Analysis of Traffic Scheduling Technique for Smart Grid Mesh Networks

Hamid Gharavi and Chong Xu

National Institute of Standards and Technology

Gaithersburg, USA

Email: {hamid.gharavi, chong.xu}@nist.gov

Abstract

In this paper, we present a multi-gate mesh network architecture that has been developed to ensure high performance and reliability under emergency conditions when a system expects to receive power outage notifications and exchanges. In order to handle the metering traffic, under time varying outage conditions we introduce a multi-gate and single-class back-pressure based scheduling algorithm, which takes into account both the hop-count, as well as the queue length of each mesh node. An important requirement of this algorithm is that all the meter nodes should always maintain a separate path to each gateway. We first quantify the stability region of the network when our novel algorithm is applied to schedule the packets. We then present a numeric analysis to prove that the overall network delay is reduced as a result of employing the proposed scheduling algorithm. Moreover, we also theoretically prove that the network is always able to remain stable as long as the arrival rate vector lies inside the stability region provided by our scheduling algorithm. Finally, we derive a distributed objective function that is adopted by the practical implementation of the packet-scheduling scheme. The simulation results indicate that under the context of the multi-gate network, our packet-scheduling scheme can indeed significantly improve the network's reliability and delay performance, which are important factors under outage conditions.

I. Introduction

One of the most important issues in Smart Grid (as defined in the US Energy Independence and Security Act of 2007 [1]) is to provide a reliable and secure two-way end-to-end communications system for the Advanced Metering Infrastructure (AMI). The AMI system aims at providing consumers with knowledge of their energy usage and the capability of monitoring and controlling the electrical system components. While networking technologies and systems have been greatly enhanced, the smart grid faces challenges in terms of reliability and security in both wired and wireless communication environments. For instance, smart home appliances represent a major part of the Smart Grid vision of improving energy efficiency and they have to communicate with entities and players in other Smart Grid domains via home area networks and neighborhood area networks. For inter-operable networks, the appropriate use of wired and wireless technologies has been the main focus for smart grid last mile communication networks. One example of the latter is Power Line Communication (PLC) [2], which is receiving considerable attention for home area networking applications. At the same time, Wireless LAN (WLAN) techniques, such as the IEEE 802.11 family of standards [3] with their maturity and cost effectiveness, have been extensively deployed for wireless access and home entertainment. However, to provide a large coverage area for AMI in residential areas, multi-hop communication is vastly favored over long-range single-hop links. Indeed, the benefit of multi-hop transmissions is that of combating the rapid decay of the received electromagnetic signal strength, as the communication distance increases. Although there has been a tremendous advancement in mesh networking, from the architectural point of view, the AMI network should be designed to ensure a high degree of reliability, self-configuration, and self-healing. Meeting these requirements depends not only on the selection of a mesh routing protocol, medium Access Control (MAC) protocols, and physical (PHY) layer, but also on the nature of the traffic at its application layer.

For example, the time-varying traffic generated under emergency situations poses a significant challenge in ensuring the reliability and timeliness of the smart grid network. In particular, outage management is one example where a system expects to receive power outage notifications and an exchange of information among all the meters. This situation tends to increase traffic load, resulting in severe network congestions. Furthermore, in a multi-hop mesh network, a meter which represents a mesh node, should not only transmit its own generated packets, but also those received from neighboring meters.

In the case of a conventional neighborhood area network, a residential area is normally divided into separate regions where meters (i.e., mesh nodes) in each region communicate with the AMI headend through their local gateway point. Under such conditions, meters closer to the local gateway (i.e., last

hope nodes) are expected to experience more severe congestion than those further away and this could create a bottleneck, especially under outage conditions. In order to allow collective participation in the routing, it is advantageous to combine all the sub-networks into a larger network with multiple gateways where meters can access to any of the gateways based on local traffic activity [4].

Multiple gateway networks, also known as any-cast services, have been an important issue for internet access. Their function is to provide a mechanism that can select one of many servers in the network [5], [6]. For mobile ad-hoc and sensor networks, any-cast communications can be applied in situations where there are multiple sinks in the network and the main strategy is to find the nearest sink [7]–[9]. With these networks another challenge arises from the shared medium if a MAC/CSMA protocol, such as IEEE 802.11, is used. In this situation, interference due to traffic flows sharing the same path as well as other traffic using different links, could affect the network throughput and delay performance. Recently, there has been increasing interest in the design of distributed CSMA algorithms at the MAC layer to maximize network performance [10]–[15]. In [12] a CSMA algorithm with a rate control has been proposed for multi-hop networks. The combined algorithm can achieve an optimal performance under ideal conditions. The authors in [12] have also expanded their analysis for any-cast and multi-cast services, which are presented in the Appendix in [13]. However, the analysis which is based on the assumption of continuous back-off time and instantaneous channel feedback, ignores the effect of collusion. Although CSMA-based algorithms have shown a throughput optimality, their delay performance for practical applications can be worse than that of the Max-Weight algorithm [10], [11], [15], [16]. The basic concept of the Max-Weight (also known as back-pressure) algorithm for a multi-hop network was first introduced in [17]. It will schedule any packet through a specific route according to the queue-length difference of each single-direction single hop link. Essentially, the scheduling algorithm presented in [17] endeavors to mitigate the queue length difference between any pair of mesh nodes in the network to the maximum extent. They provide a statistical analysis to prove that the algorithm is able to achieve the maximum stability region, albeit without providing any distributed solution. Since then, this algorithm has found its application in many areas of wired or wireless communication systems [18]–[24]. Additionally, a large number of variants of the algorithm were put forward with different objective functions (OF) in wireless multi-hop networks. For instance, in [25], the authors modify the OF so that head-of-the-line packet delays are taken into consideration. In [26], [27], a related delay-based index policy that provides exponential weight to the delay (the so-called exponential rule) is shown to be throughput-optimal. The authors of [28] consider a single transmitter connecting to a number of destinations via an ad-hoc network. A separate queue

is maintained for each destination at each relay node. However, the delay performance of the original back-pressure algorithm [17] may become uncontrollable for the following two reasons. Firstly, given F classes of flows in the entire ad-hoc network, which are distinguished by the destination, F queues remain at each mesh node, although only one queue is served at a time. Based on this structure, the complexity of maintaining the queuing data at each node increases proportionally with the number of potential destinations, which further increases the delay of the original back pressure algorithm. Secondly, due to a lack of contribution from the hop-count to the OF, it is quite possible that the original back-pressure algorithm in [17] may route some packets through a much longer route rather than the shortest path to the obliged destination.

Against this background, the main objective in this paper is to design a low complexity Max-Weight distributed routing algorithm that can achieve a low delay performance. Therefore, **our first contribution is to propose a hop-count based single-class back-pressure scheduling algorithm, which can significantly reduce the delay when compared with the original back pressure algorithm [17]**. For instance, the authors of [29] have put forward a novel OF for the centralized algorithm, which jointly takes into account the hop-count as well as the queueing data waiting in the buffer for each node's concern. However, each packet has a single destination, which doesn't change the multi-class queueing data structure maintained at each node where a separate queue is maintained for every class of packets. Since the destination of each new packet injected into the mesh network was determined and could be any of the nodes constituting the network, the hop counts to all potential destinations need to be obtained from time to time. This definitely adds to the traffic load, as well as the complexity of calculating the OF, hence increasing the delay of the proposed algorithm.

As opposed to previous works [17], [25]–[29] **our second contribution is to embed our proposed scheduling algorithm into a multi-gate network structure, where the destination of any packet injected is not fixed beforehand.** In other words, the final destination may vary as the packet passes through each relay-node and cannot be determined, until it reaches the final destination. Given this flexibility, the processing delay is significantly reduced as only one queue needs to be maintained at each node. Based on the above distinct features of the proposed algorithm compared with the original back-pressure algorithm [17], **our next contribution is to quantify the stability region and analyze the reduction of the overall network delay, which is a result of employing our scheduling algorithm.** We will analyze the contradictory impact of the key parameters employed by the OF of our algorithm to enhance the throughput and reduce the delay, which leads to the necessity of finding appropriate values

for the parameters, so that a trade off between enhancing the stability region and reducing the network delay can be maintained. Additionally, we prove that the network is able to remain stable as long as the arrival rate vector lies inside the stability region.

The scheduling solutions presented by all the above mentioned back-pressure based algorithms, i.e. [17], [29], are achieved by optimizing the centralized OFs via exhaustively searching all possible scheduling solutions. These centralized algorithms require too much time and complexity to implement in the context of ad-hoc wireless communication. Thus, a large variety of distributed algorithms, such as those of [30]–[36] were proposed to apply the OFs featuring the centralized algorithms. **Our fourth contribution is to derive the distributed OF from the centralized OF used by our scheduling algorithm**, so that it can be used for practical implementation. Finally, for our implementation we use an extended experimental test-bed developed in [4]. This test-bed consists of four gateways and 48 mesh nodes where each node is generating variable bit rate (VBR) traffic, which are then forwarded towards the master gateway as their final destination.

The organization of the paper is as follows. In Section II the multi-gate mesh network architecture is presented. In Section III, the novel multi-gate hop-count and queue-length based back-pressure algorithm is proposed. Next the corresponding stability region is defined in Section III-A, in the context of the proposed algorithm. Moreover, the ability of our algorithm to reduce the overall network delay is numerically analyzed in Section III-B. Additionally, the capability of our algorithm to stabilize the network is proved in Section III-D and the appendix. In Section IV, the OF of the distributed implementation method is derived from the OF employed by the proposed centralized algorithm. Other aspects of the implementation of the proposed algorithm, together with the simulation results, which are obtained by using a proactive tree-based routing manner, are presented in Section V. This is then followed by the conclusion and final remarks, which are presented in Section VI.

II. Multi-gate Routing Network Architecture

A neighborhood area network may consist of multiple mesh sub-networks where mesh nodes representing meters in each sub-network can only access the local gateway. Each gateway, which is also referred to as the Data Aggregation Point (DAP) by the smart grid community is connected to the master gateway (headend) via a wired or wireless link. Due to the variable nature of the traffic, some gateways may suffer from more congestion than others. Under such conditions, nodes belonging to neighboring sub-networks cannot participate in the routing to reduce the traffic load. In order to allow collective participation in the routing, it would be advantageous to combine all the sub-networks into a larger network with

multiple gateways (DAPs) where all the meters can access any of the gateways, as shown in Fig. 1. This arrangement is expected to enhance the self-healing and self-organization abilities of the network if some of the gateways and nodes become non-operational or new nodes are added to expand the network. Designing such a network would require developing a flexible multi-gate routing protocol so that each node (meter) can have a separate path to the gateways

We use symbol $\mathbb{G}$ and $\mathbb{N}$ to denote the set containing all the G gateways $\mathtt{G}_1$, $\mathtt{G}_2$, ..., $\mathtt{G}_G$ and all the N mesh nodes in the entire investigated mesh network. Given any node $\dot{\mathtt{N}} \in \mathbb{N}$, $\hat{H}(\dot{\mathtt{N}})$ stands for the number of hops from node $\dot{\mathtt{N}}$ to the nearest gateway. More specifically, $\hat{H}(\dot{\mathtt{N}})$ can be defined as

$$\hat{H}(\dot{\mathtt{N}}) = \min_{\dot{\mathtt{G}} \in \mathbb{G}} H(\dot{\mathtt{N}} \rightarrow \dot{\mathtt{G}}), \tag{1}$$

where $H(\dot{\mathtt{N}} \rightarrow \dot{\mathtt{G}})$ stands for the number of hop counts associated with the shortest path from node $\dot{\mathtt{N}}$ to an arbitrary gateway $\dot{\mathtt{G}}$. As the maximum number of hops of a loop-free route is limited by a fixed number H, the complete set $\mathbb{N}$ can be divided into H subsets, which are denoted as $\mathbb{N}_h$ with $h = 1, \ldots, H$. For any given node $\dot{\mathtt{N}} \in \mathbb{N}$, we have $\dot{\mathtt{N}} \in \mathbb{N}_h$, iff $\hat{H}(\dot{\mathtt{N}}) = h$. Additionally, the queue length of the packets inside the buffer of node $\dot{\mathtt{N}}$ at the tth moment is denoted as $Q(\dot{\mathtt{N}}, t)$.

We denote symbol $\mathbb{L}$ as the set containing all the single-hop single-direction links. Suppose the mesh network encompasses L single-hop single-direction links, which are denoted as $\mathtt{L}_1$, $\mathtt{L}_2$, $\cdots$, $\mathtt{L}_L$, respectively. Intuitively, the hop-count difference between the transmitter and receiver of any single-hop link can only be one of the following values: $+1$, 0 or -1. According to this metric, we may divide the entire link-set $\mathbb{L}$ into three subsets $\mathbb{L}_{\text{I}}$, $\mathbb{L}_{\text{II}}$ and $\mathbb{L}_{\text{III}}$. More exactly, all the links belonging to $\mathbb{L}_{\text{I}}$ are named as Type-I links or forwarding transmissions, while the transmitters of all the Type-I links have one more hop-count than the receivers. Similarly, all the links belonging to $\mathbb{L}_{\text{II}}$ are named as Type-II links or peer-level transmissions, while the transmitters of all the Type-II links have the same number of hop-counts as the receivers. Lastly, all the links belonging to $\mathbb{L}_{\text{III}}$ are named Type-III links or backwards transmission, while the transmitters of all the Type-III links have one less hop-count than the receivers.

The routing matrix is denoted as $\mathbf{R}$, having elements from the set $\{+1, -1, 0\}$, representing all the one-hop links in the wireless mesh network. The N rows of matrix $\mathbf{R}$ correspond to the N mesh nodes, and its L columns correspond to all the L single-hop single-direction links. There are only two non-zero elements in each column of $\mathbf{R}$. More specifically, in the lth column of $\mathbf{R}$, the value of the element with its row-index corresponding to the transmitter node of $\mathtt{L}_l$ is $+1$, and that corresponding to the receiver node of $\mathtt{L}_l$ is assigned to -1.

The $(L \times 1)$-element binary vector $\mathbf{s}$ represents a potential activation vector. If the ith element of $\mathbf{s}$

is equivalent to one, Link $\mathtt{L}_i$ is activated. Otherwise, Link $\mathtt{L}_i$ is not activated. The element values of $\mathbf{s}$ are determined so that they comply with the selected interference model. That is, if all the links marked by non-zero elements of $\mathbf{s}$ are activated simultaneously, there will not be any packet-loss due to mutual link interference. Given a certain interference model, there will be more than one solution satisfying the requirement given by the model, which are able to offer mutual interference-free link combinations. For brevity, all possible solutions to a certain interference model are collectively included in one set $\mathbb{S}$. Therefore, the elements in $\mathbb{S}$ are recorded as $\mathbf{s}_1$, $\mathbf{s}_2$, ..., $\mathbf{s}_{|\mathbb{S}|}$, where $|\mathbb{S}|$ is the cardinality of $\mathbb{S}$ - the number of vectors in $\mathbb{S}$. $\mathbf{e} \in \mathbb{S}$ denotes the activation vector that is ultimately selected out among all the vectors in $\mathbb{S}$ for the next time slot, based on the OF of the scheduling algorithm.

III. Novel Scheduling Algorithm

The authors of [17] presented a throughput-optimized back-pressure based Max-Weight scheduling algorithm. It does not only quantify its stability region, but also proves that the network is always capable of maintaining stable status as long as the arrival rate vector is under the stability region. However, the centralized back-pressure algorithm in [17] does not take the delay issue into account. Hence, it will cause a longer-than-average delay and even routing loop problem. This would be a crucial issue in smart grid, specially under outage conditions. Nonetheless, our algorithm can be regarded as a variant of the original back-pressure algorithm proposed in [17]. Given the same network topology however, its performance exceeds the original back-pressure algorithm in two aspects: firstly, by employing three parameters α, β and γ to quantify the metrics of Type-I, Type-II and Type-III links in respect to the hop counts, the average overall time for a packet to traverse the network from the source to the destination is dramatically reduced, as compared with [17]. Secondly by adopting a multi-gate structure the processing complexity imposed by the proposed scheduling algorithm would be much lower than the Max-Weight algorithm in [17], hence further reducing the processing delay. Each mesh-node would only require keeping one queue containing all the packets arriving at that node.

