Clique-Based Utility Maximization in Wireless Mesh Networks

−Algorithm, Simulation, and Mathematical Analysis

Erwu Liu
Dept. of Electrical and Electronic Engineering
Imperial College
London, United Kingdom
Email: erwu.liu@imperial.ac.uk

Qinqing Zhang
Applied Physics Laboratory
Johns Hopkins University
Laurel, Maryland, USA
Email: qinqing.zhang@jhuapl.edu

Kin K. Leung
Dept. of Electrical and Electronic Engineering
Imperial College
London, United Kingdom
Email: kin.leung@imperial.ac.uk

***Abstract*—Compared with cellular networks, wireless mesh networks (WMNs) need more careful design for resource allocation. To this end, we develop a clique-based proportional fair scheduling (CBPFS) algorithm for WMNs. Furthermore, we obtain a closed-form model to quantify the throughput of connection links and traffic flows in multi-hop transmissions. We use the derived analytical framework to estimate the throughput of CBPFS and compared it with simulations. It is the first time a mathematical model is developed to quantify the throughput of links and end-to-end flows in a multi-hop network where links are proportionally fair scheduled.**

Keywords- wireless mesh network; multi-hop transmission; clique-based proportional fair scheduling

I. INTRODUCTION

We consider network resource optimization in a wireless mesh network (*WMN*). A WMN consists of radio nodes organized in a mesh topology. Different from the cellular network, nodes in a WMN can communicate with each other directly or through intermediate nodes. A WMN is a special type of wireless ad-hoc network where the topology of the network is relatively static and nodes are with limited mobility [1]. We address the throughput and fairness issues in such wireless networks [2].

Resource allocation mechanism relies on a suitable performance metric. There are two types of performance metrics used in resource allocation: throughput-based performance metric and utility-based performance metric. The former is often seen in cellular networks whose objective is to maximize the sum (or weighted sum) of the throughput of all links, while the latter is typically used when the objective is to exhibit some sense of fairness criteria. Throughput-based performance metric is known to cause severe unfairness among the nodes in wireless networks, and most recent work on resource allocation has shifted to a utility-based framework. In our context, utility-based approach is to maximize the aggregate utility in the network, and we propose to use utility-based performance metric for resource allocation in this paper.

In an utility-based framework, each link l is associated a utility function $U_l(R_l)$, over the link throughput R_l. It is known that by defining $U(.)$ appropriately, different fairness criteria of interest, such as proportional or max-min fairness, can be achieved [3][4]. Radunovic and Boudec [5] prove that the max-min allocation has fundamental efficiency problem in the limit of long battery lifetime, regardless of the MAC layer, network topology, choice of routes and power constraints. This results in all links receiving the rate of the worst link, and leads to severe inefficiency. For efficiency, we consider proportional fairness in this paper. We then extend the proportional fair scheduling (*PFS*) algorithm so that the proposed framework supports higher throughput. To be specific, non-contending links are grouped into link cliques which are to be scheduled in a proportionally fair manner (refer to Section II for details).

Even in cellular networks, there are few analytical results related to the PFS algorithm. To simplify the problem, most analyses assume some kind of *i.i.d* relationship among the links [6-9]. Moreover, *linear rate model* and *logarithm rate model* are the two rate models commonly used for analyzing the performance of PFS [6-9]. The assumption of linear or logarithm rate model is a reasonable modeling convention. However, when examining throughput performance, it does not seem entirely satisfactory to assume such simplified models [10]. Work by Telatar [11] and Smith et al. [12][13] suggest that the link capacity in Rayleigh fading networks be better modeled by a Gaussian distribution.

With this Gaussian rate model, Liu [10][14] conducts mathematical analysis and provides closed-form expressions for the PFS throughput without the limitations mentioned above. The theoretical results in [10][14] fit with the simulation ones with surprisingly high accuracy.

Our ultimate objective is to develop a theoretical framework to facilitate the research on fair resource allocation for WMNs with multiple contending links and multi-hop transmissions. Towards this end, we propose a systematic method and then derive a mathematical model to estimate the throughput achieved for a multi-hop path in WMNs where links are scheduled under the proportional fair criteria. In particular, our contributions are summarized as follows.

1) In this paper, we introduce clique-based proportional fair scheduling (*CBPFS*), which is an extension of PFS. Since

This work was supported, in part, by Johns Hopkins University, Applied Physics Laboratory's internal research and development funds.

its first presence [3], the PFS algorithm has aroused considerable interest (see [14] and the references therein). Until now, virtually all analytical results on PFS are for cellular networks where links are non-contending by Time-Division-Multiple-Access (*TDMA*) technique. This paper provides a theoretical framework for the research of PFS in multi-hop or mesh networks where there are multiple contending links.

