End-to-End Fair Bandwidth Allocation in Multi-hop Wireless Ad Hoc Networks

Baochun Li

***Abstract*—**
The shared-medium multi-hop nature of wireless ad hoc networks poses fundamental challenges to the design of an effective resource allocation algorithm to maximize spatial reuse of spectrum, while maintaining basic fairness among multiple flows. When previously proposed scheduling algorithms have been shown to perform well in providing fair shares of bandwidth among *single-hop* wireless flows, they do not consider *multi-hop* flows with an end-to-end perspective when maximizing spatial reuse of spectrum. Instead, previous work attempts to break each multi-hop end-to-end flow into multiple single-hop flows for scheduling purposes. While this may be sufficient for maintaining basic fairness properties among single-hop subflows with respect to bandwidth, we show that, due to the intra-flow correlation between upstream and downstream hops, it may not be appropriate for maximizing spatial reuse of bandwidth. In this paper, we analyze the issue of increasing such spatial reuse of bandwidth from an end-to-end perspective of multi-hop flows. Through analysis and simulation results, we show that our proposed algorithm is able to appropriately distribute resources among multi-hop flows, so that end-to-end throughput may be maximized in wireless ad hoc networks, while still maintaining basic fairness across the multi-hop flows.

I. INTRODUCTION

A wireless ad hoc network consists of a collection of wireless nodes without a fixed infrastructure. Each network node serves as a router that forwards packets for other nodes. Each flow from the source to the destination traverses multiple hops of wireless links. Compared with wireline networks where flows contend only at the packet router with other simultaneous flows through the same router (contention in the time domain), the unique characteristics of medium access control protocols in wireless networks show that, flows also compete for shared channel bandwidth if they are within the transmission ranges of each other (contention in the spatial domain). This presents the problem of designing an appropriate topology-aware resource allocation algorithm such that contending flows fairly share the scarce channel capacity, while increasing spatial reuse of spectrum as much as possible to improve channel utilization.

With intuitive examples, previous work [1] has pointed out that such a topology-aware resource allocation algorithm needs to carefully arbitrate the trade-off between the two extremes: maintaining strict fairness among backlogged flows may lead to waste of bandwidth, while solely maximizing the spatial reuse of spectrum violates fairness among flows (e.g. a subset of flows may be starved). Towards the goal of reaching a balanced trade-off, Luo *et al.* [1], [2], [3] has presented several centralized or distributed scheduling algorithms in wireless ad hoc networks. These algorithms attempt to discuss the problem from both a theoretical and a practical point of view, making various degrees of trade-offs between the conflicting goals of maximizing bandwidth spatial reuse and maintaining fairness among flows. While the contributions are original and noteworthy, their definition of a *flow* is limited to *single-hop flows*, with one direct wireless link between the source and the destination. In contrast, in actual multi-hop wireless ad hoc networks, flows with *multiple hops* are commonplace, while single-hop flows are only exceptions.

With existing solutions targeting single-hop flows, a natural extension to address identical arbitration goals in the context of multi-hop flows may be to consider them as multiple independent *single-hop flows* (referred to as *subflows* henceforth). However, after breaking a multi-hop flow into multiple independent subflows, the inherent correlation between upstream and downstream subflows are lost. One of the possible consequences is the increased probability of dropping packets due to buffer overflow in the downstream, since maximizing subflow throughput may lead to a higher bandwidth allocation in the upstream than downstream subflows (in the same multi-hop flow), which leads to potential buffer overflow in the downstream routers.

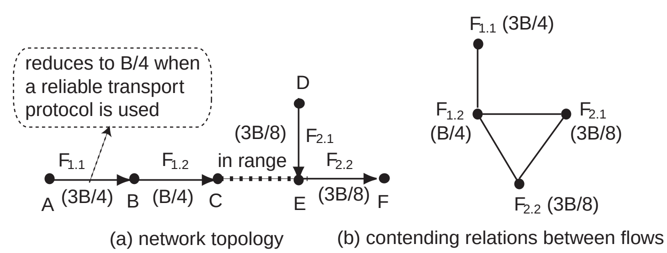


Fig. 1. Fair bandwidth allocation among multi-hop flows: the problem

To illustrate this problem, consider the example topology in Fig. 1(a), where there are two multi-hop flows: F_1 from node A to C and F_2 from D to F. If we break both multi-hop flows into subflows, and assume that two subflows contend spatially if their sources or destinations are within range, then Fig. 1(b) shows the flow contention relationships between the subflows, where $F_{i.j}$ denotes the jth subflow of a multi-hop flow F_i, counting from the source. As shown, $F_{1.1}$ contends with its immediate downstream subflow $F_{1.2}$ and $F_{1.2}$ contends with both $F_{2.1}$ and $F_{2.2}$. For convenience, we assign equal weights of 1 to all subflows.

The basic idea from previous work (e.g., [1]) is to first guarantee a basic share of $B/4$ to all four subflows, where B denotes the effective channel capacity for data transmissions[1], and then select maximum independent sets of subflows that may transmit packets concurrently. The subflows in such a set may increase

Baochun Li is affiliated with Department of Electrical and Computer Engineering, University of Toronto. His email address is *bli@eecg.toronto.edu*.

[1] For example, in a IEEE 802.11-based wireless channel of 2 Mbps, the effective capacity available for data transmissions is approximately 1.7 Mbps [4].

their shares to maximize spatial bandwidth reuse. Since $F_{2.1}$ and $F_{2.2}$ do not compete with $F_{1.1}$, it may be easily shown that spatial reuse of bandwidth is maximized if we guarantee $B/4$ to $F_{1.2}$, and assign $3B/8$ to $F_{2.1}$ and $F_{2.2}$, and finally $3B/4$ to $F_{1.1}$. However, if we consider multi-hop flow F_1, since $F_{1.2}$ is the bottleneck, packets coming from subflow $F_{1.1}$ will accumulate at node B, buffer overflow may eventually occur at node B. If a reliable transport protocol is used, the end-to-end throughput of F_1 stabilizes to $B/4$ over time (for both subflows $F_{1.1}$ and $F_{1.2}$), with a total effective throughput of $5B/4$ for all subflows. However, if we allocate $B/2$ to both $F_{1.1}$ and $F_{1.2}$, and $B/4$ to both $F_{2.1}$ and $F_{2.2}$, the total effective throughput for all subflows increases to $3B/2$, while still guarantees the basic fair shares of $B/4$ across the subflows[2].

In this paper, we study the problems illustrated by this intuitive example, based on strict definitions of *total effective throughput* (that characterizes the extent of spectrum spatial reuse) and *fairness* among *multi-hop flows* from an end-to-end perspective. The novelty of our analysis and algorithms comes from the fundamental differences between the problems presented by *single-hop* and *multi-hop* flows, when optimality is sought given strict definitions of constraints in both problem domains. Our insights shown in this work start from clear and appropriate definitions of our objective (maximizing spatial reuse of bandwidth) and fairness constraints. Based on these definitions, we show that previous approaches that consider single-hop flows may not achieve the same optimal results when multi-hop flows are considered. In order to evaluate any proposed algorithms against an upper bound in ideal situations, we proceed to propose an estimation algorithm (Sec. III) to estimate the optimal allocation strategies based on the defined notion of fairness in multi-hop flows. More realistically, we first propose a centralized algorithm to achieve our objectives, followed by a distributed algorithm (that only relies on local state information) to approximate its centralized counterpart. Finally, a distributed backoff-based scheduling algorithm is developed to achieve calculated shares of the subflows. We argue that, while studying single-hop flows may lead to theoretical insights, examining multi-hop flows leads to more general results and applicable algorithms in realistic wireless ad hoc networks.

The remainder of the paper is organized as follows. We motivate the work and present clear definitions of the objectives and constraints in this work in Sec. II. We present our idealized estimation algorithm in Sec. III, and the proposed scheduling algorithms in Sec. IV. Sec. V evaluates the performance of our algorithms. Sec. VI and VII discuss related work and conclude the paper.

