PROGRESS IN ELECTRICAL ENGINEERING
Vol. 1, No. 1, 2020, 33-54

URL : www.arfjournals.com

CROSS-LAYER CONTROL SCHEME WITH DETERMINISTIC WAITING DELAY GUARANTEES IN MULTIHOP WIRELESS NETWORKS

ShuFAN*

*Audio-Visual and Image Technology Department, Criminal Investigation Police University of China, Shenyang 110854, *Correspondence E-mail: fanshufs@sina.com*

Received: 25 March 2020 / Revised: 4 April 2020 / Accepted: 19 June 2020 / Published: 25 August 2020

Abstract: Deterministic end-to-end delay guarantees are meaningful for real-time applications. Under the assumption that the hop-count from source node to destination node of each session is limited, if the worst case waiting delay in nodes can be guaranteed, the bounds of worst case end-to-end delay of sessions are deterministic.We propose a joint congestion control, routing and scheduling algorithm which can provides deterministic bounds of waiting delay of packets in queues. This scheme can adjust transmission rates and drop packets according to the status of the average end-to-end delay of sessions that are reflected through a virtual queue. This virtual queue can also strengthen constraints on the average end-to-end delay of sessions. Theoretical proof indicates that the scheme can guarantee bounded worst case waiting delay of packets in nodes. To reduce the computational complexity, a distributed routing and scheduling scheme is designed.Rigorous theoretical analyses indicate that the utility optimality and network stability can be maintained under the proposed scheme. Simulation results show that, compared with existing schemes, this proposed scheme can improve QoS performances on throughput and average end-to-end delay.

Keywords: delay guarantees; lyapunov optimization; multihop wireless networks; cross-layer control

1. INTRODUCTION

In multihop wireless networks, end-to-end delay is an important QoS performance metric. Multimedia applications such as medical monitoring and object tracking have high demands on real-time performances [1].Therefore, besides reducing end-to-end delay of data flows, it is also necessary to design algorithms that can provide bounded worst case end-to-end delay for real-time applications to increase QoS.

There have been various algorithms developed for reduction of end-to-end delay, including scheduling algorithms [2-4], MAC (Media Access Control) schemes [5-

7], routing algorithms [8-10] and backpressure-based cross-layer schemes [11-16]. However, they can not provide deterministic bound of end-to-end delay.

Several layered algorithms can guarantee worst case bounded end-to-end delay [2,3,17]. In [2], a scheduling policy to reduce end-to-end delay for real-time video streaming over IEEE 802.11e WLAN (Wireless Local Area Networks) is proposed. In this policy, Video packets are mapped into different queues according to their priorities. A packet is scheduled and dropped according to the value of Cwhose initial value is calculated using the delay estimation and the priority of the packet. However, the newly designed scheduling scheme in [2] mainly focuses on the order of the packets in the buffers. In addition, the scheduling of queues is operated using EDCA (Enhanced Distributed Channel Access), which may results in unbalance of transmission of packets with different priorities. The scheme in [3] constrains the end-to-end delay of video packets through discarding the video packets that are expected not to be played out in time. However, this algorithm only propose the adaptive early drop scheme of packets, without designing routing and scheduling policies. In [17], a TDMA-based integrated MAC and routing protocol is proposed to provide deterministic end-to-end delay guarantees. In the algorithm it is assumed that the sink is at the center of the circular sensing area that is divided into tiers and blocks based on the radial distance of nodes and the angular distance of nodes from the sink, respectively. Slots are assigned to blocks, and nodes in even (odd)-numbered blocks can reuse the same slot without interferences.However, the strict assumptions reduce the practicability of the algorithm.

Some backpressure-based algorithms that pay attention to bound of worst case end-to-end delay have been proposed.[18]gives bounds on average delay for maximal scheduling in wireless networks with different traffic sources. [19] studies delay properties of the maximum weight scheduling algorithm for both single-hop and multi-hop sessions. In [20], a joint congestion control, routing and scheduling algorithm which can satisfy average end-to-end delay guarantees is designed.The cross-layer control schemeproposed in [21] can also guarantee average end-to-end delay constraints through utilizing a virtual queue. However, these prior works can only keep the overall average delay of traffic flowswithin bounds based on Little's theorem. In [22] and[23], algorithms that can ensuredeterministic worst case delay guaranteesof individual sessions are proposed.Under the scheme proposed in [22], transmission opportunities are distributed to packets according to waiting time of head-of-line packets in each queue. This scheme can provide deterministic bounds on end-to-end delay of each traffic flow. However, the network in [22] is assumed to

be a one-hop network.For both single-hop and multi-hop networks, the scheduling algorithm in [23] can ensure a a bounded worst case waiting delay of packets buffered in each nodethrough developing a novel virtual queue.However, the scheme in [23] suffers from its poor throughput performance due to its serious packets drop decision.

By re-designing the virtual queues, the throughput is increased significantly in our work. Besides, our algorithm can also satisfy average end-to-end delay constraints of each session. The key contributions of this paper can be summarized as follows:

- The paper proposes a cross-layer QoS scheme that can keep the buffering delay of packets within deterministic bounds. This scheme can improve performances through adjusting transmission rates and dropping packets according to the status of the average end-to-end delay of sessions.
- The algorithm constructs a virtual queue to strengthen constraints on the average end-to-end delay of sessions.
- A distributed routing and scheduling algorithm is designed to reduce the computational complexity of the cross-layer QoS scheme.
- Both rigorous theoretical analyses and simulation results are provided to demonstrate that the utility optimality and network stability can be maintained under the proposed algorithm.

The remainder of this paper is organized as follows. Section 2 introduces the system model and problem formulation. In Section 3, the algorithm is designed using Lyapunov optimization. A distributed routing and scheduling algorithm is introduced in Section 4. The performance analyses of the proposed algorithm are presented in Section 5. The simulation results are given in Section 6. Finally, the conclusions are provided in Section 7.

2. MODEL AND PROBLEM FORMULATION

2.1. Network Model

In this paper, we consider a multihop wireless network which can be modeled by graph $G(N,L)$.The network operates in slotted time denoted by $t=\{0,1,2,\ldots\}$. Ndenotes the set of nodes and Lrepresents the set of directional wireless links in the network. (i,j) denotes the link from node i to node j. The set of data sessions m in the network is denoted by M. Each session has one source-destination pair. s_m is the source node of session m, and $d_m$is the destination node of session m. The set of source nodes and the set of destination nodes are denoted by N_s and N_d, respectively.Each node in the model contains three layers including transport layer, network layer and MAC layer.

At the transport layer,newly arriving data of session *m* first enters the transport layer storage reservoir in node s_m. We assume that all backlog storage reservoirs are infinite. The arrival data rate of session *m* at transport layer is denoted by $A_m(t) \in [0, A_{\max}^{(m)}]$. $A_{\max}^{(m)}$ is the allowable upper bound of arrival rate of session *m* at the transport layer of node s_m. $r_m(t)$ is the data amount of session *m* injected into the network layer from the transport layer in time slot *t*.Obviously, we can obtain $r_m(t) \in [0, A_m(t)]$.

