce a random variable B^t, called *belief state*, which represents the information collector and the decision maker's belief of the true network state at time t based on all the collected information before time t (i.e., all Y^{t_0} with $t_0 < t$). During the system operation, instead of maintaining all the past collected information, the information collector and the decision maker only maintain the current belief state of the network. At time t, the collected information Y^t therefore depends on the true network state X^t and the belief B^t; at a particular time $t+d$, the protocol decision Z^{t+d} depends on the belief B^{t+d} which has already been updated based on the collected information Y^t. Therefore, the information collection and the decision-making strategies can be denoted by $P_{Y^t|X^t,B^t}$ and $P_{Z^{t+d}|B^{t+d}}$, respectively. In this paper, we only consider time-invariant strategies, i.e.,

$$\begin{aligned} & p(Y^t = y|X^t = x, B^t = b) \\ &= p(Y^{t-nT} = y|X^{t-nT} = x, B^{t-nT} = b) \\ \\ & p(Z^{t+d} = z|B^{t+d} = b) \\ &= p(Z^{t-nT+d} = z|B^{t-nT+d} = b) \end{aligned} \quad (28)$$

for all values of n, x, y, b, z, t. That is, when the true state is x and the belief state is b, the probability to get information y is invariant for all the information collection time instances $t, t-T, t-2T, ...$; when a decision is made after delay d with a belief state b, the probability to make decision z is also invariant over all the decision-making time instances $t+d, t-T+d, t-2T+d, ...$ Therefore, in the following, we will drop t and use $P_{Y|X,B}$ and $P_{Z^d|B^d}$ to denote the information collection and decision-making strategies, respectively.

After transmitting and receiving the collected information Y^t at time t, the information collector and the decision maker need to update their belief state to B^{t+1}, and the update rule is written as

$$B^{t+1} = s(Y^t, B^t) \quad (29)$$

From the above equation, we could see that the belief state B^t can be regarded as a summary of the past collected information Y^{t-T}_∞. In fact, B^t is a function of all the past collected information before time t (i.e., Y^{t-T}_∞).

The mapping from the sequence Y^{t-T}_∞ to B^t is implied by the belief update function $s(\cdot,\cdot)$. Although the sequence Y^{t-T}_∞ has infinite number of realizations (as the length of the sequence is infinite), the alphabet size of B^t can either be finite or infinite. When the alphabet size of B^t is finite, the function $s(\cdot,\cdot)$ implies a mapping from a set of infinite elements to a set of finite elements. By properly designing the update rule $s(\cdot,\cdot)$, we could make the belief state B^t an equivalence of the past collected information Y^{t-T}_∞, i.e., they result in the same optimal performance-rate relationship. The most naive way is to let $B^t = (Y^{t-T}, Y^{t-2T}, ...)$, by which we do not save any space for memory. However, in some special conditions (e.g., the underlying network state follows a Markov process, as in the example in Section V), we can use less space by memorizing the belief state and still achieve the optimal performance-rate relationship.

The expected system performance at a particular time $t+d$ is still denoted by G_d. As in Section III-B, for

$\vec{d} = (d_1, d_2, \cdots, d_m)$, the time average of the expected system performance over the period starting at time t is

$$G_{\vec{d}} = \frac{1}{m}\sum_{i=1}^{m} G_{d_i} \quad (30)$$

Alternatively, if we are considering decisions weighted unequally, we can also assign different weights to different decisions as in Equation (20).

As the M delays are random variables, the expected average network performance with random delays $\vec{D}$ could be calculated as in Equations (21) and (22). Furthermore, it can also be written as a weighted sum of G_d with $d \in \mathcal{D}$ as in Equation (24).

At a random selected time t, given the past memory B^t, we use $R(Y^t|B^t)$ to denote the expected number of bits required to transmit information Y^t. If we use the optimal variable-length coding, the number of bits required is bounded by

$$H(Y^t|B^t) \leq R(Y^t|B^t) < H(Y^t|B^t) + 1 \quad (31)$$

where $H(Y^t|B^t)$ is the conditional entropy of Y^t given the belief B^t.

Therefore, assume our belief state B^t is an equivalence of the past collected information Y_{∞}^{t-T}, if we want to find the optimal performance-rate relationship, we need to solve the following optimization problem with various values of λ:

$$\min_{P_{Y|X,B}, \mathbf{P}_{Z^d|B^d}, T} \lambda \frac{R(Y|B)}{T} - G_{\vec{D}} \quad (32)$$

where the notation $\mathbf{P}_{Z^d|B^d}$ represents the set $\{P_{Z^d|B^d} : d \in \mathcal{D}\}$. Here we have dropped the notation t to indicate that we are considering the expected value when the process is stationary.

Transformed Into a Markov Decision Process: When the belief state B^t is an equivalence of the past collected information Y_{∞}^{t-T}, we could show that the above process is in fact a Markov Decision Process (MDP) with an infinite horizon [23].

The belief ..., $B^{t-2T}, B^{t-T}, B^t, B^{t+T}, B^{t+2T}$... are the series of states of the MDP. The information collection strategy $P_{Y^t|X^t,b^t}$ is the action taken when the state B^t is b^t. Given the state b^t and the action $P_{Y^t|X^t,b^t}$, the probability distribution of the next state B^{t+T} only depends on the conditional probability distributions $P_{X^t|b^t}$, $P_{Y^t|X^t,b^t}$ and the belief update rule $s(\cdot,\cdot)$, and does not depend on the previous belief states (since the current belief state is equivalent to all of the past collected information) and actions (i.e., the Markov property). The expected penalty of the action $P_{Y^t|X^t,b^t}$ is then

$$\lambda \frac{R(Y^t|b^t)}{T} - G_{\vec{D},b^t} \quad (33)$$

where we use $G_{\vec{D},b^t}$ to represent the expected average network performance after delay $\vec{D}$ given the current belief state b^t.