II. Objective and Constraints

Conceptually, the objective of our work is to *maximize spatial reuse of spectrum*, while satisfying the constraint of *maintaining basic fairness among contending flows*. Obviously, the prerequisites of a solution include the definition of a metric that evaluates the extent of spatial reuse of spectrum, and the definition of *basic fairness* among multi-hop flows. We present clear and appropriate definitions of both our objective and our constraints.

[2]Detailed discussions of this example are postponed to Sec. III.

A. Preliminaries

Two active (backlogged) subflows are *contending subflows* if either the source or destination of one subflow is within the transmission range of the source or destination of the other. Two multi-hop flows are *contending flows* if any of their subflows are contending subflows. If multi-hop flows F_i and F_j are contending flows, we claim that F_i and F_j are in the same *contending flow group*, i.e., $G(F_i) = G(F_j) = \{F_i, F_j\}$. We note that if $G(F_i) = G(F_j)$ and $G(F_j) = G(F_k)$, it may be possible that F_i and F_k are not contending flows. In this case, we still claim that F_i, F_j and F_k are in the same contending flow group $\{F_i, F_j, F_k\}$. As such, all multi-hop flows in the network are essentially partitioned into several disjoint contending flow groups.

A *subflow contention graph* represents the spatial contention relationship among contending subflows, where vertices correspond to subflows and connected vertices correspond to contending subflows. Fig. 1(b) show examples of subflow contention graphs. Naturally, partitioned subgraphs in a subflow contention graph corresponds to contending flow groups.

We assume a preassigned weight w_i for each multi-hop flow F_i. We further assign $w_{i.j} = w_i$, where $w_{i.j}$ represents the weight for the subflow $F_{i.j}$.

B. Objective: Maximizing Spatial Reuse of Spectrum

When discussing single-hop flows, the objective of maximizing spatial reuse of bandwidth may naturally be translated to maximizing the aggregate channel utilization [1], or the *total effective single-hop throughput* in the network; i.e., maximizing $\sum_i u_i$, for all active (backlogged) single-hop flows F_i in the network, where u_i denotes the throughput of the single-hop flow F_i. For the case of multi-hop flows, if we revisit the previous example (Fig. 1), the problem hinges on an appropriate definition of the objective. If we reuse the objective defined in the single-hop case, the total effective single-hop throughput is $7B/4$ (that is superior than $3B/2$ of our proposed alternative solution). However, since $F_{1.2}$ is the bottleneck in F_1, the actual *end-to-end throughput* achieved in F_1 (assuming an effective reliable transport protocol) is $B/4$, leading to a total effective single-hop throughput of $5B/4$ (which is inferior to the alternative). It decreases since we have taken the end-to-end effect into consideration. Details of transport protocols aside, the end-to-end throughput of multi-hop flows is determined by the minimum throughput of its subflows, i.e., $u_i = \min(u_{ij}), j = 1, \ldots, l_i$, where u_{ij} denotes the throughput of the j^{th} subflow and l_i is the length of the flow F_i.

We define the *total effective throughput* as the total *end-to-end* throughput of all multi-hop flows, i.e., $\sum_i u_i$, for all active (backlogged) multi-hop flows F_i in the network, where u_i is given previously. When we target on maximizing the spatial reuse of spectrum, our objective is then to maximize the total effective throughput. It is subtly different from the objective in the single-hop case. We argue that considering end-to-end flow throughput is a more appropriate metric to measure the effectiveness of bandwidth spatial reuse, since packets delivered in a single hop and then dropped in downstream hops constitute a waste of bandwidth, rather than the contrary.

C. Fairness: the case of multi-hop flows

If we revisit wireline networks, the fairness constraint may be *locally* defined among backlogged flows contending for a single bottleneck link. To be more specific, for any two backlogged flows F_i and F_j contending for a bottleneck link during a specific period $[t_1, t_2]$, the aggregate services they receive over this link satisfy:

$$\left| \frac{\int_{t_1}^{t_2} r_i(t)dt/(t_2 - t_1)}{w_i} - \frac{\int_{t_1}^{t_2} r_j(t)dt/(t_2 - t_1)}{w_j} \right| < \epsilon \qquad (1)$$

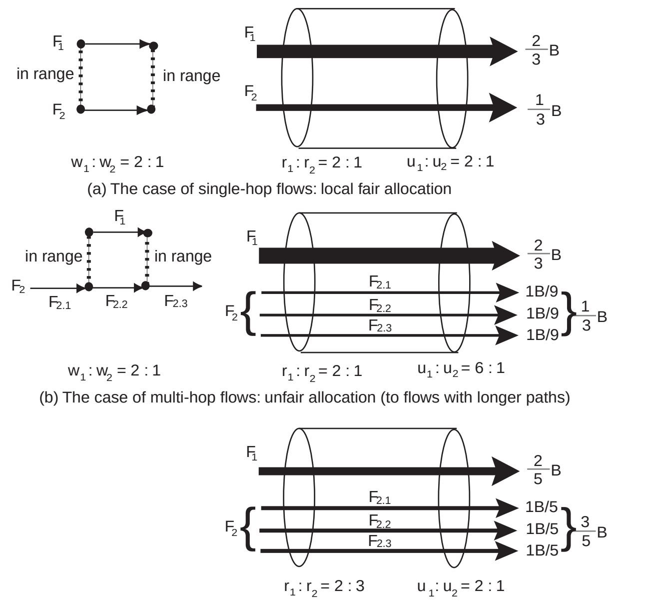
where $r_i(t)$ is the instantaneous link capacity allocated to F_i at time t. For the case where capacity allocation for F_i is constant and equals r_i, $\int_{t_1}^{t_2} r_i(t)dt = (t_2 - t_1)r_i$. If we only consider long-lived flows (F_i) with a constant (or stable) source bit rate (ρ_i), we may simplify the definition to the objective of achieving *weighted max-min fairness* across all flows contending for the same bottleneck link; i.e., an allocation strategy $(r_1, \ldots, r_n)$ is *weighted max-min fair*, if (1) both $\sum_{k=1}^{n} r_k \leq B$ and $r_i \leq \rho_i, i = 1, \ldots, n$ hold for all n contending flows; and (2) for each flow F_i, any increase in r_i would cause a decrease in the allocation r_j for some flow F_j satisfying $r_j/w_j < r_i/w_i$.

In multi-hop wireless networks, since flows contend for channel allocation in both time and spatial domains, fairness is essentially a topology-aware global property. However, if we only consider single-hop flows F_i (as in previous work) within the same local neighborhood with effective channel capacity B, if the set of contending flows are known, we may start by using the previous definition of weighted max-min fairness for local channel allocation. For the purpose of this paper, however, we make one additional simplifying assumption[3] that the sources are always greedy, i.e., $r_i < \rho_i$ for all contending flows F_i. Therefore, $u_i = r_i$. Under such an assumption, for the local effective channel capacity B, we may determine that the allocation strategy $(r_1, \ldots, r_n)$ is *fair* for n single-hop contending flows $(F_1, \ldots, F_n)$, if $\sum_{k=1}^{n} r_k \leq B$ and $|r_i/w_i - r_j/w_j| < \epsilon$ over any time period $[t_1, t_2]$. In the example shown in Fig. 2(a), if $w_1 = 2$ and $w_2 = 1$, a fair allocation strategy $(r_1, r_2) = (2B/3, B/3)$.

