Power and Delay Optimization in Multi-Hop Wireless Networks

Li Xia, Basem Shihada

Abstract

In this paper, we study the optimization problem of transmission power and delay in a multi-hop wireless network consisting of multiple nodes. The goal is to determine the optimal policy of transmission rates at various buffer and channel states, in order to minimize the power consumption and the queueing delay of the whole network. With the assumptions of interference-free links and independently and identically distributed (i.i.d.) channel states, we formulate this problem using a semi-open Jackson network model for data transmission and a Markov model for channel states transition. We derive a difference equation of the system performance under any two different policies. The necessary and sufficient condition of optimal policy is obtained. We also prove that the system performance is monotonic with respect to (w.r.t.) the transmission rate and the optimal transmission rate can be either maximal or minimal. That is, the "bang-bang" control is an optimal control. This optimality structure greatly reduces the problem complexity. Furthermore, we develop an iterative algorithm to find the optimal solution. Finally, we conduct the simulation experiments to demonstrate the effectiveness of our approach. We hope our work can shed some insights on solving this complicated optimization problem.

Keywords—**Power control, Markov decision process, queueing network, discrete event dynamic system**

I. INTRODUCTION

Wireless communication technologies have revolutionized our lifestyle by providing flexible infrastructure and ubiquitous accesses and computing. One of the most important yet challenging tasks in wireless communications is to determine the optimal transmission policy so as to reduce the power consumption without increasing the queueing delays significantly. This is a particularly challenging problem for multi-hop wireless networks, such as wireless sensor and ad-hoc networks [4], [14], [32].

This problem has its roots in the tradeoff between power consumption and queueing delay. An increase in transmission power can increase the data rate and as a result reduce this transmitting node's queueing delay. However, since the nodes are interconnected, the increase in data rate of one transmitter may increase the queueing delay of the next-hop nodes due to an increase in the data arrival rates to those nodes. Such a scheme may induce congestions in some nodes and cannot reduce the overall end-to-end queueing delay of a packet traversing multiple hops. Moreover, the actual transmission rate may fluctuate due to varying channel states, and thus also affect the transmission power consumption. In this paper, we assume the radio channels are interference free as we adopt the time division multiple access (TDMA) or orthogonal frequency division multiple access (OFDMA) to manipulatively avoid the media access contention. Our goal is to perform an efficient rate control via power allocation for all the nodes in a network according to the buffer and channel states of each node, such that a graceful trade-off between power consumption and queueing delay of the entire network can be achieved.

The optimization of transmission power and queueing delay in wireless networks has been discussed in the literature. In [4], it was discussed how to model the fading channel to optimize the transmission power under the constraint of queueing delay with 3 different models: block-fading channel, mutual information model, and telatar model. Similarly, in [11], [16], it was also showed how to optimize the transmission power under a delay constraint, while other studies investigated the delay optimization problem with a power constraint [14]. All these papers have used a Markov chain to model the data transmission process along with a dynamic programming technique for the optimization. Miao and Cassandras used a two-fold decomposing approach to optimize the energy under a constraint of queueing delay [23]. This approach was based on a structured analysis of sample paths and it was executed on an off-line basis. However, all these studies mentioned above considered only a single node and a single hop between the node and the base station, which is much simpler than the case of multi-hop networks.

For a multi-hop wireless network consisting of multiple nodes, the energy and delay optimization problem is much more complicated. In the literature, some studies focused only on the performance

metrics of a certain flow with a specific multi-hop route, not on the performance of all the nodes in the entire network [10], [26], [37]. In [13], authors assumed the links are interference free with TDMA scheme and used a cross-layer method to optimize the energy and delay for a small-scale sensor network after some simplifications. In [17], the original complicated problem was decoupled into several small sub-problems to approximately solve the energy and reliability problem in wireless sensor networks. In [22], authors formulated the energy and delay optimization problem as a game theoretical problem and used the local information to study the Nash equilibrium. In summary, the energy and delay optimization in a multi-hop wireless network is very complicated. All of the above studies did simplifications to reduce the problem complexity to a treatable level. The simplifications include the assumption of interference-free link, small-scale network, problem decoupling, etc.

The difficulty of this problem mainly comes from two factors. One is the high dimensionality caused by the multiple nodes. Every node has its own buffer status and channel status, which makes the system state high-dimensional and the complexity of the optimization problem grow exponentially. The other factor is the interconnection among multiple nodes, which makes the original complicated problem undecomposable. The traffic dynamics of data transmission along the multi-hop path greatly increases the problem complexity. For example, in a multi-hop wireless network, the transmission rate of a node can be increased by increasing its transmission power. Such an action will increase the data arrival rates of the nodes on the next-hop and may also increase the queueing delay of those next-hop nodes. Therefore, in a multi-hop wireless network, the transmission rate of a node will not only affect the current node, but also other nodes in the network. We need to schedule the transmission rate of every node by considering both the state of current node and the states of other nodes in the network.

Motivated by the above observations, we consider the following assumptions to be able to investigate this problem. We consider the static nodes [12], and therefore, the network topology is stable or relative stable compared to the time scale of our optimization process. We assume that the radio channels among nodes are interference free, which is reasonable in the situation where the nodes use TDMA or FQDMA techniques to avoid the contention [13]. The channel status of every node is i.i.d. We study a small-scale network to illustrate the merits of our work. Although our method suffer the curse of dimensionality, the analysis results of a small-scale network give some insights to solve this complicated problem. The data to be transmitted is generated at each node according to a certain probability distribution. This data is then transmitted across the network following a given routing scheme. When the data is received by a node, it will be queued in the buffer and waiting for transmission according to the scheduling policy. Our goal is to determine the transmission rates of all the nodes according to their channel and buffer

states, in order to optimize the long-run average performance including the power consumption and the queueing delay of the entire network.

In order to formulate the above problem, we use a two-level model: First, a semi-open Jackson network is used to model the data transmission process among nodes. The Jackson network model is a classical model in queueing systems which can capture the dynamics of multiple interconnected stations. It is suitable to model the traffic dynamics in a multi-hop wireless network. Second, the channel state transition process of nodes is modeled as a Markov process [28], [31]. Since a Jackson network can also be represented as a Markov process, we combine the Jackson network model and the Markov channel model to form an integrated Markov model.

The performance function jointly considers both the power consumption and the queueing delay. Based on the integrated Markov model, we use the performance potential theory [6], [7], [33], [34] to solve the formulated optimization problem. The performance potential theory is an efficient approach to optimize the performance of Markov systems from a sensitivity-based viewpoint. We first derive a difference equation of the system performance under any two sets of transmission rates. With the difference equation, some interesting optimality theorems are proved, which include that the system performance is monotonic w.r.t. the transmission rate and the optimal transmission rate can be either maximal or minimal. We further develop an online iterative algorithm to find the optimal policy. This proposed algorithm does not require the full prior knowledge of the system, such as the routing probabilities and the statistics of channel states. Such information can be learned from the sample path using an online algorithm. Note that the online implementation does not require interrupting the normal operation of the system, which is a desirable property in practice [27].

The main contribution of this paper includes the following aspects. First, our paper aims to handle the power and delay optimization for multiple nodes after reasonable simplifications, while many previous studies focused on the optimization for a single node. Since the nodes are interconnected, the transmission policies of nodes have mutual effects. We have to consider the dynamics of all the nodes to optimize the average performance of the entire network. Second, we obtain the necessary and sufficient condition of the optimal policy of this problem. We derive the structural properties of the optimal policy, which includes that the system performance is monotonic w.r.t. the transmission rate and the optimal transmission rate can be either maximal or minimal. That is, the "bang-bang" control is an optimal control for this problem [1], [2]. These structural properties greatly simplify the optimization problem and give valuable insights to further study a more complicated formulation of this problem. Third, we develop an iterative algorithm to find the optimal policy and this algorithm can be online implemented, which is an important advantage

for the practical application.

The remainder of the paper is organized as follows. In Section II, we present a detailed mathematical formulation of our problem. In Section III, we derive the performance difference equation for the formulated problem. Optimization algorithms and related optimality properties are also obtained in this section. In Section IV, we use simulations to demonstrate the effectiveness of our approach. Finally, we conclude this paper with Section V.

II. PROBLEM FORMULATION

In this section, we use a semi-open Jackson network to model the data transmission process among the nodes, and a Markov process to model the transition of channel states of the nodes. By integrating these two processes as a single Markov model, we formulate this problem as a Markov decision process (MDP).

Suppose that a wireless network consists of M communication nodes. The network is static or has little mobility. Therefore, the locations of nodes are fixed and the topology of the network is relatively stable. This assumption is reasonable for many applications of wireless sensor networks or static ad-hoc networks, such as the battlefield surveillance or environment monitoring. Each node generates data packets. We assume that all these data generation processes are independent and follow Poisson distributions with an arrival rate denoted as λ_{0i} for node i, $i = 1, 2, \cdots, M$. The data packets are relayed by the adjacent nodes per the routing mechanism. Since the packets are generated at different nodes and have different destinations, the transition paths of data packets are approximated with a probabilistic routing mechanism. From a perspective of statistics, when a packet arrives at node i, it has to be further transferred to node j with a routing probability q_{ij}. On the other hand, the packet may find that node i is exactly its final destination with probability q_{i0}, and leaves the network without further transfer. Obviously, we have $q_{ii} = 0$ and $\sum_{j=0}^{M} q_{ij} = 1$, for all $i = 1, 2, \cdots, M$. Note that these routing probabilities can be statistically estimated according to the routing data of the network. We assume that the route of the network is relatively static and the routing probability q_{ij} is constant. This is reasonable in the scenario where the network adopts a static routing table.