We assume that the hop-count of a node to the nearest gateway is known. In order to reduce the overall propagation delay, nodes with less hop-count to the nearest gateway should be the next hop. Moreover, a larger arrival rate will be accommodated if the average queue length of each node is reduced. For an arbitrary node $\mathtt{N}_{h,i} \in \mathbb{N}_h$, any of its neighbors is included in one of the sets $\mathbb{N}_{h-1}$, $\mathbb{N}_h$ and $\mathbb{N}_{h+1}$. Hence, in order to reduce the overall propagation delay of delivering the packets initiated at the node $\mathtt{N}_{h,i}$ to any of the G gateways, its neighboring nodes belonging to $\mathbb{N}_{h-1}$ have the privilege over the nodes from the sets $\mathbb{N}_h$ and $\mathbb{N}_{h+1}$ as the next hop.

In order to accommodate a larger arrival rate initiated at each node, the queue length of each node should be reduced rather than increased with time and a balance of the queue length among all nodes should be observed with time passing. Hence, the neighboring nodes of the transmitter node $\mathtt{N}_{h,i}$, having a shorter queue size, should enjoy the privilege of being a receiver. Based on the above discussion the activation vector $\mathbf{e}(t+1)$ can be calculated through maximizing the centralized OF containing both hop-count and queue-length metrics. More exactly:

$$\mathbf{e}(t+1) = \arg\max_{\mathbf{s}\in\mathbb{S}} \big\{\mathbf{d}^T(t)\bar{\mathbf{D}}\mathbf{s}\big\}, \tag{2}$$

where $\mathbf{s}$ is an arbitrary candidate activation vector in set $\mathbb{S}$. Additionally, the L elements in vector $\mathbf{d}(t)$ and the diagonal elements of the $(L\times L)$-element matrix $\bar{\mathbf{D}}$ in (2) respectively represent the queue-length difference and hop-count difference between the transmitter and receiver associated with the L links. Given the transmitting node of link $\mathtt{L}_i$ denoted as $\mathtt{T}(\mathtt{L}_i)$ and its receiver node as $\mathtt{R}(\mathtt{L}_i)$, the ith element of $\mathbf{d}(t)$ is defined as: $d_i(t) = Q\big(\mathtt{T}(\mathtt{L}_i), t\big) - Q\big(\mathtt{R}(\mathtt{L}_i), t\big)$.

In (2), the ith diagonal element of the $(L\times L)$-element diagonal matrix $\bar{\mathbf{D}}$ represents the metric of $\mathtt{L}_i$ in terms of hop counts. Hence $\bar{\mathbf{D}}$ can be considered as a function of $\mathbf{D}$, where $\mathbf{D}$ is a diagonal matrix with its ith diagonal element $D_i \in \{+1, 0, -1\}$, for all $i = 1, 2, \ldots, L$. The element D_i indicates the hop-count difference between the transmitter and receiver of link $\mathtt{L}_i$, and is defined as: $D_i = \hat{H}\big(\mathtt{T}(\mathtt{L}_i)\big) - \hat{H}\big(\mathtt{R}(\mathtt{L}_i)\big)$. Given $\mathtt{T}(\mathtt{L}_i) \in \mathbb{N}_h$, the value of D_i can only fall into one of the following three categories: if $\mathtt{L}_i$ is a Type-I link with $\mathtt{R}(\mathtt{L}_i) \in \mathbb{N}_{h-1}$, then $D_i = +1$; if $\mathtt{L}_i$ is a Type-II link with $\mathtt{R}(\mathtt{L}_i) \in \mathbb{N}_h$, then $D_i = 0$; if $\mathtt{L}_i$ is a Type-III link with $\mathtt{R}(\mathtt{L}_i) \in \mathbb{N}_{h+1}$, then $D_i = -1$. We use α, β and γ to quantify the hop-count related metric of all the Type-I, Type-II and Type-III links respectively. Since we want to reduce the overall propagation delay, less hop-count routes are preferred. Hence, the three parameters may be ranked in non-strict descending order as $\alpha \geqslant \beta \geqslant \gamma > 0$. In order to make the values of α, β and γ positive, as well as linearly proportional to the genuine hop-count difference $D_i \in \{+1, 0, -1\}$, the values of $\bar{D}_i \in \{\alpha, \beta, \gamma\}$ are set to be $\bar{\mathbf{D}} = (\mathbf{D} + \mu\mathbf{I})\nu$. If we want $\gamma > 0$, we should let $\mu > 1$ and $\nu > 0$.

A. Stability Region

According to [17], to enable the network to operate in steady status, the rate with which packets arrive at a certain mesh-node should be equal to the rate with which packets leave the mesh node. The dotted vector $\dot{\mathbf{a}}$ represents the arrival rates at the N mesh nodes at any moment. Generally speaking, any arrival rate vector $\dot{\mathbf{a}}$ belonging to the stability region $\mathbb{C}$ has to be equal or smaller than the maximum service rate that can be provided by the mesh-network. As for a mesh network, its topology is quantified by $\mathbf{R}$,

and a certain interference model is quantified by $\mathbb{S}$. Note that a server here refers to a transmission on a single-hop link carrying packets from a certain mesh-node to its neighbor. More quantitatively, suppose a service rate vector $\mathbf{f}$ is defined such that its ith element f_i quantifies the service rate of the ith server, given any arrival rate vector $\dot{\mathbf{a}}$ under stability region, it should satisfy the conservation equations, $\dot{\mathbf{a}} = -\mathbf{R}\mathbf{f}$. All the vectors $\mathbf{f}$ satisfying the conservation equation based on $\dot{\mathbf{a}}$ is named as an $\dot{\mathbf{a}}$-admissible flow vector. Additionally, all the $\dot{\mathbf{a}}$-admissible flow vectors are collectively defined by the set $\mathbb{F}_{\dot{\mathbf{a}}}$.

The matrix $\bar{\mathbf{D}}$ will generate a bias of different type of servers. As been defined previously, α is used to weigh all the Type-I servers (forwards links). Similarly, β is used to weigh all the Type-II servers (peer-level links) and γ is used to weigh all the Type-III servers (backward links). Apparently, the advantage of the forwarding service over the Type-II or Type-III service obviously causes a more frequent adoption of the Type-I service compared with the frequency of using the Type-I service given the same network topology, when the original back-pressure based scheduling algorithm in [17]. Similarly, the frequency of utilizing Type-II and Type-III services is reduced as a result of the presence of the hop-count weighting matrix $\bar{\mathbf{D}}$; more exactly β and γ in the OF (2) characterizing our scheduling algorithm. The stability region of the mesh-network is changed correspondingly, and needs to be re-quantified.

In other words, introducing the hop-count matrix $\bar{\mathbf{D}}$ in OF (2) is identical to associating every single-hop link with a metric ranging between $[0, 1]$. The metric of every Type-I link, considered by the proposed scheduling algorithm, is the same as that quantified by the original back-pressure algorithm, where the weighting factors of all the Type-I links are set to 1, which can be alternatively regarded as α/α. Additionally, the metric of every Type-II link considered by the proposed algorithm to calculate its OF is only a fraction of β/α of that required by the original algorithm in [17]. Finally, the metric of every Type-III link considered by the proposed algorithm to calculate its OF is only a fraction of γ/α of that employed by the original algorithm in [17]. The variation of the metrics of different type of links can be regarded as a result of a variation of the service rate of different types of servers. More exactly, as the metric of the Type-I link is not changed, the service rate provided by every Type-I link under our scheduling algorithm remains the same as that under the original scheduling algorithm [17]. The metric of each Type-II link and Type-III link is reduced to a factor of β/α and γ/α respectively. The reduction of the metric of the link can be equivalently regarded as the result of the reduction of the links service rate. Regardless of the queue length difference between the transmitter and receiver of each node-to-node flow, the service rate provided by every Type-II or Type-III link under the proposed algorithm is scaled down by a factor of β/α or γ/α, compared with the rate in [17].

Finally, the arrival rate vector $\dot{\mathbf{a}}$ under the stability region provided by the proposed algorithm can be quantified as:

$$\dot{\mathbf{a}} = -\mathbf{R}\bar{\mathbf{D}}'\mathbf{f} = -\frac{1}{\alpha}\mathbf{R}\bar{\mathbf{D}}\mathbf{f}. \tag{3}$$

Let $\mathbb{F}_{\dot{\mathbf{a}}}$ be the set of all flow vectors satisfying (3). Following the methodology and notation employed by [17], the set $\mathbb{C}'$ can be quantified as:

$$\begin{aligned}\mathbb{C}' =\{&\dot{\mathbf{a}} : \text{there exists } \mathbf{f} \in \mathbb{F}_{\dot{\mathbf{a}}},\ \mathbf{s} \in co(\mathbb{S}) \text{ such that for the corresponding } \mathbf{f},\\ &\text{we have } f_i < s_i \text{ if } s_i > 0 \text{ and } f_i = 0 \text{ if } s_i = 0\}.\end{aligned} \tag{4}$$

Similarly, the closure of $\mathbb{C}'$, namely $\bar{\mathbb{C}}'$ is defined as:

$$\bar{\mathbb{C}}' =\{\dot{\mathbf{a}} : \text{there exists } \mathbf{f} \in \mathbb{F}_{\dot{\mathbf{a}}},\ \mathbf{s} \in co(\mathbb{S}), \text{ such that } \mathbf{f} \preceq \mathbf{s}\}. \tag{5}$$

Finally, the stability region $\mathbb{C}$ is given so that inequation $\mathbb{C}' \subset \mathbb{C} \subset \bar{\mathbb{C}}'$

$$\mathbb{C}' \subset \mathbb{C} \subset \bar{\mathbb{C}}' \tag{6}$$

is satisfied.

B. Delay Reduction

In Section III-A, we have quantified the stability region of the network when our novel scheduling algorithm is applied. In this subsection, we will analyze the impact of our scheduling algorithm on network delay performance. Firstly, the scheduling algorithm will not generate influence on queueing delay, transmission delay and propagation delay. Additionally, as we have observed, the processing delay will be reduced at each node of the network employing our scheduling algorithm, since the reduction of the number of queues maintained at each node will significantly reduce the processing complexity, as a result of combining all the number of classes in [17] to a universal class. All in all, the delay of the network specifies how long it takes for a packet to travel across the network from its source node to its destination gateway. It usually contains two key parts: namely the overall propagation delay and overall waiting delay.

Assume that each single-hop link in the network leads to a unitary propagation delay. Thus the overall propagation delay is proportional to the number of hops adopted by a packet to travel across the network from its source node to its destination gateway. Additionally, the proposed scheduling algorithm will not have any effect on the entire waiting time. All in all, the most significant impact of our scheduling algorithm in the network delay is the overall propagation delay; more exactly, the total hop count. Let

us assume that the minimum number of hops required by a certain packet to reach its destination node from the source node is $\dot{H}$, which is regardless of the scheduling algorithm employed by the network. The total hop-counts genuinely cost by the packet to reach its destination can be quantified as:

$$H = \dot{H} + \sum_{i=1}^{l} \Delta h_i = l \tag{7}$$

where l is the total number of single-hop links actually traveled by the packet on its way to the destination. Meanwhile, Δh_i refers to the number of additional hop-counts added by the ith link taken by the current packet. More exactly, we have $\Delta h_i = 0$ if the ith link traveled by the current packet is a Type I (forwarding) link having its transmitter one hop-count further than its receiver. Then we have $\Delta h_i = 1$ if the ith link traveled by the current packet is a Type II link (peer-level transmission) with its transmitter having the same hop-counts as its receiver. Finally, $\Delta h_i = 2$ extra hop counts will be added if the ith link traveled by the current packet is a Type III link (backwards transmission) with its transmitter one less hop-count to the nearest gateway than its receiver.

Now we investigate the mean value of the incremental hop counts $E(\Delta h)$ added by the links starting from an arbitrary mesh-node, which has n_{I} forward links, n_{II} peer-level links and n_{III} backward link. Suppose that the service rate of the ith forwarding, peer-level and backward link is $f_{\text{I},i} = E[f_{\text{I},i}(t)]$, $f_{\text{II},i} = E[f_{\text{II},i}(t)]$ and $f_{\text{III},i} = E[f_{\text{III},i}(t)]$ respectively. Thus, by defining

$$\hat{f} = \sum_{i=1}^{n_{\text{I}}} f_{\text{I},i} + \sum_{i=1}^{n_{\text{II}}} f_{\text{II},i} + \sum_{i=1}^{n_{\text{III}}} f_{\text{III},i}, \tag{8}$$

the average probability of utilizing the forwarding, peer-level and backwards link, when the original back pressure algorithm [17] is applied can be quantified as:

$$P_{\text{I}} = \frac{\sum_{i=1}^{n_{\text{I}}} f_{\text{I},i}}{\hat{f}},\ P_{\text{II}} = \frac{\sum_{i=1}^{n_{\text{II}}} f_{\text{II},i}}{\hat{f}} \text{ and } P_{\text{III}} = \frac{\sum_{i=1}^{n_{\text{III}}} f_{\text{III},i}}{\hat{f}}. \tag{9}$$

As a result, the average number of additional hops added by all the links transmitting from the current node can be quantified as:

$$\begin{aligned} E(\Delta h) =& P_{\text{I}}\Delta h_{\text{I}} + P_{\text{II}}\Delta h_{\text{II}} + P_{\text{III}}\Delta h_{\text{III}} \\ =& \frac{\sum_{i=1}^{n_{\text{I}}} f_{\text{I},i}}{\hat{f}} \times 0 + \frac{\sum_{i=1}^{n_{\text{II}}} f_{\text{II},i}}{\hat{f}} \times 1 + \frac{\sum_{i=1}^{n_{\text{III}}} f_{\text{III},i}}{\hat{f}} \times 2. \end{aligned} \tag{10}$$

On the other hand, when the novel scheduling algorithm is applied, the service rate of each forwarding link is the same, i.e. α/α as the original service rate. However, as discussed in Section III-A, the actual service rate of the Type-II (and Type-III) links will be reduced by a fraction of β/α and γ/α compared

with the original service rate, when the original back pressure [17] is applied. Ergo, by defining

$$\hat{\bar{f}} = \sum_{i=1}^{n_{\mathrm{I}}} f_{\mathrm{I},i} + \frac{\beta}{\alpha}\sum_{i=1}^{n_{\mathrm{II}}} f_{\mathrm{II},i} + \frac{\gamma}{\alpha}\sum_{i=1}^{n_{\mathrm{III}}} f_{\mathrm{III},i}, \tag{11}$$

the probability of purchasing a Type-I, Type-II and Type-III link, when our scheduling algorithm is applied, will be modified respectively to:

$$\bar{P}_{\mathrm{I}} = \frac{\sum_{i=1}^{n_{\mathrm{I}}} f_{\mathrm{I},i}}{\hat{\bar{f}}},\ \bar{P}_{\mathrm{II}} = \frac{\frac{\beta}{\alpha}\sum_{i=1}^{n_{\mathrm{II}}} f_{\mathrm{II},i}}{\hat{\bar{f}}} \text{ and } \bar{P}_{\mathrm{III}} = \frac{\frac{\gamma}{\alpha}\sum_{i=1}^{n_{\mathrm{III}}} f_{\mathrm{III},i}}{\hat{\bar{f}}}. \tag{12}$$

Thus, the mean value of additional hop-counts added by the links, starting from the current node when our scheduling algorithm is applied, can be evaluated as:

$$\begin{aligned} E(\Delta\bar{h}) =& \bar{P}_{\mathrm{I}}\Delta h_{\mathrm{I}} + \bar{P}_{\mathrm{II}}\Delta h_{\mathrm{II}} + \bar{P}_{\mathrm{III}}\Delta h_{\mathrm{III}} \\ =& \frac{\sum_{i=1}^{n_{\mathrm{I}}} f_{\mathrm{I},i}}{\hat{\bar{f}}} \times 0 + \frac{\frac{\beta}{\alpha}\sum_{i=1}^{n_{\mathrm{II}}} f_{\mathrm{II},i}}{\hat{\bar{f}}} \times 1 + \frac{\frac{\gamma}{\alpha}\sum_{i=1}^{n_{\mathrm{III}}} f_{\mathrm{III},i}}{\hat{\bar{f}}} \times 2. \end{aligned} \tag{13}$$

Comparing $E(\Delta h)$ in (10) with $E(\Delta\bar{h})$ in (13), immediately, we have $\bar{P}_{\mathrm{I}} \geqslant P_{\mathrm{I}}$, $\bar{P}_{\mathrm{II}} \leqslant P_{\mathrm{II}}$ and $\bar{P}_{\mathrm{III}} \leqslant P_{\mathrm{III}}$, for all kinds of network topology. As long as $\sum_{i=1}^{n_{\mathrm{II}}} f_{\mathrm{II},i} > 0$ or $\sum_{i=1}^{n_{\mathrm{III}}} f_{\mathrm{III},i} > 0$, we will have $E(\Delta\bar{h}) < E(\Delta h)$; while $E(\Delta\bar{h}) = E(\Delta h)$ is only achieved when $\sum_{i=1}^{n_{\mathrm{II}}} f_{\mathrm{II},i} = \sum_{i=1}^{n_{\mathrm{III}}} f_{\mathrm{III},i} = 0$. That is, when the novel scheduling algorithm is applied, the average extra hop counts added by the links starting from an arbitrary mesh node in the smart grid network is less than that when the original back-pressure scheduling algorithm [17] is applied. As a result, the overall propagation delay of a packet to travel across the network from its source to destination node is ultimately reduced.