Within the local channel, we proceed to extend such a definition to the case of multi-hop flows. A straightforward extension is as follows. The allocation strategy $(r_1, \ldots, r_n)$ is *fair* for multi-hop contending flows $(F_1, \ldots, F_n)$, if $\sum_{k=1}^{n} r_k \leq B$ and $|r_i/w_i - r_j/w_j| < \epsilon$ over any time period $[t_1, t_2]$. In the example shown in Fig. 2(b), we show that such a strategy is unfair to flows with longer paths. When F_2 is a three-hop flow, the strategy allocates $r_2 = B/3$. Since the allocation is shared by the three subflows of F_2, the end-to-end throughput $u_2 = B/9$. In this case, $u_2/u_1 = 1/6$, which is inconsistent with $w_2/w_1 = 1/2$. F_2 is penalized for its longer path.

Since our focus is on the end-to-end throughput of flows, the desired fairness constraint is $|u_i/w_i - u_j/w_j| < \epsilon$ over any time period $[t_1, t_2]$. For example, in Fig. 2(c), a more appropriate allocation strategy may be $(r_1, r_2) = (2B/5, 3B/5)$, so that $(u_1, u_2) = (2B/5, B/5)$, which is fair to F_2. Generally, if $r_{i.j}$ is allocated to the subflow $F_{i.j}$, we have $u_{ij} = r_{i.j}$, thus $u_i = \min(r_{i.j})$. If we equalize channel allocations for all subflows belonging to the same flow, i.e., if $\hat{r_i} \equiv r_{i.1} = r_{i.2} = \ldots = r_{i.l_i}$ for an l_i-hop flow F_i, we have $u_i = r_{i.j} = \hat{r_i}$. As such, from the viewpoint of channel allocation, we define the fairness constraint as $|\hat{r_i}/w_i - \hat{r_j}/w_j| < \epsilon$. For the example in Fig. 2(c), $r_2 = 3B/5$, while $\hat{r_2} = r_{2.j} = B/5, j = 1, 2, 3$.

[3]This assumption usually holds due to the scarce capacity in a wireless channel. When it does not hold, the results in this work may easily be extended to the more generic case of weighted max-min fair allocations.



(a) The case of single-hop flows: local fair allocation

(b) The case of multi-hop flows: unfair allocation (to flows with longer paths)

(c) The case of multi-hop flows: local fair allocation of end-to-end throughput

Fig. 2. Fairness: the single-hop and multi-hop case

Finally, we extend our definition from the local channel to the global topology. In the strictest sense of fairness, we require that $|\hat{r_i}/w_i - \hat{r_j}/w_j| < \epsilon$ is satisfied for all flows in the network. However, such a constraint limits spatial bandwidth reuse for the flows that are not in the area of intense contention. For the interests of spatial reuse of bandwidth, we only define the fairness constraint among contending flows, rather than considering all flows in the network. *Fairness* among multi-hop flows in a wireless multi-hop network is defined as follows.

Definition: In a multi-hop wireless network, the allocation strategy $(\hat{r_1}, \ldots, \hat{r_n})$ is *fair* for contending flows $(F_1, \ldots, F_n)$ *in the same contending flow group*, if (1) within any local neighborhood (that flows contend for the same channel capacity B), $\sum_{k=1}^{n} m_k \hat{r_k} \leq B$, with m_i being the number of contending subflows of F_i in this local neighborhood; and (2) $|\hat{r_i}/w_i - \hat{r_j}/w_j| < \epsilon$ over any time period $[t_1, t_2]$.

Henceforth, we only consider the case of a single contending flow group, since multiple contending flow groups may transmit concurrently without contention. Further, we assume n multi-hop flows, $F_1, \ldots, F_n$, exist in such a contending flow group, and each flow F_i consists of l_i subflows $F_{i.1}, \ldots, F_{i.l_i}$.

D. Basic Fairness

We now turn to illustrate the differences between the single-hop and multi-hop cases, and to derive the definition of the *basic share* of a flow. With respect to end-to-end throughput of a flow, the previously defined *fairness* constraint guarantees $|u_i/w_i - u_j/w_j| < \epsilon$. The allocation strategies that satisfy such a fairness constraint are, in fact, not unique, and result in different total effective throughput. For example, we may begin by considering contending subflows of a multi-hop flow independently. By breaking multi-hop flows into subflows, we may use the allocation strategy $(\hat{r_1}, \ldots, \hat{r_n})$ that satisfies the following:

$$\sum_{i=1}^{n}\sum_{j=1}^{l_i} r_{i.j} = \sum_{i=1}^{n} \hat{r_i} l_i = B \tag{2}$$

In order to satisfy the constraint of fairness, it may be easily derived that $u_i = \hat{r_i} = w_i B / \sum_{j=1}^{n} w_j l_j$. The total effective throughput is $\sum_{i=1}^{n} u_i = \frac{(\sum_{i=1}^{n} w_i) B}{\sum_{j=1}^{n} w_j l_j}$. The general idea of this allocation strategy is to allocate B to all subflows in the same contending flow group, regardless of whether they actually contend in the same local neighborhood. A special case is the example in Fig. 2(c).

We show that, if the correlation between subflows in the same flow is considered, we may achieve a better total effective throughput. For this purpose, we first consider a multi-hop flow shown in Fig. 3(a), where nodes N_i and $N_j (j > i + 1)$ are in range. Such a flow is referred to as a flow *with a shortcut*. Fig. 3(b) shows the same flow without such a shortcut. In our analysis, we assume that all flows are without shortcuts. This is a realistic assumption, since most ad hoc routing protocols (e.g. Dynamic Source Routing) finds shortest paths.

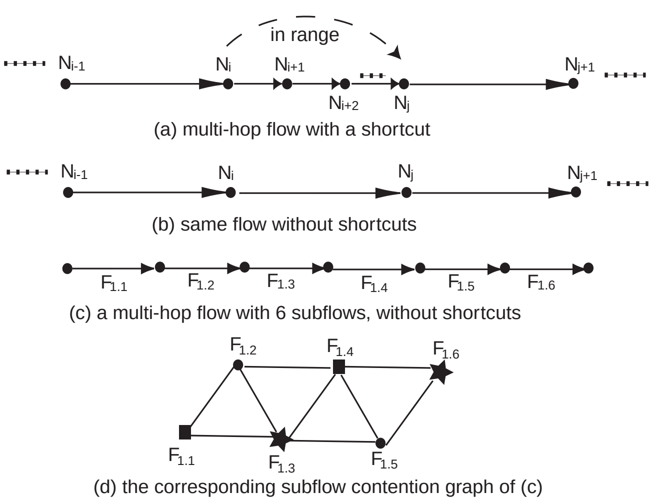


Fig. 3. Examples of multi-hop flows

Given this assumption, it may be derived that, for a flow F_i, each subflow $F_{i.k}$ only contends with its immediate upstream flow $F_{i.k-1}$ and immediate downstream flow $F_{i.k+1}$. If $l_i \geq 3$, we may classify the subflows into three independent sets, where subflows in each set may transmit concurrently: $\{F_{i.j}, j = 3k + 1, k \geq 0\}$, $\{F_{i.j}, j = 3k + 2, k \geq 0\}$, and $\{F_{i.j}, j = 3k + 3, k \geq 0\}$. This is, in fact, a graph coloring exercise in the subflow contention graph of F_i. For example, we consider a multi-hop flow with 6 subflows (but without shortcuts), illustrated in Fig. 3(c), whose subflow contention graph is shown in Fig. 3(d). We use the minimum number of colors (3) to partition the subflows into three non-contending sets: $\{F_{1.1}, F_{1.4}\}$, $\{F_{1.2}, F_{1.5}\}$, $\{F_{1.3}, F_{1.6}\}$.

