On the Complexity of System Throughput Derivation for Static 802.11 Networks

Gene Cheung, *Senior Member, IEEE,* Jeongkeun Lee, *Member, IEEE,* Sung-Ju Lee, *Senior Member, IEEE,* and Puneet Sharma, *Senior Member, IEEE*

***Abstract*—The exploding popularity of 802.11 Wireless Local Area Networks (WLAN) has drawn intense research interest in the optimization of WLAN performance through channel assignment to access points (AP), AP-client association control, and transmission scheduling—we refer to any combination of the three approaches as WLAN management. No matter what degrees of freedom are enabled in WLAN management for performance optimization in a particular WLAN setting, a fundamental question is the corresponding maximum achievable system throughput. We show that for a particular network setting, the derivation of the system throughput (where system throughput is aggregate throughput of all clients or max-min throughput), for any combination of channel assignment, association control and transmission scheduling, is NP-hard and inapproximable to a constant factor in polynomial time.**

I. Introduction

With increasing popularity of 802.11 Wireless Local Area Networks (WLAN), extensive research into optimization of WLAN system performance is being conducted. A key challenge in WLAN optimization is to properly account for signal interference, stemming from simultaneous wireless transmissions by different entities in the same channel. Broadly speaking, there are three mechanisms to mitigate interference: channel assignment to access points (AP) [1], AP-client association control [2], and transmission scheduling [3]. Depending on a particular network setting, an optimizer may perform any combination of the three—together we simply call *WLAN management*—to maximize a chosen system throughput objective function.

No matter what particular WLAN management an optimizer is to perform, a fundamental question is the maximum achievable system throughput for a given *static* network setup: namely, placements of APs and clients and their communication and interference relations. Not only can the maximum throughput serve as an upper bound against which different WLAN management schemes can be compared, it can also be used to evaluate the network setup itself, so that the setup can be recasted (e.g., relocation of APs) to increase throughput.

In this paper we prove, for our chosen network model, the following negative result: for the special case when there is only one channel and each client can only associate to one particular AP, the derivation of 802.11 system throughput, where throughput can be either *aggregate throughput* of all clients or *max-min throughput*, is NP-hard and inapproximable to a constant factor in polynomial time. As a corollary, the derivation of throughput for the general case when there are more than one channel and a client can associate to one of several APs is also NP-hard and inapproximable to a constant factor in polynomial time. Compared to previous WLAN throughput analysis, our work differs from [4] in that optimal centralized scheduling is considered for *multiple* APs when the system throughput is sought, and differs from [5] in that a graph-based interference model is adopted rather than distance-based models. Moreover, there is no discussion on the complexity of deriving optimal system throughput.

Compared to the majority of previous works studying complexity and algorithms for interference-aware wireless networking, we focus on AP-client communication in WLAN only, which is *single-hop*, while [3], [6], [7] discussed the more general *multi-hop* case. While the general multi-hop problem includes single-hop as a special case, from an algorithmic point of view, there is hope that the special case of deriving system throughput for single-hop network can be simpler in complexity. Our contribution is to demonstrate that even in the special case of single-hop network, deriving system throughput is NP-hard and inapproximable in polynomial time.

II. Model and Assumptions

We consider a set of WLAN APs, each with its own exclusive group of associated client stations (STA), all managed by a single administrative domain. (See Fig. 1 for an example.) An ideal optimizer schedules interference-free transmissions for each AP-STA pair in normalized time $0 \leq t \leq 1$ to maximize a chosen performance metric (to be defined in Section III), taking into account all STA-AP, STA-STA and AP-AP interference relationships. Finding the ideal schedule is equivalent to deriving the system throughput for a given network setup.

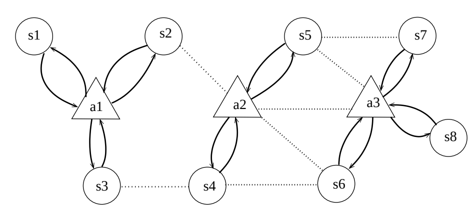

Fig. 1. Example of WLAN graph $G(\mathcal{V};\mathcal{L},\mathcal{E})$. Triangles are APs; circles are STAs; arrows are communication links; dotted lines are interference edges.

More precisely, input to an optimizer is a *WLAN graph* $G(\mathcal{V};\mathcal{L},\mathcal{E})$, containing: i) vertices $\mathcal{V} = \mathcal{A} \cup \mathcal{S}$ of APs $\mathcal{A}$ and STAs $\mathcal{S}$, ii) directed *communication links* $\mathcal{L}$ connecting each AP to its associated STAs, and iii) vertex-to-vertex *interference edges* $\mathcal{E}$. An AP a_i and any one of its associated STAs s_j share two communication links $l_{i,j}$ and $l_{j,i}$; activation of $l_{i,j}$ means data is transmitted from vertex v_i to vertex v_j. A bi-directional edge[1] $e_{i,j}$ between vertices v_i and v_j (between APs, between STAs associated with different APs, or between a STA and an AP different from its associated AP) implies that one vertex's transmission can interfere with another vertex's transmission or reception. v_i and v_j can receive at the same time, however.