C. Trade-off between Stability Region and Delay

As can also be seen from (13), the values of α, β and γ play an important role in determining the delay performance achieved by our scheduling algorithm. As discussed in Section III-A, it also plays an important role in determining the value of the stability region provided by our algorithm. Generally speaking, the stability region, which we want to increase, is roughly inversely proportional to the ratio of α/β and α/γ. Nevertheless, the average overall network delay, which we want to reduce, is also inversely proportional to the ratio of α/β and α/γ. Hence, a trade-off needs to be maintained with appropriate α/β and α/γ values.

Two extreme cases can be illustrated in more details to address this issue. Firstly, when $\alpha = \beta = \gamma$, there is no special preference on different types of routes, regardless of whether it is a forwarding link, peer-level transmission or backwards link. The average overall propagation delay cost of a packet to travel

to the destination node when the network employs our scheduling algorithm, is the same as that when the network employs the original back pressure algorithm in [17]. According to (13), with $\alpha = \beta = \gamma$, the overall network delay will reach its maximum value given $\alpha \geqslant \beta \geqslant \gamma$. Despite of the degraded delay performance the stability region of the network is able to reach its peak value, as discussed in Section III-A.

On the other hand, the opposite case occurs when $\alpha \to \infty$, and β, γ both have finite values. According to (13), with this set of values for α, β and γ, we will always have $E(\Delta\bar{h}_i) = 0$ for all $i = 1, \ldots, L$. In other words, when $\alpha \to \infty$, and β, γ both have finite values, the hop-counts taken by any packet when scheduled according to our scheduling algorithm, will always remain at its minimum hop-count benchmark value, as only forwarding links are activated on its way. Ergo, the average overall propagation delay of the network reaches its minimum value. However, under such a scenario the stability region for the arrival rate will also be reduced to its minimum value zero. In other words, network stability is not guaranteed under such conditions. As can be seen from the above discussion, the choice of whether enlarging or reducing the value of α, β and γ is contradictory in terms of either reducing the overall network delay or enhancing the stability region. Therefore, an appropriate value of α, β and γ should be selected according to the specific maximum delay tolerance and maximum throughput requirements of the mesh network investigated.

D. Proof of the Network Stability Region

The dotted vector $\dot{\mathbf{q}} = [\dot{q}_1 \quad \dot{q}_2 \quad \cdots \quad \dot{q}_N]^T$ can be used to represent the queue-length values of all the N mesh nodes at any time slot. It can be regarded as an arbitrary instantaneous sample value of the random vector $\mathbf{q}(t)$. The system is considered to be stable if the queue-lengths of all the N mesh nodes are proved to have a tendency of being reduced. The different value of $\dot{\mathbf{q}}$ can be regarded as the different states of an aperiodic, irreducible discrete-time Markov chain on a countable stage space $\mathbb{Q}$. According to the Fosters theorem [37], the Markov system is stable, if we can find a Lyapunov function $\bar{V} : \mathbb{R}^N \to \mathbb{R}$ and a finite subset $\hat{\mathbb{Q}}$ of $\mathbb{Q}$ such that the following two conditions are fulfilled. Firstly, when $\dot{\mathbf{q}}$ falls into subset $\hat{\mathbb{Q}}$, the incremental amount of the Lyapunov function with the time index will not blow to infinity. On the other hand, when $\dot{\mathbf{q}}$ falls outside $\hat{\mathbb{Q}}$, the value of the Lyapunov function will be reduced during the next time slot. More quantitatively, as suggested in [17], the Lyapunov function and the subset $\hat{\mathbb{Q}}$ can be defined as $\bar{V}(\dot{\mathbf{q}}) = \dot{\mathbf{q}}^T\dot{\mathbf{q}} = \sum_{n=1}^{N} \dot{q}_n^2$ and $\hat{\mathbb{Q}} = \{\dot{\mathbf{q}} : \dot{\mathbf{q}}^T\dot{\mathbf{q}} \leqslant b_{\mathrm{q}}\}$ respectively, where b_{q} is a positive number. Let's assume that the arrival rate vector $\dot{\mathbf{a}}$ at an arbitrary moment is Poisson distributed with a mean and

variance of $\rho_{\mathfrak{a}}$. Given a positive number ϵ, as long as $\mathbf{a}(t+1)$ belongs to the stability region $\mathbb{C}$ and we can find a positive number b_{q} as a multi-variable function of ϵ and $\rho_{\mathfrak{a}}$, such that

$$E\Big[\mathbf{q}^T(t+1)\mathbf{q}(t+1)-\mathbf{q}^T(t)\mathbf{q}(t)\Big|\mathbf{q}(t)=\dot{\mathbf{q}}\Big]<\infty \qquad \text{if } \dot{\mathbf{q}}^T\dot{\mathbf{q}}\leqslant b_{\mathrm{q}} \tag{14}$$

$$E\Big[\mathbf{q}^T(t+1)\mathbf{q}(t+1)-\mathbf{q}^T(t)\mathbf{q}(t)\Big|\mathbf{q}(t)=\dot{\mathbf{q}}\Big]\leqslant -\epsilon \qquad \text{if } \dot{\mathbf{q}}^T\dot{\mathbf{q}}> b_{\mathrm{q}} \tag{15}$$

are both able to be achieved respectively, the mesh network is considered to be stable. Based on this principle, we will propose and prove the following theorem:

***Theorem** 1:* Given the activation vector $\mathbf{e}(t+1)$ that maximizes the value of centralized OF (2) characterizing our scheduling algorithm, an $\alpha \geqslant 1$ and the stability region $\mathbb{C}$ quantified by (6) and other formulae in Section III-A, if the arrival vector $\mathbf{a}(t+1)\in\mathbb{C}$, then the network is stable.

Proof: See Appendix, Sect. B. ■

IV. Distributed Implementation

The distributed algorithm proposed in this paper leads to a practical way to realize the centralized algorithm. Firstly, a local OF derived from the OF characterizing the proposed centralized algorithm as addressed in Section III is employed by each node as a criterion to select its next hop in the network. Thus, a single-hop link, which has the node as the sender and the next hop as the receiver, will be activated. Then, if collision occurs during transmission, a re-transmission will be called for. The attempt will not stop until it fails after a pre-stipulated number of times. Before revealing the distributed approach of applying the centralized algorithm featured by the OF (2) of Section III, the definition of the Single Node Metric (SNM) and Single Link Metric (SLM) will be presented. The SNM is the product of the queue length of a certain node, multiplied by its hop count to the nearest gateway. More quantitatively, given a node $\dot{\mathtt{N}}\in\mathbb{N}$, the SNM $W(\dot{\mathtt{N}},t)$ at time t can be defined as $W(\dot{\mathtt{N}},t)=Q(\dot{\mathtt{N}}_i,t)\hat{H}(\dot{\mathtt{N}})$, where $\hat{H}(\dot{\mathtt{N}})$ has been defined in (1). On the other hand, SLM represents the metric of a single-hop direct link. The value of SLM associated with a link roughly stands for the SNM-value difference between the transmitter and the receiver. More quantitatively, for a given link $\mathtt{L}_i\in\mathbb{L}$, with $i=1,2,\cdots,L$, the SLM $W(\mathtt{L}_i,t)$ can be defined as $W(\mathtt{L}_i,t)=d(\mathtt{L}_i,t)\bar{D}(\mathtt{L}_i)$, where the scalars $d(\mathtt{L}_i,t)$ and $\bar{D}(\mathtt{L}_i)$ quantify the queue-length difference and hop-count difference respectively, both between the transmitter node $\mathtt{T}(\mathtt{L}_i)$ and the receiver node $\mathtt{R}(\mathtt{L}_i)$ at time t. More exactly, we have $\bar{D}(\mathtt{L}_i)=\Big[\hat{H}\big(\mathtt{T}(\mathtt{L}_i)\big)-\hat{H}\big(\mathtt{R}(\mathtt{L}_i)\big)+\mu\Big]\nu$, where $\mu>1$ and $\nu>0$ make a significant impact on the values of α, β and γ and have to be appropriately selected, as discussed in Section III-C. It may be worth mentioning that $d_i(t)=d(\mathtt{L}_i,t)$ is the ith element of the $(L\times 1)$-element vector $\mathbf{d}(t)$ and $\bar{D}_i=\bar{D}(\mathtt{L}_i)$ is the ith diagonal element of the $(L\times L)$-element matrix $\bar{\mathbf{D}}$,

which constitute parts of the OF in (2) characterizing the centralized scheduling algorithm described in Section III. For a G-gateway system, let us assume that a potential transmitter node $\mathtt{N}_{h,\mathrm{T}}$ has M neighbors $\mathtt{B}_1(\mathtt{N}_{h,\mathrm{T}})$, $\mathtt{B}_2(\mathtt{N}_{h,\mathrm{T}})$, ..., $\mathtt{B}_M(\mathtt{N}_{h,\mathrm{T}})$, and we denote $\mathbb{B}(\mathtt{N}_{h,\mathrm{T}})$ as the neighbor set of $\mathtt{N}_{h,\mathrm{T}}$, which is defined as: $\mathbb{B}(\mathtt{N}_{h,\mathrm{T}}) = \{\mathtt{B}_m(\mathtt{N}_{h,\mathrm{T}})|\ m = 1, 2, \ldots, M\}$. The OF of the centralized algorithm, as quantified by (2), can be equivalently interpreted using the term SLM as:

$$\mathbf{e}(\mathbb{L})(t+1) = \arg \max_{\mathbf{s}(\mathbb{L})\subset\mathbb{S}(\mathbb{L})} \left\{ \sum_{\mathtt{L}_i\in\mathbf{s}(\mathbb{L})} \big[\mathrm{SLM}(\mathtt{L}_i, t)\big] \right\} = \arg \max_{\mathbf{s}(\mathbb{L})\subset\mathbb{S}(\mathbb{L})} \left\{ \sum_{\mathtt{L}_i\in\mathbf{s}(\mathbb{L})} \big[W(\mathtt{L}_i, t)\big] \right\}$$
$$= \arg \max_{\mathbf{s}(\mathbb{L})\subset\mathbb{S}(\mathbb{L})} \left\{ \sum_{\mathtt{L}_i\in\mathbf{s}(\mathbb{L})} \left\{ \Big[Q\big(\mathtt{T}(\mathtt{L}_i), t\big) - Q\big(\mathtt{R}(\mathtt{L}_i), t\big)\Big] \cdot \Big[\hat{H}\big(\mathtt{T}(\mathtt{L}_i)\big) - \hat{H}\big(\mathtt{R}(\mathtt{L}_i)\big) + \mu\Big] \cdot \nu \right\} \right\}, \tag{16}$$

where $\mathbf{e}(\mathbb{L})(t+1)$ is the set of links that are finally activated during the next time slot in a centralized algorithm and obviously $\mathbf{e}(\mathbb{L})(t+1) \subset \mathbb{L}$. Moreover $\mathbf{s}(\mathbb{L})$ is an arbitrary set of links that could be simultaneously activated without interference. Additionally, $\mathbb{S}(\mathbb{L})$ is a second-level set containing all the sets of mutually interference-free links that could be activated simultaneously. Apparently, $\mathbf{e}(\mathbb{L})(t+1) \subset \mathbb{S}(\mathbb{L})$ is a particular element of $\mathbb{S}(\mathbb{L})$ that can maximize the OF displayed in (16).

In the distributed counterpart, for a given node $\mathtt{N}_{h,\mathrm{T}}$, the concern is to select a single destination $\mathtt{R}(\mathtt{N}_{h,\mathrm{T}})$ among all its neighbors $\mathtt{B}_1(\mathtt{N}_{h,\mathrm{T}})$, $\mathtt{B}_2(\mathtt{N}_{h,\mathrm{T}})$, ..., $\mathtt{B}_M(\mathtt{N}_{h,\mathrm{T}})$ as the next hop. Naturally, the given node $\mathtt{N}_{h,\mathrm{T}}$ constitutes a transmitter node of a certain link $\mathtt{L}_i$, and the receiver node should be selected from its neighbor-set $\mathbb{B}(\mathtt{N}_{h,\mathrm{T}})$ based on a distributed OF. Due to a lack of SNM knowledge of the nodes, other than opting for its neighbors in the distributed regime, the system is unable to jointly choose the transmitters and receivers of all the links to be simultaneously activated during the next time slot, as implied by the OF of the centralized algorithm in (16). Instead, each link with a fixed transmitter will choose its receiver independently in the distributed regime. As long as a receiver node $\mathtt{R}(\mathtt{N}_{h,\mathrm{T}})$ is selected from the neighbors of $\mathtt{N}_{h,\mathrm{T}}$, a link transmitting one packet from $\mathtt{N}_{h,\mathrm{T}}$ to $\mathtt{R}(\mathtt{N}_{h,\mathrm{T}})$ will be activated. More specifically, the next hop of a certain link $\mathtt{L}_i$ starting from $\mathtt{T}(\mathtt{L}_i)$ is selected as:

$$\mathtt{R}\big(\mathtt{T}(\mathtt{L}_i)\big)(t+1) = \arg \max_{\mathtt{R}(\mathtt{L}_i)\in\mathbb{B}(\mathtt{T}(\mathtt{L}_i))} \left\{ \Big[Q\big(\mathtt{T}(\mathtt{L}_i), t\big) - Q\big(\mathtt{R}(\mathtt{L}_i), t\big)\Big] \cdot \Big[\hat{H}\big(\mathtt{T}(\mathtt{L}_i)\big) - \hat{H}\big(\mathtt{R}(\mathtt{L}_i)\big) + \mu\Big] \cdot \nu \right\} \tag{17}$$

The equation in the brackets behind *max* in (17) is the closest approximation of the centralized OF that is equivalently displayed in (2) or (16), when the ith link $\mathtt{L}_i$ is concerned in the distributed scheduling algorithm. However, opposite to the OF of the centralized algorithm in (16), there is no element of other links $\mathtt{L}_j \neq \mathtt{L}_i$ presented in the OF of the distributed algorithm in (17) when the receiver of a certain link $\mathtt{L}_i$ is being selected. Below, we begin to simplify the distributed OF in (17) without incurring any performance loss. Since both the value of $Q(\mathtt{T}(\mathtt{L}_i), t)$ and $\hat{H}(\mathtt{T}(\mathtt{L}_i))$ makes no difference for different node

$\mathtt{R}(\mathtt{L}_i) \in \mathbb{B}\big(\mathtt{T}(\mathtt{L}_i)\big)$, the identical part related to the transmitting node in (17) can be omitted only when the receiver node is concerned. Then (17) can be equivalently written as:

$$\mathtt{R}\big(\mathtt{T}(\mathtt{L}_i)\big)(t+1) = \underset{\mathtt{R}(\mathtt{L}_i)\in\mathbb{B}(\mathtt{T}(\mathtt{L}_i))}{\arg\min}\Big\{Q(\mathtt{R}(\mathtt{L}_i),t)\big[\hat{H}(\mathtt{R}(\mathtt{L}_i))+\mu\big]\nu\Big\} = \underset{\mathtt{R}(\mathtt{L}_i)\in\mathbb{B}(\mathtt{T}(\mathtt{L}_i))}{\arg\min}\Big\{Q(\mathtt{R}(\mathtt{L}_i),t)\hat{H}(\mathtt{R}(\mathtt{L}_i))\Big\}. \tag{18}$$

Thus, the next hop of the first packet in the queue of $\mathtt{N}_{h,\mathrm{T}}$ is selected according to the following criterion:

$$\mathtt{R}(\mathtt{N}_{h,\mathrm{T}})(t+1) = \min_{\dot{\mathtt{N}}\in\mathbb{B}(\mathtt{N}_{h,\mathrm{T}})}\big\{\hat{H}(\dot{\mathtt{N}})\cdot Q(\dot{\mathtt{N}},t)\big\}. \tag{19}$$

As can be seen by the OF of (19), the distributed algorithm can only determine the next hop for a given transmitter node. When a certain node is chosen by several transmitter nodes as the next hop, the distributed manner has no way of predicting or scheduling these potential transmitting nodes. Conversely, according to the centralized algorithm given in Section III, the overall optimal arrangement will not allow one receiving node to have two transmitting nodes simultaneously. In the next subsection, we will discuss in detail the implementation procedure of the proposed routing protocol characterized by (19), as well as the practical methodology to solve the collision problem incurred by more than one simultaneous transmitting node as addressed above.