Given the same channel capacity, a flow with more than 3 hops is entitled to the same end-to-end throughput as a flow that has exactly 3 hops. We define the *virtual length* of a flow F_i, v_i, as follows:

$$v_i = \begin{cases} 3 & l_i \geq 3 \\ l_i & l_i < 3 \end{cases}$$

Subject to the fairness constraint and $\sum_{i=1}^{n} r_i = B$, we may obtain the allocation strategy $(\hat{r_1}, \ldots, \hat{r_n})$, such that $\hat{r_i} = u_i = \frac{w_i B}{\sum_{j=1}^{n} w_j v_j}$. The total effective throughput under such a strategy is $\sum_{i=1}^{n} u_i = \frac{(\sum_{i=1}^{n} w_i) B}{\sum_{j=1}^{n} w_j v_j}$. Since $v_i \leq l_i$ for any F_i, we observe that the end-to-end throughput and the total effective throughput achieved in the case of multi-hop flows is *no lower than* those in the case of single-hop flows. Nevertheless, such an allocation strategy still satisfies the fairness constraint. The possibly higher share of allocation to each flow is made possible by considering the intra-flow spatial reuse of spectrum. Hereafter, the allocation $\hat{r_i} = \frac{w_i B}{\sum_{j=1}^{n} w_j v_j}$ is referred to as the *basic share* of F_i, and the resulting throughput u_i as the *basic throughput*. When all flows receive the basic share, the total effective throughput is at least $\frac{(\sum_{i=1}^{n} w_i) B}{\sum_{j=1}^{n} w_j v_j}$. We claim that an allocation strategy satisfies the constraint of *basic fairness*, if the allocation of any flow is equal to or higher than its basic share.

Naturally, the fairness constraint is stronger, while it is advantageous to achieve a higher total effective throughput if only *basic fairness* is required. Our objective is to maximize the total effective throughput, while supplying the basic fairness property.

III. Optimal Allocation Strategies

For the purpose of evaluating the effectiveness of any proposed algorithms against solutions in the ideal case, we develop an estimation algorithm to calculate the optimal allocation strategies that achieve our objective of maximizing spatial bandwidth reuse, while satisfying (1) the fairness constraint; and (2) the basic fairness constraint.

A. Satisfying the Fairness Constraint

We first present optimal allocation strategies to satisfy the fairness constraint. Since optimality is achieved by maximizing total effective throughput, we estimate the theoretical upper bound of total effective throughput with a weighted subflow contention graph, where each node in a subflow contention graph is labeled with the weight of the corresponding subflow. Fig. 4 shows an example of the weighted flow contention graph.

In the weighted subflow contention graph, a complete subgraph is referred to as a *clique*, while a clique not contained in another clique is referred to as the *maximum clique*, which represents a set of subflows that mutually contend with each other. Assume that there are J maximum cliques in the weighted subflow contention graph. The sum of weights on all vertices in a clique is referred to as the *weighted clique size*, ω_{Ω_k}, of the corresponding clique $\Omega_k, k = 1, \ldots, J$. The maximum of weighted clique sizes of all maximum cliques is referred to as the *weighted clique number*, $\omega_\Omega = \max\ \omega_{\Omega_k}, k = 1, \ldots, J$. In addition, assume that for each flow F_i, there are $n_{i,k}$ subflows in the clique

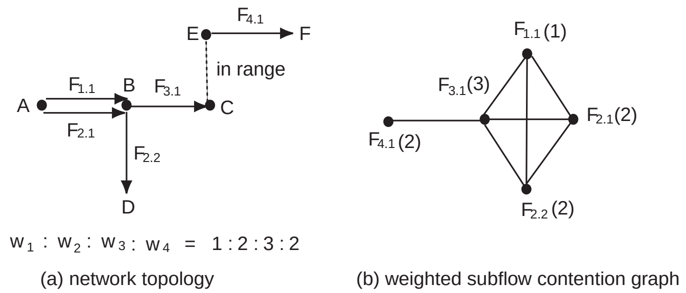


Fig. 4. Weighted subflow contention graph

Ω_k ($n_{i,k} \geq 0$). Since all subflows in the same clique contends for the channel capacity B, for contending flows $(F_1, \ldots, F_n)$ in the same contending flow group, we have

$$\sum_{i=1}^{n} (n_{i,k} \hat{r_i}) \leq B, \; k = 1, \ldots, J \tag{3}$$

Under the fairness constraint, we have $u_i/u_j = \hat{r_i}/\hat{r_j} = w_i/w_j$. With respect to the channel allocation per unit weight (denoted by $\hat{r_0}$), we have $\hat{r_i} = w_i \hat{r_0}$. We thus obtain

$$\sum_{i=1}^{n} (n_{i,k} w_i) \hat{r_0} = \omega_{\Omega_k} \hat{r_0} \leq B, \; k = 1, \ldots, J \tag{4}$$

which leads to

$$\omega_\Omega \hat{r_0} \leq B \tag{5}$$

Hence, channel allocation per unit weight is upper bounded by B/ω_Ω. Thus, the theoretical upper bound of both $\hat{r_i}$ and end-to-end throughput u_i of a flow F_i is $w_i B/\omega_\Omega$, while the upper bound of total effective throughput is $\sum_{i=1}^{n} u_i = \sum_{i=1}^{n} w_i B/\omega_\Omega$.

Proposition 1: Under the fairness constraint, the upper bound of total effective throughput is $\sum_{i=1}^{n} w_i B/\omega_\Omega$, where ω_Ω denotes the weighted clique number.

Proposition 1 sets the *theoretical* upper bound of the total effective throughput under fairness constraint. It is consistent with the calculated basic share of a flow, since in the maximum clique, there are at most v_i subflows for each flow F_i, we have $\omega_\Omega \leq \sum_{i=1}^{n} w_i v_i$. The equality occurs when all n flows have v_i subflows in the maximum clique. However, we note that in some cases, the upper bound is not achievable. For example, Fig. 5 shows a pentagon-shaped weighted flow contention graph. Obviously, $\omega_\Omega = 2$. From Proposition 1, the upper bound of total effective throughput is $5B/2$, with each flow achieving an end-to-end throughput of $B/2$. However, this is, in fact, impossible to achieve.

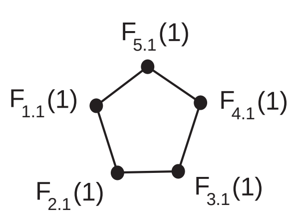


Fig. 5. weighted subflow contention graph with unachievable upper bound of total effective throughput: the *pentagon* example

B. Satisfying the Basic Fairness Constraint

In order to supply the basic fairness property, the optimal allocation strategy $(\hat{r_1}, \ldots, \hat{r_n})$ needs to satisfy the basic share constraints $\hat{r_i} \geq \frac{w_i B}{\sum_{j=1}^{n} w_j v_j}, 1 \leq i \leq n$. Further, when all flows contend for the channel capacity B in the same clique, Eq. (3) needs to be satisfied. Under such constraints, our objective is to maximize the total effective throughput $\sum_{i=1}^{n} u_i = \sum_{i=1}^{n} \hat{r_i}$. We formulate this optimization problem as the following linear programming problem:

maximize $\sum_{i=1}^{n} \hat{r_i}$
subject to

$$\sum_{i=1}^{n} n_{i,k} \hat{r_i} \leq B, \; 1 \leq k \leq J \tag{6}$$

$$\hat{r_i} \geq \frac{w_i B}{\sum_{j=1}^{n} w_j v_j}, \; 1 \leq i \leq n \tag{7}$$

Eq. (6) is identical to Eq. (3), while Eq. (7) guarantees the basic share of each flow, and supplies the basic fairness property. We proceed to show that there exists basic feasible solutions to the above optimization problem. Let $x_i = \hat{r_i} - \frac{w_i B}{\sum_{j=1}^{n} w_j v_j}, 1 \leq i \leq n$. x_i represents the additional shares that a flow may be allocated after the basic fairness constraint is satisfied. It can be straightforwardly derived that the above problem is equivalent to the following linear programming problem in a canonical form:

maximize $\sum_{i=1}^{n} x_i$
subject to

$$\sum_{i=1}^{n} n_{i,k} x_i \leq B - B\frac{\omega_{\Omega_k}}{\sum_{j=1}^{n} w_j v_j}, \; 1 \leq k \leq J \tag{8}$$

$$x_i \geq 0, \; 1 \leq i \leq n \tag{9}$$

Obviously, $\{x_i = 0, \; i = 1, \ldots, n\}$ is a basic feasible solution. With this solution, the total effective throughput is $\frac{\sum_{i=1}^{n} w_i B}{\sum_{j=1}^{n} w_j v_j}$, which is achieved when all flows enjoy their basic throughput in the network.