At the network layer of each node *n*, every session *m* maintains its packets waiting for transmission in a separate data queue $Q_n^{(m)}$. $Q_n^{(m)}(t)$ denotes the queue backlog in time slot *t*. Dynamic evolution of $Q_n^{(m)}(t)$ is as follows.

$$Q_n^{(m)}(t+1) = \max\{Q_n^{(m)}(t) - \sum_{i \in O(n)} \mu_{ni}^{(m)}(t) - D_n^{(m)}(t), 0\} + \sum_{j \in I(n)} \mu_{jn}^{(m)}(t) + 1_{\{n=s_m\}} \cdot r_m(t) \quad (1)$$

where *O*(*n*) are the set of nodes which can receive data packets from node *n*. *I*(*n*) denotes the set of nodes which can send data packets to node *n*. $\mu_{ni}^{(m)}(t)$ represents data packet amount of session *m*transmitted on link (*n*, *i*) in time slot *t*. $\mu_{jn}^{(m)}(t)$ is the the incoming traffic of session *m* from node *j* to *n* in time slot *t*. The value of the indicator function $1_{\{n=s_m\}}$ will be set to be 1 if $n = s_m$, and 0 otherwise.The data packet amount of session *m* that node *n* decides to drop in time slot *t* is denoted by $D_n^{(m)}(t) \in [0, D_{\max}]$, where $D_{\max}$ is the maximal allowable amount of packets that can be dropped by one node in a time slot. C_0 denotes the transmission capacity that any linksupports in single time slot. Therefore, $\mu_{ni}^{(m)}(t)$ can be derived as:

$$\mu_{ni}^{(m)}(t) \in \{0, \min\{Q_n^{(m)}(t), C_0\}\}, \forall (n,i) \in L, n \neq d_m, m \in M \quad (2)$$

At the MAC layer, there are two channels including common control channel and data channel which use different communication frequencies in the network. Each node can broadcast control packets consisting of channel access negotiation information, lengths of queues and weight values of nodes on the common control channel. Each node can gain control information by monitoring the control channel. The data channel is used for data communication. If link (*n*,*j*) obtains data transmission opportuinty in scheduling of time *t*, $\alpha_{nj}(t)$ will be set to be 1. Otherwise, the value of $\alpha_{nj}(t)$ wil be 0. The following constraints must beguaranteed.

$$\sum_{j:(n,j) \in L} \alpha_{nj}(t) + \sum_{i:(i,n) \in L} \alpha_{in}(t) \leq 1 \quad (3)$$

$$\alpha_{nj}(t)+\sum_{k\in N}\sum_{l}\alpha_{kl}(t)\le 1 \tag{4}$$

Constraint (3) means that each node can establish data transmission with at most one other node during one time slot. In constraint(4), l denotes nodes in the transmission range of node n. Constraint (4) ensures that new data transmision on a link can not be established if the receiving node is interfered by other on-going transmissions. In the wireless networks, a link is shared by several sessions. In the same time slot, the total data transmission amount of all sessions on the link is constrained by the data amount can be transmitted on the link in a time slot. The constraint is as follow:

$$\sum_{m\in S(a,b)}\mu_{ab}^{(m)}(t)\le C_{ab}(t)\cdot t_{slot} \tag{5}$$

where $S(a, b)$ is the set of the sessions on link (a,b) . t_{slot} is the duration of a time slot. $C_{ab}(t)$ denotes the transmission capacity of link (a,b) in time slot t . $\mu_{ab}^{(m)}(t)$ is the data transmission amount of session m on link (a,b) in time slot t. Without loss of generality, in this paper, we assume that C_0 packets can be transmitted on each link in each time slot [24].

In this paper, we define $\bar{x}$ as the time average of $x(t)$. $\bar{x}$ can be cauculated using $\bar{x}=\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}E(x(\tau))$.

2.2. Throughput Utility Optimization Problem

In this paper, we define utility function of session m $U_m(.)$ as a strictly concave, twice differentiable, and non-decreasing function with $U_m(0)=0$. $\overline{r_m}$ denotes the time average throughput of session m. $\overline{d_n^{(m)}}$ represents the time average of $D_n^{(m)}(t)$. According to the physical meaning of the above expressions, $\overline{r_m}-\sum_{n\ne d_m}\overline{d_n^{(m)}}$ is the time average net throughput of session m.The throughput utility maximization problem *P1* can be designed as follows:

$$\begin{aligned}\text{maximize}\quad & \sum_{m\in M}U_m\left(\overline{r_m}-\sum_{n\ne d_m}\overline{d_n^{(m)}}\right)\\ \text{subject to}\quad & \bar{r}\in\Lambda\\ & (2),(3),(4)\end{aligned} \tag{6}$$

where Λ denotes the capacity region of the network. $\bar{r} = (\overline{r_m}, m \in M)$. Constraint (6)is set up to guarantee the network stability.However, if $r_m(t)$ and $D_n^{(m)}(t)$ are used as the variables of $U_m(.)$ in Lyapunov optimization framework,when $U_m(.)$ is nonlinear, $\sum_{m \in M} U_m(\overline{r_m} - \sum_{n \neq d_m} \overline{d_n^{(m)}})$ can not be guaranteed to be maximized[25]. To solve this problem, the throughput utility maximization problem *P1* can be transformed into *P2* as*:*

$$\begin{aligned} \text{maximize} \quad & \sum_{m \in M} U_m(\overline{\eta_m} - \sum_{n \neq d_m} \overline{d_n^{(m)}}) \\ \text{subject to} \quad & (2),(3),(4),(6) \\ & \bar{\eta} \leq \bar{r} \end{aligned} \tag{7}$$

where $\eta_m(t) \in [0, A_{\max}^{(m)}]$ is an auxiliary variable. $\overline{\eta_m}$ denotes the the time average value of $\eta_m(t)$ .We define β_m as the maximum slope of the utility function $U_m(r_m)$. Considering that $U_m(.)$ is concave, twice differentiable and non-decreasing, wecanobtain $U_m(0) = \beta_m$ .Clearly, the following inequality can be derived:

$$U_m(\overline{r_m} - \sum_{n \neq d_m} \overline{d_n^{(m)}}) \geq U_m(\overline{r_m}) - \beta_m \sum_{n \neq d_m} \overline{d_n^{(m)}}$$

Therefore, the throughput utility maximization problem *P3*can be derived as*:*

$$\begin{aligned} \text{maximize} \quad & \sum_{m \in M} U_m(\overline{\eta_m}) - \sum_{m \in M} \sum_{n \neq d_m} \beta_m \overline{d_n^{(m)}} \\ \text{subject to} \quad & (2),(3),(4),(6),(7) \end{aligned} \tag{8}$$

2.3. Virtual queue dynamics

In this paper, three kinds of virtual queues are also constructed, including virtual queue Y_m at transport layer of source nodes, virtual delay queue X_m at source nodes, and virtual persistent service queue $Z_n^{(m)}$ in each node.

Virtual queue Y_m is used to ensure that constriant (7), $\bar{\eta} \leq \bar{r}$, is satisfied. Then the stochastic inequality constraint is transformed into a queueing stability problems [25]. At each source node s_m, there is a virtual queue Y_m maintained for session *m*. The design of Y_m is as follows.

$$Y_m(t+1) = \max\{Y_m(t) - r_m(t), 0\} + \eta_m(t) \tag{9}$$

According to the properties for queue stability [25], constraint (7) will hold if Y_m is stable. $\overline{\eta_m}$ can be the lower bound of $\overline{r_m}$.

Virtual queue X_m is constructedfor session *m*at the source node s_m to satisfy the average end-to-end delay constraints. In each time slot *t*, the queue is updated as:

$$X_m(t+1) = \max\{X_m(t) - \rho_m \cdot r_m(t), 0\} + \sum\nolimits_{n\in N} Q_n^{(m)}(t) \tag{10}$$

where ρ_m is the threshold of the average end-to-end delay of session *m*.If each virtual queue X_m is stable, the following inequality can be derived:

$$\lim_{t\to\infty} \frac{\frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{n\in N} Q_n^{(m)}(t)}{\frac{1}{t}\sum_{\tau=0}^{t-1} r_m(t)} \le \rho_m \tag{11}$$

According to Little's Theorem, inequality (11) implies that the average end-to-end delay of session *m* is constrained by ρ_m.