As the buffer size of nodes is limited, packets may be dropped due to queue overflow. We assume that the arriving rates of packets are within the stable region of the network and the system is ergodic [29]. Denote N as the upper limit of the total number of packets in the network. When N is large, the overflow probability is small enough to be ignored in such stability framework. We truncate all of these small overflow probabilities to approximate the original system. When there are already N packets in the

network, any newly generated packet will be dropped immediately. Since N is large and the dropping probability is very small, this packet dropping model can be approximated to the model where each node has their own capacity limit. As the probability of the buffer of one node being full is very small, the probability of the buffers of multiple nodes being full is high-order small. We can omit the later situation and approximately assume that the network has a total capacity N. As we will see later, this assumption offers us convenience to treat the mathematical model, with the cost of a fair approximation to the original problem.

The number of packets at node i is denoted as n_i, where $0 \leq n_i \leq N$. Since various applications may generate different packet sizes, the size of data packets is assumed exponentially distributed with a unit mean. Therefore, the transmission time of each packet is also exponentially distributed, and the mean transmission time is $1/\mu_i$ where μ_i is the transmission rate (also called service rate) of node i, $i = 1, 2, \cdots, M$. Therefore, we can formulate the data transmission process as a semi-open Jackson network [9]. The state transition of the system is modeled as a Markov process. For example, for a network with $M = 2$ nodes, the state transition diagram is illustrated in Fig. 1, where the numbers i, j in the circle represent the number of packets at node 1 and 2, respectively.
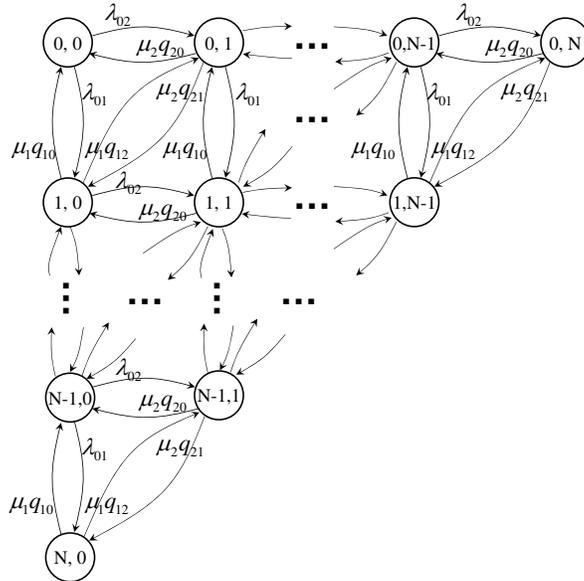

Fig. 1. An example of state transition diagram of packet transmission in a network with two nodes.

Because the customers (data packets) can enter or exit the network and the network has a capacity limit, a semi-open Jackson network is required for the modeling process. Nonetheless, a semi-open Jackson

network can be converted to a closed Jackson network model, such that the whole network does not have any external customer arrival or departure. Thus, we deal with a closed Jackson network for the original problem by performing the following transformation: The exterior is taken as a virtual server 0 (node 0) with a service rate $\mu_0 \triangleq \sum_{i=1}^{M} \lambda_{0i}$ and routing probabilities $q_{0i} \triangleq \lambda_{0i}/\mu_0$, $i = 1, 2, \cdots, M$. The queue length of server 0 is defined as $n_0 \triangleq N - \sum_{i=1}^{M} n_i$, such that the number of total customers in the network is a constant N. With this transformation, the original network is modeled as a closed Jackson network with N customers and $M + 1$ nodes, where the service rate, routing probability, and queue length of node i are μ_i, q_{ij}, and n_i, $i, j = 0, 1, \cdots, M$, respectively [9]. From the above description, we see that the Jackson network model captures the interconnection among multiple nodes in a multi-hop wireless network, especially the transfer and routing of customers (packets) among different nodes.

The fading of a wireless channel has a major effect on the wireless communication and link quality. We use a Markov process to model the transition of channel states [28], [31]. Let us define $\mathcal{H}$ as a finite set of all the possible channel states and $\hbar_i \in \mathcal{H}$ as the channel state of node i, $i = 1, 2, \cdots, M$. Please note, for simplicity we discuss the channel state of a node instead of the channel state of a link between two adjacent nodes. This is partly reasonable as we assume the channel state is i.i.d. At each time epoch, there is only one particular link of node i can communicate. This is equivalent to say that node i is completely occupying the radio channel from a view of statistics. Therefore, we focus on the channel state of node i, instead of that of links, to simplify the model of channel transition.

The transition rate of channel states at node i is denoted as $R_1(\hbar_i, \hbar_i')$, $\hbar_i, \hbar_i' \in \mathcal{H}$, and the associated matrix R_1 is called the *state transition rate matrix* of Markov process. We assume that the utilization of channels in the network is highly time-synchronized or frequency orthogonally divided, such as TDMA or OFDMA. The adjacent links do not contend for shared media and we do not consider the contention and interference among links. The assumption of interference-free link is reasonable for certain media access control schemes, such as TDMA discussed in [13]. The channel states of all nodes are assumed i.i.d. The transition rate matrices of channel states of all nodes are identical as R_1. The channel state transition of this Markov model is illustrated in Fig. 2 as an example, where 0 represents bad channel state and 1 represents good channel state.

It is well known that the transmission rate of a radio module varies with the channel state and transmission power. In this paper, we assume that adaptive modulation and coding (AMC) is employed such that the transmission rate and power can be automatically adapted per the channel fluctuation. Similar to most studies in the literature, the reconfiguration delay discussed in [8] is omitted for simplicity. In this paper, we consider the M-ary orthogonal modulation scheme as an example. With the additive white
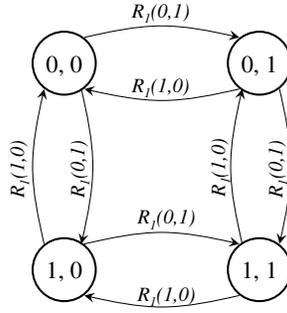
Fig. 2. An example of state transition diagram of channel states in a network with two nodes.

Gaussian noise (AWGN), it is well known that the bit error probability P_b of the M-ary modulation is

$$P_b \leq \frac{M}{2} Q\left[\sqrt{\frac{E}{N_0}}\right], \tag{1}$$

where M is the M-ary signal, with a little abuse of notation (in other places of this paper, M indicates the number of nodes), E is the symbol energy, N_0 is the spectrum density of noise under a channel gain $|\hbar|^2$, $Q[.]$ is called the Gaussian error function defined as

$$Q[x] = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt, \quad x \geq 0. \tag{2}$$

It is easy to verify that

$$Q[x] \leq \frac{1}{2} e^{-\frac{x^2}{2}}, \quad \text{if } x \geq 3. \tag{3}$$

Substituting (3) into (1), we have

$$P_b \leq \frac{M}{4} e^{-\frac{E}{2N_0}}. \tag{4}$$

For the worst case of (4), we have

$$P_b = \frac{M}{4} e^{-\frac{E}{2N_0}}. \tag{5}$$

Since

$$E = \frac{P_w \log_2 M}{\mu}, \tag{6}$$

where P_w is the transmission power, $\log_2 M$ is the number of bits for each symbol, and μ is the bit transmission rate. Substituting (6) into (5), we obtain the following formula,

$$P_w = \frac{2N_0}{\log_2 M} \ln \frac{M}{4P_b} \cdot \mu = k_1 \mu. \tag{7}$$

It is clear that the parameters k_1 is related to N_0, i.e., k_1 depends on the channel states $\hbar$, $\hbar \in \mathcal{H}$. From (7), we observe that in such situation the transmission power P_w will linearly increase w.r.t. the transmission rate μ. In this paper, we use this relation to construct the system performance measure. It is worth noting that this linearity is not universally valid and we only assume it in this specific example.

By combining the state of queue length n_i and the state of channel $\hbar_i$, we define the total state of node i as $s_i \triangleq (\hbar_i, n_i)$ to describe the whole state information of this node, $i = 1, 2, \cdots, M$. Since node 0 is a virtual node and it does not have a channel state, we define $s_0 \triangleq (n_0)$. The total system state is $\boldsymbol{s} \triangleq (s_0, s_1, \cdots, s_M)$. We also define $\boldsymbol{\hbar} \triangleq (\hbar_1, \hbar_2, \cdots, \hbar_M)$, $\boldsymbol{n} \triangleq (n_0, n_1, \cdots, n_M)$. Therefore, $\boldsymbol{s} = (s_0, s_1, \cdots, s_M)$ is equivalent to $\boldsymbol{s} = (\boldsymbol{\hbar}, \boldsymbol{n})$. The space of $\boldsymbol{\hbar}$ is $\mathcal{H}^{(M)} \triangleq \underbrace{\mathcal{H} \times \mathcal{H} \times \cdots \times \mathcal{H}}_{M}$, where $\times$ is the Cartesian product. The whole state space is $\mathcal{S} \triangleq \mathcal{H}^{(M)} \times \mathcal{N}$, where $\mathcal{N} \triangleq \left\{ \text{all } \boldsymbol{n} : \sum_{i=0}^{M} n_i = N \right\}$.

Since the virtual node 0 represents the external data arrival process and its rate μ_0 is uncontrollable, we focus on the optimization of μ_i, $i = 1, 2, \cdots, M$. We assume that the transmission rate of node i, μ_i, is dependent on the system state $\boldsymbol{s}$. That is, when the system state $\boldsymbol{s}$ (channel state $\boldsymbol{\hbar}$ or buffer state $\boldsymbol{n}$) changes, the transmission rate μ_i can also change accordingly. We denote this type of state-dependent transmission rates as $\mu_{i(\boldsymbol{s})}$, $i = 1, 2, \cdots, M$, and $\mu_{0(\boldsymbol{s})} \equiv \mu_0$ for all $\boldsymbol{s} \in \mathcal{S}$.