2) Our analysis is general and covers a broad set of scenarios. CBPFS becomes PFS when used in cellular networks. The most valuable contribution of this paper is a mathematical framework that provided quick estimates of the throughput for both links and multi-hop paths under the proportional fairness criteria.

The rest of the paper is structured as follows. In Section II, we first introduce concepts of clique-based scheduling in WMNs; we then present CBPFS algorithm and derive the mathematical framework for evaluating the throughput of a given link or flow in a WMN. Closed-form models of the throughput for both links and flows are also developed to facilitate the mathematical analysis. Due to space limitation, we do not provide detailed proofs in this conference version. In Section III, we compare analytical results with simulation experiments to validate the theoretical findings in Rayleigh fading scenarios. We conclude the paper in Section IV.

II. NETWORK MODEL

A. System Model and Notations

Consider a WMN that is represented by a connected graph, $\boldsymbol{G}=(\boldsymbol{N},\boldsymbol{L})$, which has a node set $\boldsymbol{N}$ (with cardinality $|\boldsymbol{N}|$), a link set $\boldsymbol{L}$ (with cardinality $|\boldsymbol{L}|$), and $|\boldsymbol{F}|$ source-destination node pairs $\{s_1, d_1\},\ldots, \{s_{|F|}, d_{|F|}\}$. We refer to a source-destination pair $\{s_f, d_f\}$ as a flow, denoted by f, and denote the set of flows by $\boldsymbol{F}$. Let ζ_f be the end-to-end path which is the set of wireless links crossed by flow $f\in\boldsymbol{F}$ and $\{x_l[t]\}$ the arrival process for link $l\in\boldsymbol{L}$, i.e., $x_l[t]$ is the number of packets that transmitted over link l in slot t. Let λ_f denote the average throughput of flow f. For any link $l\in\boldsymbol{L}$, let C_l and $E[C_l]$ denote the capacity and average capacity, R_l and $E[R_l]$ denote the throughput and average throughput of link l, respectively. At each node, a packet queue is maintained for each destination. We call a queue *stable* if its length does not increase infinitely. Obviously, the network is *stable* if all queues are stable; and *unstable* otherwise. For the queues to be stable, it requires $E[R_l] \geq \lambda_f$ for any link $l\in\zeta_f$. Let Λ denote the *stability region* of the network. To find Λ, subject to resource constraints such as capacity constraints or power constraints, one needs to choose a suitable performance metric for resource allocation.

We assume that each link experiences independent (but not necessarily identical) Rayleigh fading in WMNs. Similar to our previous research on PFS, for the analysis on CBPFS, we use Gaussian rate model for link capacity.

B. Link Contention

In WMNs, due to undesired interference and resource contention, not all links in a wireless network can transmit simultaneously.

For our discussion, we assume that each node in WMNs is equipped with one single antenna and operates in half-duplex transmit/receive mode. We consider a general link contention model formulation specified by a set of pairs of links that contend with each other: we say that two links *contend* if their concurrent transmissions need to access the same radio resources. There are five kinds of link contention: multiple links transmitting to the same node, multiple links receiving from the same node, the transmitting link and receiving link of the same node, *intra-flow* link contention where different links of the same flow heavily interference with each other, *inter-flow* link contention where different links of different flows heavily interference with each other. The level and size of the contention in a WMN is determined by the node position, and each node's communication, interference and sensing range.

In our analysis, TDMA-based scheduling is used: Time is divided into small scheduling intervals called slots. In each time slot, the system schedules a number of non-contending links to transmit simultaneously. The scheduled links could be different from slot to slot. The links that are to be scheduled are chosen in the current slot and the chosen links transmit their packets in the next slot.

C. Clique-Based Proportional Fair Scheduling (CBPFS)

Refer to Figure 1, from the network topology represented by a graph $\boldsymbol{G}=(\boldsymbol{N},\boldsymbol{L})$ together with the set of flows $\boldsymbol{F}$, we generate the 1st *link contention graph* $\boldsymbol{G}^{LC\text{-}1}$ that captures the contention among links in such a way that each link is a vertex in this graph and two links that contend are adjacent. Refer to the left-side plot in Figure. 1, there are 11 nodes, 6 active flows f_1-f_6, and 13 links L_1-L_{13} in the network. The right-side plot in Figure 1 is the corresponding link contention graph $\boldsymbol{G}^{LC\text{-}1}$. According to the definition of link contention, link L_1 contends with links L_2, L_6; link L_8 contends with links L_2, L_{11}, L_3, L_{13}. The edge between L_1 and L_6 indicates an *intra-flow* link contention over flow f_2; The edge between L_1 and L_8 indicates an *inter-flow* link contention.