Therefore, for a given value of T, solving Problem (32) is equivalent to minimizing the expected penalty of each step of the MDP.

Deterministic Strategies: Similar to the previous sections, we could easily see that the Pareto optimal decision-making strategies are still deterministic. The best decision at time $t+d$ is chosen by maximizing the expected network performance given the belief state B^{t+d}.

For the information collection strategy, it is slightly different. Given the decision-making strategies and the current belief state B^t, to minimize the objective function in Problem (32), one cannot simply map the network state X^t to the codeword Y^t which minimizes the penalty (i.e., the weighted sum of the codeword length and the expected network performance) within the period starting at t. This is because the transmitted information now not only affects the performance within the current period, but also influences the performance in the future, by affecting the future belief states. However, we could still show that the Pareto optimal information collection strategy should be deterministic.

As $P_{Y^t|X^t,b^t}$ is probabilistic, it is equivalent to a probability distribution of several deterministic actions, each of which, denoted by f_i, maps a true network state X_t to a codeword Y_t given state b^t, i.e., $Y_t = f_i(X^t, b^t)$. Each function f_i is in fact a deterministic information collection strategy. According to the theory of Markov Decision Process [23], for a given state b^t, the same action should always be taken to minimize the average penalty of each step in the long-term. Therefore, we can always choose a deterministic information collection strategy for any state b^t to obtain the optimal performance-rate relationship.

Memory Improves Network Performance: By comparing the Problems (26) and (32), we could see obviously that exploiting memory can achieve at least the same optimal performance-rate relationship as in the memoryless case, i.e., given the same amount of information collected per time unit, the maximum achievable network performance with memory exploited is at least the same as that of the memoryless case. The solution search space of Problem (32) is larger than that of Problem (26): any memoryless information collection strategy $P_{Y|X}$ could be written in the form of a memory dependent strategy $P_{Y|X,B}$; similarly, any memoryless decision-making strategy $P_{Z^d|Y}$ could also be written in a memory dependent form.

D. Summary

So far, we have analyzed three cases: state information with a random delay, the memoryless periodic information update scheme, and the memory-exploited periodic information update scheme. We have provided the formulations to solve the optimal performance-rate relationship. We have also shown that the Pareto optimal information collection and decision-making strategies should always be deterministic. By using the derived optimal performance-rate relationship, we can then obtain the optimal resource allocation between state information collection and data transmission.

IV. Example: Memoryless Information Update

In this section, we use a multiuser scheduling problem to illustrate the application of our framework. We consider the case where state information is collected periodically and no memory is used to store the past collected information. We first introduce the application scenario in Section IV-A, and then apply the framework to analyze the problem in Section IV-B, and finally present the results in Section IV-C with discussions.

A. Scenario

Consider the case where a controller has downlink data to transmit to N users. Assume time is slotted and the length of each slot is L bit time ($L \geq N$), where 1 bit time is the time taken to transmit 1 data bit. The channel between the controller and an arbitrary user is modeled as a stationary two-state Markov chain, i.e., the channel state at any time is one of two states $\{Good, Bad\}$. This model has been used widely in the literature to represent a channel with bursty noises [8] [24]. Assume the channel state transits at the beginning of each slot, and the transition matrix is symmetric, i.e., $Pr(Good|Good) = Pr(Bad|Bad) = p$ and $Pr(Bad|Good) = Pr(Good|Bad) = 1 - p$. Let $p \geq 0.5$, that is, the states of the channel are positively correlated over time. It is easy to see that the stationary probability distribution of the channel state is $Pr(Good) = Pr(Bad) = 0.5$. We assume that the channels between the controller and the users are independent Markov chains with identical statistics.

Assume the controller always has data to send to any user, and it can transmit data to only one user each time. In this case, we consider collecting only the channel state information. Assume time slots are grouped into frames, where each frame contains T slots. At the beginning of each frame, the controller collects some channel state information from the users (e.g., by probing the channels or requesting the users to report) and makes scheduling decisions for this frame based on the newly collected information (i.e., past information is not utilized).

The performance metric used in this example is the *throughput* of the data transmission duration (i.e., excluding the time used for information collection), and is defined as

$$\text{throughput} = \frac{\text{total number of data bits transmitted}}{\text{total transmission period (in bit time)}} \quad (34)$$

When a channel is *Good*, it can transmit 1 data bit per bit time; no data bit is transmitted otherwise.

The target is to find strategies for channel state information collection and scheduling to optimize the overall resource efficiency.

B. Analysis

First we need to identify the variables X, Y and Z in this problem. Assume Slot t_0 is the first slot of a frame. Let X_i denote the channel state of User i in Slot t_0, and $X_i = 0$ and 1 represent the events that User i's channel is in *Bad* and *Good* states, respectively. States of all the users are then $X = (X_1, X_2, ..., X_N)$. The collected information from User i at time t_0 is denoted by Y_i, and $Y = (Y_1, Y_2, ..., Y_N)$ represents all the collected information. Let Z^d be the scheduling decision made by the controller for Slot $t_0 + d$, and $Z^d = i$ means User i is scheduled for Slot $t_0 + d$. We also use X^d to denote network states at Slot $t_0 + d$. To find the optimal performance-rate relationship, we need to solve the following problem:

$$\min_{P_{Y|X}, \mathbf{P}_{Z^d|Y}, T} \lambda \frac{R(Y)}{T} - G_{\vec{D}} \quad (35)$$

Note that delays of the scheduling decisions in this case are fixed, i.e., $\vec{D} = (0, 1, ..., T-1)$ with probability 1. This is possible for cellular networks, as the base station and the mobile terminals are usually synchronized (e.g., as in the 3GPP standards).