By considering the physical meanings of closed Jackson networks, we can obtain its state transition rate matrix R_2. For instance, for a simple closed Jackson network with 2 servers and 2 customers, it has 3 states $\boldsymbol{s}_1, \boldsymbol{s}_2, \boldsymbol{s}_3$, which are $(2, 0), (1, 1), (0, 2)$, respectively. Here, the system state $\boldsymbol{s}$ only includes the buffer state $\boldsymbol{n}$, does not include the channel state $\boldsymbol{\hbar}$. According to the queueing theory, it is well known that the transition rate matrix R_2 is written as follows [9].

$$
R_2 = \begin{bmatrix}
-\mu_{1(\boldsymbol{s}_1)} & \mu_{1(\boldsymbol{s}_1)} q_{12} & 0 \\
\mu_{2(\boldsymbol{s}_2)} q_{21} & -\mu_{1(\boldsymbol{s}_2)} - \mu_{2(\boldsymbol{s}_2)} & \mu_{1(\boldsymbol{s}_2)} q_{12} \\
0 & \mu_{2(\boldsymbol{s}_3)} q_{21} & -\mu_{2(\boldsymbol{s}_3)}
\end{bmatrix},
$$

where actually $q_{12} = q_{21} = 1$ since there are only two servers. On the other hand, since the channel state transition at each node is independent and its transition rate matrix is R_1, we obtain the transition rate matrix of the total channel state $\boldsymbol{\hbar}$ and denote it as $R_1^{(M)}$. We have $R_1^{(M)}(\boldsymbol{\hbar}, \boldsymbol{\hbar}) = -\sum_{i=1}^{M} R_1(\hbar_i, \hbar_i)$; $R_1^{(M)}(\boldsymbol{\hbar}, \boldsymbol{\hbar}'_i) = R_1(\hbar_i, \hbar'_i)$, $i = 1, 2, \cdots, M$; $R_1^{(M)}(\boldsymbol{\hbar}, \boldsymbol{\hbar}') = 0$, for other $\boldsymbol{\hbar}'$; where $\boldsymbol{\hbar}'_i \triangleq (\hbar_1, \cdots, \hbar'_i, \cdots, \hbar_M)$ is called the neighboring channel state of $\boldsymbol{\hbar}$ which means only the ith element of $\boldsymbol{\hbar}$ is changed. Please note, since the channel states obey i.i.d. Markov process, the probability that the channel states of more than one node have change simultaneously in a period Δ is in the order $o(\Delta)$, $\Delta \to 0$. This accounts for the case $R_1^{(M)}(\boldsymbol{\hbar}, \boldsymbol{\hbar}') = 0$ for other $\boldsymbol{\hbar}'$. Therefore, we further obtain

the transition rate matrix R for the whole state $\boldsymbol{s} = (\hbar, \boldsymbol{n})$. The element of R is as follows, $R(\boldsymbol{s}, \boldsymbol{s}) = -\sum_{i=0}^{M} \mathbf{1}_{n_i > 0} \mu_{i(\boldsymbol{s})} - R_1^{(M)}(\hbar, \hbar)$; $R(\boldsymbol{s}, \boldsymbol{s}') = \mathbf{1}_{n_i > 0} \mu_{i(\boldsymbol{s})} q_{ij}$, if $\boldsymbol{s}' = (\hbar, \boldsymbol{n}_{-i+j})$, $i, j = 0, 1, \cdots, M$; $R(\boldsymbol{s}, \boldsymbol{s}') = R_1^{(M)}(\hbar, \hbar_i')$, if $\boldsymbol{s}' = (\hbar_i', \boldsymbol{n})$, $i = 1, 2, \cdots, M$; $R(\boldsymbol{s}, \boldsymbol{s}') = 0$, for other $\boldsymbol{s}' \in \mathcal{S}$; where $\boldsymbol{n}_{-i+j} \triangleq (n_0, \cdots, n_i - 1, \cdots, n_j + 1, \cdots, n_M)$ is called the neighboring state of $\boldsymbol{n}$ with $n_i > 0$, and $\mathbf{1}_{n_i > 0}$ is an indicator function defined as $\mathbf{1}_{n_i > 0} = 1$ (or 0) if $n_i > 0$ (or $n_i \leq 0$). The above values of elements in R also follow the constraint that the probability of more than one buffer state $\boldsymbol{n}$ or channel state $\hbar$ having changes simultaneously is infinitesimal. This is similar to a Poisson process where the probability of more than one arrival happening in a period Δ is in the order $o(\Delta)$, $\Delta \to 0$.

From the description above, we have obtained the transition rate matrix R for our formulated Markov process. For convenience, a continuous-time Markov process is discretized as a discrete-time Markov chain using the *uniformization* technique [24]. Define a constant b satisfying $b \geq \max |R(\boldsymbol{s}, \boldsymbol{s})|$, for all $\boldsymbol{s}$. The original Markov process with R is equivalent to a Markov chain with *transition probability matrix* $P = I + R/b$, where I is an identity matrix with the same dimension as R. The transition probability matrix of our problem is represented as (8), for all $\boldsymbol{s} = (\hbar, \boldsymbol{n}) \in \mathcal{S}$. Therefore, our original model has been transformed to a discrete-time Markov chain whose system state $\boldsymbol{s}$ will evolve step by step according to the transition probability matrix P. We further denote $\boldsymbol{s}_t$ as the system state at time epoch t, where $t = 1, 2, \cdots$, and $\boldsymbol{s}_t \in \mathcal{S}$.

$$P(\boldsymbol{s}, \boldsymbol{s}') = \begin{cases} 1 - \sum_{i=0}^{M} \mathbf{1}_{n_i > 0} \mu_{i(\boldsymbol{s})}/b - \sum_{i=1}^{M} R_1(\hbar_i, \hbar_i)/b, & \text{if } \boldsymbol{s}' = \boldsymbol{s} = (\hbar, \boldsymbol{n}); \\ \mathbf{1}_{n_i > 0} \mu_{i(\boldsymbol{s})} q_{ij}/b, & \text{if } \boldsymbol{s}' = (\hbar, \boldsymbol{n}_{-i+j}), \ i, j = 0, 1, \cdots, M; \\ R_1(\hbar_i, \hbar_i')/b, & \text{if } \boldsymbol{s}' = (\hbar_i', \boldsymbol{n}), \ i = 1, 2, \cdots, M; \\ 0, & \text{for other } \boldsymbol{s}' \in \mathcal{S}. \end{cases} \tag{8}$$

The value domain of all service rates $\mu_{i(\boldsymbol{s})}$, $i = 1, 2, \cdots, M$, $\boldsymbol{s} \in \mathcal{S}$, is denoted as $\mathcal{D} \triangleq \{\text{all } \mathcal{D}_{i(\boldsymbol{s})}, \ i = 1, 2, \cdots, M, \boldsymbol{s} \in \mathcal{S}\}$, where $\mathcal{D}_{i(\boldsymbol{s})}$ is the value domain of $\mu_{i(\boldsymbol{s})}$. For simplicity, we assume that the value domain of every node at every state is identical, i.e., $\mathcal{D}_{i(\boldsymbol{s})} = \mathcal{D}_{j(\boldsymbol{s}')}$, $i, j = 1, 2, \cdots, M$, $\boldsymbol{s}, \boldsymbol{s}' \in \mathcal{S}$. We use the vector $\boldsymbol{\mu_s} \triangleq (\mu_{1(\boldsymbol{s})}, \mu_{2(\boldsymbol{s})}, \cdots, \mu_{M(\boldsymbol{s})})$ to represent the transmission rates of all nodes when the state is $\boldsymbol{s}$. The optimization variables are transmission rates $\boldsymbol{\mu_s}$ at each state $\boldsymbol{s} \in \mathcal{S}$. Note that with the terminology of MDP, $\boldsymbol{\mu_s}$ is the "action" at state $\boldsymbol{s}$ and the "policy" $\mathcal{L} : \mathcal{S} \to \mathcal{D}$ is a rule of choosing $\boldsymbol{\mu_s}$ at each state $\boldsymbol{s}$. In other words, when the state is $\boldsymbol{s}$, the transmission rates will be $\boldsymbol{\mu_s} = \mathcal{L}(s)$ according to the policy $\mathcal{L}$. We denote the set of all the possible policies as the policy space Ψ, i.e., $\Psi \triangleq \{\text{all } \mathcal{L}\}$.

The cost function f (column vector) is related to the policy $\mathcal{L}$ and we define it as

$$f^{\mathcal{L}}(\boldsymbol{s}) = \sum_{i=1}^{M} n_i + \beta \sum_{i=1}^{M} \mathbf{1}_{n_i > 0} c_{\hbar_i} \mu_{i(\boldsymbol{s})}, \tag{9}$$

where β is a trade-off coefficient between the queueing delay and the power consumption, $c_{\hbar_i}$ is the power consumption of unit transmission rate of a node at the channel state $\hbar_i$. With (7), $c_{\hbar_i}$ can be set as k_1. The first term of right-hand-side in (9) represents the queue length of all nodes at state $\boldsymbol{s}$, which is generally proportional to the data queueing delay. Actually, the real queueing delay can be formulated as $n_i/\mu_{i(\boldsymbol{s})}$, instead of n_i in (9). But such definition makes the objective more complicated and some desirable optimality properties derived later will be lost, such as Theorem 3 and Theorem 4. Thus, for simplicity, we use (9) as the objective function in this paper. The second term of right-hand-side in (9) represents the transmission power consumption of all nodes at state $\boldsymbol{s}$. Please note, the objective function (9) focuses on the controllable delay caused by different scheduling policies. The inherent delay which cannot be reduced is not our optimization target. For example, the delay caused by the size and number of time slots in TDMA is inherent and we do not include such factors in the objective function. The coefficient β indicates the relative importance between the energy efficiency and the transmission delay. A larger value of β places more importance on the weight of energy efficiency. This problem can be viewed as a multi-objective optimization problem including the objectives of queueing delay and energy consumption. The optimization results under different values of β are *Pareto* optimum of the multi-objective optimization problem. Different β's balance the weights of the multiple objectives, as illustrated later by the delay/energy curve in Fig.5.