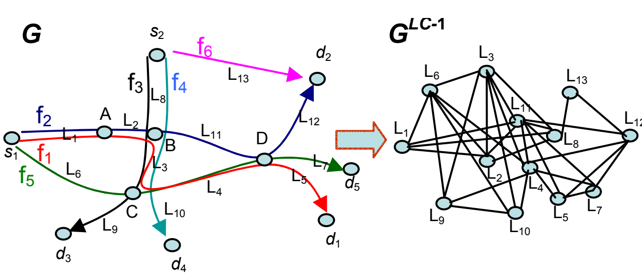

Figure 1. Generation of flow contention graph $\boldsymbol{G}^{LC}$

From the 1st link contention graph $\boldsymbol{G}^{LC\text{-}1}$, we generate the 1st *clique allocation graph* $\boldsymbol{G}^{CA\text{-}1}$ which is a *maximal complete* sub-graph of the *complement* or *inverse* graph of the 1st link contention graph $\boldsymbol{G}^{LC\text{-}1}$. A clique $\boldsymbol{V}$ represents a maximum number of concurrent links in the link contention graph, and is simply the set of vertices in the clique allocation graph. Refer to Figure 2, the 1st clique $\boldsymbol{V}_1=\{L_1, L_3, L_5, L_{13}\}$. Here, the complement or inverse graph $\boldsymbol{G}^I$ of a graph $\boldsymbol{G}$ is constructed in such a way that two vertices in $\boldsymbol{G}^I$ are adjacent if and only if they are not adjacent in $\boldsymbol{G}$.

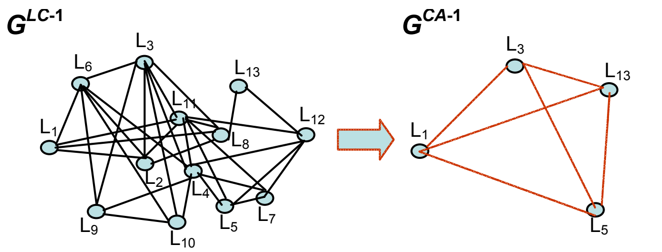

Figure 2. Generation of clique allocation graph $\boldsymbol{G}^{CA\text{-}1}$

Refer to Figure 3, we then remove from $\boldsymbol{G}^{LC\text{-}1}$ the vertices in $\boldsymbol{G}^{CA\text{-}1}$ to produce the 2nd link contention graph $\boldsymbol{G}^{LC\text{-}2}$. Similarly, we generate the 2nd clique allocation graph $\boldsymbol{G}^{CA\text{-}2}$ and have the 2nd clique $\boldsymbol{V}_2$={L_2, L_9, L_{12}}.

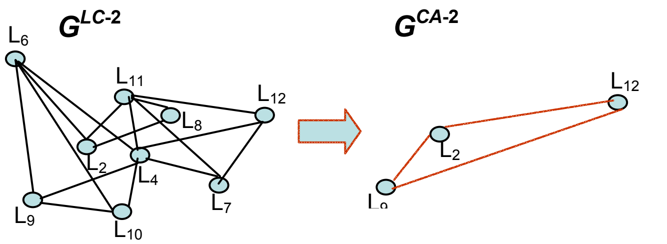

Figure 3. Generation of clique allocation graph $\boldsymbol{G}^{CA\text{-}1}$

We repeat the above procedure to produce the Kth link contention graph $\boldsymbol{G}^{LC\text{-}K}$, the Kth clique allocation graph $\boldsymbol{G}^{CA\text{-}K}$ and the Kth clique $\boldsymbol{V}_K$, until all links are in cliques.

Once we have allocated all links in cliques {$\boldsymbol{V}_i$, i=1, 2, …, K}, we then generate a *clique scheduling graph* $\boldsymbol{G}^{CS}$ that represents each link clique as an entity to be scheduled. In each slot, one and only one clique $\boldsymbol{V}_i$ (i=1, 2, …, K) is scheduled. Once a link clique $\boldsymbol{V}_i$ is chosen to transmit according to the scheduling criterion, all links in link clique $\boldsymbol{V}_i$ will be scheduled to transmit simultaneously. Refer to Figure 4, there are 5 cliques. When $\boldsymbol{V}_1$ is chosen to transmit, links L_1, L_3, L_5 and L_{13} can transmit at the same time.