V. Implementation Model and Simulation Results

For implementation of the back-pressure routing scheme, we used the mesh network architecture shown in Fig. 1. A tree-base, multi-gate routing protocol developed in [4] was considered. The routing scheme is an extension of the Hybrid Wireless Mesh Protocol (HWMP) of the IEEE 802.11s [3]. Due to the static nature of the mesh network, we only consider the proactive part of this hybrid protocol. As mentioned earlier, in this network every mesh node in a tree not only generates its own packets, but also relays packets from its children nodes (except for the leaf nodes). As a result, the aggregated traffic in the upstream link tends to increase as the hop-count reduces. As can be seen in this figure each gateway ($\mathtt{G}_1$, $\mathtt{G}_2$, ..., $\mathtt{G}_G$) at the root of a tree periodically broadcasts root announcements to set up its tree. We use a randomization technique to avoid collision of root enhancement messages. It is important to note that a gateway in this network represents the last hop node in the upstream link where the master gateway (headend) is the final destination node. The MAC addresses of $\mathtt{G}_1$, $\mathtt{G}_2$, ..., $\mathtt{G}_G$ are employed as the unique identifications that correspond to the root announcements from the routing trees. In contrast to single gateway, a node in a multi-gate topology has multiple entries in its tree-table representing a separate path to each gateway [4]. With respect to the implementation of the proposed back-pressure scheme, the main objective would be the selection of the next hop based on the SNM value. Therefore, in obtaining the

SNM value, only the parent nodes may be used in the calculation. This is consistent with the OF (2) of the centralized algorithm, as the elements corresponding to the parents nodes in hop-count matrix $\bar{\mathbf{D}}$ are assigned with a higher value, as detailed in Section III. According to (19) in the proposed back-pressure scheme, the queue-length and hop-count have to be calculated by a transmitting node, before scheduling its packet to the next hop's node. To implement this, we consider using beacon frames, which are primarily used by a node to update its neighbors about its current route to the destination [3]. For example, when a node receives a beacon frame (or an association request) from its neighboring mesh node, it creates (or updates) a neighbor list according to the information in the beacon frame. The period of sending a beacon frame should depend on the traffic model. At a high packet rate, a faster update for calculating (19) may be needed and this would be at the expense of higher overhead [4]. It is important to emphasize that the proposed back-pressure scheme relies on the multi-path routing described earlier. Every node should possess an active path to all the gateways (or at least a few neighboring gateways for a large network). When a mesh node (meter) receives a packet from its upper layer (self generated packet) or a neighboring node (relayed packet), it checks its neighbor list and compares the corresponding metrics. As the parent list is updated by a root announcement process, the neighbors' list is updated and maintained through the beacon frames. In our implementation, the second smallest value in (19) will be selected from the same list to represent the next hop node in the case of link failure. In this back-pressure scheme the route error message is blocked in order to reduce the overhead in the network. For instance, when a link from node A to node B is broken due to consecutive packet losses, node A will re-schedule all the packets from its data queue by selecting the second best neighbor as its next hop destination. The main difference between the distributed approximation algorithm and centralized algorithm is that the interference between links cannot be ignored by the distributed algorithm. For the distributed network, the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) access protocol is commonly used by the IEEE 802.11 family. This protocol controls access to the shared wireless medium, which makes it very sensitive to interference caused by other active nodes. As a result, the stability region will be reduced as the interference would cause transmission failure of some links, which have been activated to mitigate the queue-length difference. This is despite the fact that the OF employed by the distributed algorithm is derived from our centralized algorithm. Similarly, as a result of failures in transmission caused by interference, the delay associated with the distributed algorithm is hence increased, compared with the centralized algorithm.

In our simulation model, a multi-gate network is constructed that comprises four gateways ($\mathtt{G}_1$, $\mathtt{G}_2$, $\mathtt{G}_3$,

G_4) and a master gateway (Headend) with 48 meters, which are uniformly distributed. The gateways are connected via wireless links to the Headend that represents the final destination in the upstream link. In our simulations the input data generated at a Variable Bit Rate (VBR), is encapsulated into fixed 512-byte user datagram packet (UDP) packets [4]. In the physical layer the IEEE 802.11b standard is used and the data-rate is 2 Mbps, while gateways are assumed to have an unlimited bit rate. A free space propagation channel with a path loss factor of 4 has been used in these experiments. Further details of the testbed can be found in [4].

In tree-based proactive routing, each gateway as the root of a tree, periodically floods the network by broadcasting a root announcement message. The period in which this message is generated depends on the nature of the application. For instance, in the case of the smart grid, it should be sufficiently long enough to reduce excessive overheads, but short enough to handle changes in the network structure, such as adding new meters or handling malfunctioning nodes (e.g., self healing). In our simulation, the root announcement is transmitted every 32 seconds. This is marginal when compared with the overhead associated with the beacon frame. In our experiments we observed that a beacon interval of 0.8s can achieve the best results. For the sake of comparison we also compare the performance of the combined multi-gate network operating both with and without packet scheduling. In the absence of packet scheduling, the shortest path leading to the nearest gateway is selected. Fig. 4 shows the effective throughput versus input bit rate performance at the final destination point (headend). This clearly indicates that a combination of multi-gate and back-pressure schemes can work very well together in achieving a higher performance. The next step in our evaluation is to compare the delay performance of the multi-gate scheme with and without packet scheduling. Fig. 5 shows their respective average end-to-end delay performances. As can be observed, the back-pressure scheme shows a significant improvement in delay performance as compared with the single best-path scheme. This is mainly due to the fact that the main cause of delay in the best path routing approach is to do with the path failure phenomenon. The link failure is the result of unsuccessful transmission/retransmissions of packets between two neighboring hops along the best path from a meter to the destination. Under these conditions, a new multihop path discovery process has to be initiated and this would consequently result in a sharper increase in delay as the traffic load increases. In contrast, the link failure phenomenon does not exist in the case of a distributed back-pressure scheme, where the packet delivery is performed on a hop-by-hop basis. From these results we can clearly observe that the flexibility of the back-pressure scheme combined with the multi-path feature of the multi-gate network structure, can be effectively distributed amongst the gateways.

VI. Conclusion

In this paper we propose a so-called multi-gate network structure and, under this architecture, we develop a novel packet-scheduling algorithm aimed at providing reliable two-way communications from meters to the AMI head-end, which can maintain a trade-off between maximizing the throughput and minimizing the average overall network delay. We quantify the stability region of the network when the packets are scheduled according to the proposed centralized algorithm. Numeric analysis of the propagation delay justifies the capability of our centralized scheduling algorithm in reducing the average overall network delay. We also prove that the network will always remain stable, regardless of the topology, time index or the current queue length status, as long as the average arrival rate vector stays inside the stability region.

Ultimately, we derive a novel distributed OF from the centralized algorithm and implement it with simulation. The simulation results further justify the ability of the proposed distributed scheduling algorithm to raise the network throughput and reduce the overall delay. Bear in mind that low latency is a crucial factor in delivering outage messages to the outage management system in order to fix the problem.

Appendix

A. Preparative Lemmas and Theorem

Before stating the proposition of the theorem, we first define $\hat{\mathtt{N}}_t$ as the node having the longest queue length among all the N mesh nodes at time t. Suppose $\hat{\mathtt{N}}_t \in \mathbb{N}_{\bar{H}_t}$, where $1 \leqslant \bar{H}_t \leqslant H$, we then define $\hat{\mathbb{N}}_{\bar{H}_t,t}$ as the set containing all the $\bar{H}_t$ nodes having the longest queue in each of the $\bar{H}_t$ subsets $\mathbb{N}_h$, $\forall h = 1, 2, \ldots . \bar{H}_t$ at the tth moment. More explicitly, we have $\hat{\mathbb{N}}_{\bar{H}_t,t} = \{\hat{\mathtt{N}}_{\bar{H}_t,t}, \ldots, \hat{\mathtt{N}}_{2,t}, \hat{\mathtt{N}}_{1,t}\}$, with $\hat{\mathtt{N}}_{h,t}$ representing the node having the longest queue among all the nodes in set $\mathbb{N}_h$ at the tth time slot.

1) Five Lemmas and One Corollary:

Lemma 1: Given a node $\mathtt{N}_{h,\mathrm{T}} \in \mathbb{N}_h$, with $h = 2, 3 \ldots, H$, and an arbitrary neighbor of it in $\mathbb{N}_{h-1}$ denoted as $\mathtt{N}_{h-1,\mathrm{R}}$, if at the $(t+1)^{\text{th}}$ time slot no single-hop link is activated, which takes $\mathtt{N}_{h,\mathrm{T}}$ as the transmitter and $\mathtt{N}_{h-1,\mathrm{R}}$ as the receiver, then the node-pair $(\mathtt{N}_{h,\mathrm{T}}, \mathtt{N}_{h-1,\mathrm{R}})$ can only fall in one of the following four statuses, corresponding to the conditions to fulfil Lemmas 2 to 5 respectively.

Proof of Lemma 1: When considered as a potential transmitter during the $(t+1)^{\text{th}}$ time slot, $\mathtt{N}_{h,\mathrm{T}}$ has only three possible statuses, which can be defined as: $\mathtt{S}_{\mathrm{T},1} = \{$ $\mathtt{N}_{h,\mathrm{T}}$ transmits a packet to $\mathtt{N}_{h-1,\mathrm{R}}\}$; $\mathtt{S}_{\mathrm{T},2} = \{$ $\mathtt{N}_{h,\mathrm{T}}$ transmits a packet, but not to $\mathtt{N}_{h-1,\mathrm{R}}\}$; $\mathtt{S}_{\mathrm{T},3} = \{$ $\mathtt{N}_{h,\mathrm{T}}$ does not transmit any packets$\}$. $\mathbb{S}_{\mathrm{T}}$ can be defined as the set containing all three elements, namely $\mathbb{S}_{\mathrm{T}} = \{\mathtt{S}_{\mathrm{T},1}$, $\mathtt{S}_{\mathrm{T},2}$ and $\mathtt{S}_{\mathrm{T},3}\}$.

Similarly, under the consideration of being a potential receiver during the $(t+1)^{\text{th}}$ time slot, $\mathtt{N}_{h-1,\mathrm{R}}$ has only three possible statuses, which can be defined as: $\mathtt{S}_{\mathrm{R},1} = \{$ $\mathtt{N}_{h-1,\mathrm{R}}$ receives a packet from $\mathtt{N}_{h,\mathrm{T}}\}$; $\mathtt{S}_{\mathrm{R},2} = \{$ $\mathtt{N}_{h-1,\mathrm{R}}$ receives a packet, but not from $\mathtt{N}_{h,\mathrm{T}}\}$; $\mathtt{S}_{\mathrm{R},3} = \{$ $\mathtt{N}_{h-1,\mathrm{R}}$ does not receive any packets$\}$. The set of these three elements can be written as $\mathbb{S}_{\mathrm{R}} = \{\mathtt{S}_{\mathrm{R},1}$, $\mathtt{S}_{\mathrm{R},2}$ and $\mathtt{S}_{\mathrm{R},3}\}$.

Ergo, there are nine ways to choose one element from each set of $\mathbb{S}_{\mathrm{T}}$ and $\mathbb{S}_{\mathrm{R}}$ as $|\mathbb{S}_{\mathrm{T}}| \times |\mathbb{S}_{\mathrm{R}}| = 9$. Among the nine distinct combinations, the following four combinations are illogical: $(\mathtt{S}_{\mathrm{T},1}, \mathtt{S}_{\mathrm{R},2})$, $(\mathtt{S}_{\mathrm{T},1}, \mathtt{S}_{\mathrm{R},3})$, $(\mathtt{S}_{\mathrm{T},2}, \mathtt{S}_{\mathrm{R},1})$, $(\mathtt{S}_{\mathrm{T},3}, \mathtt{S}_{\mathrm{R},1})$.

We define the set containing all the logical combinations as $\mathbb{P} = \{(\mathtt{S}_{\mathrm{T},1}, \mathtt{S}_{\mathrm{R},1})$, $(\mathtt{S}_{\mathrm{T},2}, \mathtt{S}_{\mathrm{R},2})$, $(\mathtt{S}_{\mathrm{T},2}, \mathtt{S}_{\mathrm{R},3})$, $(\mathtt{S}_{\mathrm{T},3}, \mathtt{S}_{\mathrm{R},2})$, $(\mathtt{S}_{\mathrm{T},3}, \mathtt{S}_{\mathrm{R},2})\}$. $\mathbb{P}$ can be further divided into two subsets: $\mathbb{P}_1$ and $\bar{\mathbb{P}}_1$, where $\mathbb{P}_1$ denotes the scenario when a packet is delivered from $\mathtt{N}_{h,\mathrm{T}}$ to $\mathtt{N}_{h-1,\mathrm{R}}$. $\bar{\mathbb{P}}_1$ is the complement of $\mathbb{P}_1$ and $\bar{\mathbb{P}}_1 = \{(\mathtt{S}_{\mathrm{T},2}, \mathtt{S}_{\mathrm{R},2})$, $(\mathtt{S}_{\mathrm{T},2}, \mathtt{S}_{\mathrm{R},3})$, $(\mathtt{S}_{\mathrm{T},3}, \mathtt{S}_{\mathrm{R},2})$, $(\mathtt{S}_{\mathrm{T},3}, \mathtt{S}_{\mathrm{R},2})\}$. As can be easily observed, the four elements of $\bar{\mathbb{P}}_1$, correspond to the conditions for fulfilling Lemma 2 to Lemma 5 respectively. ■

***Lemma** 2:* If Node $\mathtt{N}_{h,\mathrm{T}}$ is the transmitter $\mathtt{T}_1$ of a receiver $\mathtt{R}_1 \in \mathbb{N}_h \cup \mathbb{N}_{h+1}$ and Node $\mathtt{N}_{h-1,\mathrm{R}}$ is the receiver $\mathtt{R}_2$ of another transmitter $\mathtt{T}_2 \in \mathbb{N}_{h-1} \cup \mathbb{N}_h \cup \mathbb{N}_{h+1}$ at the $(t+1)^{\text{th}}$ time slot; then the sufficient condition for satisfying $Q(\mathtt{T}_2, t) > Q(\mathtt{R}_1, t)$ is that $\mathtt{N}_{h,\mathrm{T}} = \mathtt{T}_1 = \hat{\mathtt{N}}_{h,t}$.

***Proof of Lemma 2**:* The basic assumption for Lemma 2 to be verified lies in that: $\mathtt{T}_1 = \mathtt{N}_{h,\mathrm{T}} \in \mathbb{N}_h$, $\mathtt{R}_1 \in \mathbb{N}_h \cup \mathbb{N}_{h+1}$, $\mathtt{T}_2 \in \mathbb{N}_h \cup \mathbb{N}_{h-1} \cup \mathbb{N}_{h-2}$, $\mathtt{R}_2 = \mathtt{N}_{h-1,\mathrm{R}} \in \mathbb{N}_{h-1}$. Furthermore, the set satisfying the above condition can be further divided into two sub-cases, namely Case I and Case II.