Under a policy $\mathcal{L}$, the steady-state distribution of the system is denoted as a row vector $\pi^{\mathcal{L}}$, and $\pi^{\mathcal{L}}(\boldsymbol{s})$ means the probability of state $\boldsymbol{s}$ when the system is steady under policy $\mathcal{L}$. The objective is to select a policy $\mathcal{L}^*$ which minimizes the system average cost $\eta^{\mathcal{L}} \triangleq \pi^{\mathcal{L}} f^{\mathcal{L}}$. That is

$$\mathcal{L}^* = \arg\min_{\mathcal{L} \in \Psi} \pi^{\mathcal{L}} f^{\mathcal{L}} = \arg\min_{\mathcal{L} \in \Psi} E_t\{f^{\mathcal{L}}(\boldsymbol{s}_t)\} = \arg\min_{\mathcal{L} \in \Psi} \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} f^{\mathcal{L}}(\boldsymbol{s}_t), \tag{10}$$

where the last equality holds when the system is ergodic, which is true for most cases, such as the system is stable under the stability condition proposed in [29].

Until now, we have formulated this problem as an MDP with adjustable transmission rates of each node. In the next section, an efficient online optimization method based on the performance potential theory [6], [7] is developed to solve this optimization problem.

III. METHOD AND ALGORITHMS

In this section, we introduce the performance potential theory to solve the formulated power and delay optimization problem. Performance potential theory is an efficient method to optimize the performance of Markov systems. It uses a difference equation to describe the difference of system performance under any two policies. Based on the difference equation, an iterative algorithm is developed to find the optimal policy. We also derive some useful optimality properties of this problem, which can help us schedule the transmission power in an efficient way.

A. Performance Potential Theory

Performance potential theory is an extension of perturbation analysis (PA) methodology [6], [7], and it provides a new sensitivity-based perspective to solve the MDP problem. The key idea is to extract the performance sensitivity information of Markov systems by deriving the performance difference equation under two arbitrary policies. With the difference equation, policy iteration can be developed directly and this algorithm can be implemented with an online manner. We give a brief introduction as follows.

Consider a discrete-time Markov chain $\Im = \{s_t, t = 0, 1, \cdots\}$, where $s_t \in \mathcal{S}$ is the system state at time t and the state space $\mathcal{S}$ is assumed finite. The state transition probability matrix is denoted as P and the steady state distribution is denoted as a row vector π. We have $Pe = e$ and $\pi P = \pi$, where e is a column vector whose elements are all one. By denoting the system cost function as a column vector f, the long-run average performance of the system is $\eta = \pi f$.

We define the *performance potential g* to evaluate the "potential" contribution of an initial state to the overall system performance η [7]. g is a column vector whose element $g(s), s \in \mathcal{S}$, is defined as

$$g(s) = \lim_{T \to \infty} E \left\{ \sum_{t=0}^{T} [f(s_t) - \eta] \Big| s_0 = s \right\}. \tag{11}$$

Thus, the above equation can be rewritten as follows.

$$g(s) = f(s) - \eta + \sum_{s' \in \mathcal{S}} P(s, s') \lim_{T \to \infty} E \left\{ \sum_{t=1}^{T} [f(s_t) - \eta] \Big| s_1 = s' \right\}. \tag{12}$$

With a matrix form, we obtain the following Poisson equation [6].

$$(I - P)g + \eta e = f. \tag{13}$$

We consider the performance difference of a Markov system when the transition probability matrix changes from P to P' with the same state space. We use prime "'" to denote the parameter values of

the system with P'. By multiplying both sides of (13) with π' and with some transformations, we can obtain the following performance difference equation [6], [7]

$$\eta' - \eta = \pi'[(P' - P)g + (f' - f)].\tag{14}$$

Equation (14) provides important sensitivity information of the system, which can be used in the subsequent optimization process. Since each element of π' is positive (for ergodic Markov chain) and the object in the square bracket of (14) is a column vector, we only need to estimate the value of g of the current system and choose a proper P' to make the value of each element of the column vector represented by the square bracket non-positive. Thus, we can make sure that $\eta' \leq \eta$, and the system performance is improved (for the minimization problem), where the estimation for the exact value of π' can be completely avoided in the comparison between η and $\eta' = \pi'f'$. Note that estimating π' is not realistic since it needs to be enumerated under every possible P', which is equivalent to the brute-force approach.

B. Optimality Properties and Algorithms

Since the network-wide optimization for power and delay has been modeled as an MDP, we use the performance potential theory to solve the formulated problem. We derive the necessary and sufficient conditions for the optimal policy. Some structural properties of optimal policy are also derived. We prove that the system performance is monotonic w.r.t. the transmission rate and the optimal transmission rate is either maximal or minimal. That is, the "bang-bang" control is an optimal control for this problem. Based on the optimization theory, we further develop an iterative algorithm which can be implemented online.

As described in Section II, the transition probability matrix P is determined by (8). When we change the transmission rates $\mu_{i(s)}$, P will be changed and the cost function f will also be changed. Therefore, we apply the difference equation (14) to analyze the change of system performance when a different set of transmission rates is adopted.

First, we consider a situation where the transmission rate of a particular node i at a particular state s changes from $\mu_{i(s)}$ to $\mu'_{i(s)}$. Since the transmission rate $\mu_{i(s)}$ changes to $\mu'_{i(s)}$, the associated P and f will also change. Substituting (8) and (9) into the difference equation (14), we obtain

$$\begin{aligned}
\eta' - \eta &= \pi'\left[(P' - P)g + (f' - f)\right] \\
&= \pi'(\boldsymbol{s})\frac{\mu'_{i(\boldsymbol{s})} - \mu_{i(\boldsymbol{s})}}{b}\left\{\sum_{j=0}^{M}\mathbf{1}_{n_i>0}q_{ij}[g(\boldsymbol{\hbar}, \boldsymbol{n}_{-i+j}) - g(\boldsymbol{\hbar}, \boldsymbol{n})] + \beta\mathbf{1}_{n_i>0}c_{\hbar_i}b\right\}.
\end{aligned}\tag{15}$$

Define the aggregated performance potential as

$$\tilde{g}(i, \boldsymbol{s}) \triangleq \sum_{j=0}^{M} \mathbf{1}_{n_i>0} q_{ij}[g(\boldsymbol{\hbar}, \boldsymbol{n}_{-i+j}) - g(\boldsymbol{\hbar}, \boldsymbol{n})]. \tag{16}$$

We obtain

$$\eta' - \eta = \frac{\pi'(\boldsymbol{s})}{b} \left[\mu'_{i(\boldsymbol{s})} - \mu_{i(\boldsymbol{s})} \right] [\tilde{g}(i, \boldsymbol{s}) + \beta \mathbf{1}_{n_i>0} c_{\hbar_i} b]. \tag{17}$$

Thus, we obtain the performance difference equation when only one transmission rate $\mu_{i(\boldsymbol{s})}$ is changed. If multiple nodes change their transmission rates at multiple states, i.e., $\mu_{i(\boldsymbol{s})}$'s change to $\mu'_{i(\boldsymbol{s})}$'s, $i = 1, 2, \cdots, M$, $\boldsymbol{s} \in \mathcal{S}$, we can obtain the following difference equation with a similar analysis.

$$\begin{aligned}
\eta' - \eta &= \sum_{\boldsymbol{s} \in \mathcal{S}} \pi'(\boldsymbol{s}) \sum_{i=1}^{M} \frac{\mu'_{i(\boldsymbol{s})} - \mu_{i(\boldsymbol{s})}}{b} \left\{ \sum_{j=0}^{M} \mathbf{1}_{n_i>0} q_{ij}[g(\boldsymbol{\hbar}, \boldsymbol{n}_{-i+j}) - g(\boldsymbol{\hbar}, \boldsymbol{n})] + \beta \mathbf{1}_{n_i>0} c_{\hbar_i} b \right\} \\
&= \frac{1}{b} \sum_{\boldsymbol{s} \in \mathcal{S}} \sum_{i=1}^{M} \pi'(\boldsymbol{s}) \left[\mu'_{i(\boldsymbol{s})} - \mu_{i(\boldsymbol{s})} \right] [\tilde{g}(i, \boldsymbol{s}) + \beta \mathbf{1}_{n_i>0} c_{\hbar_i} b].
\end{aligned} \tag{18}$$

Comparing (18) with (17), we observe that the difference equation holds a linear superposition property when multiple transmission rates change.

For (18), since $\pi'(\boldsymbol{s})$ is always positive, $\mu_{i(\boldsymbol{s})}$, β, $\mathbf{1}_{n_i>0}$, $c_{\hbar_i}$, and b are known parameters, we only need to choose a proper $\mu'_{i(\boldsymbol{s})}$ to make the product of these two square brackets non-positive, then the new performance η' is not larger than the current performance η. Thus, we obtain the following theorem.