Through the persistent service virtual queue, data queues with larger backlogs can gain higher transmission priorities.The persistent service queue proposed in this paper mostly relates to the queue $G_n^{(m)}$ designed in [23], which is updated in each time slot as follows:

$$G_n^{(m)}(t+1) = \max\{G_n^{(m)}(t) + 1_{\{Q_n^{(m)}(t)>0\}} \cdot (\varepsilon - \sum\nolimits_{i\in O(n)} \mu_{ni}^{(m)}(t)) - D_n^{(m)}(t) - 1_{\{Q_n^{(m)}(t)=0\}} \cdot \mu_n^{\max,out}, 0\} \tag{12}$$

where $\varepsilon > 0$ are pre-specified constants. $\mu_n^{\max,out}$ is defined as the maximal allowable packets amount that node *n* can send out in one time slot.Obviously, $\sum_{m\in M} \mu_{ni}^{(m)}(t)$ must be less than $\mu_n^{\max,out}$. However, Since $Q_n^{(m)}(t) > 0$ in most time slots, $G_n^{(m)}$ may maintain large which leads to dropping of packets according to the packet drop decision scheme. As a result, the throughput utilities are reduced significantly.To solve this issue, we propose a new persistent service queue denoted by $Z_n^{(m)}$, whose dynamic evolution is as follows:

$$Z_n^{(m)}(t+1) = \begin{cases} \max[Z_n^{(m)}(t) + \varepsilon_1 - D_n^{(m)}(t) - \sum\limits_{i\in O(n)} \mu_{ni}^{(m)}(t), 0] & if\ X_m(t) > 0 \\ \max[Z_n^{(m)}(t) + \varepsilon_2 - D_n^{(m)}(t) - \sum\limits_{i\in O(n)} \mu_{ni}^{(m)}(t), 0] & if\ X_m(t) = 0 \end{cases} \tag{13}$$

where constant $\varepsilon_1 > \varepsilon_2 > 0$ and $Z_n^{(m)}(0) = 0$. We can note that, the new queue $Z_n^{(m)}$ is updated considering the value of X_m which can reflect the degree of congestion more accurately than $Q_n^{(m)}$. Compared with the algorithm in [23], the average value of $Z_n^{(m)}$ should be smaller, and the number of packets dropped should decrease.

In addition, Theorem 1 can be obtained.

Theorem 1(Worst case Delay): Under FIFO(First-In-First-Out)queueing discipline, if $Q_n^{(m)}$ and $Z_n^{(m)}$ are bounded by constants $Q_n^{(m),\max}$ and $Z_n^{(m),\max}$ in any time slot, the waiting delay of packets in queues of session *m* at node *n* can be bounded by a deterministic constant $W_n^{(m),\max}$, which can be derived as follows:

$$W_n^{(m),\max} = \left\lceil (Q_n^{(m),\max} + Z_n^{(m),\max}) / \varepsilon_2 \right\rceil \tag{14}$$

where $\lceil x \rceil$ represents the smallest integer that is not less than*x*.

Proof: The proof is based on but different from the proof of Lemma 1 in [23].

If the theorem holds, all incoming data in $Q_n^{(m)}(t)$ should be either served or dropped in or before time slot $t + W_n^{(m),\max}$. The theorem is proved in three cases.

Case 1: If $X_m > 0$ in time slots $\tau \in \{t+1, ..., t+W_n^{(m),\max}\}$, obviously we can get:

$$Z_n^{(m)}(t+1) \geq Z_n^{(m)}(t) + \varepsilon_1 - \sum_{i \in O(n)} \mu_{ni}^{(m)}(t) - D_n^{(m)}(t)$$

Through summing the above expression over $\tau \in \{t+1, ..., t+W_n^{(m),\max}\}$, the following can be derived :

$$Z_n^{(m)}(t+1+W_n^{(m),\max}) - Z_n^{(m)}(t+1) \geq \varepsilon_1 \cdot W_n^{(m),\max} - \sum_{\tau=t+1}^{t+W_n^{(m),\max}} [\sum_{i \in O(n)} \mu_{ni}^{(m)}(t) + D_n^{(m)}(t)] \tag{15}$$

According to $Z_n^{(m)}(t+1+W_n^{(m),\max}) \leq Z_n^{(m),\max}$ and $Z_n^{(m)}(t+1) \geq 0$, the following can be derived from (15):

$$\varepsilon_1 . W_n^{(m),\max} - Z_n^{(m),\max} \leq \sum_{\tau=t+1}^{t+W_n^{(m),\max}} [\sum_{i \in O(n)} \mu_{ni}^{(m)}(t) + D_n^{(m)}(t)] \tag{16}$$

Accordint to $W_n^{(m),\max} = \left\lceil (Q_n^{(m),\max} + Z_n^{(m),\max}) / \varepsilon_2 \right\rceil \geq (Q_n^{(m),\max} + Z_n^{(m),\max}) / \varepsilon_2$ and $\varepsilon_1 > \varepsilon_2 > 0$, we can get:

$$\varepsilon_1 . W_n^{(m),\max} - Z_n^{(m),\max} \geq Q_n^{(m),\max} + Z_n^{(m),\max} - Z_n^{(m),\max} = Q_n^{(m),\max} \tag{17}$$

Clearly, the following can be obtain:

$$\sum_{\tau=t+1}^{t+W_n^{(m),\max}}[\sum_{i\in O(n)}\mu_{ni}^{(m)}(t)+D_n^{(m)}(t)]\geq Q_n^{(m),\max}\geq Q_n^{(m)}(t+1) \tag{18}$$

According to FIFO queueing discipline and (18), all the data packets buffered in $Q_n^{(m)}$ in time slot t + 1 can leave $Q_n^{(m)}$ before time slot $t+W_n^{(m),\max}$. Therefore, in this case, the upper bound of waiting delay of packets in queues of session m at node n is:

$$W_{n,case1}^{(m),\max}=\left\lceil (Q_n^{(m),\max}+Z_n^{(m),\max})/\varepsilon_2 \right\rceil \tag{19}$$

Case 2: If $X_m=0$ from time slot $t+1$ to $t+W_n^{(m),\max}$,it is easy to obtain the following inequality:

$$Z_n^{(m)}(t+1+W_n^{(m),\max})-Z_n^{(m)}(t+1)\geq \varepsilon_2\cdot W_n^{(m),\max}-\sum_{\tau=t+1}^{t+W_n^{(m),\max}}[\sum_{i\in O(n)}\mu_{ni}^{(m)}(t)+D_n^{(m)}(t)]$$

Using the procedure from (15) to (19) in case 1 as a guide, the upper bound of waiting delay of packets in queues of session m at node n in case 2 is:

$$W_{n,case2}^{(m),\max}=\left\lceil (Q_n^{(m),\max}+Z_n^{(m),\max})/\varepsilon_2 \right\rceil$$

Case 3: We assume that $n_1+n_2=W_n^{(m),\max}$. If from time slot $t+1$ to $t+W_n^{(m),\max}$, there are n_1 time slots with $X_m>0$ and n_2 time slots with $X_m=0$, the following can be derived:

$$Z_n^{(m)}(t+1+W_n^{(m),\max})-Z_n^{(m)}(t+1)\geq \varepsilon_*\cdot W_n^{(m),\max}-\sum_{\tau=t+1}^{t+W_n^{(m),\max}}[\sum_{i\in O(n)}\mu_{ni}^{(m)}(t)+D_n^{(m)}(t)]$$

where $\varepsilon_*=(\varepsilon_1\cdot n_1+\varepsilon_2\cdot n_2)/(n_1+n_2)$. It is obvious that $\varepsilon_2<\varepsilon_*<\varepsilon_1$. The upper bound of waiting delay of packets in queues of session m at node $n$in case 3 is:

$$W_{n,case3}^{(m),\max}=\left\lceil (Q_n^{(m),\max}+Z_n^{(m),\max})/\varepsilon_2 \right\rceil$$

According to the above three cases, $W_n^{(m),\max}$ can be calculated as:

$$W_n^{(m),\max}=\left\lceil (Q_n^{(m),\max}+Z_n^{(m),\max})/\varepsilon_2 \right\rceil$$