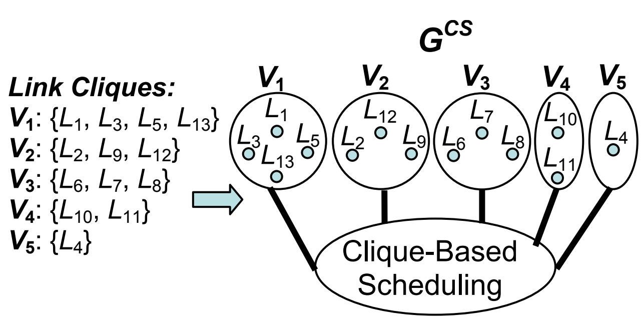

Figure 4. Generation of clique scheduling graph $\boldsymbol{G}^{CS}$

In light of all the superior features of proportional fairness over other fairness criteria, we use the concept of proportional fairness together with the above clique-based scheduling method as our *clique-based proportional fair scheduling* (*CBPFS*) policy. Just as PFS in cellular networks schedules a link/node at each slot, CBPFS in WMNs schedules a link clique at each slot.

It is known that the PFS algorithm maximizes the aggregate utility of all links in a TDMA cellular network [3]. Similarly, CBPFS maximizes the aggregate utility of all link cliques in a TDMA wireless mesh network. In other words, the CBPFS algorithm is the solution to the following optimization problem:

$$\max \sum_{i=1}^{K} \ln\left(\gamma_i[t]\right). \tag{1}$$

s.t.

$$\gamma_i[t] = \sum_{\forall l_{i,j} \in V_i} R_{i,j}[t]. \tag{2}$$

$$\chi_i[t+1] = \sum_{\forall l_{i,j} \in V_i} C_{i,j}[t+1]. \tag{3}$$

$$I_i[t+1] = \begin{cases} 1, & clique\ V_i\ \ is\ scheduled\ \ at\ next\ slot \\ 0, & else \end{cases}. \tag{4}$$

$$R_{i,j}[t+1] = \left(1 - \frac{1}{k}\right) R_{i,j}[t] + I_i[t+1] \times \frac{C_{i,j}[t+1]}{k}. \tag{5}$$

where K is the total number of cliques, t is the current slot, $I_i[t+1]$ is the indicator function of the event that link clique $\boldsymbol{V}_i$ is scheduled to transmit in time slot t+1, $R_{i,j}$ and $C_{i,j}$ are the throughput and capacity of the jth link $l_{i,j}$ in the ith clique.

Simulations [6][8] indicates that each link $l_{i,j}$ in a PFS-enabled cellular network has the same average probability of being scheduled (In fact, this property is also proven in our research on PFS and the detail analyses will be in the future papers). From the generation of link cliques, we know that this interesting property also exists for any link in a CBPFS-enabled WMN. Since the same average scheduling probability means the same number of slots allocated to any link, this property (i.e., equal-time-share) makes CBPFS an efficient yet fair scheduling algorithm for WMNs.

Refer to Figure 1~3, we would like to point out that the network throughput could increase if: we do not remove the vertices in $\boldsymbol{V}_1$ to produce $\boldsymbol{G}^{LC\text{-}2}$. (and $\boldsymbol{G}^{LC\text{-}3}$, …, etc.,) when we have the 1st clique $\boldsymbol{V}_1$={L_1, L_3, L_5, L_{13}} from $\boldsymbol{G}^{LC\text{-}1}$. By this, the resulting cliques may overlap, i.e., a link may belong to multiple cliques. Though in this way, the network throughput could be higher than that in CBPFS due to higher space re-use, this scheme raises fairness issue and is thus not desirable as a link belonging to multiple cliques will be allocated more time slots than other non-overlapping links.

D. *Mathematical Analysis of CBPFS*

For a network topology and the set of flows, the clique scheduling graph $\boldsymbol{G}^{CS}$ is finally generated with the scheme described in sub-section C. We assume that there are K vertices in such graph denoted as link cliques $\boldsymbol{V}_1$, $\boldsymbol{V}_2$,…$\boldsymbol{V}_K$. Link clique $\boldsymbol{V}_i$ = {$l_{i,1}$, $l_{i,2}$,…$l_{i,|Vi|}$} ($l_{i,j} \in \boldsymbol{V}_i \subseteq \boldsymbol{L}$, i=1,2,…K; j=1,2,…|$\boldsymbol{V}_i$|) contains |$\boldsymbol{V}_i$| simultaneous links. From sub-section C, a link l

belongs to one and only one link cliques, i.e., $\boldsymbol{V_i} \cap \boldsymbol{V_j} = \varnothing$, $\forall i \neq j$. We also have $\boldsymbol{V_1} \cup \boldsymbol{V_2} \cup \ldots \cup \boldsymbol{V_K} = \boldsymbol{L}$.