Theorem 1: If $\left[\mu'_{i(\boldsymbol{s})} - \mu_{i(\boldsymbol{s})} \right] [\tilde{g}(i, \boldsymbol{s}) + \beta \mathbf{1}_{n_i>0} c_{\hbar_i} b] \leq 0$ for all $i = 1, 2, \cdots, M$ and $\boldsymbol{s} \in \mathcal{S}$, then $\eta' \leq \eta$. Additionally, if the inequality strictly holds for at least one $\boldsymbol{s}$ and i, then $\eta' < \eta$.

Since $\pi'(\boldsymbol{s})$ is always positive for all $\boldsymbol{s} \in \mathcal{S}$ and b is a known positive number, this theorem is straightforward based on the difference equation (18). We omit the detailed proof for simplicity.

Furthermore, based on the difference equation (18) and Theorem 1, we obtain the following necessary and sufficient condition of the optimal policy.

Theorem 2: A policy $\mathcal{L}^* = \{\boldsymbol{\mu}^*_{\boldsymbol{s}}, \boldsymbol{s} \in \mathcal{S}\}$ is optimal if and only if

$$\left[\mu'_{i(\boldsymbol{s})} - \mu^*_{i(\boldsymbol{s})} \right] [\tilde{g}^*(i, \boldsymbol{s}) + \beta \mathbf{1}_{n_i>0} c_{\hbar_i} b] \geq 0 \tag{19}$$

for all $\mathcal{L}' = \{\boldsymbol{\mu}'_{\boldsymbol{s}}, \boldsymbol{s} \in \mathcal{S}\} \in \Psi$, $i = 1, 2, \cdots, M$, and $\boldsymbol{s} \in \mathcal{S}$, where $\tilde{g}^*(i, \boldsymbol{s})$ is the aggregated performance potential under policy $\mathcal{L}^*$.

Proof: The sufficiency of (19) is straightforward based on the difference equation (18). The necessity of (19) can be proved by contradiction. Assume that the current policy $\mathcal{L}^*$ is optimal. For a particular i and $\boldsymbol{s}$, if (19) is not satisfied for a service rate $\mu'_{i(\boldsymbol{s})}$, then we can define a new policy $\mathcal{L}'$ whose $\mu_{i(\boldsymbol{s})}$ is

set as $\mu'_{i(s)}$ and all the other transmission rates of $\mathcal{L}'$ are the same as those of $\mathcal{L}^*$. With (18), we obtain $\eta' < \eta^*$ and the new policy $\mathcal{L}'$ is better than $\mathcal{L}^*$. This is conflicted with our previous assumption. Thus, the necessity of (19) is proved. ∎

Theorem 1 and Theorem 2 are helpful to develop the optimization algorithm, which we will discuss later. Theorem 2 defines the optimality criterion which identifies the stopping condition when the algorithm finds the optimal policy. Moreover, with the difference equation (17), we prove that the system performance is monotonic w.r.t. the transmission rate, which is described by the following theorem.

Theorem 3: For any particular transmission rate $\mu_{i(s)}$, $i = 1, 2, \cdots, M$ and $s \in \mathcal{S}$, the system performance η is monotonic w.r.t. $\mu_{i(s)}$.

Proof: For a particular transmission rate, when the transmission rate is changed from $\mu_{i(s)}$ to $\mu'_{i(s)}$, the performance difference is quantified by (17). On the other hand, when the transmission rate is changed from $\mu'_{i(s)}$ to $\mu_{i(s)}$ and all the other transmission rates remain the same, the performance difference is as follows.

$$\eta - \eta' = \frac{\pi(s)}{b} \left[\mu_{i(s)} - \mu'_{i(s)} \right] \left[\tilde{g}'(i, s) + \beta \mathbf{1}_{n_i > 0} c_{\hbar_i} b \right]. \tag{20}$$

Comparing (17) and (20), we have

$$\frac{\tilde{g}(i, s) + \beta \mathbf{1}_{n_i > 0} c_{\hbar_i} b}{\tilde{g}'(i, s) + \beta \mathbf{1}_{n_i > 0} c_{\hbar_i} b} = \frac{\pi(s)}{\pi'(s)} > 0, \tag{21}$$

where the steady state probability $\pi(s)$ and $\pi'(s)$ are always positive. Equation (21) indicates that the sign of $\tilde{g}(i, s) + \beta \mathbf{1}_{n_i > 0} c_{\hbar_i} b$ is unchanged when we only change $\mu_{i(s)}$. Furthermore, with (17), we can obtain the following derivative equation

$$\frac{d\eta}{d\mu_{i(s)}} = \frac{\pi(s)}{b} \left[\tilde{g}(i, s) + \beta \mathbf{1}_{n_i > 0} c_{\hbar_i} b \right]. \tag{22}$$

Since the sign of $\tilde{g}(i, s) + \beta \mathbf{1}_{n_i > 0} c_{\hbar_i} b$ is unchanged when $\mu_{i(s)}$ is changed, and $\pi(s)$ is always positive, it is obvious that the sign of derivative $\frac{d\eta}{d\mu_{i(s)}}$ is unchanged. Therefore, the average performance η is monotonic w.r.t. the transmission rate $\mu_{i(s)}$. This theorem is proved. ∎

Theorem 3 describes an important property of this optimization problem. With this monotone property, the optimization problem becomes easier to handle. One of the direct conclusions of this monotone property is the following theorem.

Theorem 4: The optimal transmission rates $\mu^*_{i,s}$ can be either maximum or minimum in their value domain $\mathcal{D}_{i(s)}$, for all $i = 1, 2, \cdots, M$ and $s \in \mathcal{S}$.

The conclusion in this theorem is fairly straightforward with Theorem 3. We omit the proof for simplicity.

Theorem 4 describes another important structural property of the optimal policy. This theorem is powerful for its concise conclusion. It means that the optimal transmission rate (or transmission power) can be found by considering only the maximal or minimal values. This theorem can help us reduce the searching space from the original $D_{i(\boldsymbol{s})}$, which is a continuous and infinite space, to only the maximal and minimal values of $D_{i(\boldsymbol{s})}$. That is, the action search space is reduced from $|\mathcal{D}_{i(\boldsymbol{s})}|$ to 2, for all $i = 1, 2, \cdots, M$ and $\boldsymbol{s} \in \mathcal{S}$. All of the intermediate values of transmission rates can be ignored. Therefore, this structural property greatly simplifies the optimization complexity. Actually, the scenario that the optimal values of the controllable parameters are either maximal or minimal is common in practice. In the theory of optimal control, such control is called the "bang-bang" control, which always adopts the controllable parameters at their boundary values [1], [2], [35]. The "bang-bang" control is easy to implement in the real world for its simple form, including some controls adopted in wireless communication [18], [25]. Theorem 4 shows that the "bang-bang" control is an optimal control for our optimization problem.

In summary, Theorems 2, 3, and 4 describe some optimality structures of the optimal control policy. The difference equations (17) and (18) perform a very critical role during the analysis and derivation of these theorems. With difference equation (18) as the basis, we further develop the following iterative algorithm to find the optimal policy for this optimization problem.

Algorithm 1 Iterative algorithm for optimizing the transmission power and delay of wireless networks.

Initialization

- Choose arbitrary values of transmission rates $\mathcal{L}^0 = \{\boldsymbol{\mu_s}, \boldsymbol{s} \in \mathcal{S}\} \in \Psi$ as an initial policy.

Improvement

- At the jth iteration with the policy denoted as $\mathcal{L}^j = \{\boldsymbol{\mu_s}, \boldsymbol{s} \in \mathcal{S}\}$, estimate the aggregated performance potentials $\tilde{g}(i, \boldsymbol{s})$, $i = 1, 2, \cdots, M$, $\boldsymbol{s} \in \mathcal{S}$, under the current policy $\mathcal{L}^j$.
- Choose the policy of the next (the $(j + 1)$th) iteration as $\mathcal{L}^{j+1} = \{\boldsymbol{\mu'_s}, \boldsymbol{s} \in \mathcal{S}\}$ with

$$
\begin{aligned}
&\text{if } \tilde{g}(i, \boldsymbol{s}) + \beta \mathbf{1}_{n_i > 0} c_{\hbar_i} b > 0, \ \mu'_{i,\boldsymbol{s}} = \min\{\mathcal{D}_{i(\boldsymbol{s})}\}; \\
&\text{else,} \qquad\qquad\qquad\qquad\qquad \mu'_{i,\boldsymbol{s}} = \max\{\mathcal{D}_{i(\boldsymbol{s})}\};
\end{aligned}
\tag{23}
$$

for all $i = 1, 2, \cdots, M$ and $\boldsymbol{s} \in \mathcal{S}$.

Stopping Rule

- If $\mathcal{L}^{j+1} \neq \mathcal{L}^j$, set $j \triangleq j + 1$ and return to step 2; otherwise, stop.

From the procedure of Algorithm 1, we observe that the system performance is improved at each iteration. Since the search space is finite, it is obvious that the algorithm will converge to the optimal policy within a finite number of iterations.

In order to implement Algorithm 1, we need to estimate the aggregated performance potentials $\tilde{g}(i, \boldsymbol{s})$, $i = 1, 2, \cdots, M$, $\boldsymbol{s} \in \mathcal{S}$. With the definition (11), we first introduce how to estimate the difference between any two potentials, $g(\boldsymbol{s})$ and $g(\boldsymbol{s}')$, $\boldsymbol{s}, \boldsymbol{s}' \in \mathcal{S}$.