3. DYNAMIC ALGORITHM VIA LYAPUNOV OPTIMIZATION

In this paper, the throughput utility maximization problem *P3* is solved through Lyapunov optimization. $\Theta(t)=[Q(t),Y(t),Z(t),X(t)]$ denotes the network state vector

of time slot t. We define the Lyapunov function as a quadratic function of data queues $Q_n^{(m)}$ and virtual queues Y_m, $Z_n^{(m)}$ and X_m as:

$$L(\Theta(t)) = \tfrac{1}{2}[\sum_{m\in M}(Y_m(t))^2 + \sum_{n\neq d_m}\sum_{m\in M}(Q_n^{(m)}(t))^2 + \sum_{m\in M}(X_m(t))^2 + \sum_{n\neq d_m}\sum_{m\in M}(Z_n^{(m)}(t))^2] \tag{20}$$

Then, in the Lyapunov optimization framework, the drift-plus-penalty function which is used to maximize $\sum_{m\in M} U_m(\overline{\eta_m}) - \sum_{m\in M}\sum_{n\neq d_m} v_m \overline{d_n^{(m)}}$ can be derived as:

$$\Delta_V(\Theta(t)) = E\{L(\Theta(t+1)) - L(\Theta(t)) - V\cdot[\sum_{m\in M}U_m(\eta_m(t)) - \sum_{m\in M}\sum_{n\neq d_m}\beta_m D_n^m(t)] \mid \Theta(t)\} \tag{21}$$

where *V*represents the weight parameter of the utility in the optimization. Expectation of $\Delta_V(\Theta(t))$ satisfies that:

$$E\{\Delta_V(\Theta(t))\} \le B - \Psi_1(t) - \Psi_2(t) - \Psi_3(t) - \Psi_4(t) + \Psi_5(t) \tag{22}$$

where $\Psi_1(t)$, $\Psi_2(t)$, $\Psi_3(t)$, $\Psi_4(t)$ and $\Psi_5(t)$ are derived as:

$$\Psi_1(t) = \sum_{m\in M}(V\cdot U_m(\eta_m(t)) - \eta_m(t)\cdot Y_m(t)) \tag{23}$$

$$\Psi_2(t) = \sum_{m\in M} r_m(t)\cdot[Y_m(t) - Q_n^{(m)}(t)\cdot 1_{\{n=s_m\}} + X_m(t)\cdot\rho_m] \tag{24}$$

$$\Psi_3(t) = \sum_{n\neq d_m}\sum_{m\in M}\sum_{i\in O(n)}\mu_{ni}^{(m)}(t)\cdot[Q_n^{(m)}(t) - Q_i^{(m)}(t) + Z_n^{(m)}(t)] \tag{25}$$

$$\Psi_4(t) = \sum_{n\neq d_m}\sum_{m\in M} D_n^{(m)}(t)\cdot[Q_n^{(m)}(t) + Z_n^{(m)}(t) - V\cdot\beta_m] \tag{26}$$

$$\Psi_5(t) = \sum_{m\in M}\sum_{n\neq d_m} Z_n^{(m)}(t)\cdot\varepsilon_1 + \sum_{m\in M}\sum_{n\neq d_m} Q_n^{(m)}(t)\cdot X_m(t) \tag{27}$$

where *B* satisfies:

$$B \ge \tfrac{1}{2}\sum_{m\in M}[(\eta_m(t))^2 + (r_m(t))^2 + (\rho_m\cdot r_m(t))^2 + (\sum_{m\in M}Q_n^{(m)}(t))^2]$$

$$+\tfrac{1}{2}\sum_{m\in M}\sum_{n\neq d_m}[\varepsilon_1 - \sum_{i\in O(n)}\mu_{ni}^{(m)}(t) - D_n^{(m)}(t)]^2 + \tfrac{1}{2}\sum_{m\in M}\sum_{n\neq d_m}[(\sum_{j\in I(n)}\mu_{jn}^{(m)}(t) + 1_{\{n=s_m\}}\cdot r_m(t))^2$$

$$+(\sum_{i\in O(n)}\mu_{ni}^{(m)}(t) + D_n^{(m)}(t))^2] \tag{28}$$

Since $0 \le \eta_m(t) \le A_{\max}^{(m)}$, $0 \le r_m(t) \le A_{\max}^{(m)}$, $0 \le \mu_{ni}^{(m)}(t) \le C_0$, $0 \le D_n^{(m)}(t) \le D_{\max}$ and $Q_n^{(m)}(t) \le Q_n^{(m),\max}$ can be obtained, *B*can be regarded as a constant.

In the Lyapunov optimization framework, the network stability and utilities maximization can be obtained through minimization of the right-hand-side of (22).

The cross-layer algorithm CCWD (**C**ross-Layer **C**ontrol with **W**orst Case **D**elay Guarantees) algorithm congestion control scheme, routing and scheduling scheme and packet drop decision scheme.

Through the maximization of $\Psi_1(t)$ in (22), the value of virtual auxiliary variable $\eta_m(t)$ can be decided. The concaveoptimization problem with linear constraintsto choose $\eta_m(t)$ is derived as:

$$\text{maximize} \quad \Psi_1(t) \tag{29}$$

$$\text{subject to} \quad 0 \le \eta_m(t) \le A_{\max}^{(m)}.$$

If $U_m(.)$ is astrictly concave and twice differentiable function, $U_m(.)$'s first order derivative, which is denoted by $U_m^{'}(.)$, should be monotonic. Therefore, $U_m^{'}(.)$'s inverse function denoted by $U_m^{'-1}(.)$ can exist. Value of $\eta_m(t)$ can be decided through:

$$\eta_m(t) = \max\{\min\{U^{'-1}(Y_m(t)/V), A_{\max}^{(m)}\}, 0\}$$

$\Psi_2(t)$ in (22) is maximized by congestion control scheme which can be transformed into the following linear optimization problem as:

$$\text{maximize} \quad \Psi_2(t) \tag{30}$$

$$\text{subject to} \quad 0 \le r_m(t) \le A_m(t).$$

This problem can be solved as follows.

$$r_m(t) = \begin{cases} A_m(t) & \text{if } Y_m(t) > Q_{s_m}^{(m)}(t) - X_m(t)\cdot\rho_m \\ 0 & \text{if } Y_m(t) \le Q_{s_m}^{(m)}(t) - X_m(t)\cdot\rho_m \end{cases} \tag{31}$$

$\Psi_3(t)$ in (22) is maximized by routing and scheduling scheme which can be transformed into the following optimization problem as:

$$\text{maximize} \quad \Psi_3(t) \tag{32}$$

$$\text{subject to} \quad (2),(3),(4)$$

As the first step, the transmission capacityof link (n,i) should be distributed to the session m^* that satisfies:

$$m^* = \arg\max_{m\in M}\{Q_n^{(m)}(t) - Q_i^{(m)}(t) + Z_n^{(m)}(t)\} \tag{33}$$

The weight of link (n,i) distributed to session m^* is:

$$w_{ni} = Q_n^{(m^*)}(t) - Q_i^{(m^*)}(t) + Z_n^{(m^*)}(t) \tag{34}$$

Then, the optimization problem (32) can be transformed to a new problem as:

$$\text{maximize} \quad \sum_{n \neq d_m} \sum_{i \in O(n)} \mu_{ni}^{(m^*)}(t) \cdot w_{ni} \tag{35}$$

$$\text{subject to} \quad (2),(3),(4)$$

The solution variables of (35) is $\mu_{ni}^{(m)}(t)$ which implies transmission rates of session m^* on link (n,i). Problem (35) is a convex optimization problem, as its optimization objective function is linear and the constraint space of variables is convex.Problem (35) can be sloved through centralized algorithms whose complexity is $O(|N|^3)$, where $|N|$denotes the number of nodes in the network [26].

$\Psi_4(t)$ in (22) is maximized by packet drop decision scheme which can be transformed into the following linear optimization problem as:

$$\text{maximize} \quad \Psi_4(t) \tag{36}$$

$$\text{subject to} \quad 0 \le D_n^{(m)}(t) \le D_{\max}.$$

This problem can be solved as follows.

$$D_n^{(m)}(t) = \begin{cases} D_{\max} & if \ Q_n^{(m)}(t) + Z_n^{(m)}(t) > V \cdot \beta_m \\ 0 & if \ Q_n^{(m)}(t) + Z_n^{(m)}(t) \le V \cdot \beta_m \end{cases} \tag{37}$$

In each time slot, $Q(t)$, $Y(t)$, $X(t)$ and $Z(t)$ are updated according to(1), (9), (10) and (13).