Proposition 2: The solution to the above linear programming problem constitutes the optimal allocation strategy $(\hat{r_1}, \ldots, \hat{r_n})$, while supplying the basic fairness property. Such an allocation strategy maximizes the total effective throughput.

The process of constructing the linear programming problem essentially proves the correctness of the proposition. It is known that there exist polynomial-time algorithms to solve such a linear programming problem; however, in most cases, it is sufficient to solve the problem with the Simplex algorithm.

Mostly, there exist feasible schedules to achieve the optimal allocation strategy $(\hat{r_1}, \ldots, \hat{r_n})$ and to allocate the calculated shares of bandwidth. However, there are still exceptional cases, where there is no feasible schedules to achieve the derived optimal allocation strategy. In the *pentagon* example, shown in Fig. 5, the optimal allocation strategy is $B/2$ for each flow, where there exist no feasible schedules to achieve such a strategy. In this case, we use the calculated optimal allocation strategy as a new set of *weight factors* to drive our algorithm proposed in the next section. Such weights are referred to as the *allocated share*, to distinguish from the original weights (w_i) of the flows. The *allocated shares* incorporate the original weight information, since the basic fair share is guaranteed during the derivation, and obtained based on the original weight. Further, they reflect the situation of contention. Therefore, although in exceptional situations the optimal allocation strategy is not feasible, it shows the appropriate weight ratio among the flows that may lead to the best total effective throughput.

We revisit the example shown in Fig. 1, with the objective of maximizing spatial reuse of spectrum (i.e., the total effective throughput). If we are to satisfy the fairness constraint, according to our analysis, the allocation strategy $(\hat{r_1}, \hat{r_2}) = (u_1, u_2) = (B/3, B/3)$ for (F_1, F_2), which leads to a total effective throughput of $2B/3$. In comparison, if we are to satisfy the basic fairness constraint, the solution of the linear programming problem:

maximize $\hat{r_1} + \hat{r_2}$
subject to

$$\begin{aligned} 2\hat{r_1} &\leq B \\ \hat{r_1} + 2\hat{r_2} &\leq B \\ \hat{r_1} &\geq B/4 \\ \hat{r_2} &\geq B/4 \end{aligned}$$

leading to the optimal allocation strategy $(\hat{r_1}, \hat{r_2}) = (u_1, u_2) = (B/2, B/4)$ for (F_1, F_2), which amounts to a total effective throughput of $3B/4$, and this optimal allocation strategy has a feasible scheduling associated with it.

We compare the above allocation strategies with results from previous work [1], where the objective is to maximize total throughput of all single-hop flows, while guaranteeing basic fairness among single-hop flows. In the same example topology, such a different objective that focuses on single-hop flows yields an allocation strategy $(r_{1.1}, r_{1.2}, r_{2.1}, r_{2.2}) = (3B/4, B/4, 3B/8, 3B/8)$. With respect to end-to-end throughput of multi-hop flows (F_1, F_2), we have $(u_1, u_2) = (B/4, 3B/8)$. The total effective throughput is thus $5B/8$, which is inferior to the optimal solution obtained in our analysis ($3B/4$). However, the total single-hop throughput obtained in previous work ($7B/4$) exceeds that achieved by our allocation strategy ($3B/2$). This comparative study shows the importance of considering end-to-end throughput of multi-hop flows, and the effectiveness of our solutions.

IV. Achieving Allocation Strategies: Algorithms

Building on insights derived from our theoretical analysis, we propose a two-phase algorithm to achieve and implement near-optimal allocation strategies that maximize total effective throughput for multi-hop flows, while still supplying the basic fairness property. The first phase determines the allocation strategy for subflows at each of the nodes, i.e., $(\hat{r_1}, \ldots, \hat{r_n})$ for $(F_1, \ldots, F_n)$. We propose both centralized and distributed alternatives of the algorithm. In particular, the distributed form only depends on local information to approximate the achieved optimality of the centralized form, while still guaranteeing the basic fair share of each flow. The second phase of the algorithm is fully distributed, and seeks to implement the calculated allocation strategy for each of the subflows.

A. First Phase: The Centralized Form

To implement the centralized algorithm in the first phase, we need to assume a centralized node that processes per-flow information and constructs the weighted subflow contention graph. Though such a centralized node may not be achievable in ad hoc networks, it is feasible when considering a hybrid network that includes both ad hoc and infrastructure modes (e.g. base stations in cellular modes). In the latter case, we implement the centralized algorithm in the base station.

To assist centralized processing, each node collects information about outgoing subflows originating from itself, which is delivered to the centralized node. If a node is the source of a multi-hop flow F_i, it is able to calculate the virtual length, v_i, of F_i, from information derived from routing protocols (such as Dynamic Source Routing), or by a combination of (1) overhearing the existence of subflows from neighboring nodes; and (2) local information exchanges with neighboring nodes. Since end-to-end paths longer than 3 has a virtual length of 3, no further information beyond a two-hop neighborhood is required for determining v_i. Further, we assume that the source of a flow F_i possesses its weight w_i. After relevant information is collected and delivered, the centralized node may then construct the weighted subflow contention graph. By solving the linear programming problem presented previously (e.g. with the Simplex algorithm), the allocation strategy for each subflow may then be finalized. Finally, the centralized node broadcasts the allocation strategy to all nodes in the network.

B. First Phase: The Distributed Form

Obviously, a centralized algorithm is not suitable in wireless ad hoc networks. We propose an alternative, fully distributed algorithm. The algorithm depends on local flow information obtained by information exchange between neighboring nodes, and between neighboring upstream and downstream nodes in multi-hop flows. The overhead of such a scheme is minimal, since such information may be piggybacked with data packets or handshake control packets among the nodes on the path of flows. Each node determines the allocation strategies for local flows, based on the principles of optimizing the total effective throughput for all flows in the local neighborhood, while guaranteeing basic shares for these flows. We will show that such local optimization may generate a slightly higher basic share for flows, and the total effective throughput for the entire network is slightly lower than the centralized form of the algorithm.

For a particular node, the *local optimization* problem is to maximize the total effective throughput of multi-hop flows that may be overheard by the node itself (including information learned from neighboring nodes), while satisfying constraints with respect to local basic fairness (local counterpart of Eq. (7)) and local channel capacity B (Eq. (6)). The proposed distributed algorithm consists of the following steps.

Construction of local cliques: Each node is able to construct *local cliques* by (1) overhearing the existence of subflows within its transmission range (by overhearing both control packets such as RTS and CTS, or data packets); and (2) exchange overheard information with immediate neighbors. Previous work [5] has shown that, by exchanging subflow information with immediate neighbors, cliques involving only local subflows may be constructed. For space limitations, details of proving such feasibility are omitted in this paper.

Intra-flow exchange of constraints: Once local cliques have been identified, with available virtual lengths v_i of flows (obtained as in the centralized form), the local channel capacity constraint (Eq. (6)) and the local basic fairness constraint (Eq. (7)) are thus known locally. Since we require that subflows from the same multi-hop flow obtain equal allocations of the channel, a node needs to propagate locally obtained constraints to all of its upstream and downstream nodes along the same multi-hop flow. Such propagation of constraints may take the form of an array of coefficients and flow identifiers — $(n_{i,k}, i)$ — of a particular flow F_i. As such, each node along an end-to-end path may eventually possess all the constraints that include the corresponding flow. We observe that the constraints that a node constructs are a subset of *global* constraints, which may only be constructed given the complete subflow contention graph. This is realized in the centralized form of the algorithm, but impossible to achieve in the distributed form.