Case I: $\mathtt{T}_2 \in \mathbb{N}_{h-1}$, $\mathtt{R}_1 \in \mathbb{N}_h$ and $\mathtt{R}_1$ is within the single-hop range of $\mathtt{T}_2$; as shown in Fig. 2.

Under this circumstance, a link between $\mathtt{T}_2$ and $\mathtt{R}_1$ could be established, but is not activated at the end. The reason is that the sum of the metrics of the two links $\mathtt{T}_2 \to \mathtt{R}_2$ and $\mathtt{T}_1 \to \mathtt{R}_1$ that are finally activated is bigger than the sum of the metrics of the two links $\mathtt{T}_1 \to \mathtt{R}_2$ and $\mathtt{T}_2 \to \mathtt{R}_1$ that are not activated. More quantitatively, if the link is from $\mathtt{T}_2$ to $\mathtt{R}_1$, we will have $\alpha[Q(\mathtt{T}_1, t) - Q(\mathtt{R}_2, t)] + \gamma[Q(\mathtt{T}_2, t) - Q(\mathtt{R}_1, t)] < \beta[Q(\mathtt{T}_2, t) - Q(\mathtt{R}_2, t)] + \beta[Q(\mathtt{T}_1, t) - Q(\mathtt{R}_1, t)]$. Otherwise, if the link is from $\mathtt{R}_1$ to $\mathtt{T}_2$, we will have $\alpha[Q(\mathtt{T}_1, t) - Q(\mathtt{R}_2, t)] + \alpha[Q(\mathtt{R}_1, t) - Q(\mathtt{T}_2, t)] < \beta[Q(\mathtt{T}_2, t) - Q(\mathtt{R}_2, t)] + \beta[Q(\mathtt{T}_1, t) - Q(\mathtt{R}_1, t)]$. Below we will show that both equations will lead to $Q(\mathtt{T}_2, t) > Q(\mathtt{R}_1, t)$.

a) When the other non-activated link is from $\mathtt{T}_2 \in \mathbb{N}_{h-1}$ to $\mathtt{R}_1 \in \mathbb{N}_h$, we have $\alpha[Q(\mathtt{T}_1, t) - Q(\mathtt{R}_2, t)] + \gamma[Q(\mathtt{T}_2, t) - Q(\mathtt{R}_1, t)] < \beta[Q(\mathtt{T}_2, t) - Q(\mathtt{R}_2, t)] + \beta[Q(\mathtt{T}_1, t) - Q(\mathtt{R}_1, t)]$. Since we have $Q(\mathtt{T}_1, t) > Q(\mathtt{R}_2, t)$ and $\alpha > \beta > \gamma$, we can then write $0 < (\alpha - \beta)[Q(\mathtt{T}_1, t) - Q(\mathtt{R}_2, t)] < (\beta - \gamma)[Q(\mathtt{T}_2, t) - Q(\mathtt{R}_1, t)] < Q(\mathtt{T}_2, t) - Q(\mathtt{R}_1, t)$. Finally, we arrive at $Q(\mathtt{T}_2, t) > Q(\mathtt{R}_1, t)$.

b) When the other non-activated link is from $\mathtt{R}_1 \in \mathbb{N}_h$ to $\mathtt{T}_2 \in \mathbb{N}_{h-1}$, we shall have $\alpha[Q(\mathtt{T}_1,t) - Q(\mathtt{R}_2,t)] + \alpha[Q(\mathtt{R}_1,t) - Q(\mathtt{T}_2,t)] < \beta[Q(\mathtt{T}_2,t) - Q(\mathtt{R}_1,t)] + \beta[Q(\mathtt{T}_1,t) - Q(\mathtt{R}_2,t)]$. It can be further derived as: $(\alpha-\beta)[Q(\mathtt{T}_1,t)-Q(\mathtt{R}_2,t)] < (\alpha+\beta)[Q(\mathtt{T}_2,t)-Q(\mathtt{R}_1,t)]$. Since we have $Q(\mathtt{T}_1,t) > Q(\mathtt{R}_2,t)$ and $\alpha > \beta$, we therefore have $Q(\mathtt{T}_2,t) > Q(\mathtt{R}_1,t)$.

Case II: All the other scenarios match the condition of Lemma 2, but do not satisfy the condition of Case I.

Under this situation, no single-hop direct link can be established between nodes $\mathtt{T}_2$ and $\mathtt{R}_1$. According to the scheduling principle, the superposition of the metrics of the two links $\mathtt{T}_2 \to \mathtt{R}_2$ and $\mathtt{T}_1 \to \mathtt{R}_1$ that are finally activated is bigger than the metric of the non-activated link $\mathtt{T}_1 \to \mathtt{R}_2$. For the sake of simplicity, in the scope of proof of Lemma 2 we use another variable η_1 to alternatively denote α, β or γ. Similarly, we use another variable η_2 to alternatively represent either β or γ. More quantitatively, we will have $\alpha[Q(\mathtt{T}_1,t) - Q(\mathtt{R}_2,t)] < \eta_1[Q(\mathtt{T}_2,t) - Q(\mathtt{R}_2,t)] + \eta_2[Q(\mathtt{T}_1,t) - Q(\mathtt{R}_1,t)]$, where $\eta_1 \in \{\alpha,\beta,\gamma\}$ and $\eta_2 \in \{\beta,\gamma\}$. Below we will prove the validation of $Q(\mathtt{T}_2,t) > Q(\mathtt{R}_1,t)$ under the following two scenarios.

a) When $\eta_1 = \alpha$ corresponding to $\mathtt{T}_2 \in \mathbb{N}_h$

According to the OF addressed in (2), we will always have $\alpha[Q(\mathtt{T}_1,t) - Q(\mathtt{R}_2,t)] < \alpha[Q(\mathtt{T}_2,t) - Q(\mathtt{R}_2,t)] + \beta[Q(\mathtt{T}_1,t) - Q(\mathtt{R}_1,t)]$. Hence, we arrive at $(\alpha-\beta)[Q(\mathtt{T}_1,t)-Q(\mathtt{T}_2,t)] < \beta[Q(\mathtt{T}_2,t)-Q(\mathtt{R}_1,t)]$. Since $\alpha > \beta$, the sufficient condition for $Q(\mathtt{T}_2,t) - Q(\mathtt{R}_1,t) > 0$ is $Q(\mathtt{T}_1,t) > Q(\mathtt{T}_2,t)$. Ultimately, when $\mathtt{T}_1$ and $\mathtt{T}_2 \in \mathbb{N}_h$, the sufficient condition of validating $\mathtt{N}_{h,\mathrm{T}} = Q(\mathtt{T}_1,t) > Q(\mathtt{T}_2,t)$ is

$$\mathtt{N}_{h,\mathrm{T}} = \mathtt{T}_1 = \hat{\mathtt{N}}_h. \tag{20}$$

b) When $\eta_1 \in \{\beta,\gamma\}$ corresponding to $\mathtt{T}_2 \in \{\mathbb{N}_{h-1} \cup \mathbb{N}_{h-2}\}$

According to the OF addressed in (2), we will always have $\alpha[Q(\mathtt{T}_1,t) - Q(\mathtt{R}_2,t)] < \eta_1[Q(\mathtt{T}_2,t) - Q(\mathtt{R}_2,t)] + \eta_2[Q(\mathtt{T}_1,t) - Q(\mathtt{R}_1,t)] < \beta[Q(\mathtt{T}_2,t) - Q(\mathtt{R}_2,t)] + \beta[Q(\mathtt{T}_1,t) - Q(\mathtt{R}_1,t)]$. Similarly, the relationship $(\alpha-\beta)[Q(\mathtt{T}_1,t) - Q(\mathtt{R}_2,t)] < \beta Q(\mathtt{T}_2,t) - \beta Q(\mathtt{R}_1,t)$ follows. Considering that $Q(\mathtt{T}_1,t) > Q(\mathtt{R}_2,t)$ and $\alpha > \beta$, we will finally arrive at $Q(\mathtt{T}_2,t) > Q(\mathtt{R}_1,t)$. Hence (20) would be the sufficient condition to achieve $Q(\mathtt{T}_2,t) > Q(\mathtt{R}_1,t)$, which ultimately demonstrates Lemma 2. ■

***Lemma** 3:* If Node $\mathtt{N}_{h,\mathrm{T}}$ is the transmitter $\mathtt{T}_1$ of a receiver $\mathtt{R}_1 = \mathtt{R}(\mathtt{N}_{h,\mathrm{T}}) \in \mathbb{N}_h \cup \mathbb{N}_{h+1}$ and Node $\mathtt{N}_{h-1,\mathrm{R}}$ is not receiving at the $(t+1)^{\text{th}}$ time slot; then $Q(\mathtt{N}_{h-1,\mathrm{R}},t) > Q(\mathtt{R}_1,t)$.

***Proof of Lemma 3**:* Node $\mathtt{N}_{h,\mathrm{T}} \in \mathbb{N}_h$ will transmit data to another receiver $\mathtt{R}_1$ either belonging to $\mathbb{N}_h$ or $\mathbb{N}_{h+1}$, but not $\mathtt{N}_{h-1,\mathrm{R}}$ during the $(t+1)^{\text{th}}$ time slot. Hence the difference of hop-count related metrics between the transmitter and the receiver of the link $\mathtt{N}_{h,\mathrm{T}} \to \mathtt{R}_1$ activated during the $(t+1)^{\text{th}}$ time slot can only be β or γ. For the sake of expression brevity, in the scope of proof of Lemma 3, we use another

variable η to alternatively denote either β or γ. On this basis, we will have $\eta[Q(\mathtt{N}_{h,\mathrm{T}},t)-Q(\mathtt{R}_1,t)]>\alpha[Q(\mathtt{N}_{h,\mathrm{T}},t)-Q(\mathtt{N}_{h-1,\mathrm{R}},t)]$, where $\eta\in\{\beta,\gamma\}$. Since $\alpha>\beta>\gamma$, we will have $\alpha[Q(\mathtt{N}_{h,\mathrm{T}},t)-Q(\mathtt{R}_1,t)]>\eta[Q(\mathtt{N}_{h,\mathrm{T}},t)-Q(\mathtt{R}_1,t)]>\alpha[Q(\mathtt{N}_{h,\mathrm{T}},t)-Q(\mathtt{N}_{h-1,\mathrm{R}},t)]$. Thus, $Q(\mathtt{N}_{h,\mathrm{T}},t)-Q(\mathtt{R}_1,t)>Q(\mathtt{N}_{h,\mathrm{T}},t)-Q(\mathtt{N}_{h-1,\mathrm{R}},t)$, where $Q(\mathtt{R}_1,t)<Q(\mathtt{N}_{h-1,\mathrm{R}},t)$ can be readily obtained. ■

***Lemma** 4:* If Node $\mathtt{N}_{h,\mathrm{T}}$ is not transmitting and Node $\mathtt{N}_{h-1,\mathrm{R}}$ is the receiver $\mathtt{R}_2$ of another transmitter $\mathtt{T}_2\in\mathbb{N}_{h-2}\cup\mathbb{N}_{h-1}\cup\mathbb{N}_h$ during the $(t+1)^{\text{th}}$ time slot; then $Q(\mathtt{T}_2,t)>Q(\mathtt{N}_{h,\mathrm{T}},t)$.

***Proof of Lemma 4**:* Given $\mathtt{R}_2\in\mathbb{N}_{h-1}$ as the receiving node, $\mathtt{T}_2$ can only be included in one of the three sets: $\mathbb{N}_h$, $\mathbb{N}_{h-1}$ and $\mathbb{N}_{h-2}$. According to the algorithm quantified by the objective function (OF) (2), we will have $\eta\big[Q(\mathtt{T}_2,t)-Q(\mathtt{R}_2,t)\big]>\alpha\big[Q(\mathtt{N}_{h,\mathrm{T}},t)-Q(\mathtt{R}_2,t)\big]$, where $\eta\in\{\alpha,\beta,\gamma\}$. Therefore, we will have $\alpha\big[Q(\mathtt{T}_2,t)-Q(\mathtt{R}_2,t)\big]\geqslant\eta\big[Q(\mathtt{T}_2,t)-Q(\mathtt{R}_2,t)\big]>\alpha\big[Q(\mathtt{N}_{h,\mathrm{T}},t)-Q(\mathtt{R}_2,t)\big]$, and finally $Q(\mathtt{T}_2,t)>Q(\mathtt{N}_{h,\mathrm{T}},t)$ follows. ■

***Lemma** 5:* If Node $\mathtt{N}_{h,\mathrm{T}}$ is not transmitting and Node $\mathtt{N}_{h-1,\mathrm{R}}$ is not receiving at the $(t+1)^{\text{th}}$ time slot; then $Q(\mathtt{N}_{h-1,\mathrm{R}},t)>Q(\mathtt{N}_{h,\mathrm{T}},t)$.

***Proof of Lemma 5**:* According to the scheduling policy described in Section III, the only circumstance that satisfies the condition of Lemma 5 is that $Q(\mathtt{N}_{h-1,\mathrm{T}},t)-Q(\mathtt{N}_{h,\mathrm{R}},t)<0$, namely $Q(\mathtt{N}_{h,\mathrm{T}},t)<Q(\mathtt{N}_{h-1,\mathrm{R}},t)$. ■

***Corollary** 1:* If Node $\mathtt{N}_{h,\mathrm{T}}$ is not transmitting and Node $\mathtt{N}_{h-1,\mathrm{R}}$ is not receiving at the $(t+1)^{\text{th}}$ time slot and $\mathtt{N}_{h,\mathrm{T}}=\hat{\mathtt{N}}_{h,t}$; then $Q(\hat{\mathtt{N}}_{h-1,t},t)\geqslant Q(\mathtt{N}_{h-1,\mathrm{R}},t)>Q(\hat{\mathtt{N}}_{h,t},t)$.

The proof of Corollary 1 is omitted for the sake of brevity, as it can be readily achieved given Lemma 5.

2) Queue-Length Theorem:

***Theorem** 2:* For any given activation vector $\mathbf{e}(t+1)$ determined at the tth moment, we can always find M_{t+1} links ($\mathtt{L}_{i,t+1}=\mathtt{T}_{i,t+1}\rightarrow\mathtt{R}_{i,t+1}$, $i=1,\ldots,M_{t+1}$) so that the first link starts from transmitter node $\hat{\mathtt{N}}_t$, the last link is $\mathtt{L}_{M_{t+1},t+1}=\mathtt{T}_{M_{t+1},t+1}\rightarrow\dot{\mathtt{G}}$, with $\dot{\mathtt{G}}$ being the closest gateway to $\mathtt{T}_{M_{t+1},t+1}$ and

$$\bar{\Delta}_{i,t+1}=-Q(\mathtt{R}_{i,t+1},t)+Q(\mathtt{T}_{i+1,t+1},t)\geqslant 0. \tag{21}$$

is achieved for all $i=1,\ldots,M_{t+1}-1$.

***Proof of Theorem 2**:* We will prove (21) by demonstrating the two propositions (Propositions 1 and 2) with mathematical induction starting from $h=\bar{H}_t$ to $h=1$ with an incremental step of -1.

***Proposition** 1:* When $h=\bar{H}_t$, we will have $\mathtt{T}_{1,t+1}=\hat{\mathtt{N}}_t$ and $Q(\mathtt{R}_{1,t+1},t)\leqslant Q(\hat{\mathtt{N}}_{\bar{H}_t-1,t},t)$.