4. DISTRIBUTED ROUTING AND SCHEDULING ALGORITHM

To reduce the computational complexity of the joint routing and scheduling scheme, we design a distributed routing and scheduling algorithm, which can be implemented at each node.

The details are as follows: (i) In the networks, there are two channels including common control channel and data channel. The packets carrying control information are switched on the common control channel. The control information of weight value is classified into three types, which are “candidate”, “send” and “receive”. (ii) In each time slot t, each node $n \neq d_m$ calculates the value of $w_{ni}(i \in N_{nei}(n))$, where $N_{nei}(n)$ is the set of neighbor nodes of node n. Each node monitors the common control channel and records all the received weight w_r whose type is “receive”. Choose node $i^* = \arg\max_{i \in N_{nei}(n)} |w_{ni}|$. If $w_{ni^*} > 0$ and the value of w_{ni^*} is larger than that of each

recorded w_r, w_{ni^*} is classified to be "candidate" and broadcasted on the common control channel by node *n*, as well as node *i* is notified that it has been chosen as the candidate target node of node *n*. Otherwise, each nodekeeps on sensing the common control channel and monitoring the control information from its neighbor nodes. (iii) After receiving the w_{ni^*} that is "candidate" from the neighbor nodes, node i^* chooses the node $n^* = \arg\max_{n \in R(i^*)} w_{ni^*}$, where $R(i^*)$ denotes the set of neighbor nodes that choose node i^* as their candidate target node. Node i^* broadcasts value of $w_{n^*i^*}$ which is set as "receive". Node i^* listens to the common control channel. If the value of $w_{n^*i^*}$ is larger than that of each received weight whose type is "send", node i^* will notify node n^* to send data packets. (iv) When receiving the notification of sending data from node i^*, node n^* sets $w_{n^*i^*}$ as "send" with broadcasting it on the common control channel, and send data to node i^* on the data channel. (v) If the node is not notified to send data after broadcasting weight information classified as "candidate", it will keep waiting till the end of the time slot.

In essence, the distributed routing and scheduling algorithm plays the same role to the GMS (Greedy Maximal Scheduling) method [14].

5. PERFORMANCE ANALYSIS

5.1. Overhead Analysis

X_m is updated at s_m using information of all $Q_n^{(m)}$, and at node n, $Z_n^{(m)}$ is updated according to X_m. Therefore, the overhead induced by queue length message will be increased. Each node can broadcasts the queue length message on the common control channel at the beginning of time slot. Total quantity of queue length message to be broadcasted is $2|M|(|N|+1)$ bytes where the number of nodes is $|N|$, the number of sessions is $|M|$, and quantity of queue length message of each session at each node is 2 bytes. If the bandwidth of the control channel is enough high, the ratio of duration of broadcasting the queue length message to the duration of a time slot will be low, and CCWD scheme can be carried out successfully in the networks.

5.2. Queue Length Analysis

Theorem 2(Bounded Queues): If $D_{\max} \geq \max\{\varepsilon_1, A_{\max}^{(m)} + \mu_n^{\max,in}\}$, in networks using CCWD, $Q_n^{(m)}$, $Z_n^{(m)}$ and Y_m can always be bounded by constants $Q_n^{(m),\max}$, $Z_n^{(m),\max}$ and $Y_m^{\max}$, respectively. Here, $\mu_n^{\max,in}$ is the maximal allowable packets amount that node *n* can receive in one time slot. $Q_n^{(m),\max}$, $Z_n^{(m),\max}$ and $Y_m^{\max}$ are constants as:

$$Y_m^{\max} = V \cdot \beta_m + A_{\max}^{(m)} \tag{38}$$

$$Q_n^{(m),\max} = V \cdot \beta_m + \mu_n^{\max,in} + 1_{\{n=s_m\}} \cdot A_{\max}^{(m)} \tag{39}$$

$$Z_n^{(m),\max} = V \cdot \beta_m + \varepsilon_1 \tag{40}$$

Proof: We use induction method to prove this theorem.Under induction method, if we can prove that $R(t+1) \le R_{\max}$ from the assumption of $R(t) \le R_{\max}$, we can ensure that $R(t) \le R_{\max}$ for all time slots.

We first assume that $Y_m(t) \le V \cdot \beta_m \le Y_m^{\max}$. According to (9) and $\eta_m(t) \in [0, A_{\max}^{(m)}]$, we can derive that:

$$Y_m(t+1) = \max\{Y_m(t) - r_m(t), 0\} + \eta_m(t) \le Y_m(t) + A_{\max}^{(m)} \le Y_m^{\max}$$

Then we assume that $V.\beta_m < Y_m(t) \le Y_m^{\max}$. Because we have $U_m(0) = \beta_m$ and $U_m(.)$ is strictly concave and twice differentiable, we can derive that:

$$V \cdot U_m(\eta_m(t)) - \eta_m(t) \cdot Y_m(t) \le V \cdot U_m(0) + V \cdot \beta_m \cdot \eta_m(t) - \eta_m(t) \cdot Y_m(t) = V \cdot U_m(0) + \eta_m(t) \cdot (V \cdot \beta_m - Y_m(t)) \le V \cdot U_m(0) \tag{41}$$

Under CCWD algorithm, $V \cdot U_m(\eta_m(t)) - \eta_m(t) \cdot Y_m(t)$ should be maximized. According to (41), it is obvious that $\eta_m(t)$ should be set to be 0 to maximize $V \cdot U_m(\eta_m(t)) - \eta_m(t) \cdot Y_m(t)$. According to (9) and $\eta_m(t) = 0$, we can get $Y_m(t+1) \le Y_m(t) \le Y_m^{\max}$. Therefore, Y_m is bounded by constant $Y_m^{\max}$ for all time slots.

We first assume that $Q_n^{(m)}(t) \le V.\beta_m$. According to (1), the definition of $\mu_n^{\max,in}$ and $r_m(t) \in [0, A_m(t)]$, we can get that:

$$Q_n^{(m)}(t+1) \le Q_n^{(m)}(t) + \sum_{j \in I(n)} \mu_{jn}^{(m)}(t) + 1_{\{n=s_m\}} \cdot r_m(t) \le V \cdot \beta_m + \mu_n^{\max,in} + 1_{\{n=s_m\}} \cdot A_{\max}^{(m)} = Q_n^{(m),\max}$$

Then we assume that $V \cdot \beta_m < Q_n^{(m)}(t) \le Q_n^{(m),\max}$. According to (37), $D_n^{(m)}(t)$ should be $D_{\max}$. Considering $D_{\max} \ge \max\{\varepsilon_1, A_{\max}^{(m)} + \mu_n^{\max,in}\}$ and (1), we can obtain that:

$$Q_n^{(m)}(t+1) \le Q_n^{(m)}(t) - D_{\max} + A_{\max}^{(m)} + \mu_n^{\max,in} \le Q_n^{(m)}(t) \le Q_n^{(m),\max}$$

Therefore, $Q_n^{(m)}$ is bounded by constant $Q_n^{(m),\max}$ for all time slots.

We first assume that $Z_n^{(m)}(t) \le V \cdot \beta_m$. According to (13), the following can be derived:

$$Z_n^{(m)}(t+1) \le Z_n^{(m)}(t) + \varepsilon_1 \le V \cdot \beta_m + \varepsilon_1 = Z_n^{(m),\max}$$

Then we assume that $V \cdot \beta_m < Z_n^{(m)}(t) \le Z_n^{(m),\max}$.According to (37), $D_n^{(m)}(t)$ should be $D_{\max}$. Considering $D_{\max} \ge \max\{\varepsilon_1, A_{\max}^{(m)} + \mu_n^{\max,in}\}$ and (13), we can derive that:

$$Z_n^{(m)}(t+1) \le Z_n^{(m)}(t) - D_{\max} + \varepsilon_1 \le Z_n^{(m)}(t) \le Z_n^{(m),\max}$$

Therefore, $Z_n^{(m)}$ is bounded by constant $Z_n^{(m),\max}$ for all time slots.