Next we give a further analysis of $P_{Y|X}$, $R(Y)$, $\mathbf{P}_{Z^d|Y}$, and $G_{\vec{D}}$ in the problem.

Information Collection Strategy $P_{Y|X}$: $P_{Y|X}$ in this case can be further simplified. As discussed in Section III-B, the Pareto optimal solution should be a deterministic mapping from X to Y. Furthermore, we assume each user's information is collected independently. Therefore, we only need to consider deterministic mappings from X_i to Y_i. As X_i only has two states, there are only two possible ways to collect Y_i: (1) $Y_i = X_i$ (complete information is collected from User i) [2]; (2) $Y_i = \_\_$ (no information is collected from User i).

A strategy $P_{Y|X}$ that collects complete information from K users will output a vector $Y = (Y_1, Y_2, ..., Y_N)$ where K of the components have values 0 or 1, and $N-K$ of the components have values $\_\_$. As the users are homogeneous, all $P_{Y|X}$ that collect complete information from K users result in the same performance and rate. Therefore, instead of optimizing $P_{Y|X}$, we only need to search for the optimal value of K.

Information Rate $R(Y)$: When $Y_i = X_i$, the minimum codeword length $R(Y_i) = 1$; when $Y_i = \_\_$, the minimum length $R(Y_i) = 0$. Therefore, when complete channel state information of K users are collected, $R(Y) = \sum_i R(Y_i) = K$.

Decision-making Strategy $\mathbf{P}_{Z^d|Y}$: Using Equation (13), given collected information y, decision that maximizes throughput at time $t_0 + d$ is

$$z_o^d = \arg\max_i \sum_{x^d} p(x^d|y) g(x^d, i) \quad (36)$$

where $g(x^d, i)$ represents the network performance at time $t_0 + d$ given $X^d = x^d$ and $Z^d = i$. We have $g(x^d, i) = 1$ if $x_i^d = 1$; $g(x^d, i) = 0$ otherwise. Therefore, we have

$$\begin{aligned} z_o^d &= \arg\max_i \sum_{x^d : x_i^d = 1} p(x^d|y) \\ &= \arg\max_i p(X_i^d = 1|y) \end{aligned} \quad (37)$$

That is, at $t_0 + d$, we wish to schedule User i that has the highest probability to be *Good* given y.

As the states of the Markov channel is positively correlated over time, it is easy to see that the following strategy satisfies Equation (37):

- If $y_i = 1$ for some i, let $z_o^d = i$ where i satisfies the condition $y_i = 1$, i.e., choose a user that is *Good* at t_0;

[2] In fact, any scheme that maps $X_i = 0$ and $X_i = 1$ to distinct values of Y_i collects complete information from User i. For example, User i can report $Y_i = 1$ if $X_i = 1$, and use silence (i.e., $Y_i = \_\_$) to indicate $X_i = 0$. Complete information is collected in this case since X_i can be uniquely determined by Y_i. Without loss of generality, we assume $Y_i = X_i$ for complete information collection in the analysis.

- If $y_i = 0$ or __ for all i, let $z_o^d = i$ where i satisfies the condition $y_i =$ __ if $N > K$ (i.e., randomly select one from the remaining $N - K$ users), or let z_o^d be any i if $N = K$.

Performance $G_{\vec{D}}$: Finally, we need to characterize $G_{\vec{D}}$. Assume K users' information collected at t_0 and the above decision-making strategy is used. Using Equation (14), the expected throughput at $t_0 + d$ is calculated by

$$\begin{aligned} G_d &= \sum_y p(y) \sum_{x^d} p(x^d|y) g(x^d, z_o^d) \\ &= \sum_y p(y) p(X^d_{z_o^d} = 1|y) \\ &= \sum_{y: y_i = 1, \exists i} p(y) p(X^d_{z_o^d} = 1|y) + \sum_{y: y_i \neq 1, \forall i} p(y) p(X^d_{z_o^d} = 1|y) \end{aligned} \tag{38}$$

When $y_i = 1$ for some i, $p(X^d_{z_o^d} = 1|y)$ is the probability that a channel is still *Good* at $t_0 + d$ given that it is *Good* at t_0; when $y_i \neq 1$ for all i, $p(X^d_{z_o^d} = 1|y)$ is the probability that the randomly selected channel is *Good* at $t_0 + d$ when none of the K channels are *Good* at t_0. We use $\pi^d_{g \to g}$ and π^d_g to denote the former and the latter, respectively. Therefore, Equation (38) becomes

$$\begin{aligned} G_d &= \pi^d_{g \to g} \sum_{y: y_i = 1, \exists i} p(y) + \pi^d_g \sum_{y: y_i \neq 1, \forall i} p(y) \\ &= \alpha \pi^d_{g \to g} + (1 - \alpha) \pi^d_g \end{aligned} \tag{39}$$

where $\alpha = 1 - 0.5^K$ is the probability that at least one of the K channels is *Good* at t_0.

Based on the transition matrix of the Markov chain, we have the following:

$$\begin{aligned} \pi^d_{g \to g} &= 0.5 + 0.5(2p - 1)^d \\ \pi^d_{b \to g} &= 0.5 - 0.5(2p - 1)^d \\ \pi^d_g &= 0.5 \quad (\text{for } N > K) \\ \pi^d_g &= \pi^d_{b \to g} \quad (\text{for } N = K) \end{aligned} \tag{40}$$

where $\pi^d_{b \to g}$ is the probability that a channel is *Good* at $t_0 + d$ given that it is *Bad* at t_0. The above relations are derived directly from properties of the Markov chain. As space is limited, we leave the derivations in the supplementary materials.