Proof: When $h=\bar{H}_t$, according to the policy, $\hat{\mathtt{N}}_t$ must be the transmitter of $\mathtt{L}_{1,t+1}$. Suppose an arbitrary neighbor of $\hat{\mathtt{N}}_t$ in set $\mathbb{N}_{\bar{H}_t-1,t}$ is $\mathtt{N}_{\bar{H}_t-1,\mathrm{R}}$. a) If the receiver $\mathtt{R}_{1,t+1}\in\mathbb{N}_{\bar{H}_t-1}$, we will have $Q(\mathtt{R}_{1,t+1},t)\leqslant$

$Q(\hat{\mathtt{N}}_{\bar{H}_t-1,t}, t)$ naturally. b) if the receiver $\mathtt{R}_{1,t+1} \in \mathbb{N}_{\bar{H}_t} \cup \mathbb{N}_{\bar{H}_t+1}$, the condition of Lemma 3 is satisfied, we will have $Q(\mathtt{N}_{\bar{H}_t-1,\mathrm{R}}, t) > Q(\mathtt{R}_{1,t+1}, t)$, hence we have $Q(\mathtt{R}_{1,t+1}, t) < Q(\mathtt{N}_{\bar{H}_t-1,\mathrm{R}}, t) \leqslant Q(\hat{\mathtt{N}}_{\bar{H}_t-1,t}, t)$. ■

***Proposition** 2:* When $h = \dot{h}$, with $\dot{h} = \bar{H}_t - 1, \bar{H}_t - 2, \ldots, 2$, we assume that m links have been found hence (21) was valid for all $i = 1, \ldots, m-1$. If

$$Q(\mathtt{R}_{m-1,t+1}, t) \leqslant Q(\hat{\mathtt{N}}_{h,t}, t), \tag{22}$$

firstly we will locate Δm ($\Delta m \in \{0, 1, 2\}$) number of extra links and if $\Delta m > 0$ we will prove (21) is achieved with $i = m, \ldots, m + \Delta m - 1$. Secondly we will arrive at $Q(\mathtt{R}_{m+\Delta m,t+1}, t) \leqslant Q(\hat{\mathtt{N}}_{h-1,t}, t)$.

Proof: Regardless of the value of $\mathbf{e}(t+1)$, for the node $\hat{\mathtt{N}}_{h,t}$ with any $h \in \{\bar{H}_t - 1, \bar{H}_t - 2, \ldots, 2\}$, the joint scenarios of whether it is a transmitter and whether any of its neighbors $\mathtt{N}_{h-1,\mathrm{R}} \in \mathbb{N}_{h-1}$ is a receiver can only fall into one of the five cases below:

a If $\Delta m = 1$, $\mathtt{T}_{m+1,t+1} = \hat{\mathtt{N}}_{h,t}$, $\mathtt{R}_{m+1,t+1} = \mathtt{N}_{h-1,\mathrm{R}}$, we have $Q(\mathtt{R}_{m+\Delta m,t+1}, t) \leqslant Q(\hat{\mathtt{N}}_{h-1,t}, t)$. According to (22), we also have $\bar{\Delta}_{m,t+1} \geqslant 0$.

b If $\Delta m = 1$, $\mathtt{T}_{m+1,t+1} = \hat{\mathtt{N}}_{h,t}$, $\mathtt{R}_{m+1,t+1} \neq \mathtt{N}_{h-1,\mathrm{R}}$, the condition of Lemma 3 is satisfied. According to (22), $\bar{\Delta}_{m,t+1} \geqslant 0$ follows, According to the conclusion of Lemma 3 we will have $Q(\mathtt{R}_{m+\Delta m,t+1}, t) < Q(\mathtt{N}_{h-1,\mathrm{R}}, t) \leqslant Q(\hat{\mathtt{N}}_{h-1,t}, t)$.

c If $\Delta m = 2$, $\mathtt{T}_{m+1,t+1} = \hat{\mathtt{N}}_{h,t}$ and $\mathtt{R}_{m+2,t+1} = \mathtt{N}_{h-1,\mathrm{R}}$, the condition of Lemma 2 is satisfied. Based on (22), $\bar{\Delta}_{m,t+1} \geqslant 0$ follows. According to the conclusion of Lemma 2, we have $Q(\mathtt{R}_{m+1,t+1}, t) < Q(\mathtt{T}_{m+2,t+1}, t)$, which leads to $\bar{\Delta}_{m+1,t+1} \geqslant 0$. As $\mathtt{R}_{m+2,t+1} = \mathtt{N}_{h-1,\mathrm{R}} \in \mathbb{N}_{h-1}$ we have $Q(\mathtt{R}_{m+\Delta m,t+1}, t) \leqslant Q(\hat{\mathtt{N}}_{h-1,t}, t)$.

d If $\Delta m = 1$, $\mathtt{T}_{m+1,t+1} \neq \hat{\mathtt{N}}_{h,t}$ and $\mathtt{R}_{m+1,t+1} = \mathtt{N}_{h-1,\mathrm{R}}$ the condition of Lemma 4 is satisfied. Based on (22), we immediately have $Q(\mathtt{R}_{m+\Delta m,t+1}, t) \leqslant Q(\hat{\mathtt{N}}_{h-1,t}, t)$. According to the conclusion of Lemma 4, we have $Q(\mathtt{T}_{m+1,t+1}, t) > Q(\hat{\mathtt{N}}_{h,t}, t)$, hence $\bar{\Delta}_{m+1,t+1} \geqslant 0$ follows.

e If $\Delta m = 0$, the condition of Lemmas 5 and 1 is satisfied. According to the conclusion of Lemmas 5 and (22), $Q(\hat{\mathtt{N}}_{h-1,t}, t) > Q(\hat{\mathtt{N}}_{h,t}, t)$ follows. ■

According to Proposition 1 and Proposition 2 we may prove that (21) is achieved for $i = 1, \ldots, M_{t+1} - 2$. As the bandwidth of the gateway is much wider than the ordinary mesh node in the network, $\hat{\mathtt{N}}_1(t)$ must be the transmitter of the M_{t+1}^{th} link, which leads to $\mathtt{L}_{M_{t+1}} = \mathtt{T}_{M_{t+1}} \to \dot{\mathtt{G}}$. Based on (22) with $m = M_{t+1}$ and $h = 1$, $\bar{\Delta}_{M_{t+1}} \geqslant 0$ follows. So far, we have proved that (21) is valid for $i = 1, \ldots, M_{t+1} - 1$. Thus, Theorem 2 is proved. ■

3) Queue-Length Corollary:

***Corollary** 2:* If Theorem 2 is achieved, we will have $\mathbf{d}^T(t)\mathbf{e}(t+1) \geqslant Q(\hat{\mathtt{N}}_t, t)$.

Proof of Corollary 2: According to Theorem 2, we will have $Q(\mathtt{T}_{1,t+1},t)-Q(\mathtt{R}_{1,t+1},t)+Q(\mathtt{T}_{2,t+1},t)-Q(\mathtt{R}_{2,t+1},t)+\cdots+Q(\mathtt{T}_{M_{t+1}-1,t+1},t)-Q(\mathtt{R}_{M_{t+1}-1,t+1},t)+Q(\mathtt{T}_{M_{t+1}},t)=Q(\mathtt{T}_{1,t+1},t)+\sum_{i=1}^{M_{t+1}}\bar{\Delta}_{i,t+1}\geqslant Q(\mathtt{T}_{1,t+1},t)=Q(\hat{\mathtt{N}}_t,t)$. If we denote the set containing all the single-hop direct links that will be activated during the next time slot as $\mathbb{L}_{\mathrm{A},t+1}$ and further denote the subset of $\mathbb{L}_{\mathrm{A},t+1}$ that contains all the links included in Theorem 2 as $\hat{\mathbb{L}}_{\mathrm{A},t+1}$ and the subset containing the rest of elements as $\bar{\mathbb{L}}_{\mathrm{A},t+1}=\mathbb{L}_{\mathrm{A},t+1}-\hat{\mathbb{L}}_{\mathrm{A},t+1}$, $\mathbf{d}^T(t)\mathbf{e}(t+1)$ can be alternatively rewritten as: $\mathbf{d}^T(t)\mathbf{e}(t+1)=\sum_{\mathtt{L}_i\in\mathbb{L}_{\mathrm{A},t+1}}d(\mathtt{L}_i,t)=\sum_{\mathtt{L}_i\in\hat{\mathbb{L}}_{\mathrm{A},t+1}}d(\mathtt{L}_i,t)+\sum_{\mathtt{L}_i\in\bar{\mathbb{L}}_{\mathrm{A},t+1}}d(\mathtt{L}_i,t)$. According to the scheduling algorithm given in Section III, for each activated link, the queue length of the transmitter is always bigger or equal to that of the receiver. In this case we will have $\sum_{\mathtt{L}_i\in\bar{\mathbb{L}}_{\mathrm{A},t+1}}d(\mathtt{L}_i,t)\geqslant 0$. Hence, $\mathbf{d}^T(t)\mathbf{e}(t+1)=\sum_{\mathtt{L}_i\in\hat{\mathbb{L}}_{\mathrm{A},t+1}}d(\mathtt{L}_i,t)+\sum_{\mathtt{L}_i\in\bar{\mathbb{L}}_{\mathrm{A},t+1}}d(\mathtt{L}_i,t)\geqslant Q(\hat{\mathtt{N}}_t,t)+0=Q(\hat{\mathtt{N}}_t,t)$ can be proved. ■

B. Proof of the Network Stability Under Stability Region

Proof of Theorem 1: Let's assume that the arrival rate vector $\dot{\mathbf{a}}$ at any moment is Poisson distributed with a mean and variance of $\rho_{\mathfrak{a}}$. Following the statement given in Section III-D, given a positive number ϵ, as long as $\mathbf{a}(t+1)$ belongs to the stability region $\mathbb{C}$ and we can find a positive number b_{q} as a multi-variable function of ϵ and $\rho_{\mathfrak{a}}$, such that

$$E\Big[\mathbf{q}^T(t+1)\mathbf{q}(t+1)-\mathbf{q}^T(t)\mathbf{q}(t)\Big|\mathbf{q}(t)=\dot{\mathbf{q}}\Big]<\infty \qquad \text{if } \dot{\mathbf{q}}^T\dot{\mathbf{q}}\leqslant b_{\mathrm{q}} \tag{23}$$

$$E\Big[\mathbf{q}^T(t+1)\mathbf{q}(t+1)-\mathbf{q}^T(t)\mathbf{q}(t)\Big|\mathbf{q}(t)=\dot{\mathbf{q}}\Big]\leqslant -\epsilon \qquad \text{if } \dot{\mathbf{q}}^T\dot{\mathbf{q}}> b_{\mathrm{q}} \tag{24}$$

are both able to be achieved, then the mesh network is considered to be stable.

As can be observed from (23) and (24), under both situations, we need to calculate the value of the common component $E\big\{\mathbf{q}^T(t+1)\mathbf{q}(t+1)-\mathbf{q}^T(t)\mathbf{q}(t)\big\}$ shared by both inequations. Given the current queue length vector $\mathbf{q}(t)$, the queue length vector of the next time slot can be quantified as $\mathbf{q}(t+1)=\mathbf{q}(t)+\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)$. Hence, the common part of both equations can be further expressed as:

$$\begin{aligned}&E\Big\{\mathbf{q}^T(t+1)\mathbf{q}(t+1)-\mathbf{q}^T(t)\mathbf{q}(t)\Big\}=E\Big\{\big[\mathbf{q}(t+1)+\mathbf{q}(t)\big]^T\big[\mathbf{q}(t+1)-\mathbf{q}(t)\big]\Big\}\\
=&E\Big\{\big[2\mathbf{q}(t)+\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)\big]^T\big[\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)\big]\Big\}\\
=&2E\Big\{\mathbf{q}^T(t)\big[\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)\big]\Big\}+E\Big\{\big[\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)\big]^T\big[\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)\big]\Big\},\end{aligned} \tag{25}$$

where $\mathbf{a}(t+1)$ is an instantaneous sample of the N-dimensional vector-valued random variable $\mathfrak{a}_N$. Each element of $\mathfrak{a}_N$ is an independent and identically distributed (i.i.d.) random variable $\mathfrak{a}$, which follows a Poisson distribution. So if its mean is $E(\mathfrak{a})=\rho_{\mathfrak{a}}$, the variance is also going to be $\mathrm{var}(\mathfrak{a})=E\big\{[\mathfrak{a}-E(\mathfrak{a})]^2\big\}=\rho_{\mathfrak{a}}$, which leads to $E(\mathfrak{a}^2)=\rho_{\mathfrak{a}}(\rho_{\mathfrak{a}}+1)$.

Thus, the second item of (25) can be further derived as: $E\Big\{\big[\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)\big]^T\big[\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)\big]\Big\}=\sum_{n=1}^{N}E\Big\{\big[\mathbf{r}_n(\mathbf{R})\mathbf{e}(t+1)\big]^2\Big\}+2\sum_{n=1}^{N}E\Big\{\big[\mathbf{r}_n(\mathbf{R})\mathbf{e}(t+1)\big]a_n(t+1)\Big\}+\sum_{n=1}^{N}E\big\{a_n^2(t+1)\big\}$, where $\mathbf{r}_n(\mathbf{R})$ is the nth row vector of the $(N\times L)$-element routing matrix $\mathbf{R}$ and $a_n(t+1)$ is the nth element of the $(N\times 1)$-element arriving rate vector $\mathbf{a}(t+1)$. Obviously, the scalar $\big[\mathbf{r}_n(\mathbf{R})\mathbf{e}(t+1)\big]$ is upper bounded by the number of single-hop direct links connected to $\mathtt{N}_n$. More precisely, even if all the L single-hop links existing in the mesh network are connected to $\mathtt{N}_n$, and all of them are activated, $\big[\mathbf{r}_n(\mathbf{R})\mathbf{e}(t+1)\big]=L$. Hence, we will have $\big[\mathbf{r}_n(\mathbf{R})\mathbf{e}(t+1)\big]\leqslant L,\quad\forall n=1,\ldots,N$. Subsequently, we will have $\sum_{n=1}^{N}\big[\mathbf{r}_n(\mathbf{R})\mathbf{e}(t+1)\big]^2\leqslant NL^2$. Therefore, the second term of (25) can be further improved as:

$$\begin{aligned}&E\Big\{\big[\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)\big]^T\big[\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)\big]\Big\}\\ \leqslant& NL^2+2L\sum_{n=1}^{N}E\Big[a_n(t+1)\Big]+\sum_{n=1}^{N}E\Big[a_n^2(t+1)\Big]\leqslant NL^2+2LN\rho_{\mathfrak{a}}+N\rho_{\mathfrak{a}}(\rho_{\mathfrak{a}}+1)\end{aligned}\tag{26}$$

For the first term of the right hand side (RHS) of (25), we have the following:

$$2E\Big\{\mathbf{q}^T(t)\big[\mathbf{R}\mathbf{e}(t+1)+\mathbf{a}(t+1)\big]\Big\}=2E\big\{\mathbf{q}^T(t)\mathbf{R}\mathbf{e}(t+1)\big\}+2E\big\{\mathbf{q}^T(t)\big\}E\big\{\mathbf{a}(t+1)\big\},\tag{27}$$

where $\mathbf{q}^T(t)\mathbf{R}$ is a $(1\times L)$-element vector, and its lth element corresponds to $\mathbf{q}^T(t)\mathbf{c}_l(\mathbf{R})=-Q\big[\mathtt{T}(\mathtt{L}_l)\big](t)+Q\big[\mathtt{R}(\mathtt{L}_l)\big](t)=-d(\mathtt{L}_l,t)$, where $\mathbf{c}_l(\mathbf{R})$ stands for the lth column of the $(N\times L)$-element routing matrix $\mathbf{R}$. Therefore, we have

$$\mathbf{q}^T(t)\mathbf{R}=-\mathbf{d}^T(t).\tag{28}$$

Below, we will prove (23) and (24) in Sections B1 and B2 respectively.