5.3. Utility Performance Analysis

Theorem 3: We define that $\varphi(r,D) = \sum_{m\in M} U_m(r_m) - \sum_{m\in M}\sum_{n\ne d_m} \beta_m D_n^{(m)}$. The optimization problem *P4*is defined as:

$$\text{maximize} \quad \varphi(r,D)$$

$$\text{subject to} \quad (2),(3),(4)$$

r_m^* and $D_n^{*(m)}$ are defined as the solutions of optimization problem *P4*. Then, we define φ^* as:

$$\varphi^* = \sum_{m\in M} U_m(r_m^*) - \sum_{m\in M}\sum_{n\ne d_m} \beta_m D_n^{*(m)}$$

In networks with CCWD algorithm we proposed, we can obtain that:

$$\sum_{m\in M} U_m(\overline{r_m}) - \sum_{m\in M}\sum_{n\ne d_m} \beta_m \overline{d_n^{(m)}} \ge \varphi^* - B/V$$

Proof: According to Lemma 4 in [23], for time slots $\tau = \{0,1,...,T-1\}$, we can derive that:

$$\sum_{\tau=0}^{T-1} E\{L(\Theta(t+1)) - L(\Theta(t))\} - V \cdot \sum_{\tau=0}^{T-1}(\sum_{m\in M} U_m(\eta_m(\tau)) - \sum_{m\in M}\sum_{n\ne d_m} \beta_m D_n^{(m)}(\tau))$$

$$\le B\cdot T - V\cdot T\cdot(\sum_{m\in M} U_m(\eta_m^*) - \sum_{m\in M}\sum_{n\ne d_m} \beta_m D_n^{*(m)}) - \sum_{\tau=0}^{T-1}\sum_{m\in M} Y_m(\tau)[r_m^* - \eta_m^*]$$

$$-\sum_{\tau=0}^{T-1}\sum_{n\ne d_m}\sum_{m\in M} Q_n^{(m)}(\tau)[\sum_{i\in O(n)} \mu_{ni}^{*(m)}(\tau) + D_n^{*(m)}(\tau) - \sum_{j\in I(n)} \mu_{jn}^{*(m)}(\tau) - 1_{\{n=s_m\}} r_m^*(\tau)]$$

$$-\sum_{\tau=0}^{T-1}\sum_{n\ne d_m}\sum_{m\in M} Z_n^{(m)}(\tau)[D_n^{*(m)}(\tau) + \sum_{i\in O(n)} \mu_{ni}^{*(m)}(\tau) - \varepsilon_1]$$

$$-\sum_{\tau=0}^{T-1}\sum_{m\in M} X_m(\tau)[\rho_m \cdot r_m^* - \sum_{n\in N} Q_n^{(m)}(t)] \tag{42}$$

According to Theorem 5.8 in [27], we can obtain the following inequality:

$$\sum_{m\in M} U_m(\overline{\eta_m}) - \sum_{m\in M}\sum_{n\neq d_m} \beta_m \overline{d_n^{(m)}} \geq \varphi^* - B/V \tag{43}$$

where B is a constant which satisfies (28). Since $\overline{\eta} \leq \overline{r}$ and $U_m(.)$ is a non-decreasingfunction, (44) can be derived from (43):

$$\sum_{m\in M} U_m(\overline{r_m}) - \sum_{m\in M}\sum_{n\neq d_m} \beta_m \overline{d_n^{(m)}} \geq \varphi^* - B/V \tag{44}$$

(44) implies that in networks using CCWD, the achieved overall throughput utility can arbitrarily close to the optimal value.

6. SIMULATION

6.1. Simulation Setup

For power control is not considered in the scheme, there is no physical unit for the parameters in simulations. The unit of transmission data amount is set as packet. The network in simulations includes 20 nodes. These nodes are randomly distributed in a square of 40×40.The transmission distance of any node is 25. In the simulation, nodes do not move. Each node is aware of the locations of other nodes in the network. The message broadcasted on the common control channel by any node can be received by any other node.Four unicast sessions are generated. Source and destination nodes of each sesssion will berandomly chosen from nodes in the network. Data are injected at the source nodes following Poisson arrivals. The simulation time lasts 10000 time slots. The transmission capacity of any link is 10packets/slot. All initial queue sizes are 0. Similar to [23], the throughput utility function is $U(x) = \log(x+1)$, and β_m is 1. $A_{\max}^{(m)} = A_m(t) + 0.1$.

In the simulation, the performance of CCWD is compared with that of NeelyOpportunistic [23] and PDA-PMF [2].

In CCWD, $\varepsilon_1 = 2$, $\varepsilon_2 = 1$, $D_{\max} = \max\{\varepsilon_1, A_{\max}^{(m)} + \mu_n^{\max,in}\}$, $\mu_n^{\max,in} = C_0 = 10$. The threshold of the average end-to-end delay of session m, $\rho_m = 200$. In NeelyOpportunistic, $\varepsilon = 2$ and $D_{\max} = \max\{\varepsilon_1, A_{\max}^{(m)} + \mu_n^{\max,in}\}$. To make PDA-PMF scheme more comparable with CCWD and Neely Opportunistic, PDA-PMF scheme is modified as follows:

the scheduling priorities of packets are allocated according to the average delay of packets in queues. AODV is used as the routing scheme of PDA-PMF. The allowed bound of waiting delay in any buffer queue is calculated according to (14), (39) and (40). In addition, NeelyOpportunistic uses the distributed algorithm proposed in section 4 as routing and scheduling scheme, and PDA-PMF uses the distributed algorithm as scheduling scheme.

6.2. Performance under Different Average Data Arrival Rate

In Fig 1a, 1b, 1c, V is set to be 50 and the average data arrival rate is set to be from 1 to 10 packets/slot.

The average throughput achieved by CCWD, NeelyOpportunistic, and PDA-PMF are compared in Fig 1a. From the figure we can see that the average throughput of CCWD stops increasing when the average data arrival rate is higher than 5 packet/ slot.The average throughput achieved by CCWD remains higher than that of NeelyOpportunistic and PDA-PMF.In Fig 1b the average packet loss ratio under CCWD, NeelyOpportunistic and PDA-PMF are compared. From Fig 1b, it can be seen that the average packet loss ratio of CCWD remains lower than that of Neely Opportunistic and PDA-PMF. The reason is that, for the porposed novel persistent service virtual queue $Z_n^{(m)}$ updates according to X_m which can reflect whether the average end-to-end delay of session m meets the delay constraint ρ_m, the rate control and packet drop decision-making of CCWD is more effective, which reduces the average packet loss ratio. Therefore, the amount of packets dropped are reduced and the average throughput is increased.

In Fig 1c the average end-to-end delay under CCWD, NeelyOpportunistic and PDA-PMF are compared. The average end-to-end delay of CCWD is close to that of NeelyOpportunistic, and lower than that of PDA-PMF. The reason is that in CCWD and NeelyOpportunistic, the bound of waiting delay in a queue is deterministic. Fig 1c also shows that, as the average data arrival rate increases, the average end-to-end delay of CCWD and NeelyOpportunistic reduce. The reason is that, in backpressure-based algorithm, the packets are pushed from the source nodes to the destination nodes with the “gradient” induced by the differences of queue length of nodes. When the average data arrival rate is low, it needs more time to create the “gradient” from the source nodes to the destination nodes which causes the average end-to-end delay of sessions to be increased.