Since the information collection requires K bit time, the data transmission duration of this frame is $TL - K$ bit time. The expected throughput is

$$G_{\vec{D}} = \frac{L \sum_{d=0}^{T-1} G_d - K G_0}{TL - K} \tag{41}$$

which is a weighted average of G_d with different delay d.

Optimal Performance-rate Relationship: Based on the above analysis, Problem (35) can be converted to the following:

$$\min_{K,T} \lambda \frac{K}{T} - \frac{L \sum_{d=0}^{T-1} G_d - K G_0}{TL - K} \tag{42}$$

By changing the value of λ, we can get the optimal performance-rate relationship.

Net Data Rate: To find the optimal resource allocation scheme, we calculate the net data rate R_e

$$R_e = \frac{G_{\vec{D}}(TL - K)}{TL} \tag{43}$$

The optimal resource allocation scheme is then obtained by maximizing R_e in the above equation.

C. Results

Temporal Decline of the Value of Information: As shown in Figure 1(a), the expected throughput declines as the delay increases when $p < 1$. The decline rate is larger with a smaller value of p since the channel state changes more rapidly in this case. When $p = 0.5$, the channel states of consecutive slots are independent, and G_d drops to 0.5 for $d \geq 1$ indicating that the collected information becomes completely useless when it is independent from the channel states. When $p = 1$, the channel stays in the same state forever, and the expected throughput remains constant.

The Performance-Rate Relationship: Figure 1(b) illustrates the relationship between $R(Y)/T$ and $G_{\vec{D}}$. The red solid line with dotted markers represents the highest $G_{\vec{D}}$ that could be achieved given $R(Y)/T$ bits information per time slot are collected (i.e., the optimal performance-rate relationship). Figure 1(b) also plots the relationship curves with $T = 1, 3$ and 5.

In general, the highest achievable $G_{\vec{D}}$ increases as more information is collected. However, when the value of T is fixed (e.g., $T = 3$), the average expected throughput can decrease if too much information is collected. This looks counter-intuitive. The value of $G_{\vec{D}}$ drops here because the information collection process postpones the information utilization, which increases the information delay, thus reducing the information's value to network protocols.

The Optimal Resource Allocation Scheme: By using the optimal performance-rate relationship, we calculate the optimal resource allocation scheme that maximizes the net data rate. In Tables I and II, we give the optimal values of T and K, denoted by T^* and K^*, respectively, under different network conditions.

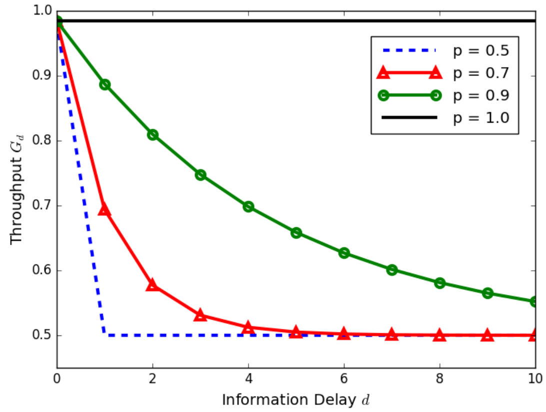
TABLE I: $N = 10$, $L = 15$

	T^*	K^*	R_e
$p = 0.75$	1	2	0.7583
$p = 0.85$	2	3	0.7781
$p = 0.95$	4	4	0.8384

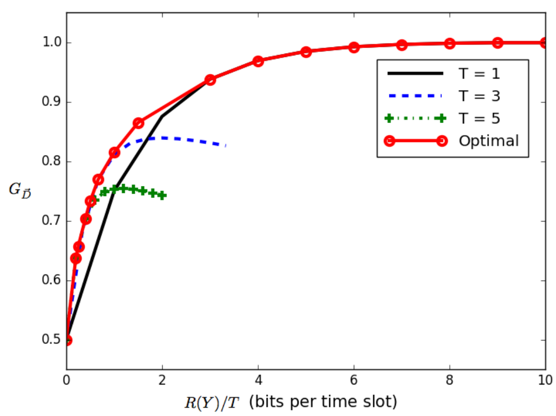
TABLE II: $N = 5$, $p = 0.8$

	T^*	K^*	R_e
$L = 5$	3	2	0.6283
$L = 12$	2	3	0.7328
$L = 50$	1	4	0.8912

In Table I, with N and L fixed and the transition probability p increasing, both of T^* and K^* increases. This means, when the network states change more rapidly (smaller p), the optimal information collection scheme is to update less information each time as the information will become useless quickly, and to update more frequently to keep track of the network states. When the network condition is more stable, the benefit of the collected information can last for a longer period, thus the information update frequency can be lessened. The net data rate also increases with the transition probability p. This is



(a) G_d vs. d



(b) $G_{\vec{D}}$ vs. $R(Y)/T$

Fig. 1: (a) G_d as a function of d ($K = 5$ and $N = 10$); (b) $G_{\vec{D}}$ as a function of $R(Y)/T$ ($L = 15$, $N = 10$, $p = 0.85$)

because, when p is larger, the network state becomes more predictable; thus less information overhead is required and the significance of collecting information decreases slowly.

When the duration of the slot becomes longer, Table II shows that more information should be collected (i.e., increase the update frequency and the amount of information updated each time), as the relative overhead of collecting information decreases as L grows. In addition, we can see that the relatively small overhead improves the net data rate of the system.