1) when $\dot{\mathbf{q}}^T\dot{\mathbf{q}}\leqslant b_q$: we will prove (23). Firstly, we define the superposition of all the elements in an instantaneous vector $\dot{\mathbf{q}}$ as $\Sigma_{\dot{\mathbf{q}}}$ and the expectation of the superposition of all elements in $\mathbf{q}(t)$ as $E\{\Sigma_{\mathbf{q}(t)}\}$ respectively. We further assume that the number of nodes N is bigger than one, which is always the case in a network. When $\Sigma_{\dot{\mathbf{q}}}\geqslant N$, we will always have $\Sigma_{\dot{\mathbf{q}}}/\sqrt{N}\leqslant\sqrt{\dot{\mathbf{q}}^T\dot{\mathbf{q}}}$. Hence, we have $E\{\Sigma_{\mathbf{q}(t)}\}/\sqrt{N}\leqslant\sqrt{E\{\mathbf{q}^T(t)\mathbf{q}(t)\}}\leqslant\sqrt{b_{\mathrm{q}}}$, which leads to $E\{\Sigma_{\mathbf{q}(t)}\}\leqslant\sqrt{Nb_{\mathrm{q}}}$. When $\Sigma_{\dot{\mathbf{q}}}\leqslant N$ and $b_{\mathrm{q}}\geqslant N$, we have $E\{\Sigma_{\mathbf{q}(t)}\}\leqslant N\leqslant\sqrt{Nb_{\mathrm{q}}}$. When $\Sigma_{\mathbf{q}}\leqslant N$ and $b_{\mathrm{q}}\leqslant N$, we have $E\{\Sigma_{\mathbf{q}(t)}\}\leqslant N$. We can always find a maximum limit $\hat{\Sigma}_{\dot{\mathbf{q}}}$, which is either $\sqrt{Nb_{\mathrm{q}}}$ or N, for the value of $E\{\Sigma_{\mathbf{q}(t)}\}$. Therefore, the first and second term of (27) can be further derived as: $2E\big[\mathbf{q}^T(t)\mathbf{R}\mathbf{e}(t+1)\big]\leqslant 2L\hat{\Sigma}_{\dot{\mathbf{q}}}$ and $2E\big[\mathbf{q}^T(t)\big]E\big[\mathbf{a}(t+1)\big]\leqslant 2\rho_{\mathfrak{a}}\hat{\Sigma}_{\dot{\mathbf{q}}}$. Based on the above discussions, we will have $E\big[\mathbf{q}^T(t+1)\mathbf{q}(t+1)-\mathbf{q}^T(t)\mathbf{q}(t)\big]\leqslant 2L\hat{\Sigma}_{\dot{\mathbf{q}}}+2\rho_{\mathfrak{a}}\hat{\Sigma}_{\dot{\mathbf{q}}}+NL^2+2LN\rho_{\mathfrak{a}}+N\rho_{\mathfrak{a}}(\rho_{\mathfrak{a}}+1)<\infty$. Hence (23) is proved.

2) when $\dot{\mathbf{q}}^T\dot{\mathbf{q}} > b_q$*,:* we will prove (24). When $\dot{\mathbf{q}}^T\dot{\mathbf{q}} > b_{\mathrm{q}}$, we will have $\sum_{n=1}^{N}\dot{q}_n^2 > b_{\mathrm{q}}$. Therefore, the maximum element $\max_{n=1}^{N}\dot{q}_n$ of vector $\dot{\mathbf{q}}$ satisfies the following relationship $\max_{n=1}^{N}\dot{q}_n > \sqrt{b_{\mathrm{q}}}/\sqrt{N}$. According to Theorem 2, the following relationship can be further obtained:

$$\mathbf{d}^T(t)\mathbf{e}(t+1) \geqslant Q(\hat{\mathtt{N}}_t, t) = \max_{n=1}^{N}\dot{q}_n > \sqrt{\frac{b_{\mathrm{q}}}{N}}. \tag{29}$$

As long as $\mathbf{a}(t+1)$ is under the stability region, it can be represented by relationship (3). Additionally, in (3) the flow vector $\mathbf{f} \in co(\mathbb{S})$. According to the definition of the convex hull, if $\mathbf{f} \in co(\mathbb{S})$, $\mathbf{f}$ should satisfy the following relationship:

$$\mathbf{f} \leqslant \sum_{i=1}^{|\mathbb{S}|}\theta_i\mathbf{s}_i, \tag{30}$$

subject to $\sum_{i=1}^{|\mathbb{S}|}\theta_i = 1$. Hence, we can find a $\delta > 1$, such that

$$\mathbf{f} = \frac{1}{\delta}\sum_{i=1}^{|\mathbb{S}|}\theta_i\mathbf{s}_i = \sum_{i=1}^{|\mathbb{S}|}\lambda_i\mathbf{s}_i, \tag{31}$$

where $\lambda_i = \theta_i/\delta$ and $\sum_{i=1}^{|\mathbb{S}|}\lambda_i \leqslant 1$. By replacing the arrival rate vector $\mathbf{a}(t+1)$ with (3) as well as (31), and by employing (28), the second term at the RHS of (27) can be expanded as:

$$E\Big\{\mathbf{q}^T(t)\Big\}E\Big\{\mathbf{a}(t+1)\Big\} = -\frac{1}{\alpha}E\Big\{\mathbf{q}^T(t)\Big\}\mathbf{R}\bar{\mathbf{D}}\mathbf{f} = \frac{1}{\alpha}E\Big\{\mathbf{d}^T(t)\Big\}\bar{\mathbf{D}}\sum_{i=1}^{|\mathbb{S}|}\lambda_i\mathbf{s}_i \tag{32}$$

As the activation vector $\mathbf{e}(t+1) = \arg\max_{\mathbf{s}_i\in\mathbb{S}}\mathbf{d}^T(t)\bar{\mathbf{D}}\mathbf{s}_i$, ergo we have $\mathbf{d}^T(t)\bar{\mathbf{D}}\mathbf{s}_i \leqslant \mathbf{d}^T(t)\bar{\mathbf{D}}\mathbf{e}(t+1)$ achieved for all $\mathbf{s}_i \in \mathbb{S}$. Therefore, (27) can be further developed as:

$$\begin{aligned}&2E\Big\{\mathbf{q}^T(t)\big[\mathbf{R}\mathbf{e}(t+1) + \mathbf{a}(t+1)\big]\Big\}\\ \leqslant &-2(1-\frac{1}{\alpha}\sum_{i=1}^{|\mathbb{S}|}\lambda_i)E\Big[\mathbf{d}^T(t)\mathbf{e}(t+1)\Big] \leqslant -2(1-\frac{1}{\alpha}\sum_{i=1}^{|\mathbb{S}|}\lambda_i)\sqrt{\frac{b_{\mathrm{q}}}{N}}.\end{aligned} \tag{33}$$

Finally, by substituting the first and the second item of (25) with (33) and (26), we will have

$$E\Big\{\mathbf{q}^T(t+1)\mathbf{q}(t+1) - \mathbf{q}^T(t)\mathbf{q}(t)\Big\} \leqslant -2(1-\frac{1}{\alpha}\sum_{i=1}^{|\mathbb{S}|}\lambda_i)\sqrt{\frac{b_{\mathrm{q}}}{N}} + NL^2 + 2LN\rho_{\mathfrak{a}} + N\rho_{\mathfrak{a}}(\rho_{\mathfrak{a}}+1). \tag{34}$$

According to (24), when $\mathbf{q}^T(t)\mathbf{q}(t) > b_{\mathrm{q}}$, the condition for the network to be stable is the existence of a positive number ϵ so that $E\big[\mathbf{q}^T(t+1)\mathbf{q}(t+1) - \mathbf{q}^T(t)\mathbf{q}(t)\big] \leqslant -\epsilon$ is always achieved. According to (34), if we want $E\big[\mathbf{q}^T(t+1)\mathbf{q}(t+1) - \mathbf{q}^T(t)\mathbf{q}(t)\big] \leqslant -\epsilon$ to be achieved all the time, we should find a positive value for the queue-length boundary b_{q} to validate

$$-2(1-\frac{1}{\alpha}\sum_{i=1}^{|\mathbb{S}|}\lambda_i)\sqrt{\frac{b_{\mathrm{q}}}{N}} + NL^2 + 2LN\rho_{\mathfrak{a}} + N\rho_{\mathfrak{a}}(\rho_{\mathfrak{a}}+1) = -\epsilon. \tag{35}$$

As we have $1-\frac{1}{\alpha}\sum_{i=1}^{|\mathbb{S}|}\lambda_i > 0$ achieved for all $\alpha > 1$, a positive value of b_{q} can be easily obtained from (35) as

$$b_{\mathrm{q}} = N\left(\frac{NL^2 + 2LN\rho_{\mathfrak{a}} + N\rho_{\mathfrak{a}}(\rho_{\mathfrak{a}}+1)+\epsilon}{2(1-\frac{1}{\alpha}\sum_{i=1}^{|\mathbb{S}|}\lambda_i)}\right)^2. \tag{36}$$

Given a positive number ϵ, a corresponding b_{q} can be evaluated; such that as long as the arrival rate vector $\mathbf{a}(t+1)$ is under the stability region even when the norm of the queue length of all the mesh network is bigger than the boundary, i.e. $\mathbf{q}^T(t)\mathbf{q}(t) > b_{\mathrm{q}}$, the network is still going to be stable since $E\big[\mathbf{q}^T(t+1)\mathbf{q}(t+1) - \mathbf{q}^T(t)\mathbf{q}(t)\big] < -\epsilon$ is always achieved. ■

REFERENCES

[1] Public Law 110–140 – Energy Independence and Security Act of 2007. Available on-line: http://www.gpo.gov/fdsys/pkg/PLAW-110publ140/content-detail.html

[2] S. Galli, A. Scaglione and Z. Wang, "For the grid and through the grid: the role of power line communications in the smart grid," *Proceedings of the IEEE*, vol. 99, no. 6, pp. 998–1027, June, 2011.

[3] IEEE 802.11 Standard Working Group, *Standard for Information Technology Telecommunications and Information Exchange Between Systems LAN/MAN Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, ANSI/ IEEE Std 802.11*, first Ed., 1999.

[4] H. Gharavi and B. Hu, "Multigate communication network for smart grid," *Proceedings of the IEEE*, vol. 99, no. 6, pp. 1028–1045, May 2011.

[5] C. Partridge, T. Mendez and W. Milliken, "RFC 1546 - Host Any casting Service," http://www.ietf.org/rfc/rfc1546.txt?number=1546, 1993.

[6] J. Abley, A. Canada and K. Lindqvist, "RFC 4786 - Operation of Anycast Services," http://www.ietf.org/rfc/rfc4786.txt?number=4786, 2006.

[7] R. Baumann and S. Heimlicher, "Routing in large-scale wireless mesh networks using temperature fields," *IEEE Network*, vol. 22, no. 1, pp. 25–31, January-February, 2008.

[8] W. Xiaonan and H. Kejing, "An anycast model of implementing communication between ad-hoc networks and IPv6 networks," *Proceedings of the IEEE International Conference on Information and Automation*, Zhuhai/Macau, China, June 22-25, 2009, pp. 324–329.

[9] Y. Amir, C. Danilov, R. Musdloiu-Elefter and N. Rivera, "An inter-domain routing protocol for multi-homed wireless mesh networks," *Proceedings of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM07)*, Helsinki, Finland, June 2007, pp. 1–10.

[10] S. C. Liew, C. Kai, J. Leung, and B. Wong, "Back-of-the-envelope computation of throughput distributions in CSMA wireless networks," *IEEE Transaction on Mobile Computing*, vol. 9, no. 9, pp. 1319–1331, Sep. 2010.

[11] J. Shin and D. Shah, *Randomized Scheduling Algorithm for Queueing Networks*, Apr. 2010, Online Available: http://arxiv.org/pdf/0908.3670.

[12] L. B. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," *IEEE/ACM Transaction on Network*, vol. 18, no. 3, pp. 960–997, Jun. 2010.

[13] L. Jiang and J. Walrand, “A distributed CSMA algorithm for throughput and utility maximization in wireless networks,” *UC Berkeley, Tech. Rep.*, 2009, Online Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-124.html.

[14] C.-K. Chau, M. Chen, and S. C. Liew, “Capacity of large-scale CSMA wireless networks,” *IEEE/ACM Transaction on Network*, vol. 19, no. 3, pp. 893–906, June 2011.

[15] J. Ni, B. Tan, and R. Srikant, “Q-CSMA: Queue-length based CSMA/CA algorithms for achieving maximum throughput and low delay in wireless networks,” in *Proceedings of IEEE INFOCOM*, March 2010.

[16] S. T. Maguluri, B. Hajek, and R. Srikant, “The Stability of Longest-Queue-First Scheduling With Variable Packet Sizes,” in *Proceedings of IEEE CDC 2011*, April 2011, available: http://hdl.handle.net/2142/26514.

[17] L. Tassiulas and A. Ephremides, “Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks,” *IEEE Transaction on Automatic Control*, vol. 37, no. 12 pp. 1936–1948, December, 1992.

[18] N. McKeown, A. Mekkittikul, V. Anantharam and J. Walrand, “Achieving 100% throughput in an input-queued switch,” *IEEE Transaction on Communications*, vol. 47, no. 8, pp. 1260–1267, Aug. 1999.

[19] M. J. Neely, E. Modiano, and C. E. Rohrs, “Dynamic power allocation and routing for time varying wireless networks,” in *Proceedings of the IEEE INFOCOM*, San Francisco, Apr. 2003.

[20] X. Lin and N. B. Shroff, “Joint rate control and scheduling in multihop wireless networks,” in *Proceedings of the IEEE Conference on Decision and Control*, Paradise Island, Bahamas, Dec. 2004.

[21] X. Lin and N. B. Shroff, “The impact of imperfect scheduling on cross-layer rate control in multihop wireless networks,” in *Proceedings of the IEEE INFOCOM 2005*, Miami, USA, Mar, 2005.

[22] L. Chen, S. H. Low, M. Chiang and J. C. Doyle, “Cross-layer congestion control, routing and scheduling design in ad-hoc wireless networks,” in *Proceedings of the IEEE INFOCOM 2006*, Barcelona, Spain, Apr. 2006.

[23] E. Yeh and R. Berry, “Throughput optimal control of wireless networks with two-hop cooperative relaying,” in *Proceedings of the IEEE ISIT 2007*, Nice, France, Jun. 2007.

[24] A. T. Hoang, W.-L. Yeow, P.-Y. Kong, C.-K. Tham,“Throughput performance of back-pressure scheduling in wireless cooperative networks,” in *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Tokyo, Japan, Sep, 2009.

[25] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, P. Whiting and R. Vijayakumar, “Providing quality of service over a shared wireless link,” *IEEE Communication Magazine*, vol. 39, pp. 150–153, Feb. 2001.

[26] S. Shakkottai and A. Stolyar, “Scheduling for multiple owners sharing a time-varying channel: The exponential rule,” *Translations of the AMS*, 2001, a volume in memory of F. Karpelevich.

[27] S. Shakkottai and A. Stolyar, “Scheduling of a shared a time-varying channel: The exponential rule stability,” in *Proceedings of the INFORMS Applied Probability Conference*, New York, Jul. 2001.

[28] R. Buche and H. J. Kushner, “Control of mobile communication systems with time-varying channels via stability methods,” *IEEE Transactions on Automatic Control*, vol. 49, no. 11, pp. 1954–1962, Nov. 2004.

[29] L, Ying, S. Shakkottai and A. Reddy, “On combining shortest and back-pressure routing over multihop wireless networks,” in *Proceedings of the IEEE INFOCOM*, Rio de Janeiro, Brazil, April 2009, pp. 1674–1682.

[30] E. Modiano, D. Shah and G. Zussman, “Maximizing throughput in wireless network via gossipping,” in *Proceedings of the ACM Sigmetrics*, Saint-Malo, France, 2006, pp. 27–38.

[31] S. Sanghavi, L. Bui and R. Srikant, “Distributed link scheduling with constant overhead,” in *Proceedings of the ACM Sigmetrics*, San Diego, CA, 2007, pp. 313–324.

[32] A. Stolyar, “Dynamic distributed scheduling in random access network,” *Journal on Applied Probability*, vol. 45, no. 2, pp. 297–313, Jun. 2008.

[33] P. Gupta, A. Stolyar, “Optimal throughput allocation in general random-access networks,” in *Proceedings of the 38th IEEE Conference on Information Sciences and Systems*, pp. 1254–1259, Princeton, NJ, Mar. 2006.

[34] C. Joo and N. B. Shroff, “Performance of random access scheduling schemes in multihop wireless networks,” in *Proceedings of the 40th Asilomar Conf. Signals, Systems and Computers*, Pacic Grove, CA, Oct. 2006.

[35] X. Lin and S. Rasool, “Constant-time distributed scheduling policies for ad hoc wireless networks,” in *Proceedings of the IEEE Conference of Decision and Control*, San Diego, CA, 2006, pp. 1258–1263.

[36] J. Liu, A. L. Stolyar, M. Chiang and H. V. Poor, “Queue back-pressure random access in multihop wireless networks: optimality and stability,” *IEEE Transactions on Information Theory*, vol. 55, no. 9, pp. 4087–4098, Sep. 2009.