An optimizer schedules transmissions of each communication link. An *ideal schedule* $\pi_{i,j}$ for each link $l_{i,j}$, is given in the form of a set of $T_{i,j}$ tuples:

$$\pi_{i,j} = \{[s_{i,j}(1), t_{i,j}(1)), \ \ldots, \ [s_{i,j}(T_{i,j}), t_{i,j}(T_{i,j}))\} \quad (1)$$

where for each tuple k, $[s_{i,j}(k), t_{i,j}(k))$, link $l_{i,j}$ becomes active (v_i transmits) from time $s_{i,j}(k)$ to $t_{i,j}(k)$. Without loss of generality, we assume the tuples are non-overlapping and ordered in time, i.e., $s_{i,j}(k) < t_{i,j}(k) \leq s_{i,j}(k+1)$. In addition, we assume that the ideal schedule is normalized to 1, i.e., $0 \leq s_{i,j}(1)$ and $t_{i,j}(T_{i,j}) \leq 1$.

Mathematically, we say that two schedules $\pi_{i,j}$ and $\pi_{i',j'}$ overlap if the following is true:

$$\begin{aligned} \exists k, k' \quad \text{s.t.} \quad & s_{i',j'}(k') \leq s_{i,j}(k) < t_{i',j'}(k') \quad \text{or} \\ & s_{i',j'}(k') < t_{i,j}(k) \leq t_{i',j'}(k') \\ \exists k, k' \quad \text{s.t.} \quad & s_{i,j}(k) \leq s_{i',j'}(k') < t_{i,j}(k) \quad \text{or} \\ & s_{i,j}(k) < t_{i',j'}(k') \leq t_{i,j}(k) \end{aligned} \quad (2)$$

III. Problem Formulations

A. Client Throughput Model

Let d_i be the *transmission and reception duration* of a STA v_i. STA v_i shares two communication links, $l_{i,j}$ and $l_{j,i}$, with its associated AP v_j. Mathematically, we calculate d_i using transmission schedules $\pi_{i,j}$ and $\pi_{j,i}$ as follows:

$$d_i = \sum_{k=1}^{T_{i,j}} (t_{i,j}(k) - s_{i,j}(k)) + \sum_{k=1}^{T_{j,i}} (t_{j,i}(k) - s_{j,i}(k)) \quad (3)$$

We use d_i as a metric to quantify client throughput (uplink plus downlink) for STA $v_i \in \mathcal{S}$. An alternative definition of client throughput d_i is the volume of uplink traffic only, derived using schedule $\pi_{i,j}$. One can easily verify that our claim of NP-hardness and inapproximability for derivation of aggregate client throughput and max-min client throughput holds equally true for this alternative definition of client throughput using the same proofs in Section IV.

B. Objective Functions

The goal of an optimizer is to find interference-free transmission schedules that maximize a given performance objective. We consider two objectives: maximize aggregate client throughput (`MaxSum`), and maximize the minimum client throughput (`MaxMin`). `MaxSum`, maximizing the sum of throughput of all STAs in the network, can be written simply:

$$\max \sum_{\forall v_i \in \mathcal{S}} d_i \quad (4)$$

Taking fairness into consideration, `MaxMin` identifies the minimum throughput STA and maximizes its performance:

$$\max \left\{ \min_{\forall v_i \in \mathcal{S}} d_i \right\} \quad (5)$$

[1]The interference relation can be derived by a measurement-based estimation [8] or by a distance-based interference model [5].

IV. NP-Hardness Proofs

A. Maximizing Aggregate Throughput

We show that `MaxSum` (4) is NP-hard via a reduction from a known NP-hard problem *independent set* (`IS`). `IS` optimization problem is to find the largest subset of nodes $U' \subset U$ in an undirected graph $Q = (U, E)$ such that there does not exist an edge $e_{i,j} \in E$ between any two nodes u_i, u_j in U', and the cardinality of $|U'|$ is maximized.

We show that solving `MaxSum` is equivalent to solving `IS`, hence `MaxSum` is also NP-hard. For each instance of `IS`, we construct an instance of `MaxSum` as follows. First, the set of nodes U in `IS` will be reused as the set of STAs $\mathcal{S}$ in `MaxSum`. Second, each STA will have its own AP serving it and shared by no other STAs, hence $|\mathcal{S}| = |\mathcal{A}|$, and there exist interference edges from each AP to all other APs. Third, the same set of edges E among nodes in `IS` will be reused as interference edges among STAs in `MaxSum`.