V. Example: Information Update with Memory

In Section IV, we discussed the multiuser scheduling problem without utilizing the past collected information. In this part, we consider utilizing the past information, and study how the deployment of memory affects the relationship between $R(Y)/T$ and $G_{\vec{D}}$ and the optimal resource allocation scheme.

A. Scenario

The scenario is basically the same as that in Section IV-A. The difference is that, the information collection and the decision-making now depend on the past collected information. At the beginning of a frame, the controller first looks at all the information collected in the past, based on which it selects a subset of the channels to collect state information and then assigns a channel for data transmission by considering all the available information.

B. Analysis

As discussed in Section III-C, we could use the belief state to represent all the past collected information. Let $Y^t = (Y_1^t, Y_2^t, ..., Y_N^t)$ denote the collected information at Slot t. We use $B^t = (B_1^t, B_2^t, ..., B_N^t)$ to represent the controller's belief about the channel states of the N users in Slot t. To find the optimal performance-rate relationship, we first need to have the belief update function $s(\cdot, \cdot)$, and then solve Problem (32).

Memory Update Policy: Assume given all the information collected before Slot t (i.e., excluding the information collection at Slot t), the controller believes that Channel i has a probability B_i^t to be *Good* in Slot t. The controller updates the belief states at the beginning of each time slot (before information collection, if any). Consider Channel i at Slot t:

- If no information is collected about Channel i at Slot $t-1$:
  The controller estimates the state of Channel i based on the property of the Markov Chain, and
$$B_i^t = pB_i^{t-1} + (1-p)(1 - B_i^{t-1}) \tag{44}$$
- If information about Channel i is collected at Slot $t-1$: The controller updates the belief based on the collected information at time $t-1$ and the property of the Markov Chain, and we have
$$B_i^t = \begin{cases} p & \text{if } Y_i^{t-1} = 1 \\ 1-p & \text{if } Y_i^{t-1} = 0 \end{cases} \tag{45}$$
  where $Y_i^{t-1} = 1$ and 0 represent the cases where Channel i is *Good* and *Bad* at Slot $t-1$, respectively.

It is obvious that maintaining the belief state vector is equivalent to memorizing all the collected state information in the past. For Channel i, due to the Markov property, we only need to remember the latest collected state information.

Similar to Section IV-B, we analyze the information collection strategy $P_{Y^t|X^t,B^t}$ and decision-making strategy $\mathbf{P}_{Z^d|B^d}$.

Information Collection Strategy $P_{Y^t|X^t,B^t}$: In this paper, we consider the following strategy to collect information from K channels. If information is updated at Slot t:

- The controller chooses the K channels that have the highest belief states in B^t (with random tie-breaking) to collect information. That is, the controller selects a subset $\mathcal{A} \subset \mathcal{I} = \{1, 2, ..., N\}$ such that $|\mathcal{A}| = K$ and $B_i^t \geq B_j^t$ for any $i \in \mathcal{A}$ and $j \in \mathcal{A}^c = \mathcal{I} \backslash \mathcal{A}$.

Decision-making Strategy $\mathbf{P}_{Z^{t+d}|B^{t+d}}$: As shown in Equation (37), the controller should choose the user that has the highest probability to be *Good* given all the collected information. Let Z^{t+d} be the scheduling decision at Slot $t+d$. We have the following decision-making strategy:
$$Z^{t+d} = \arg\max_i B_i^{t+d} \tag{46}$$

If multiple users have the same belief states, the controller randomly chooses one.

Note 1. As discussed, the memory-exploited information update scheme forms a Markov Decision Process (MDP) problem. The information collection strategy presented above is a greedy policy, which means it only tries to maximize the average expected throughput of the frame immediately after the information collection. The greedy policy is generally a suboptimal solution of MDP. Greedy policy is shown to maximize the long-term reward in [25] for a multiuser scheduling problem. However, their problem setting is different from ours. As the focus of this paper is to analyze the trade-off between information collection and data transmission, we do not attempt to search for the best policy to collect K users' channel state information; we only use the mentioned greedy policy to illustrate the performance-rate trade-off and the resource allocation between information collection and data transmission.

Calculation of $R(Y^t|B^t)$ and $G_{\vec{D}}$: Again, when state information of K channels is collected, the average information per time slot is $R(Y^t|B^t) = K/T$.

Similar to Section IV, we use G_d to denote the expected system throughput at Slot $t+d$, where Slot t is the first slot of a generic frame. The calculation of G_d is the same as in Equation (39), except the calculations of α and π_g^d:

$$\begin{aligned} \alpha &= 1 - \prod_{i \in \mathcal{A}} (1 - B_i^t) \\ \pi_g^d &= B_j^t \pi_{g\to g}^d + (1 - B_j^t)\pi_{b\to g}^d \quad (\text{for } N > K) \end{aligned} \tag{47}$$

where $\pi_{g\to g}^d$ and $\pi_{b\to g}^d$ are defined in Equation (40), respectively, and B_j^t is the belief state of the chosen channel j at time t given none of the selected K channels are *Good*.

The expected throughput of the system $G_{\vec{D}}$ is then calculated as Equation (41), and the net data rate R_e is derived from Equation (43).

C. Results

We use computer simulations to estimate the performance-rate relationship and determine the optimal resource allocation scheme. With fixed K and T, for the j^{th} simulation run, we calculate the average throughput $\hat{G}_{\vec{D}}^j$ at steady state (around 40000 frames after start). We conduct $n = 51$ simulation runs with independent initializations, and estimate $G_{\vec{D}}$ as $G_{\vec{D}} = \frac{1}{n}\sum_j \hat{G}_{\vec{D}}^j$. The computed 95% confidence intervals of the results have lengths smaller than 0.002.