Achieving locally optimal allocation strategies: Each node (with local outgoing flows) uses information obtained from previous steps to construct a linear programming problem, the solution of which amounts to allocation strategies that are optimal locally.

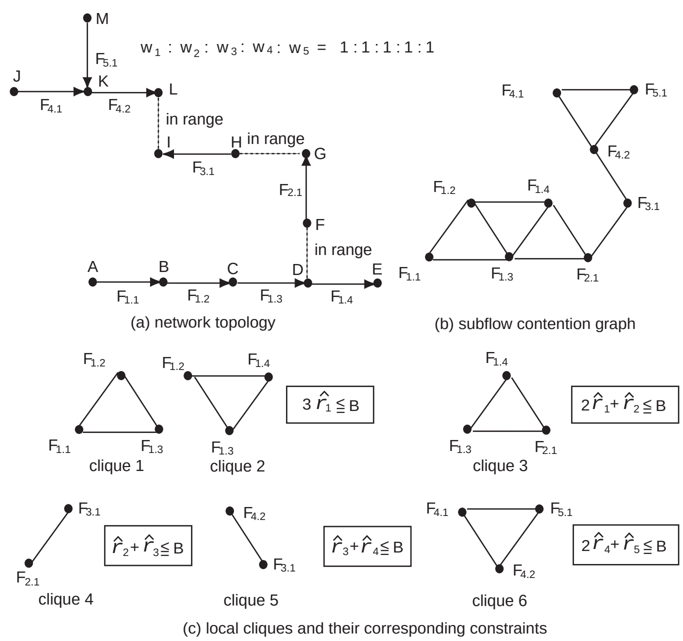


Fig. 6. First phase: the centralized and distributed form

We use an example topology shown in Fig. 6 to illustrate the details of the first phase of our proposed algorithm. The objective is to achieve optimal allocation strategies, using either the centralized or distributed alternatives of the algorithm. In its centralized form, the algorithm uses a centralized node to collect information from all nodes in the network, and achieves global optimality by solving the following problem:

maximize $\hat{r}_1 + \hat{r}_2 + \hat{r}_3 + \hat{r}_4 + \hat{r}_5$
subject to

$$\begin{aligned} 3\hat{r}_1 &\leq B \\ 2\hat{r}_1 + \hat{r}_2 &\leq B \\ \hat{r}_2 + \hat{r}_3 &\leq B \\ \hat{r}_3 + \hat{r}_4 &\leq B \\ 2\hat{r}_4 + \hat{r}_5 &\leq B \\ \hat{r}_1 &\geq B/8 \\ \hat{r}_2 &\geq B/8 \\ \hat{r}_3 &\geq B/8 \\ \hat{r}_4 &\geq B/8 \\ \hat{r}_5 &\geq B/8 \end{aligned}$$

The achieved optimal allocation strategy is thus $(\hat{r}_1, \hat{r}_2, \hat{r}_3, \hat{r}_4, \hat{r}_5) = (B/3, B/3, 2B/3, B/8, 3B/4)$.

In its distributed form, the algorithm, executed on each node, only has a partial view of the flow contention group, hence the constraints from locally constructed cliques are only a subset of those from the centralized algorithm. For example, node A is only aware of cliques Ω_1, Ω_2 and Ω_3, hence there are only two constraints associated with node A (Ω_1 and Ω_2 lead to the same constraint). Ω_3 is propagated from node D to other nodes on flow F_1. In addition, the basic share for each flow involved is higher than that in the centralized algorithm, since only part of the multi-hop flows are overheard by a node and included in the process of maximizing total effective throughput. Table I shows the local cliques on each node, and presents the local optimization problem as well as its solution.

C. Second Phase: Scheduling

So far, an optimal or near-optimal allocation strategy has been calculated. A scheduling algorithm is required to implement such a strategy by allocating the calculated shares to each flow. We note that the allocation strategy for the subflows has already attempted to maximizing spatial reuse of spectrum. It reflects the degree of contention among flows, so that flows with less contention in its neighborhood may enjoy a higher allocation of bandwidth. During scheduling, we propose to use the calculated allocation strategy (*allocated share*) as the new weights for the subflows. In the example topology shown in Fig. 4, the original subflow weights are $(F_{1.1}, F_{2.1}, F_{2.2}, F_{3.1}, F_{4.1}) = (1, 2, 2, 3, 2)$. However, to maximize total effective throughput with basic fairness guarantees, a better allocation strategy is $(r_{1.1}, r_{2.1}, r_{2.2}, r_{3.1}, r_{4.1}) = (3B/10, B/5, B/5, 3B/10, 7B/10)$, which may be obtained by the centralized first-phase algorithm solving the following problem:

maximize $\hat{r}_1 + \hat{r}_2 + \hat{r}_3 + \hat{r}_4$
subject to

$$\begin{aligned} \hat{r}_1 + 2\hat{r}_2 + \hat{r}_3 &\leq B \\ \hat{r}_3 + \hat{r}_4 &\leq B \\ \hat{r}_1 &\geq B/10 \\ \hat{r}_2 &\geq B/5 \\ \hat{r}_3 &\geq 3B/10 \\ \hat{r}_4 &\geq B/5 \end{aligned}$$

TABLE I
LOCAL OPTIMIZATION IN THE DISTRIBUTED ALGORITHM

Nodes	Local cliques	Constraints	$\hat{r_1}$	$\hat{r_2}$	$\hat{r_3}$	$\hat{r_4}$	$\hat{r_5}$
$A, B, C, D(F_1)$	$\Omega_1, \Omega_2, \Omega_3$	maximize $\hat{r_1} + \hat{r_2}$ subject to $3\hat{r_1} \leq B$ $2\hat{r_1} + \hat{r_2} \leq B$ $\hat{r_1} \geq B/3$ $\hat{r_2} \geq B/3$	**B/3**	B/3			
$F(F_2)$	Ω_3, Ω_4	maximize $\hat{r_1} + \hat{r_2} + \hat{r_3}$ subject to $2\hat{r_1} + \hat{r_2} \leq B$ $\hat{r_2} + \hat{r_3} \leq B$ $\hat{r_1} \geq B/5$ $\hat{r_2} \geq B/5$ $\hat{r_3} \geq B/5$	2B/5	**B/5**	4B/5		
$H(F_3)$	Ω_4, Ω_5	maximize $\hat{r_2} + \hat{r_3} + \hat{r_4}$ subject to $\hat{r_2} + \hat{r_3} \leq B$ $\hat{r_3} + \hat{r_4} \leq B$ $\hat{r_2} \geq B/4$ $\hat{r_3} \geq B/4$ $\hat{r_4} \geq B/4$		3B/4	**B/4**	3B/4	
$J, K, M(F_4, F_5)$	Ω_5, Ω_6	maximize $\hat{r_3} + \hat{r_4} + \hat{r_5}$ subject to $\hat{r_3} + \hat{r_4} \leq B$ $2\hat{r_4} + \hat{r_5} \leq B$ $\hat{r_3} \geq B/4$ $\hat{r_4} \geq B/4$ $\hat{r_5} \geq B/4$			3B/4	**B/4**	**B/2**

The allocated share of subflows may then become $(3B/10, B/5, B/5, 3B/10, 7B/10)$.

Due to lack of centralized coordination, scheduling packets for subflows from different nodes requires the following tasks to be implemented:

Intra-node coordinations: Packets from different subflows are queued separately; yet the scheduling algorithm needs to select the next packet to transmit from the head of the queues, obeying the allocated share. For example, at node A in Fig. 4, two queues exist for $F_{1.1}$ and $F_{2.1}$, respectively. The packet selection process needs to guarantee that the ratio of transmissions for $F_{1.1}$ and $F_{2.1}$ should be $3/10 : 1/5$.