For a Markov chain $\Im = \{\boldsymbol{s}_t, t = 0, 1, \cdots\}$, suppose the initial state is $\boldsymbol{s}_0 = \boldsymbol{s}$. We define $T(\boldsymbol{s}', t_0) \triangleq \min\{t : \boldsymbol{s}_t = \boldsymbol{s}', t > t_0\}$. Therefore, $T(\boldsymbol{s}', 0)$ is the time when the Markov chain $\Im$ reaches the state $\boldsymbol{s}'$ at the first time. With (11), the difference between $g(\boldsymbol{s})$ and $g(\boldsymbol{s}')$ can be written as follows [6].

$$g(\boldsymbol{s}) - g(\boldsymbol{s}') = E\left\{ \sum_{t=0}^{T(\boldsymbol{s}',0)-1} [f(\boldsymbol{s}_t) - \eta] \mid \boldsymbol{s}_0 = \boldsymbol{s} \right\}. \tag{24}$$

In order to develop an online estimation algorithm based on (24), we define $u_0^{\boldsymbol{s}} \triangleq 0$ and $u_{k+1}^{\boldsymbol{s}} \triangleq \min\{t : \boldsymbol{s}_t = \boldsymbol{s}, t > u_k^{\boldsymbol{s}}\}$, $k = 0, 1, \cdots$. We know that $u_k^{\boldsymbol{s}}$ is the regenerative points of state $\boldsymbol{s}$ for Markov chain $\Im$. Under the ergodic Markov chain scenario, we rewrite (24) as follows.

$$g(\boldsymbol{s}) - g(\boldsymbol{s}') = \lim_{K \to \infty} \frac{1}{K} \left\{ \sum_{k=1}^{K} \sum_{t=u_k^{\boldsymbol{s}}}^{T(\boldsymbol{s}', u_k^{\boldsymbol{s}})-1} [f(\boldsymbol{s}_t) - \eta] \right\}, \tag{25}$$

where η can be estimated by

$$\eta = \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} f(\boldsymbol{s}_t). \tag{26}$$

With the above formulas and the definition (16), we further propose an online estimation approach for $\tilde{g}(i, \boldsymbol{s})$, $i = 1, 2, \cdots, M$, $\boldsymbol{s} \in \mathcal{S}$.

First, w.r.t. the state vector $\boldsymbol{s} = (\boldsymbol{h}, \boldsymbol{n})$, we denote the operator $\boldsymbol{s}(n_i)$ as the queue length of node i at state $\boldsymbol{s}$. For a particular $\boldsymbol{s}$ and i, we define $u_0^{\boldsymbol{s},i} \triangleq 0$ and $u_{k+1}^{\boldsymbol{s},i} \triangleq \min\{t : \boldsymbol{s}_t = \boldsymbol{s}, \boldsymbol{s}_{t+1}(n_i) = \boldsymbol{s}_t(n_i) - 1, t > u_k^{\boldsymbol{s},i}\}$, i.e., $u_k^{\boldsymbol{s},i}$ is the kth time when the Markov chain $\Im$ reaches $\boldsymbol{s}$ and the node i completes a data transmission at the same time. Similar to (25), the definition formula (16) can be rewritten as follows.

$$\tilde{g}(i, \boldsymbol{s}) = \lim_{K \to \infty} \frac{1}{K} \left\{ \sum_{k=1}^{K} \sum_{t=u_k^{\boldsymbol{s},i}+1}^{T(\boldsymbol{s}, u_k^{\boldsymbol{s},i})-1} [f(\boldsymbol{s}_t) - \eta] \right\}. \tag{27}$$

With (27) and (26), we can estimate $\tilde{g}(i, \boldsymbol{s})$ by observing the sample path $\boldsymbol{s}_t$ of the system. Comparing the estimation formula (27) with (16), we find that the routing probabilities q_{ij} disappear because this information is reflected on the sample path and (27) has already included it during the calculation. Therefore, it is easy to implement the estimation of $\tilde{g}(i, \boldsymbol{s})$ based on a sample path, without requiring

the explicit values of routing probabilities, statistics of channel states, or arrival rates of packets. This is because our algorithm can learn these information from the sample path. This feature makes our algorithm implementable with an online manner. Combining the estimation of $\tilde{g}(i, s)$ and Algorithm 1, we implement the optimization algorithm online. A typical scenario of the algorithm execution is described as follows.

Procedure 1 Online optimization procedure.

- For the current transmission rate setting, observe the sample path s_t and estimate $\tilde{g}(i, s)$ with (27) and (26).
- Use (23) of Algorithm 1 to generate a better transmission rate setting.
- Run the system under the newly generated transmission rate setting and repeat the step 1 and 2 above, until the transmission rate setting cannot be improved.

From the description above, our approach operates in a centralized control mode. It is feasible to implement a central controller on the gateway of the network [25]. Since the transmission rates are adjustable based on the states of buffer and channel, the central controller has to know the states of every node and also send the decision of next transmission rates to every node. These information exchange can be implemented by specific schemes, with very slight resource consumption. For instance, there exist such schemes in communication networks (CDMA cellular network) where nodes periodically report their local information (link quality and buffer status) to the base station [3]. Another method is that we may carry such information in the header fields of packets (such as TCP segment header or IP datagram header). We may use one or two fields to store the channel state and buffer state for the packets going from nodes to base station. We may also use the same fields to store the optimized value of transmission rates for the packets coming from base station to nodes. According to the TCP/IP architecture, such design is easy to implement by observing the fact of bi-directional packet communication (e.g., for TCP data stream, almost every outgoing data packet will accompany an incoming ACK packet). Therefore, such centralized control architecture is implementable with very slight extra overheads and energy consumptions.

In general, a centralized control architecture has the problem of scalability. When the size of the network increases, the burden of the central controller will become heavy rapidly. Moreover, the failure of the central controller will make the entire network collapse. In fact, the scalability problem is not severe

for multi-hop wireless network as we notice the fact that such network always has a limited number of hops to maintain the network throughput. However, decentralizing our approach is of interest and brings significant challenges. With decentralization, we can greatly improve the scalability and flexibility. A good design of decentralized algorithm can achieve an individual optimum, with the sacrifice of social optimum. Our preliminary work shows that our approach may be still applicable in a decentralized scenario after some simplifications (in [36], we assume that each node adjusts the transmission rate only according to its own buffer status, without considering the channel status). Studying this decentralized optimization problem considering both the buffer status and channel status deserves further researches.

In this paper, we study the scenario of multi-hop network and formulate it as a Jackson network model. From the aspect of network topology, multi-hop network (or Jackson network model) is general. Our approach does not have special requirements on the network topology. For other network topologies, such as circle, tandem, or star networks, the basic framework of our approach remains valid.

Although Algorithm 1 has the ability to find the optimal policy, our approach suffers the high complexity. First, the Markov model of this problem has "the curse of dimensionality" [24]. The size of state space is $|\mathcal{S}| = \begin{pmatrix} M + N \\ M \end{pmatrix} \times |\mathcal{H}|^M$, which grows exponentially when the system size M or N increases. On the other hand, the size of action space is $|A| = 2^M$, even after the significant reduction with Theorem 4. The total size of the policy space is $|\Psi| = |A|^{|S|}$, which is highly exponential w.r.t. M and N. The computation complexity of Algorithm 1 is the same as that of policy iteration, which is studied in the MDP theory [19], [20]. There is no tight bounds for the computation complexity of policy iteration in the existing literature. In the worst case, the complexity of policy iteration is approximately the same as the size of policy space, $O(|A|^{|S|})$. Therefore, the computation complexity of Algorithm 1 is highly exponential. This indicates that Algorithm 1 is not applicable to the large-scale problem. Therefore, in this paper we only focus on the small-scale network, as we claimed earlier in the Introduction. We hope our approach for the small-scale problem may give some insights and motivations to the complicated case of large-scale problem. The combination with the approximation techniques, such as the neuro-dynamic programming [5], [21] or the aggregation techniques [15], [30], to handle the large-scale problem is a future research topic.

IV. NUMERICAL SIMULATION

In this section, we conduct numerical experiments to demonstrate the effectiveness of our approach. Simulation results show that our algorithm has the ability to find the optimum with a fast convergence

speed. Moreover, we study the relation between the power consumption and the queue length under different trade-off coefficients β.

As we mentioned before, in this paper we study a small-scale network to clearly demonstrate the main idea of our approach. We study a network with 3 communication nodes and 2 channel states. Thus, we have $M = 3$ and $\mathcal{H} = \{0, 1\}$ where 0 represents the bad channel state and 1 represents the good channel state. We choose the network capacity $N = 4$ which is a small number preventing the computation complexity from explosion. Here, $N = 4$ may represent a large buffer size, such as 4 Kilo packets and each customer represents 1 Kilo packets. Although this is a coarse-grained simulation, the final result has similarity to the fine-grained simulation with large value N. The data generation rates of each node are $\lambda_{01} = 100$, $\lambda_{02} = 100$, and $\lambda_{03} = 50$, respectively. That is, $\mu_0 = \lambda_{01} + \lambda_{02} + \lambda_{03} = 250$, $q_{01} = 0.4$, $q_{02} = 0.4$, and $q_{03} = 0.2$. The routing probabilities of node 1, 2, and 3 are $q_{10} = 0.3, q_{12} = 0.3, q_{13} = 0.4; q_{20} = 0.1, q_{21} = 0.4, q_{23} = 0.5; q_{30} = 0.2, q_{31} = 0.3, q_{32} = 0.5$. The channel state transition rate matrices of all nodes are identical as $R_1 = [-80, 80; 40, -40]$. The value domains of transmission rates of all nodes are identical. Transmission rates can be chosen as any value between $\mu_{min} = 100$ and $\mu_{max} = 1000$. The power cost coefficient is $c_{\hbar=0} = 0.1$ and $c_{\hbar=1} = 0.005$. The power and delay trade-off coefficient is $\beta = 0.3$. Since the constant b for uniformization is defined as $b \geq |R(\boldsymbol{s}, \boldsymbol{s})|$ for all $\boldsymbol{s}$ under all possible policies, we choose $b = \max_{\boldsymbol{s} \in \mathcal{S}} \left\{ \sum_{i=0}^{M} \mu_{i(\boldsymbol{s})} \right\} + M \times \max\{|R_1|\} = 3490$. The cost function is the same as (9). The objective is to find the optimal transmission rates $\mu_{i(\boldsymbol{s})}$, $i = 1, 2, 3$, $\boldsymbol{s} \in \mathcal{S}$, to minimize the average system cost.