The Performance-Rate Relationship: Figure 2 shows the relationship between the average information rate and the average throughput. The figure is similar to Figure 1(b) in Section IV where past information is not utilized. In general, when the update period T is not fixed, the average throughput increases as more information is collected per slot. However, if T is fixed, collecting more state information may not be beneficial for performance improvement, and it may even hurt the system performance, due to the inaccuracy caused by the information delay.

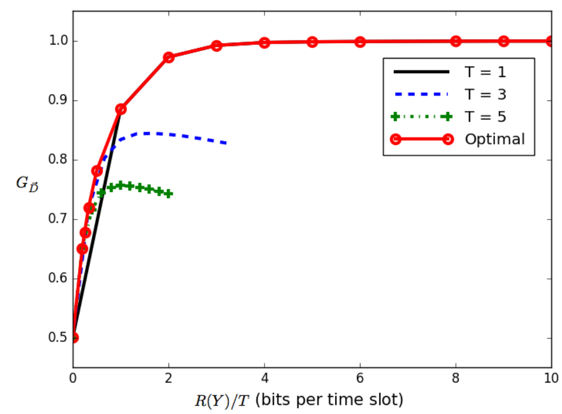


Fig. 2: The relationship between $R(Y^t|B^t)$ and $G_{\vec{D}}$ with memory at the controller ($L = 15$, $N = 10$, $p = 0.85$)

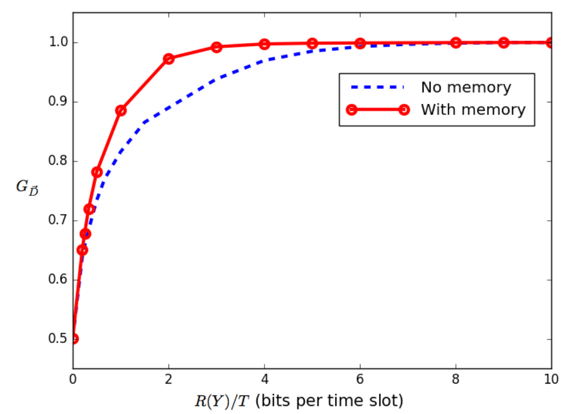


Fig. 3: The effect of having memory at the controller on the performance-rate relationship ($L = 15$, $N = 10$, $p = 0.85$)

Effect of Memory: Figure 3 compares the performance-rate relationships with and without utilizing past information, and demonstrates the effect of memory. As we could see, given a fixed value of information rate, the use of memory effectively improves the average throughput of the system.

The Optimal Resource Allocation Scheme: We have also determined the optimal resource allocation schemes, and presented the results in Tables III and IV. By comparing the results with Tables I and II, we can observe how the existence of memory affects the optimal resource allocation. In general, when memory is used, the information update frequency should be increased, and less information needs to be collected each time. This strategy effectively utilizes the correlations between the information updates at consecutive frames. Furthermore, having memory installed at the controller improves the effective data rate of the system.

VI. Example: Information With A Random Delay

In this section, we use a routing problem as an example to analyze the optimal performance-rate relationship for network state information with a random delay.

TABLE III: $N = 10$, $L = 15$

	T^*	K^*	R_e
$p = 0.75$	1	2	0.8182
$p = 0.85$	1	2	0.8425
$p = 0.95$	2	2	0.8955

TABLE IV: $N = 5$, $p = 0.8$

	T^*	K^*	R_e
$L = 5$	1	1	0.6825
$L = 12$	1	1	0.7841
$L = 50$	1	2	0.9034

We consider a wireless grid network with $N \times N$ nodes. A node has coordinate (i, j) if it is in the i^{th} row and the j^{th} column. Figure 4 gives a 2×2 network and coordinates of the nodes are also given. Node (i, j) has direct communication links with Nodes $(i-1, j)$, $(i+1, j)$, $(i, j-1)$ and $(i, j+1)$ (if the coordinates exist in the network). We assume Node $(1, 1)$ is Source, and Node (N, N) is Destination.

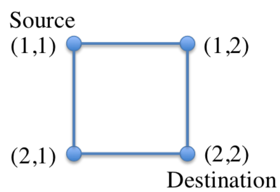


Fig. 4: A 2×2 grid network

All the nodes except the Source and the Destination are relay nodes, which may be unreachable for data transmission sometimes (e.g., in power saving mode or due to interference). We assume time is slotted, and the states of a node over time follows the same Markov chain as in Section IV, and the states of the nodes are independent Markov chains with identical statistics.

Let $X = (X_1, X_2, ..., X_{N^2-2})$ be the states of the relay nodes at Slot t, and $X_i = 1$ means Relay i is reachable. $Y = (Y_1, Y_2, ..., Y_{N^2-2})$ is the collected information about the node states. We assume the Source receives a data transmission request at a random time $t+D$ after collecting the information, where D is a random integer uniformly distributed within interval $[d_1, d_2]$ where d_1 and d_2 are also integers. Given $D = d$, let X^d be the states of relay nodes at $t+d$, the Source searches a route Z^d, which is a sequence of $(2N-3)$ relay nodes to send data to the Destination. In this problem, we consider the routes where nodes only forward data downward or to the right (i.e., no turning back). Furthermore, since data transmission takes time, we assume it takes one time slot to transmit data on each hop. When all nodes along a route are reachable during data transmission, the route is said to be *active*. The performance metric in this problem is the packet delivery ratio, i.e., the probability that Source chooses an active route.