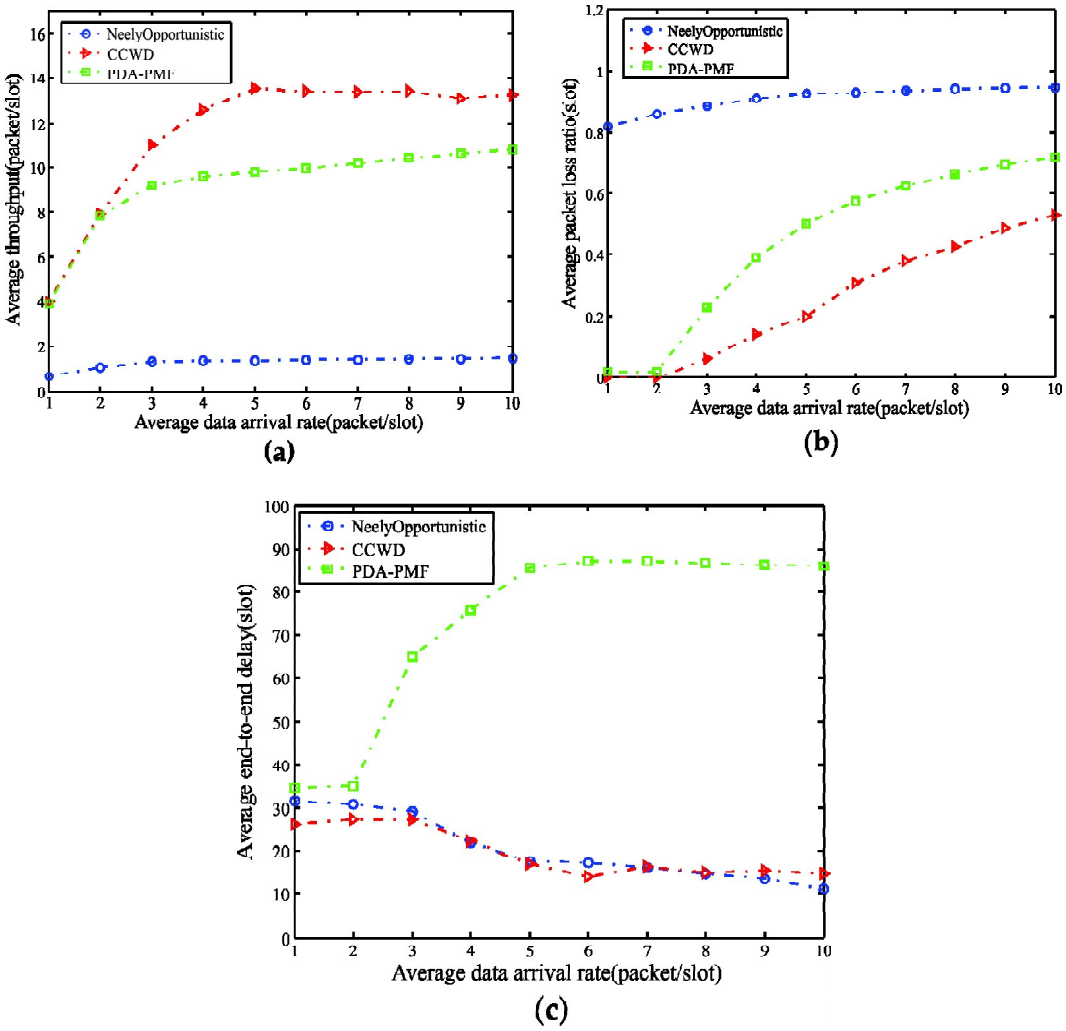

Figure 1: QoS performance versus average data arrival rate: (a) average throughput; (b) average packet loss ratio; (c) average end-to-end delay

6.3. Impact of V

In this section, parameter V is set as V = [50 100 150 200]. $A_m(t)$ of each session is 7packets/slot. The theoretical max size of Q queue, Y queue and Z queue are equal to $Q_n^{(m),\max}$, $Y_m^{\max}$ and $Z_n^{(m),\max}$, respectively.

In Fig 2a, 2b, 2c, it can be seen that under CCWD the max size of Q, Y and Z queue all increase linearly with value of V. The max size of Q, Y and Z queue are

obviously lower than $Q_n^{(m),\max}$, $Y_m^{\max}$ and $Z_n^{(m),\max}$, respectively. The results of Fig 2a,2b,2c verify Theorem 2.

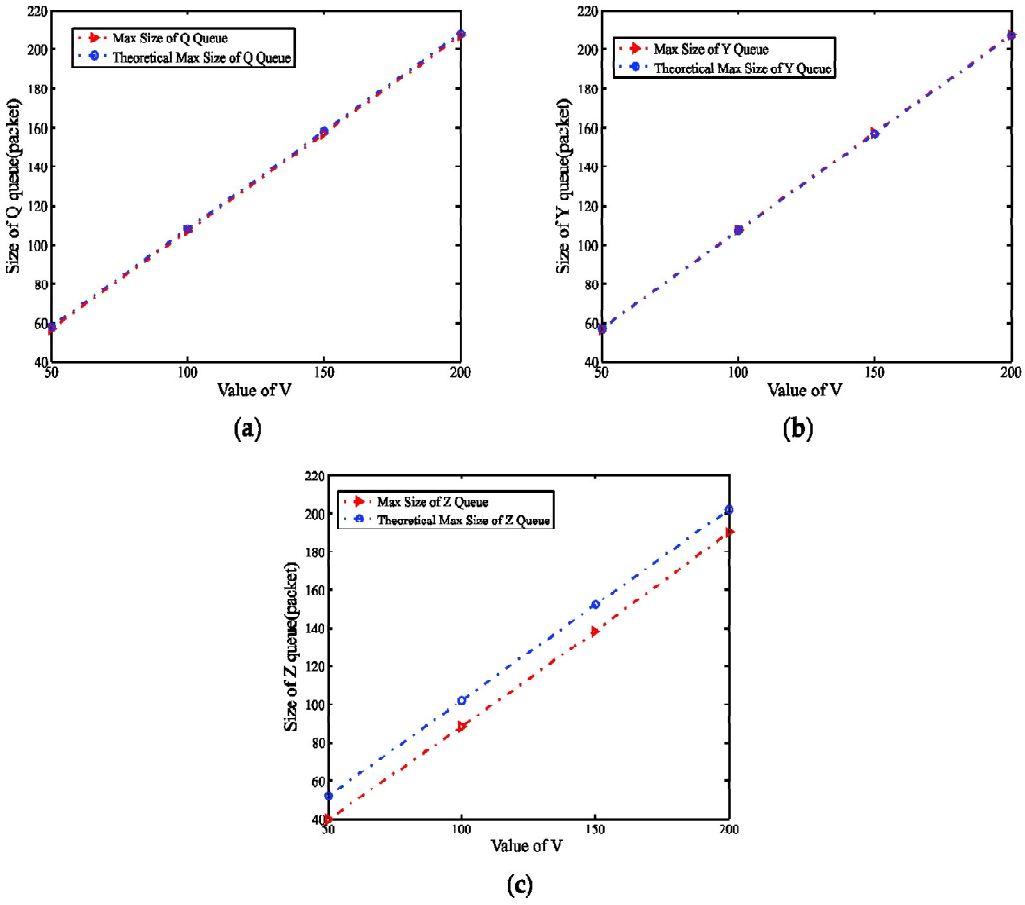

Figure 2: Queue length versus value of V: (a) size of Q queue; (b) size of Y queue; (c) size of Z queue

For φ^* in Theorem 3 is hard to calculate, and utility function $U_m(\cdot)$ is non-decreasing, the theoretical maximal utility in Fig 3 is calculated by using $\varphi^{**} = \sum_{m \in M} \log(A_{\max}^{(m)} + 1)$. Obviously, $\varphi^{**} \geq \varphi^*$ can be got. Fig 3 shows that the average overall utility increases with value of *V*, with approaching the value of theoretical maximal utility. The results of Fig 3 verify Theorem 3.