As introduced in Section III, Algorithm 1 takes the advantage of the policy iteration to optimize the transmission rates (or equivalently, to optimize the transmission power) of all the nodes. The initial policy (transmission rates) is chosen arbitrarily from the policy space, e.g., we set all the transmission rates of nodes as μ_{max} initially. The sample path for the estimation of (27) is long enough to get an adequately accurate estimate.

With the iterative procedure of Algorithm 1, the algorithm takes only 5 iterations to converge to the optimal solution. The performance improvement of the solution at each iteration is shown in Fig. 3. Besides the average performance derived from (9), the power consumption and the average queue length are also illustrated in Fig. 3, where we observe that these two metrics are conflicted and our optimal policy is to obtain a tradeoff between them. Moreover, Fig. 3 shows that the performance improvement at the first 2 iterations is significant. This can be explained that the difference equation provides great sensitivity information for optimization. Therefore, it is promising to use Algorithm 1 to approach to a near optimal solution even in a tough environment, such as the estimate of aggregated potentials is not
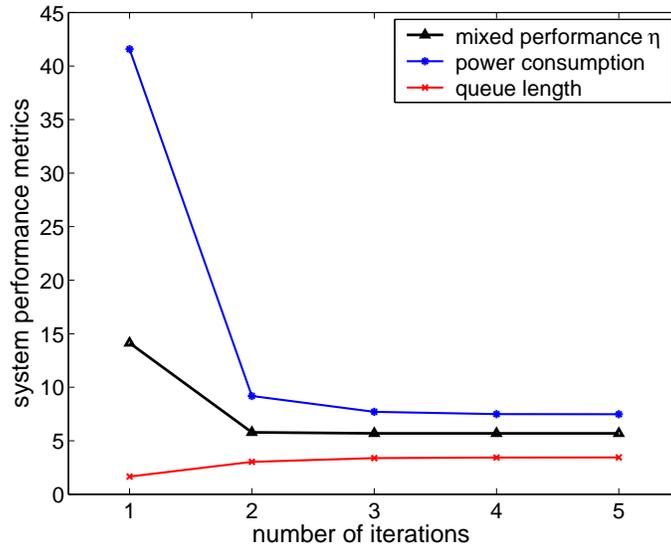
very accurate.

Fig. 3. The improvement of system performance w.r.t. the number of iterations.

The optimal transmission rates of node 1 are illustrated in Fig. 4. The results for the other 2 nodes are similar and omitted for the limit of space. For a clear representation, we use the index on axis Y and X to represent the two-dimensional state $(\boldsymbol{\hbar}, \boldsymbol{n})$. Specifically, the index on axis X, 1-35, represents the state of queue lengths $\boldsymbol{n} = (0,0,0,4), (0,0,1,3), \cdots, (4,0,0,0)$ with a binary ascending order; the index on axis Y, 1-8, represents the channel states $\boldsymbol{\hbar} = (0,0,0), (0,0,1), \cdots, (1,1,1)$ with a binary ascending order. The circles in Fig. 4 represent that the transmission rate of node 1 is 0. This is because that the queue length of node 1 at these states is 0 and node 1 does not need to transmit any packet. From the distribution of optimal transmission rates in Fig. 4, we can find that the optimal transmission rates do not have a simple structure, such as the threshold type. However, we observe that when the channel is bad, such as $\boldsymbol{\hbar} = (0,0,0)$, the optimal transmission rates should be minimum; when the channel is good, such as $\boldsymbol{\hbar} = (1,1,1)$, the optimal transmission rates are mostly maximum. It means that we should schedule more transmission tasks when the channel condition is better. This rule complies with our common sense.

Furthermore, we study the relation between the optimal power and queue length under different trade-off coefficients $\beta = 0.04, 0.05, 0.08, 0.1, 0.3, 2$, respectively. With Algorithm 1, we obtain the optimal solutions under each β. Based on the cost function (9), the average power consumption is studied by the measure $\sum_{i=1}^{M} \mathbf{1}_{n_i>0} c_{h_i} \mu_{i(\boldsymbol{s})}$, while the average queue length is studied by the measure $\sum_{i=1}^{M} n_i$. The relation is illustrated in Fig. 5, which is a convex curve that describes the conflictive nature between the
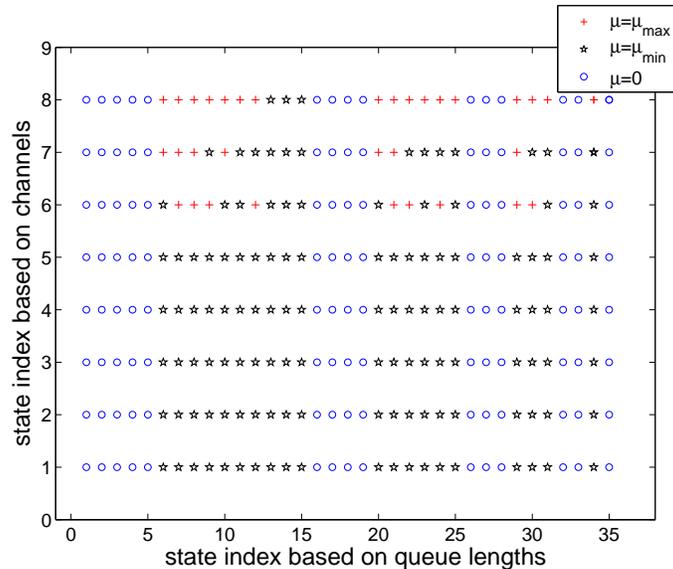
Fig. 4. The optimal transmission rates of node 1 under different states.

power consumption and the queue length (or data delay). Actually, the optimal solution under a certain value of β is a Pareto optimum of a multi-objective optimization problem, which includes the objectives of power consumption and queue length.

Although in this paper we focus on the static wireless network, mobility is truly an important feature of wireless network and it will affect the network topology greatly. We conduct a simple scenario to illustrate the effect of the change of network topology on our approach. For ease of demonstration, we roughly assume that the change of network topology can be reflected by the change of routing probabilities. For example, suppose that node 1 has mobility and it moves on a direction to the base station. This movement will induce that node 2 and 3 more probably forward their packets to node 1 and node 1 more probably forwards its packets to exterior (base station). This will affect the routing probabilities. Assume that we have $q_{10} = \delta, q_{12} = 0.45 - 0.5\delta, q_{13} = 0.55 - 0.5\delta; q_{20} = 0.4 - 0.5\delta, q_{21} = 1.5\delta, q_{23} = 0.6 - \delta; q_{30} = 0.4 - 0.5\delta, q_{31} = \delta, q_{32} = 0.6 - 0.5\delta$, where δ is a factor reflecting the movement of node 1. We execute our algorithm under different values of δ and obtain Fig. 6. It is illustrated that the optimization procedures are similar under different routing probabilities. When the network topology is not varying frequently, it is possible for the algorithm to converge before the topology is largely changed. Here we only make a rough exploration about the network topology and mobility, this topic needs more future investigations.
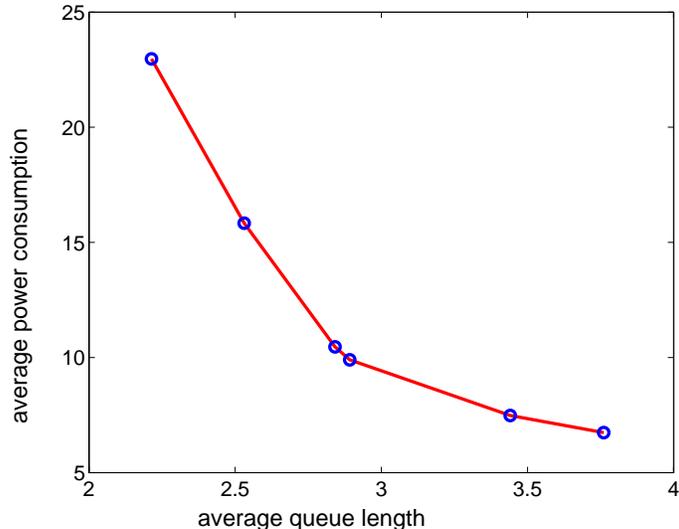
Fig. 5. The optimal power consumption versus queue length under different $\beta = 0.04, 0.05, 0.08, 0.1, 0.3, 2$.

V. CONCLUSION

In this paper, we formulate the power and delay optimization problem in a multi-hop wireless network as an integrated Markov model, where channel states and transition of data packets among interconnected nodes are jointly considered. By applying the performance potential theory, a performance difference equation is derived for this problem. Some important structural properties of optimal policy are also derived. We further develop an iterative optimization algorithm to solve this problem. This algorithm is online implementable based on the system sample path. The effectiveness of our approach has been demonstrated by simulation experiments. Although our approach is developed under some assumptions, our work gives a beneficial attempt to handle the power and delay optimization problem of multiple nodes, which is a very complicated problem in the literature. We hope our work can shed some lights on the optimization of this problem with other more general formulations, such as considering the interference link, the varying topology, and the large-scale scenario.