Let's consider the problem where these $|\boldsymbol{L}|$ links wishing to transmit data, and the rates of transmission are randomly varying due to channel fluctuations. We assume that channel fading keeps constant over each slot, and varies independently from slot to slot. The selection of the links to schedule is based on a balance between the current possible rates and fairness. As described in sub-section C, the CBPFS algorithm performs this by comparing the ratio of the capacity for each link clique to its throughput tracked by an exponential moving average, which is defined as the preference metric.

The link clique with the maximum preference metric will be selected for transmission in the next scheduling slot. This is described mathematically as follows. The end of slot t is called *time t*. In next time slot t+1, the capacity of link $l_{i,j} \in \boldsymbol{V_i}$ is $C_{i,j}[t+1]$. $C_{i,j}$ is a random variable whose distribution depends on the characteristics of the underlying Rayleigh fading process. For a link $l_{i,j}$, its k-point moving average rate (i.e., throughput) up to *time t* is denoted by $R_{i,j;k}[t]$. The preference metric of link clique $\boldsymbol{V_i}$ is denoted by $Q_{i;k}[t+1] = \Sigma_{li,j\in \boldsymbol{Vi}} (C_{i,j}[t+1])/\Sigma_{li,j\in \boldsymbol{Vi}} (R_{i,j;k}[t])$. For ease of exposition, in what follows, we will drop the subscript k from the notations $R_{i,j;k}[.]$ and $Q_{i;k}[.]$ where there is no confusion.

Link clique $\boldsymbol{V_i} = \arg_{\boldsymbol{Vj}} \max Q_j[t+1] = \arg_{\boldsymbol{Vj}} \max \Sigma_{li,j\in \boldsymbol{Vi}} (C_{i,j}[t+1])/\Sigma_{li,j\in \boldsymbol{Vi}} (R_{i,j}[t])$ is scheduled to transmit, which means all links $l_{i,j} \in \boldsymbol{V_i}$ are scheduled to simultaneously transmit in next time slot t+1. The moving average rate of link $l_{i,j}$ up to *time* t+1 is updated by (5).

Similar to [3], one can prove that the CBPFS algorithm in sub-section C is the solution to the optimization problem formulated by (1). For the analysis, our objective is to derive a closed-form expression for the average throughput of links (and also end-to-end flows) that are scheduled under the CBPFS policy. As we are dealing with per-link scheduling and the scheduling metric is directly related to link capacity C, it is desirable to incorporate in the analysis a stochastic model of link capacity in WMNs.

In the following, we provide accurate estimate of throughput for WMNs. As suggested in Section I, Gaussian rate model is used for link capacity. We have the following theorems:

Theorem 2.1: Under CBPFS, when each link clique in $\boldsymbol{G^{CS}}$ contains one and only one link $l_j \in \boldsymbol{L}$ (j=1,2,…$|\boldsymbol{L}|$), assuming the capacity C_j of link l_j is statistically independent Gaussian, if the standard deviation σ_j of C_j is a monotonically increasing, concave function of its mean value $E[C_j]$, the average throughput $E[R_j]$ of link l_j can be approximated by

$$E\left[R_j\right] \approx \frac{E\left[C_j\right]}{|L|} \times \left(1 - \left[\phi\left(-M_j\right)\right]^{|L|}\right) + \int_{-M_j}^{\infty} y\sigma_j \rho(y) \times \left[\phi(y)\right]^{|L|-1} dy \quad (6)$$

where $\rho(.)$, $\phi(.)$ are zero mean, unit variance standard normal probability density function and distribution function (i.e., *pdf* and *cdf*), respectively.