[37] S. Asmussen, *Applied Probability and Queues*, New York: Wiley, 1987.

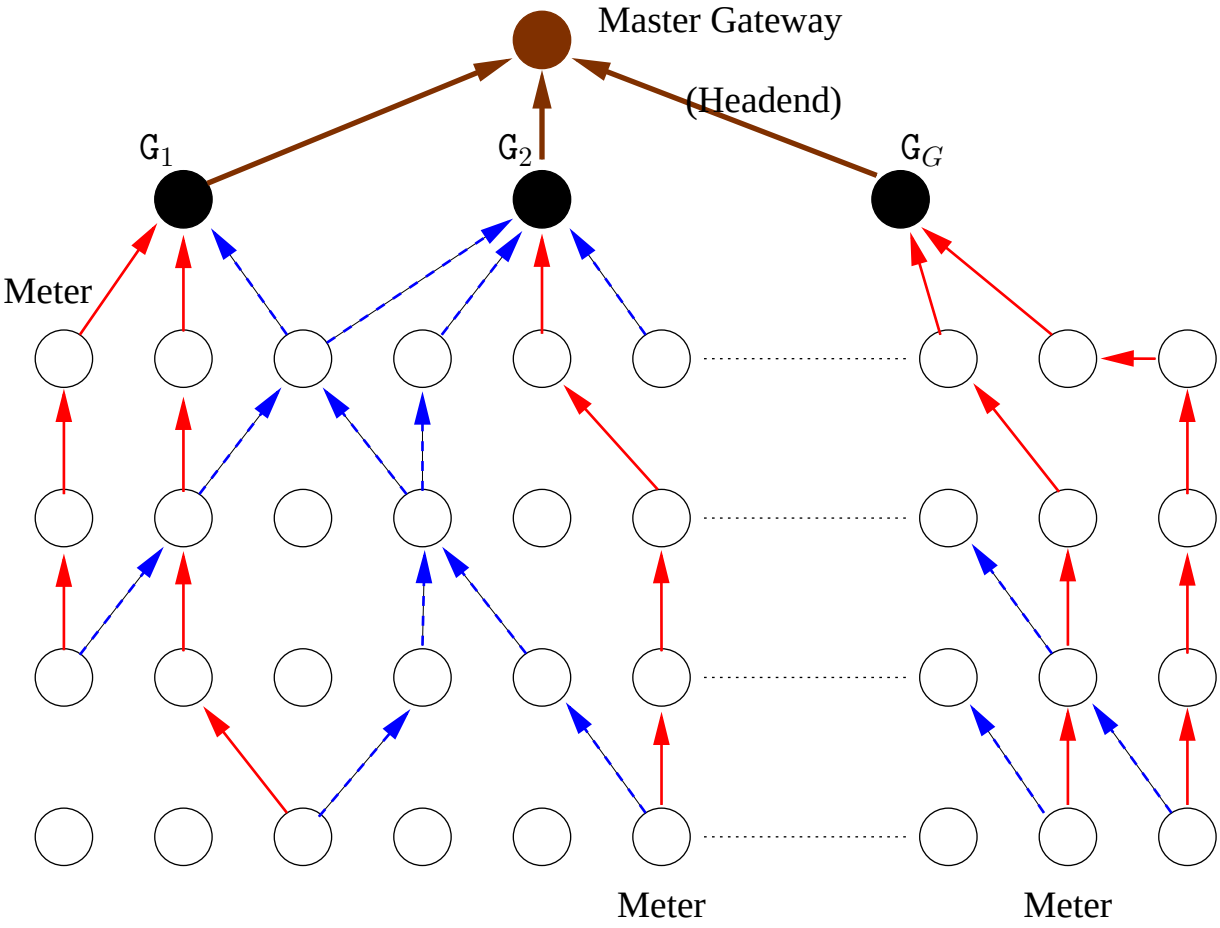

Fig. 1: An example of a multiple-path network structure consisting of G gateways ($\mathtt{G}_1$, $\mathtt{G}_2$, ..., $\mathtt{G}_G$).

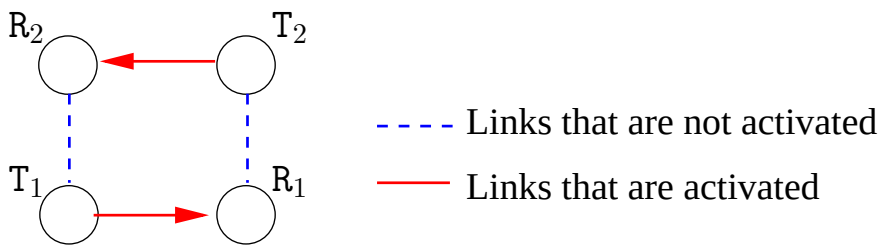

Fig. 2: Case I of the scenarios satisfying the assumption of Lemma 2.

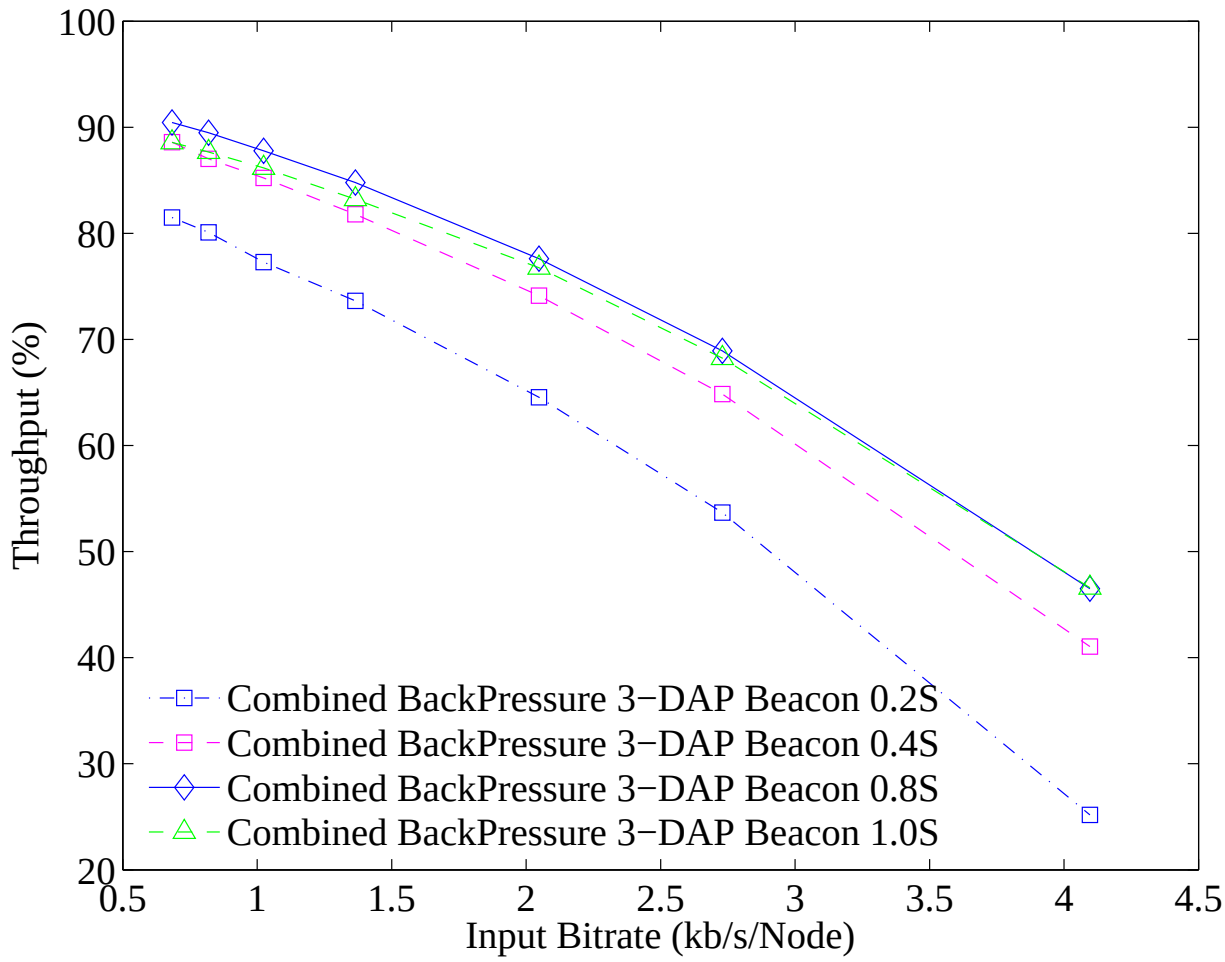

Fig. 3: Performance evaluation of the three-gateway network operating with back-pressure algorithm using a set of beacon interval values.

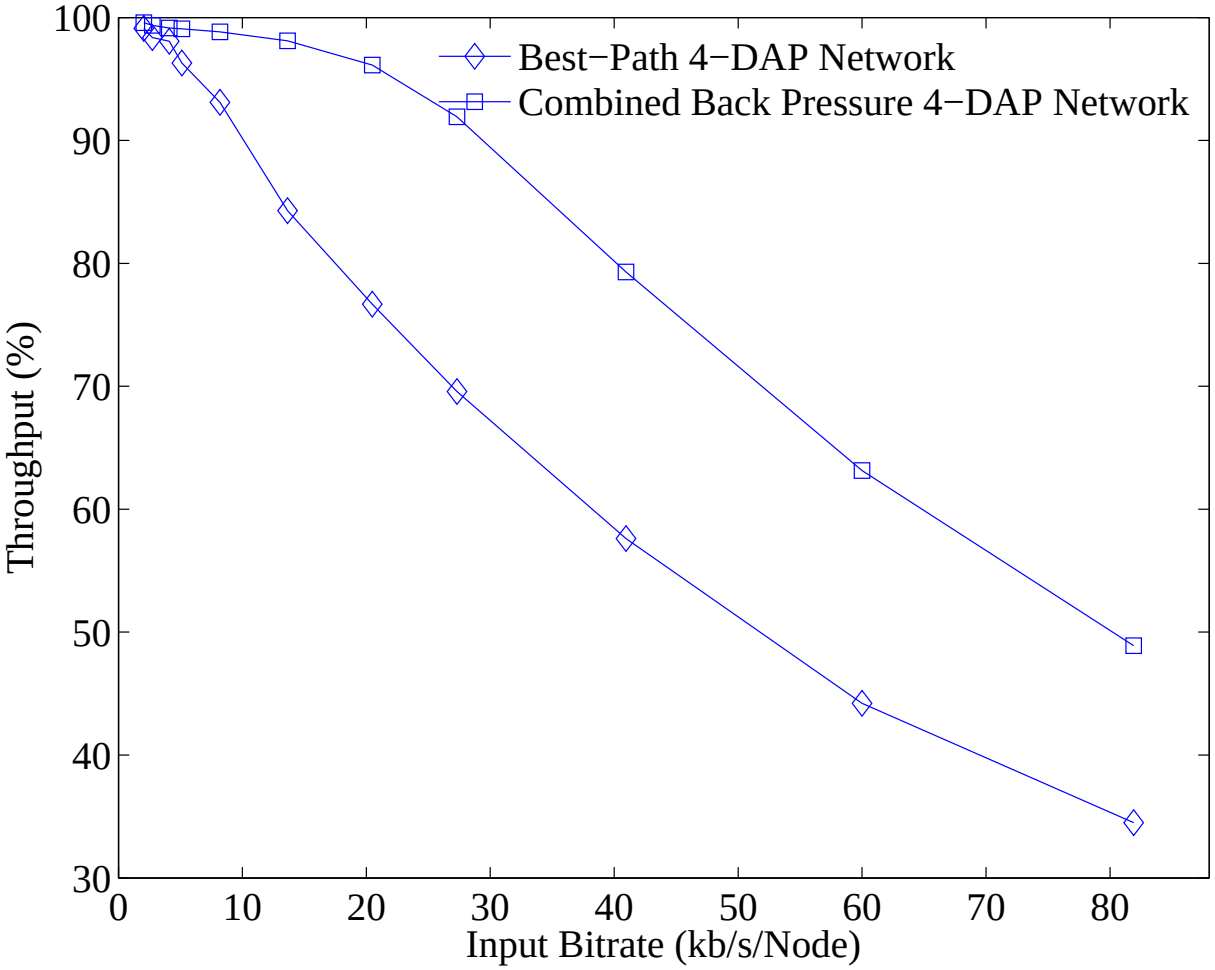

Fig. 4: Performance evaluation of the four-gateway network operating both with and without using back-pressure algorithm.

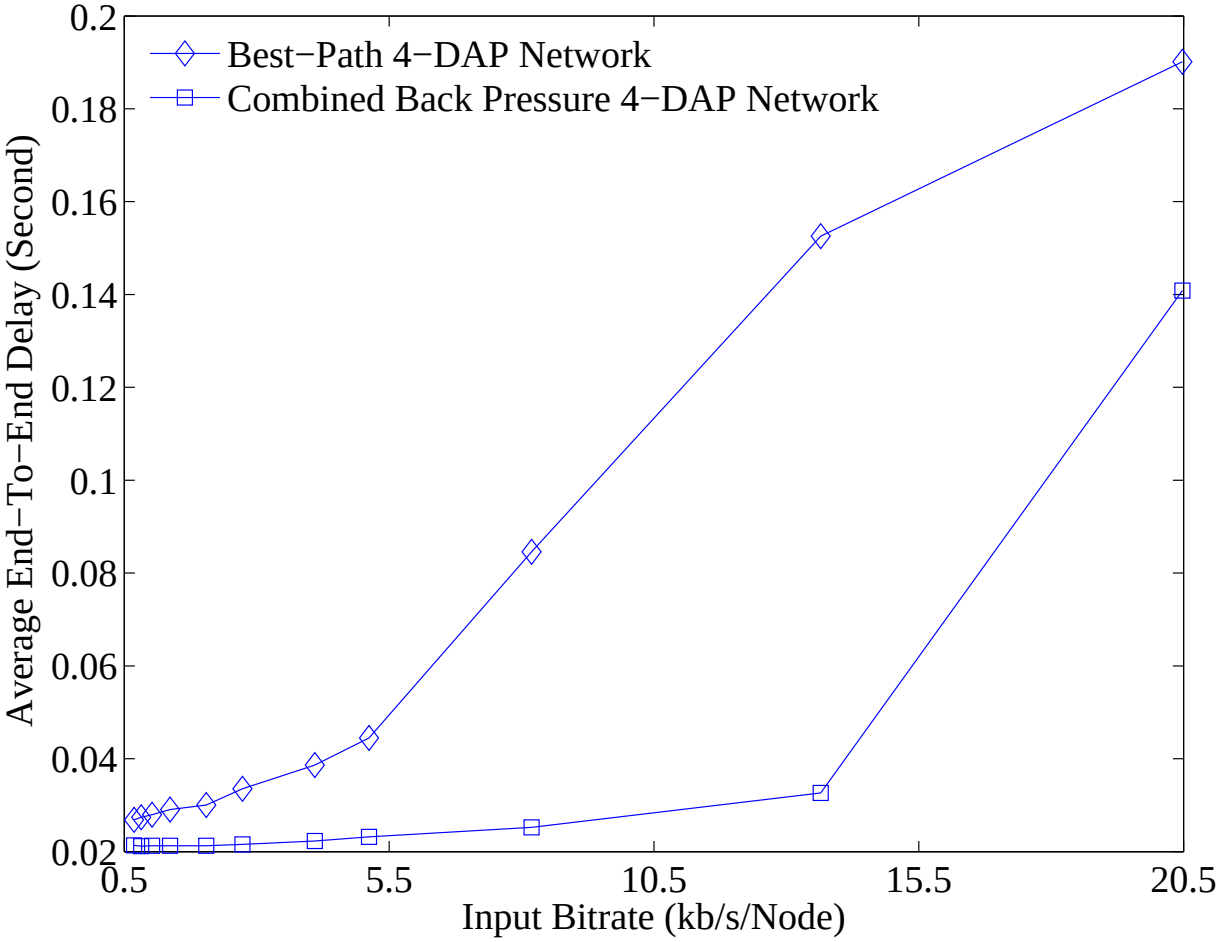

Fig. 5: Average end-to-end delay of the four-gateway network operating both with and without using back-pressure algorithm.