Information Collection Strategy $P_{Y|X}$: Similar to the previous section, information from different relay nodes are collected independently, and each node has only binary states. Therefore, instead of searching all possible $P_{Y|X}$, we only need to search for $I = (I_1, I_2, ..., I_{N^2-2})$, where $I_i = 1$ means complete information is collected from Relay i, i.e., $Y_i = X_i$ if $I_i = 1$ and $Y_i = \_\_$ if $I_i = 0$.

Information Rate $R(Y)$: The amount of collected information $R(Y)$ is then the number of 1s in I.

Decision-making Strategy $\mathbf{P}_{Z^d|Y}$: Let $z^d = (n_1, n_2, ..., n_{2N-3})$, where $1 \leq n_i, n_j \leq N^2 - 2$ and $n_i \neq n_j$. Assume Relay n_j has coordinates (u_{n_j}, v_{n_j}). To make sure z^d is a route, the following constraints should be satisfied: 1) $1 \leq u_{n_j}, v_{n_j} \leq N$; 2) $u_{n_{j+1}} + v_{n_{j+1}} = u_{n_j} + v_{n_j} + 1$.

As we assume data transmission takes time, the performance does not only depend on X^d, but also on X^{d+1}, $X^{d+2}, ...$. To apply our framework in this setting, we introduce a new variable called *the modified network state* $\tilde{X}^d$ to calculate the expected performance, where $\tilde{X}_i^d = X_i^{d+k-1}$ if Relay i is a k-hop neighbor of Source, i.e., $u_i + v_i = k + 2$.

We have $g(\tilde{x}^d, z^d) = 1$ if $\tilde{x}_{n_j}^d = 1$ for all j. Using Equation (13), given information y, the decision that maximizes the packet delivery ratio is

$$\begin{aligned} z_o^d &= \arg\max_{z^d} \sum_{\tilde{x}^d} p(\tilde{x}^d|y) g(\tilde{x}^d, z^d) \\ &= \arg\max_{(n_1, n_2, ..., n_{2N-3})} \sum_{\tilde{x}^d : \tilde{x}_{n_j}^d = 1} p(\tilde{x}^d|y) \\ &= \arg\max_{(n_1, n_2, ..., n_{2N-3})} p(\tilde{X}_{n_1}^d = 1, ..., \tilde{X}_{n_{2N-3}}^d = 1|y) \end{aligned} \tag{48}$$

where the maximization is constrained such that z^d is a route. That is, we pick the route that has the highest probability to be active given information y. The value of $p(\tilde{X}_{n_1}^d = 1, ..., \tilde{X}_{n_{2N-3}}^d = 1|y)$ could be easily derived from the property of a Markov chain, thus the calculation is omitted here.

The Performance $G_{\vec{D}}$: The expected packet delivery ratio $G_{\vec{D}} = \sum_d p(d) G_d$, where $p(d) = \frac{1}{d_2 - d_1 + 1}$. Using Equation (14), we can calculate G_d as follows:

$$\begin{aligned} G_d &= \sum_y p(y) \sum_{\tilde{x}^d} p(\tilde{x}^d|y) g(\tilde{x}^d, z_o^d) \\ &= \sum_y p(y) p(\tilde{X}_{n_1^*}^d = 1, ..., \tilde{X}_{n_{2N-3}^*}^d = 1|y) \end{aligned} \tag{49}$$

where $z_o^d = (n_1^*, ..., n_{2N-3}^*)$.

The Performance-Rate Relationship: Based on the above analysis, we can solve the following problem:

$$\min_{I : I_i \in \{0,1\}} \lambda R(Y) - G_{\vec{D}} \tag{50}$$

Figure 5 gives the result for a 3×3 network, with a random delay in intervals $[0, 0]$ (i.e., no delay), $[0, 2]$ and $[2, 4]$. It is easy to see that as the average delay becomes larger, the expected performance $G_{\vec{D}}$ decreases given the same amount of information collected.

VII. Conclusions

In this paper, we discussed the impact of information delay on the relationship between the amount of collected state information and network performance, and hence its effect on the optimal resource allocation scheme. We first considered the optimal performance-rate relationship when there is a random information delay. Then we extended the framework by assuming a periodic information collection strategy, and discussed both memoryless and memory-exploited scenarios. For all the three cases, we have analyzed the optimal relationship between the average information collected per time

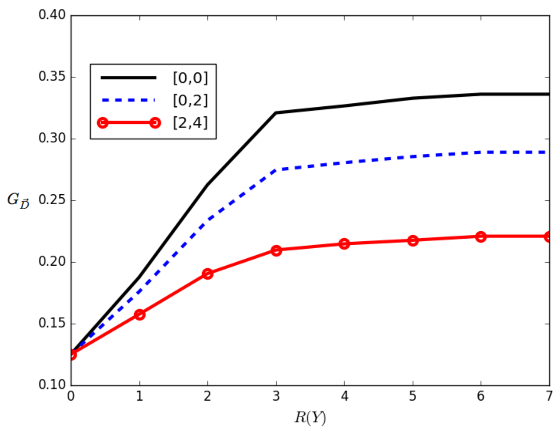


Fig. 5: $G_{\vec{D}}$ vs. $R(Y)$ with different delay intervals ($p = 0.9$ for the Markov chains)

unit and the expected network performance over the update period, and the structures of the solutions that achieve the optimal relationship. Based on the relationship obtained, the optimal resource allocation scheme (i.e., the optimal information update frequency and the amount of information updated each time) could be derived.

Application of the framework is illustrated with examples of multiuser scheduling and multi-hop routing, and some observations about the optimal resource allocation scheme in time-varying networks are obtained. First, when there is information delay, collecting more information may not result in better network performance given a fixed information update period. Second, for a network with rapidly changing states, information should be updated frequently while only a small amount of information should be collected each time. In addition, as the slot length increases, more information should be collected. Finally, memorizing past information can indeed improve the network performance, and it is better to have a higher information update frequency with less information collected each time to utilize the correlations between consecutive updates.