Proof: Due to space limitations, we omit here the details of this proof. In fact, CBPFS is PFS in this case and one can refer to [14] for the detailed proof. □

Theorem 2.2: Under CBPFS, assuming the capacity $C_{i,j}$ of each link $l_{i,j}$ is statistically independent Gaussian, let $\chi_i = \Sigma_{\forall li,j\in \boldsymbol{Vi}} (C_{i,j})$ denote the capacity of link clique $\boldsymbol{V_i}$, if the standard deviation $\sigma_{Ci,j}$ of $C_{i,j}$ is a monotonically increasing, concave function of its mean value $E[C_{i,j}]$, the average throughput $E[R_{i,j}]$ of link $l_{i,j}$ can be approximated by

$$E\left[R_{i,j}\right] \approx \frac{E\left[C_{i,j}\right]}{E\left[\chi_i\right]} \times \left[\frac{E\left[\chi_i\right]}{K} \times \left(1 - \left[\phi\left(-M_{\chi_i}\right)\right]^K\right) + \int_{-M_{\chi_i}}^{\infty} y\sigma_{\chi_i} \rho(y) \times \left[\phi(y)\right]^{K-1} dy\right] \quad (7)$$

where $E\left[\chi_i\right] = \sum_{\forall l_{i,j} \in V_i} E\left[C_{i,j}\right]$ is the average throughput of link clique $\boldsymbol{V_i}$, $\sigma_{\chi_i} = \sqrt{\sum_{\forall l_{i,j} \in V_i} \sigma_{i,j}^2}$ is the standard deviation of the throughput of link clique $\boldsymbol{V_i}$, $M_{\chi_i} = E\left[\chi_i\right] / \sigma_{\chi_i}$.

Proof: The capacity $\chi_i = \Sigma_{\forall li,j\in \boldsymbol{Vi}} (C_{i,j})$ of each link clique is statistically independent Gaussian. By considering all links in link clique $\boldsymbol{V_i}$ as a virtual link, with Theorem 2.1, we obtain the average throughput of the i^{th} virtual link. Inside a virtual link, all links transmit at the same time slot, the average throughput of link $l_{i,j}$ is then proportional to its average capacity $E[C_{i,j}]$. We then have (7). One can verify that Theorem 2.1 is a special case of Theorem 2.2.□

E. Flow Throughput in CBPFS-enabled WMNs

Flows between sources and destinations are mapped to paths, according to the routing algorithm (for example, shortest path routing) using some routing metric. In WMNs, multiple flows may share a single link l. Theorem 2.2 provides a closed-form expression to quick evaluate the average throughput $E[R_l]$ of any link l in CBPFS-enabled WMNs. Once $E[R_l]$ is determined, one can use some algorithm to allocate $E[R_l]$ among all flows sharing link l. Let $\tau_{l,f}$ denote the average data rate allocated by link l to flow f. Let N_l denote the total number of flows sharing link l. Given an end-to-end path ζ_f of flow f in a multi-hop network, it is known that the average end-to-end throughput λ_f is [15],

$$\lambda_f = \underset{\forall l \in \zeta_f}{Min} \left[\tau_{l,f}\right]. \quad (8)$$

Here we use a simple algorithm to share throughput among all flows on link l: $\tau_{l,f} = E[R_l]/N_l$, i.e., all flows on link l have the same share of throughput. In practice, this can be implemented by using link l to transmit data for one of the N_l flows in a round-robin manner, when link l is scheduled.

With this average allocation method, we have the average end-to-end throughput of flow f,

$$\lambda_f = \underset{\forall l \in \zeta_f}{Min}\left[E[R_l]/N_l\right]. \quad (9)$$

where $E[R_l]$ is determined by (7).

III. SIMULATION STUDIES

In this section, we present simulation results that validate our analyses. Due to space limitations, we would not go into details about the simulation parameters and setup. For all simulations, the CBPFS scheduling algorithm together with the throughput sharing algorithm shown by (9) are used.

To evaluate the theoretical results presented in Section II, $E[C_{i,j}]$ and $\sigma_{i,j}$ of link capacity should be available beforehand. For example, we can estimate $E[C_{i,j}]$ and $\sigma_{i,j}$ by measuring link $l_{i,j}$ over a period of time. Specifically, according to [12], for a single-input-single-output (SISO), Rayleigh fading link, we have to the following form,

$$E[C] = W\int_0^{\infty} \log(1+SINR\times\lambda)\times e^{-\lambda}d\lambda \quad (10)$$

$$\sigma_C^2 = W^2\int_0^{\infty} \left(\log(1+SINR\times\lambda)\right)^2 \times e^{-\lambda}d\lambda - W^2\left(\int_0^{\infty} \log(1+SINR\times\lambda)\times e^{-\lambda}d\lambda\right)^2 \quad (11)$$

where W is the bandwidth, $SINR$ is the signal to inference-plus-noise ratio, C is the link capacity, $E[C]$ and σ_C are the mean value and the standard deviation of C.