Inter-node coordinations: The scheduling algorithm on each node needs to determine the backoff timer for its ready-to-send packets, in order to coordinate with other nodes. If we think of all the subflows on one node as one virtual flow, while the aggregated allocated share of the subflows as the virtual flow share (or, *node share*), each node will try to determine an appropriate backoff timer that is inversely proportional to the virtual flow share on the node. In other words, nodes need to collectively adjust their contention window to be proportional to their own node share. In the example of Fig. 4, node A needs to adjust its contention windows to properly estimate its node share ($F_{1.1}+F_{2.1} = B/2$) compared to other neighboring nodes such as B (its node share being $F_{2.2} + F_{3.1} = B/2$).

We present the details of our scheduling algorithm as follows, and evaluate its performance in Sec. V. The transmission of data packets follows the standard RTS-CTS-DATA-ACK handshaking protocol as in IEEE 802.11 [6], [7] to acquire the floor. Each node, i, is required to maintain a virtual clock, $v_i(t)$, as well as a *local table* to keep track of the service tags of all its one-hop neighbor subflows. In order to maintain the local table, the RTS, CTS and ACK packets are used to piggyback the new service tag of the currently transmitting data packet. Any neighbors on hearing the tag will update its own local table.

Assume that a node i has J subflows with their corresponding queues denoted by queue(i, j), and the allocated shares for each subflow at node i denoted by c_i^j, $j = 1, 2, \ldots, J$. In addition, the node share is denoted by $c_i = \sum_{j=1}^{J} c_i^j$, *i.e.*, it is the sum of allocated shares of subflows that originate from this node.

In this setting, our scheduling algorithm is outlined as follows:

(1) When a packet arrives at node i, it enqueues in its own subflow queue;

(2) When a packet $P_i^{j,k}$ (the kth packet for subflow j) with size $L_i^{j,k}$ reaches the head of its queue, three tags are assigned:

- *Start tag:* $S_i^{j,k} = v_i(t_i^{j,k})$, $t_i^{j,k}$ being the real time when the packet k of subflow j reaches the head of its queue;
- *Internal finish tag:* $I_i^{j,k} = S_i^{j,k} + L_i^{j,k}/c_i^j$. This tag is used to determine the next-to-send packet;
- *External finish tag:* $E_i^{j,k} = S_i^{j,k} + L_i^{j,k}/c_i$. This tag is used to determine the contention backoff timer. Note that we use the node share (not the allocated share for each subflow) in the tag calculation.

(3) The sender i will estimate an approximate backoff value Q for $P_i^{j,k}$ in its local table which is defined as $Q = \sum_{m \in T}(S_i^{j,k} - r_m) \cdot \alpha$ for its subflows, where T includes all the subflows stored in the local table, r_m is the start tag of subflow from node m in the local table, and α is a tunable parameter to decide the strictness of short-term fairness. On the other hand, the receiver of the

data packet also estimates a backoff value, R, in its table and carry this information in the ACK packet to the sender for future packet scheduling. We define $R = \sum_{m \in T, m \neq i}(r_i - r_m) \cdot \alpha$. The actual backoff timer to be set for $P_i^{j,k}$ at node i is uniformly distributed in $[0, \mathrm{CW}_{min} + \max(Q, R, 0)]$, where CW_{min} is the minimum contention window.

(4) When a packet is successfully sent, sender i will update its virtual clock as the external finish tag of the previous packet. The scheduling algorithm then selects the packet from all the head-of-line packets of the queues that have the smallest internal finish tag. The backoff timer is then reset.

In the algorithm, we use the internal finish tag to determine the packet to send locally, which is calculated based on the allocated share of the subflow. We then use the external finish tag to determine the backoff interval using the node share. The probability of buffer overflow is low for a multi-hop flow using our algorithm, since subflows from the same flow will receive approximately the same channel share.

V. Performance Evaluation

We have implemented our two-phase algorithm (referred to as *2PA* in this section) within the ns-2 2.1b8a network simulator, and have performed simulations to evaluate the effectiveness of our proposed algorithm. We present simulation results in two network scenarios: (1) a simpler topology shown in Fig. 1; and (2) a more elaborate topology shown in Fig. 6. With respect to these two example scenarios, we compare the performance of 2PA with (1) standard IEEE 802.11 MAC [7]; and (2) the *two-tier fair scheduling algorithm* (abbreviated as *two-tier*) in previous work [1], which guarantees basic fairness among single-hop flows and maximizes single-hop total effective throughput.

We adopt the ns-2 standard physical layer implementation with a channel capacity of 2Mbps with Two Ray Ground Reflection as the propagation model. Each node is assigned a maximum transmission range and an interference range of 250 meters. We use Dynamic Source Routing (DSR) as the routing protocol. Data at source nodes are generated at a constant bit rate (CBR) of 200 packets per second with a packet size of 512 bytes. In order to perform side-by-side comparisons of our proposed algorithm (2PA) with IEEE 802.11 MAC and two-tier, we use identical weights of 1 for each flow (both multi-hop flows and its subflows) for both simulation scenarios. The length of each simulation session is $T = 1000$ seconds. We are interested in evaluating the following parameters: (1) During the entire simulation session, the number of packets successfully delivered for each of the flows (including subflows). This is to evaluate the allocated share to each of the flows and subflows (*i.e.*, $\hat{r_i} \cdot T$ and $r_{i,j} \cdot T$). (2) The total number of successfully delivered packets during the entire simulation session. This is to evaluate the extent of spatial spectrum reuse, *i.e.*, the total effective throughput $\sum_{i=1}^{n} \hat{r_i} \cdot T$. (3) the total number of packets lost, as well as the packet loss ratio. This is to evaluate the unevenness among subflows belonging to the same multi-hop flow. Packets delivered in upstream subflows and then dropped in downstream subflows represents an allocation strategy that is not optimal or adequate.

CW_{min} and α are set to 31 and 0.0001 respectively.

A. Scenario 1

We use the network topology in Fig. 1 in this scenario. The simulation results are shown in Table II.

TABLE II

Simulation results, topology as in Fig. 1

Parameters	802.11	two-tier	2PA
$r_{1.1}\,T$	16079	66658	111773
$r_{1.2}\,T\ (\hat{r_1}T)$	952	60992	111084
$r_{2.1}\,T$	156517	65507	56404
$r_{2.2}\,T\ (\hat{r_2}T)$	151533	65507	56404
$\sum_{i=1}^{2} \hat{r_i} \cdot T$	152485	126499	167488
lost packets	20111	5666	689
loss ratio	0.132	0.045	0.004

We may observe that, the throughput ratio among the subflows of 2PA approximates the allocated share calculated in the first phase, i.e., $r_{1.1} : r_{1.2} : r_{2.1} : r_{2.2} \approx 1/2 : 1/2 : 1/4 : 1/4$. In addition, 2PA achieves a higher total effective throughput, surpassing both IEEE 802.11 and two-tier in this scenario. Since IEEE 802.11 MAC protocol does not consider the fair allocation of bandwidth among subflows, a higher packet loss ratio occurs in between $F_{1.1}$ and $F_{1.2}$ due to the fact that the contending subflows $F_{2.1}$ and $F_{2.2}$ always occupy the channel. Two-tier attempts to schedule single-hop subflows so that channel is shared fairly, and to take advantage of the spatial bandwidth reuse when the subflow receiving the least service has started to transmit. However, since such a scheduling decision allocates more shares for $F_{1.1}$ than $F_{1.2}$, buffer overflow occurs on node B. This leads to the fact that while 2PA has lost only 689 packets, two-tier has lost ten times more. These results demonstrate that considering spatial spectrum reuse from an end-to-end perspective is not only beneficial, but also essential.