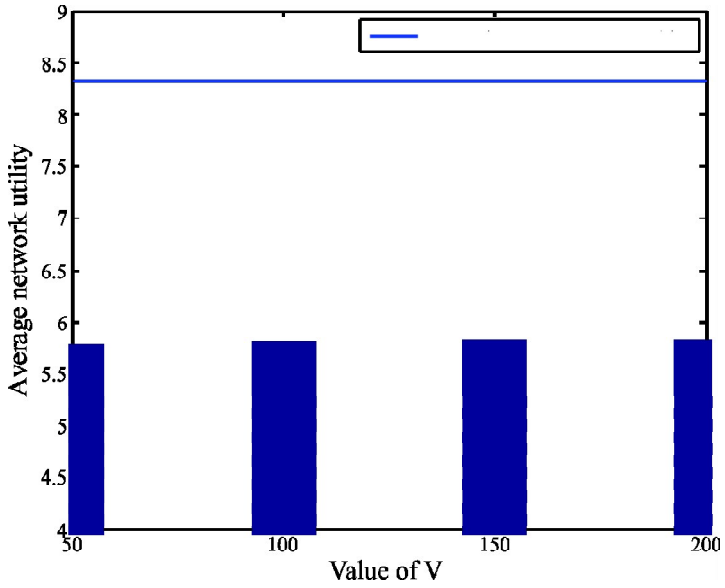

Figure 3: Average network utility versus value of V

7. CONCLUSIONS

This paper proposed a cross-layer QoSscheme, which can provide worst case waiting delay guarantees in nodes of wireless multihop networks. The scheme makesdecisions of rate control and packet dropping more effectively by updating queues according to whether the average end-to-end delay of sessions meets delay constraints.Rigorous theoretical analyses demonstrate the network stability and throughput utility optimality of the algorithm. Compared with existing works, the proposed algorithm achieves higher average throughput and lower average end-to-end delay. For future study, we plan to combine this policy with on-demand video streaming.

REFERENCES

[1] Mohammed,Z.H.;Hussain, A.; Fadi, A.A survey on multipath routing protocols for QoS assurances in real-time wireless multimedia sensor networks.*IEEE Communications Survey & Tutorials*, **2017**, *19*, 1424-1456.

[2] Wang, H.; Liu, G. Priority and delay aware packet management framework for real-time video transport over 802.11e WLANs.*Multimedia Tools and Applications*, **2014**,*69*, 621-641.

[3] Lee,S.;Yang,G.;Yoon,Y.; Lee, L.;Hyun, S. J. Delay-constrained adaptive early drop for real-time video delivery over IEEE 802.11 wireless networks.In Proceeding of International

Conference on Ubiquitous Information Management and Communication, Kota Kinabalu, Malaysia, 17-19 January 2013; pp. 1-5.

[4] Chen,W. Q.; Guan,Q. S.; Jiang, S. M.; Guan,Q. X.; Huang, T. C. Joint QoS provisioning and congestion control for multi-hop wireless networks.*EURASIP Journal on Wireless Communications and Networking*, **2016**.

[5] Doudou, M.; Djenouri,D.; Ordinas, J. M. B.; et al. Delay-efficient MAC protocol with traffic differentiation and runtime parameter adaptation for energy-constrained wireless sensor networks.*Wireless Networks*, **2016**, *22*, 1-24.

[6] Anbagi, I. A.; Kantarci, M. E.; Mouftah, H. T. Priority and delay-aware medium access for wireless sensor networks in the smart grid. *IEEE Systems Journal*, **2014**, *8*, 608-618.

[7] Raza, M.; Leminh, H.; Aslam, N.; Hussain, S. A novel MAC proposal for critical and emergency communications in industrial wireless sensor networks. *Computer and Electrical Engineering*, **2018**, *72*, 976-989.

[8] Hamid, Z.; Hussain, F. B.; Pyun, J. Y. Delay link utilization aware routing protocol for wireless multimedia sensor networks. *Multimedia Tools and Applications*, **2016**, *75*, 8195-8216.

[9] Li, X. J.; Liu, A. F.; Xie, M. D.; et al. Adaptive aggregation routing to reduce delay for multi-layer wireless sensor networks. *Sensors*, **2018**, *18*, 1-28.

[10] Sarka, A.; Murugan, T. S. Cluster head selection for energy efficient and delay-less routing in wireless sensor network. *Wireless Networks*, **2019**, *25*, 303-320.

[11] Xiong, H. Z.; Li,R.; Eryilmaz, A.; Ekici, E. Delay-aware cross-layer design for network utility maximization in multi-hop networks.*IEEE Journal on Selected Areas in Communications*, **2011**, *29*, 951-959.

[12] Ji, B.; Joo. C. H.; Shroff, N. B. Delay-based back-pressure scheduling in multihop wireless networks. *IEEE/ACM Transactions on Networking*, **2013**, *21*, 1539-1552.

[13] Huang, L. B.; Moeller, S.; Neely, M. J.; et al. LIFO-backpressure achieves near optimal utility-delay tradeoff," *IEEE/ACM Transactions on Networking*, **2013**, 21, 831-844.

[14] Jiao, Z. Z.; Yao, Z.; Zhang, B. X.; et al. An efficient network-coding based back-pressure algorithm for wireless multi-hop networks.*International Journal of Commnunication Systems*, **2016**, [Online] Available: http://doi.org/10.1002/dac.3180.

[15] Wu, J.; Ghosal, D.; Zhang, M.; et al. Delay-based traffic signal control for throughput optimality and fairness at isolated intersection.*IEEE Transactions on Vehicular Technology*, **2018**, *67*, 896-909.

[16] Hai, L.; Gao, Q. H.; Wang, J.; et al. Delay-optimal back-pressure routing algorithm for multihop wireless networks. *IEEE Transactions on Vehicular Technology*, **2018**, *67*, 2617-2630.

[17] Shanti, C.; Sahoo, A. DGRAM: A delay guaranteed routing and MAC protocol for wireless sensor networks. *IEEE Transactions on Mobile Computing*, **2010**, *9*, 1-9.

[18] Neely, M. J. Delay analysis for maximal scheduling with flow control in wireless networks with bursty traffic. *IEEE/ACM Transactions on Networking*, **2009**, *17*, 1146-1159.

[19] Le, L. B.; Jagannathan, K.; Modiano, E. Delay analysis of maximum weight scheduling in wireless ad hoc networks. In Proceedings of 43rd Annual Conference on Information Sciences and Systems, Baltimore, USA, 18-20 March 2009;pp. 389-394.

[20] Xue, D. Y.; Ekici, E. Delay-guaranteed cross-layer scheduling in multihop wireless networks. *IEEE/ACM Transactions on Networking*, **2013**, *21*, 1696–1707.

[21] Huynh, T.; Pham, N. T.; Lee, S. H.; et al. Dynamic control policy for delay guarantees in multi-hop wireless networks. *Wireless Personal Communications*, **2015**, *80*, 647-670.

[22] Neely, M. J. Delay-based network utility maximization. *IEEE/ACM Transactions on Networking*, **2013**, *21*, 41-54.

[23] Neely, M. J. Opportunistic scheduling with worst case delay guarantees in single and multihop networks.In Proceeding of IEEE INFOCOM, Shanghai, China, 30 June 2011; pp. 1728-1736.

[24] Li, H. X.; Huang, W.; Wu, C.; Li, Z. P.; Lau, F. C. M. Utility-maximizing data dissemination in socially selfish cognitive radio networks.In Proceedings of IEEE International Conference on Mobile Ad-hoc and Sensor Systems, Valencia, Spain, 17-22 Octobor 2011; pp. 212-221.

[25] Neely, M. J. Optimizing Time Average. In *Stochastic Network Optimization with Application to Communication and Queueing Systems*; Morgan & Claypool Publishers: San Rafael, CA, USA,2010; pp. 45-48, ISBN 9781608454563.

[26] Lin, X.;Shroff, N. B. The impact of inperfect scheduling on cross-layer congestion control in wireless networks. *IEEE/ACM Transactions on Networking*, **2006**, *14*, 302-315.

[27] Georgiadis, L.; Neely, M. J.; Tasiulas, L. Resource allocation and cross-layer control in wireless networks. *Foundations and Trends in Networking*, 2006, *1*, 1-144.