Some aspects of this approach need further investigations. First, in this paper the transmission rates of nodes are state-dependent. This requires knowing the state of all the nodes so that their transmission rates can be adjusted. Although our approach is feasible to be implemented at the gateway which can obtain the global information, adapting this to a distributed multi-hop wireless network deserves further investigations. Moreover, in this paper we assume the network is relatively static. However, the mobility is a common feature in wireless networks. How to handle the effect of mobility on the scheduling policy
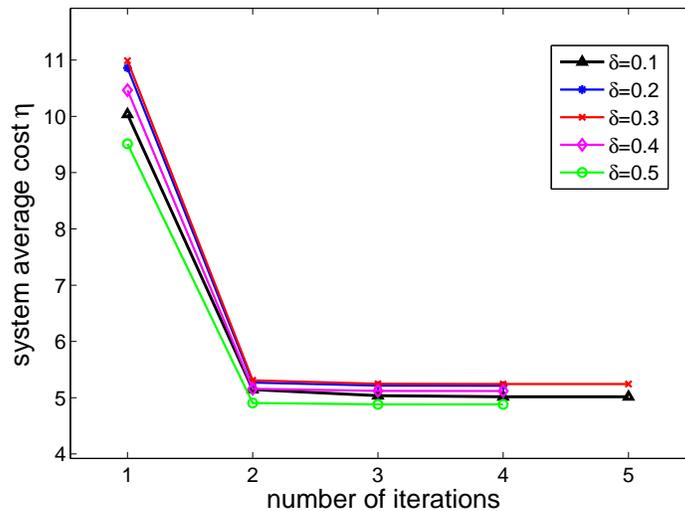
Fig. 6. The optimization procedures under different routing probabilities.

is a valuable research topic. Another issue needing further investigations is "the curse of dimensionality". This is a well-known problem of MDP theory: as the system size increases, the problem complexity will increase explosively. We will further investigate how to develop or utilize the approximation techniques to overcome this scalability difficulty.

REFERENCES

[1] Z. Artstein, "Discrete and continuous bang-bang and facial spaces, or: Look for the extreme points," *SIAM Review*, Vol. 22, pp. 172-185, 1980.

[2] R. Bellman, I. Glicksberg, and O. Gross, "On the bang-bang control problem," *Quarterly of Applied Mathematics*, Vol. 14, pp. 11-18, 1956.

[3] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushayana, and A. Viterbi, "CDMA/HDR: A bandwidth-efficient high-speed wireless data service for nomadic users," *IEEE Communications Magazine*, Vol. 38, pp. 70-77, 2000.

[4] R. A. Berry and R. G. Gallager, "Communication over Fading Channels with Delay Constraints," *IEEE Transactions on Information Theory*, Vol. 48, pp. 1135-1149, 2002.

[5] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, Belmont, MA: Athena Scientific, 1996.

[6] X. R. Cao, *Stochastic Learning and Optimization – A Sensitivity-Based Approach*, New York: Springer, 2007.

[7] X. R. Cao and H. F. Chen, "Potentials, perturbation realization, and sensitivity analysis of Markov processes," *IEEE Transactions on Automatic Control*, Vol. 42, pp. 1382-1393, 1997.

[8] G. D. Celik and E. Modiano, "Scheduling in networks with time-varying channels and reconfiguration delay," *Proceedings of IEEE INFOCOM 2012*, pp. 990-998.

[9] H. Chen and D. D. Yao, *Fundamentals of Queueing Networks: Performance, Asymptotics, and Optimization*, New York: Springer Verlag, 2001.

[10] W. Chen, M. J. Neely, and U. Mitra, "Delay-constrained energy-efficient scheduling over a multihop link," *Proceedings of IEEE International Symposium on Information Theory 2007*, pp. 1406-1410.

[11] B. E. Collins and R. O. Cruz, "Transmission policies for time varying channels with average delay constraints," *Proceedings of 37th Annual Allerton Conference on Communications, Control, and Computing*, Monticello, IL, pp. 709-717, 1999.

[12] R. Cruz and A. Santhanam, "Optimal routing, Link scheduling, and power control in multi-hop wireless networks," *Proceedings of IEEE INFOCOM 2003*, San Francisco, CA, pp. 702-711.

[13] S. Cui, R. Madan, A. J. Goldsmith, and S. Lall, "Cross-layer energy and delay optimization in small-scale sensor networks," *IEEE Transactions on Wireless Communications*, Vol. 6, No. 10, pp. 3688-3699, 2007.

[14] M. Goyal, A. Kumar, and V. Sharma, "Power Constrained and Delay Optimal Policies for Scheduling Transmission over a Fading Channel," *Proceedings of IEEE INFOCOM 2003*, pp. 311-320.

[15] Q.-S. Jia, "On state aggregation to approximate complex value functions in large-scale Markov decision processes," *IEEE Transactions on Automatic Control*, Vol. 56, pp. 333-344, 2011.

[16] A. K. Karmokar, D. V. Djonin, and V. K. Bhargava, "Delay constrained rate and power adaptation over correlated fading channels," *Proceedings of IEEE Globecom 2004*, pp. 3448-3453.

[17] H. Kwon, T. H. Kim, S. Choi, and B. G. Lee, "A cross-layer strategy for energy-efficient reliable delivery in wireless sensor networks," *IEEE Transactions on Wireless Communications*, Vol. 5, No. 12, pp. 3689-3699, 2006.

[18] J. Li, N. B. Shroff, and E. K. P. Chong, "A reduced-power channel reuse scheme for wireless packet cellular networks," *IEEE/ACM Transactions on Networking*, Vol. 7, pp. 818-832, 1999.

[19] M. L. Littman, T. L. Dean, and L. P. Kaelbling, "On the complexity of solving Markov decision problems," *Proceedings of the 11th International Conference on Uncertainty in Artificial Intelligence 1995*, pp. 394-402.

[20] Y. Mansour and S. Singh, "On the complexity of policy iteration," *Proceedings of the 15th International Conference on Uncertainty in Artificial Intelligence 1999*, pp. 401-408.

[21] P. Marbach, O. Mihatsch, and J. N. Tsitsiklis, "Call admission control and routing in integrated services networks using neuro-dynamic programming," *IEEE Journal On Selected Areas In Communications*, Vol. 18, pp. 197-208, 2000.

[22] F. Meshkati, H. V. Poor, S. C. Schwartz, and R. V. Balan, "Energy-efficient resource allocation in wireless networks with quality-of-service constraints," *IEEE Transactions on Communications*, Vol. 57, No. 11, pp. 3406-3414, 2009.

[23] L. Miao and C. G. Cassandras, "Optimal transmission scheduling for energy-efficient wireless networks," *Proceedings of IEEE INFOCOM 2006*, pp. 732-742.

[24] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, John Wiley & Sons, 1994.

[25] L. Qian, N. Song, D. R. Vaman, X. Li, and Z. Gajic, "Power control and proportional fair scheduling with minimum rate constraints in clustered multihop TD/CDMA wireless ad hoc networks," *Proceedings of the 2006 IEEE Wireless Communications and Networking Conference (WCNC 2006)*, Vol. 2, pp. 763-769.

[26] D. Rajan, "Power efficient delay allocation in multihop wireless networks," *IEEE Transactions on Vehicular Technology*, Vol. 56, pp. 1813-1825, 2007.

[27] S. Shakkottai and T. S. Rappaport, "Research challenges in wireless networks: A technical overview," *Proceedings of the Fifth International Symposium on Wireless Personal Multimedia Communications 2002*, Honolulu, HA, pp. 12-18.

[28] C. C. Tan and N. C. Beaulieu, "On first-order Markov modeling for the Rayleigh fading channel," *IEEE Transactions on Communications*, Vol. 48, pp. 2032-2040, 2000.

[29] L. Tassiulas and A. Ephremides, "Stability properties of constrained queuing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, Vol. 37, pp. 1936-1948, 1992.

[30] Y. W. Wan and X. R. Cao, "The control of a two-level Markov decision process by time aggregation," *Automatica*, Vol. 42, pp. 393-403, 2006.

[31] H. S. Wang and N. Moayeri, "Finite-state Markov channel - a useful model for radio communication channels," *IEEE Transactions on Vehicular Technology*, Vol. 44, pp. 163-171, 1995.

[32] K. Witheephanich, J. M. Escano, and M. J. Hayes, "Ambulatory wireless sensor network power management using constrained explicit generalised predictive control," *International Journal of Control*, Vol. 84, pp. 1450-1466, 2011.

[33] L. Xia and X. R. Cao, "Performance optimization of queueing systems with perturbation realization," *European Journal of Operational Research*, Vol. 218, pp. 293-304, 2012.

[34] L. Xia, X. Chen, and X. R. Cao, "Policy iteration for customer-average performance optimization of closed queueing systems," *Automatica*, Vol. 45, No. 7, pp. 1639-1648, 2009.

[35] L. Xia and B. Shihada, "Max-min optimality of service rate control in closed queueing networks", *IEEE Transactions on Automatic Control*, Vol. 58, pp. 1051-1056, 2013.

[36] L. Xia and B. Shihada, "Decentralized control of transmission rates in energy-critical wireless networks," *Proceedings of the 2013 American Control Conference (ACC2013)*, June 17-19, 2013, Washington, DC., USA, pp. 113-118.

[37] R. Zhang, J. M. Gorce, and K. Jaffrs-Runser, "Energy-delay bounds analysis in wireless multi-hop networks with unreliable radio links," Technical Report 6598, ARES / INRIA, 2008.