B. Scenario 2

The second scenario shown in Fig. 6 is more complex with longer multi-hop flows, and local nodes may not have complete flow contention information about all the multi-hop flows in its flow contention group. The calculated allocated share $(\hat{r_1}, \hat{r_2}, \hat{r_3}, \hat{r_4}, \hat{r_5})$ in the centralized form (2PA-C) and distributed form (2PA-D) of the first phase are $(1/3, 1/3, 2/3, 1/8, 3/4)$ and $(1/3, 1/5, 1/4, 1/4, 1/2)$, respectively, with the assumption that all flows have the same weight 1. The channel allocation in centralized and distributed forms are quite different due to the fact that the local optimization problem contains only part of the constraints that are in global optimization. For example, node F is only aware of cliques Ω_3 and Ω_4 that include F_2 (as shown in Table I). Other flows involved in the local optimization such as F_1, F_3 may be subject to other constraints that are not aware of by node F. As a result, the channel allocation for F_2 $(1/5)$ in the distributed form is lower than that from a global optimization perspective $(1/3)$. Similar arguments apply to F_3 and F_5 as well.

Table III shows the results using both the centralized and distributed first-phase algorithms. The effectiveness of the second phase scheduling algorithm in 2PA ensures that the throughput of each flow under 2PA is proportional to their allocated shares. Since the centralized form uses global optimization, it generally has a higher total effective throughput than that obtained with the

TABLE III

SIMULATION RESULTS, TOPOLOGY AS IN FIG. 6

Parameters	802.11	two-tier	2PA-C	2PA-D
$r_{1.1}\,T$	72150	49551	53992	67381
$r_{1.2}\,T$	53590	41731	53745	67189
$r_{1.3}\,T$	53127	39574	52955	67189
$r_{1.4}\,T$ ($\hat{r_1}T$)	53127	39574	52955	67189
$r_{2.1}\,T$ ($\hat{r_2}T$)	8345	14802	54694	42457
$r_{3.1}\,T$ ($\hat{r_3}T$)	197911	163809	112520	57321
$r_{4.1}\,T$	49966	18865	29365	62036
$r_{4.2}\,T$ ($\hat{r_4}T$)	24495	18053	28022	60855
$r_{5.1}\,T$ ($\hat{r_5}T$)	159326	157887	173971	124520
$\sum_{i=1}^{5}\hat{r_i}\cdot T$	443204	394125	422162	352341
lost packets	44494	10789	2380	1374
loss ratio	0.100	0.027	0.006	0.004

distributed form. Unlike 802.11, $F_{2.1}$ under 2PA is able to obtain a fair share of the channel since 2PA restricts $F_{3.1}$ so that it does not utilize the channel too aggressively. With respect to the total effective throughput, though 2PA-C surpasses two-tier, we note that due to the lack of constraints in the local optimization, the allocation strategy for F_2, F_3 and F_5 in 2PA-D is not as optimal. Since the topology is particularly tuned to show the differences between 2PA-C and 2PA-D, the results are expected. Further, we note that, with respect to throughput, the results of 2PA-D is *not comparable* with that of two-tier, since the latter is a centralized algorithm, and the former is a fully distributed algorithm. Finally, we observe that the packet loss ratio under 2PA is minimal compared with that under 802.11 and two-tier.

VI. RELATED WORK

Huang *et al.* [5] focuses on providing max-min fair scheduling. Rather than assigning weights in advance to flows, it attempts to compute the appropriate bandwidth share for each flow based on its surroundings. Though the approach is similar to ours, there exist two major differences: (1) as in other previous work, [5] only considers single-hop independent flows, while we consider multi-hop flows from an end-to-end perspective; (2) there are no pre-assigned weights for flows in [5], while our work takes pre-assigned weights for each flow into consideration. Huang *et al.* implicitly acknowledges that contending flows have the same weight, and depending on the contention situation, each flow will receive its appropriate share without any flow starvation. In our case, by using the preassigned weights as a guideline for the scheduling policy, we allocate basic fair shares for each flow, and attempt to maximize the total effective throughput at the same time. Similar to [5], Tassiulas *et al.* [8] also concentrates on providing max-min fair scheduling in wireless networks. However, the single-hop contention constraint are different from ours in that it assumes that any two single-hop flows not sharing a node can transmit packets simultaneously, whereas in our model, any two single-hop flows within two hops are contending with each other.

Nandagopal *et al.* [9] designs a general analytical framework and mechanism for arbitrarily specified fairness model via a utility function, and concentrates on achieving such a given fairness model by an appropriate MAC layer design. Vaidya *et al.* [10] provides a distributed fair scheduling algorithm for wireless LANs, and proposes several mapping techniques for better determination of backoff timers. Our second-phase algorithm also attempts to improve fairness by tuning the backoff timers. The work by Kanodia *et al.* [11] may be the most similar to our approach. It proposes a scheduling scheme referred to as *multi-hop coordinated scheduling* to provide better QoS guarantees regarding end-to-end delays of multi-hop flows. Similarly, we also treat each multi-hop flows from an end-to-end perspective and emphasize the coordination between upstream and downstream subflows. However, we do not discuss end-to-end delays or propose algorithms to address these issues, our objective is different in that we focus on allocating spatially reused bandwidth among multi-hop flows from an end-to-end perspective, thus improving the overall channel bandwidth utilization in the network.

VII. CONCLUSIONS

In this paper, we have extensively studied the issue of end-to-end fairness in wireless ad hoc networks. Unlike previous works that break a multi-hop flow into multiple single-hop flows, we analyze the issue of increasing the spatial reuse of bandwidth from an end-to-end prospective, and propose estimation algorithms that satisfy the fairness constraint and the basic fairness constraint. A two-phase algorithm is presented to approximate the optimal allocation strategies, building upon results of theoretical analysis. Results of performance evaluation in ns-2 have demonstrated the effectiveness of our algorithm compared to the two-tier fair scheduling algorithm and the IEEE 802.11 MAC protocol, To the best of our knowledge, there exists no previous work that examines similar problems *from an end-to-end perspective* of multi-hop flows.

REFERENCES

[1] H. Luo, S. Lu, and V. Bharghavan, “A New Model for Packet Scheduling in Multihop Wireless Networks,” in *Proceedings of ACM MobiCom*, 2000, pp. 76–86.

[2] H. Luo and S. Lu, “A Topology-Indepent Fair Queueing Model in Ad Hoc Wireless Networks,” in *Proceedings of IEEE International Conference on Network Protocols*, 2000.

[3] H. Luo, P. Medvedev, J. Cheng, and S. Lu, “Self-coordinating Localized Fair Queueing in Wireless Ad Hoc Networks,” *IEEE Trans. on Mobile Computing*, vol. 3, no. 1, pp. 86–98, 2004.

[4] J. Li, C. Blake, D. S.J. De Couto, H. I. Lee, and R. Morris, “Capacity of Ad Hoc Wireless Networks,” in *Proceedings of ACM Mobicom*, 2001.

[5] X. L. Huang and B. Bensaou, “On max-min fairness and scheduling in wireless ad-hoc networks: analytical framework and implementation,” in *Proceedings of ACM Mobihoc*, 2001.

[6] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, “MACAW: A media access protocol for wireless lan’s,” in *Proceedings of ACM SIGCOMM*, 1994.

[7] “IEEE 802.11 Standard,” *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, 1999.

[8] L. Tassiulas and S. Sarkar, “Maxmin Fair Scheduling in Wireless Networks,” in *Proceedings of IEEE INFOCOM*, 2002.

[9] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan, “Achieving MAC Layer Fairness in Wireless Packet Networks,” in *Proceedings of ACM Mobicom*, 2000.

[10] N. H. Vaidya, P. Bahl, and S. Gupta, “Distributed Fair Scheduling in a Wireless LAN,” in *Proceedings of ACM Mobicom*, 2000.

[11] V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, and E. Knightly, “Distributed Multi-Hop Scheduling and Medium Access with Delay and Throughput Constraints,” in *Proceedings of ACM MobiCom*, 2001.