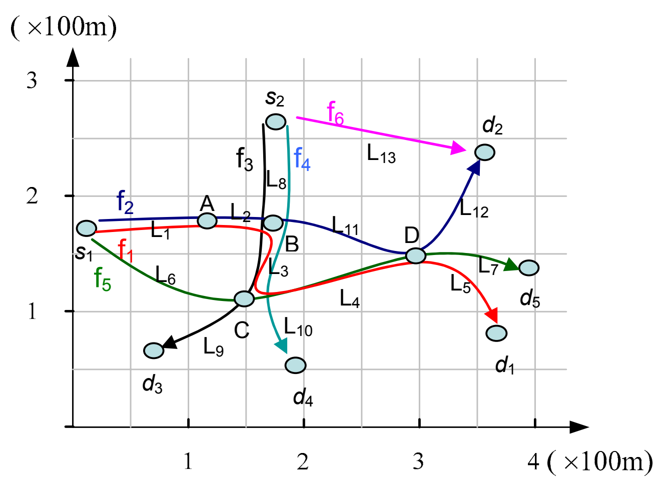

Figure 5. Network topology

In the simulation, we use standard method to simulate Rayleigh fading, i.e., link capacity $C_l=W\times Log[1+SINR_l\times|h_l|^2]$, where the channel gain h_l for link l is a normalized complex Gaussian random variable, W is the bandwidth. All links experience independent fading. System parameters are: bandwidth W=10MHz, path loss exponent α=2.5, reference distance d_0=50m, reference $SINR$ at d_0 is $SINR_{d0}$=25dB. For simplicity, the received average $SINR$ at link l is given by $SINR_l=SINR_{d0}-10\alpha\times\text{Log}_{10}[d_l/r_0]$, where d_l is link distance. More accurate $SINR$ models could also be used but that will unnecessarily complicate the simulation. All simulations run for T=4000 scheduling slots, and we use k=500 for k-point moving average calculation.

Refer to Figure 5, 11 nodes are placed in an area of 300m×400m. There are 13 links L_1~L_{13}, 6 flows f_1~f_6.

We obtain 5 cliques V_1~V_5 as shown in Figure 4. For comparison with the simulation results, we use (7) and (9) to determine the theoretical average throughput of links and flows, which are presented in Table I.

TABLE I. THEORETICAL AVERAGE THROUGHPUT (Mbps)

V_1				V_2			V_3			V_4		V_5
L_1	L_3	L_5	L_{13}	L_2	L_9	L_{12}	L_6	L_7	L_8	L_{10}	L_{11}	L_4
11.9	15.4	13.14	7.5	16.83	13.1	11.8	9.16	12.83	13.35	15.6	10.8	10.5

f_1	f_2	f_3	f_4	f_5	f_6
5.133	5.95	5.133	5.133	5.25	7.5

From Table I, we note that flows f_1, f_3 and f_4 have the same average end-to-end throughput. This indicates that link L_3 is the bottleneck of these three flows. Network provider can allocate more resource to link L_3.to resolve such issue. Preferably, one can use more intelligent throughput sharing algorithm to improve end-to-end throughput without the need for precious radio resource. As a direction for future work, intelligent throughput sharing methods other than the one suggested by (9) will integrate with the CBFPS framework.

The simulated throughput and the theoretical average throughput of links are plotted in Figure 6. For ease of presentation, only L_1, L_3, L_4 and L_6 are shown. The analytical results provide very good estimates of the simulation ones.

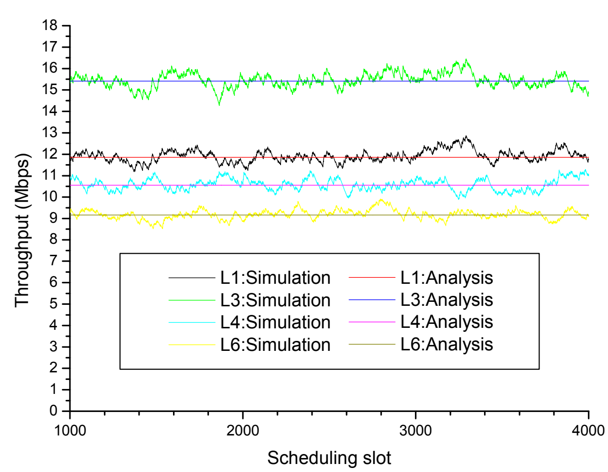

Figure 6. CBPFS Perfromance: Simulation and Analysis

Figure 7 depicts the number of slots allocated to each link in our 4000-slot simulation. Obviously, each link has the same share of time slots in CBPFS. This property makes CBPFS a promising scheduling method for WMNs.