In the future, we wish to analyze the scenario where the collected information experiences heterogeneous delays. For distributed networks, when the information delay is heterogeneous, different nodes will have different sets of knowledge about the network states. We would like to discuss the optimal resource allocation scheme in this scenario.

References

[1] J. Chuai and V.O.K. Li, "A framework for balancing information collection and data transmission," in *Proceedings of IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, pp. 2208–2212, IEEE, 2013.

[2] ——, "The temporal value of information to network protocols An analytical framework," in *Proceedings of IEEE 26th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, pp. 1707-1711, IEEE, 2015.

[3] K. Seada, A. Helmy, and R. Govindan, "On the effect of localization errors on geographic face routing in sensor networks," in *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, pp. 71–80, ACM, 2004.

[4] J. Hong and V.O.K. Li, "Effect of information on routing performance in multi-hop wireless networks," in *Proceedings of IEEE 71st Vehicular Technology Conference (VTC 2010-Spring)*, pp. 1–5, IEEE, 2010.

[5] H. Viswanathan, "Capacity of Markov channels with receiver CSI and delayed feedback," *IEEE Transactions on Information Theory*, vol. 45, no. 2, pp. 761–771, 1999.

[6] G. Caire, N. Jindal, M. Kobayashi, and N. Ravindran, "Multiuser MIMO achievable rates with downlink training and channel state feedback," *IEEE Transactions on Information Theory*, vol. 56, no. 6, pp. 2845–2866, 2010.

[7] U. Basher, A. Shirazi, and H.H. Permuter, "Capacity region of finite state multiple-access channels with delayed state information at the transmitters," *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3430–3452, 2012.

[8] S. Murugesan, P. Schniter, and N.B. Shroff, "Multiuser scheduling in a Markov-modeled downlink using randomly delayed ARQ feedback," *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 1025–1042, 2012.

[9] Y. Jia, I. Nikolaidis, and P. Gburzynski, "Multiple path routing in networks with inaccurate link state information," in *Proceedings of IEEE International Conference on Communications*, vol. 8, pp. 2583–2587, IEEE, 2001.

[10] L. Ying and S. Shakkottai, "On throughput optimality with delayed network-state information," *IEEE Transactions on Information Theory*, vol. 57, no. 8, pp. 5116–5132, 2011.

[11] B. Hassibi and B.M. Hochwald, "How much training is needed in multiple-antenna wireless links?" *IEEE Transactions on Information Theory*, vol. 49, no. 4, pp. 951–963, 2003.

[12] M. Kobayashi, G. Caire, and N. Jindal, "How much training and feedback are needed in MIMO broadcast channels?" in *Proceedings of IEEE International Symposium on Information Theory*, pp. 2663–2667, IEEE, 2008.

[13] P. Chaporkar, A. Proutiere, H. Asnani, and A. Karandikar, "Scheduling with limited information in wireless systems," in *Proceedings of the Tenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 75–84, ACM, 2009.

[14] Z. Ji, Y. Yang, J. Zhou, M. Takai, and R. Bagrodia, "Exploiting medium access diversity in rate adaptive wireless LANs," in *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking*, pp. 345–359, ACM, 2004.

[15] S. Guha, K. Munagala, and S. Sarkar, "Jointly optimal transmission and probing strategies for multichannel wireless systems," in *Proceedings of IEEE 40th Annual Conference on Information Sciences and Systems*, pp. 955–960, IEEE, 2006.

[16] J. Hong and V.O.K. Li, "The effect of information on scheduling performance in multi-hop wireless networks," *IEEE Transactions on Wireless Communications*, vol. 9, no. 10, pp. 3238–3246, 2010.

[17] J. Hong and V.O.K. Li, "Minimum required information to achieve a performance target in a network with memoryless states," *IEEE Transactions on Communications*, vol. 63, no. 10, pp. 3754–3764, 2015.

[18] L. Tassiulas and A. Ephremides, "Dynamic server allocation to parallel queues with randomly varying connectivity," *IEEE Transactions on Information Theory*, vol. 39, no. 2, pp. 466–478, 1993.

[19] K. Liu and Q. Zhao, "Indexability of restless bandit problems and optimality of Whittle's index for dynamic multichannel access," *IEEE Transactions on Information Theory*, vol. 56, no. 11, pp. 5547–5567, 2010.

[20] R.D. Smallwood and E.J. Sondik, "The optimal control of partially observable Markov processes over a finite horizon," *Operations Research*, vol. 21, no. 5, pp. 1071–1088, 1973.

[21] R.M. Gray and D.L. Neuhoff, "Quantization," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2325–2383, 1998.

[22] N.T. Gaarder and D. Slepian, "On optimal finite-state digital transmission systems," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 167–186, 1982.

[23] M.L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*, John Wiley & Sons, 2014.

[24] P. Bhagwat, P. Bhattacharya, A. Krishna, and S.K. Tripathi, "Enhancing throughput over wireless LANs using channel state dependent packet scheduling," in *Proceedings of IEEE Fifteenth Annual Joint Conference of the IEEE Computer Societies. Networking the Next Generation (INFOCOM)*, vol. 3, pp. 1133–1140, IEEE, 1996.

[25] S.H.A. Ahmad and M. Liu, "Multi-channel opportunistic access: A case of restless bandits with multiple plays," in *47th Annual Allerton Conference on Communication, Control, and Computing*, pp. 1361–1368, IEEE, 2009.