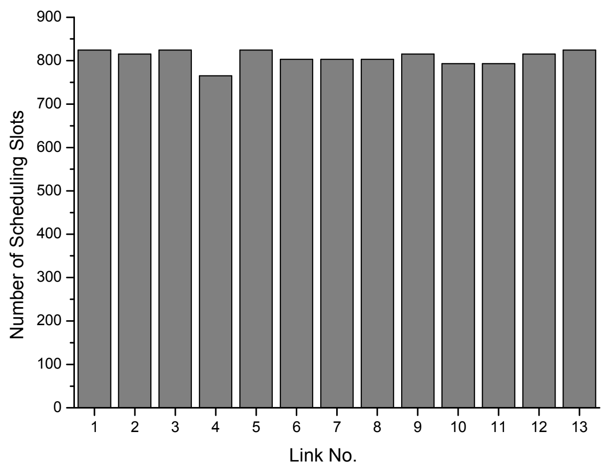

Figure 7. The number of slots allocated to each link

IV. CONCLUSIONS

The PFS algorithm is proposed for cellular networks; in this paper, we proposed a Clique-Based PFS (CBPFS) scheme, where links are grouped into link cliques which are proportionally fair scheduled to achieve maximized utility. CBPFS is a PFS extension and can be used in both cellular networks and WMNs. For the proposed CBPFS algorithm, this paper also analyzed its performance and provided closed-form expressions to evaluate the throughput of both links and flows in Rayleigh fading environments. The theoretical analyses match quite well with simulations.

For future research, we will analyze CBPFS in various fading environments and consider intelligent throughput sharing algorithms with CBPFS.

REFERENCES

[1] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks,* vol. 47, No. 4, March 2005, pp. 445-487.

[2] J. Tang, G. Xue, and W. Zhang, "Cross-layer design for end-to-end throughput and fairness enhancement in multi-channel wireless mesh networks," *IEEE Trans. Wireless Commu.*, vol.6, no.10, October. 2007, pp.3482-3486.

[3] F. Kelly, "Charging and Rate Control for Elastic Traffic", *Eur. Trans. on Telecommun.,* February 1997, pp. 33-37.

[4] X. Huang and B. Bensaou, "On max-min fairness and scheduling in wireless ad-hoc networks: Analytical framework and implementation," in *Proc. ACM MobiHoc,* Long Beach, California, October 2001.

[5] B. Radunovic and J-Y L. Boudec, "Rate performance objectives of multi-hop wireless networks," in *Proc. IEEE INFOCOM,* Hong Kong, March 2004.

[6] J. M. Holtzman, "Asymptotic analysis of proportional fair algorithm," in *Proc. IEEE PIMRC,* San Diego, CA, 2001, pp. 33-37.

[7] H. J. Kushner and P. A. Whiting, "Asymptotic Properties of Proportional-Fair Sharing Algorithms: Extensions of the Algorithm," in *Proc. of the Annual Allerton Conference on Communication, Control and Computing,* vol. 41, 2003, pp. 303-311.

[8] S. Borst, "User-level performance of channel-aware scheduling algorithms in wireless data networks," in *Proc. INFOCOM,* San Francisco, 2003, pp. 321-331.

[9] J-G. Choi and S. Bahk, "Cell-Throughput Analysis of the Proportional Fair Scheduler in the Single-Cell Environment," *IEEE Trans. Veh. Technol.,* vol. 56, no. 2, pp. 766-778, March 2007.

[10] E. Liu and K. K. Leung, "Fair resource allocation under rayleigh and/or rician fading environments," in *Proc. IEEE PIMRC,* Cannes, France, September 2008.

[11] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Transactions on Telecommunications*, Vol.10, No.6, pp.585-595, November/December 1999.

[12] P. J. Smith and M. Shafi, "On a gaussian approximation to the capacity of wireless MIMO systems," in *Proc. IEEE ICC*, New York, April 2002, pp. 406-410.

[13] P. J. Smith, S. Roy, and M. Shafi, "Capacity of MIMO systems with semi-correlated flat fading," *IEEE Trans. Inform. Theory,* vol. 49, no. 10, October 2003.

[14] E. Liu, Q. Zhang, and K. K. Leung, "Opportunistic resource allocation for frequency-selective fading, multi-carrier systems with fairness constraints," in *Proc. IEEE ICC,* Dresden, Germany, June 2009. to appear.

[15] Y. Gao, D-M. Chiu, and J. C. S. Lui, "Determining the end-to-end throughput capacity in multi-hop networks: methodology and applications," *SIGMetrics/Performance,* Saint Malo, France, June 2006.