We claim that there exists an independent set of size K in an instance of `IS` if and only if there is an aggregate throughput of K in a corresponding instance in `MaxSum`, hence solving `MaxSum` is equivalent to `IS`. We show this in both directions. If there is an independent set of size K in `IS`, then the same corresponding set of STAs in `MaxSum` can transmit simultaneously for all time $0 \leq t \leq 1$ without interference, resulting in an aggregate throughput of K.

Conversely, if there is an aggregate throughput of K in `MaxSum`, we know that there exists an instant t, $0 \leq t \leq 1$, where there are at least K simultaneously transmitting vertices (otherwise throughput of K cannot be achieved). At such instant t, the K transmitting vertices must be K non-interfering STAs (a single transmitting AP will prevent all other APs from transmitting or receiving data). Hence the K nodes in `IS` corresponding to the K simultaneously transmitting STAs will form a size-K independent set.

Since both directions have been proven, we have shown that solving `MaxSum` is equivalent to solving `IS`, and hence `MaxSum` is NP-hard. □

B. Maximizing Minimum Client Throughput

We show that `MaxMin` (5) is NP-hard via a reduction from a known NP-hard problem K*-coloring* (`COL`). Recall that `COL` optimization problem is to find the smallest number K of distinct colors required to color each node in an undirected graph $Q = (U, E)$ so that no two nodes $u_i, u_j \in U$ connected by an edge $e_{i,j} \in E$ are of the same color.

We show that solving `MaxMin` is equivalent to solving `COL`, hence `MaxMin` is also NP-hard. For each instance of `COL`, we construct a corresponding instance of `MaxMin` as follows. First, the set of nodes U in `COL` will be reused as the set of STAs $\mathcal{S}$ in `MaxMin`. Second, the set of edges E connecting nodes in `COL` will be reused as the set of interference edges connecting corresponding STAs in `MaxMin`. Third, each STA will have its own AP for communication, and there exist interference edges from each AP to all other APs.

We claim that graph Q of `COL` is K-colorable if and only if the corresponding `MaxMin` instance has max-min throughput of $\geq 1/K$. We prove this claim for both directions. It is clear that if Q is K-colorable, then STAs of `MaxMin` corresponding to nodes of the same color in Q can be scheduled for transmission simultaneously without causing interference. If each set of STAs of `MaxMin` corresponding to the same-color nodes in `COL` are scheduled for transmission in turn for the same duration each, then the throughput of each STA is exactly $1/K$, and the max-min throughput of `MaxMin` is also $1/K$.

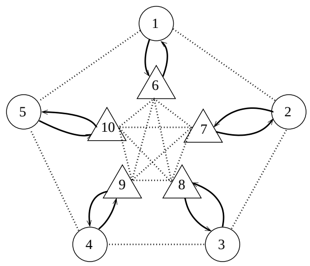

Fig. 2. Example of an `MaxMin` instance constructed from an `COL` instance.

We now show that if the max-min throughput of the corresponding instance in `MaxMin` is $\geq 1/K$, then original Q of `COL` is K-colorable. Without loss of generality, we first define a schedule $\pi = \{C_1, C_2, \dots, C_B\}$ that results in max-min throughput for the given `MaxMin` instance. B is the number of STA sets C_i's in schedule π, where each C_i is scheduled for transmission in turn for the same duration $1/B$. Let M be the minimum number of times any STA appears as a transmitter in sets C_i's of schedule π. The max-min throughput using schedule π is hence M/B. Let K be the smallest integer such that:

$$\frac{1}{K} \leq \frac{M}{B} \tag{6}$$

As an example, consider an `MaxMin` instance in Fig. 2 with five STAs and five associated APs. One optimal transmission schedule π would be $\{\{l_{1,6}, l_{3,8}\}, \{l_{2,7}, l_{4,9}\}, \{l_{3,8}, l_{5,10}\}, \{l_{4,9}, l_{1,6}\}, \{l_{5,10}, l_{2,7}\}\}$. It is easy to see that $B = 5$ and $M = 2$ and max-min throughput is $2/5$. Smallest integer K satisfying (6) is 3.

We prove by contradiction. Suppose schedule π has max-min throughput of M/B and K such that $1/K \leq M/B$, but graph Q of `COL` is not K-colorable. Q being not K-colorable means there are $\geq K+1$ nodes in Q. Let us first consider the special case when the number of nodes in Q is exactly $K+1$. Having $K+1$ nodes in Q and Q not K-colorable means $K+1$ nodes form a clique. We know each STA must appear in the schedule π at least M times. But each STA cannot appear in a same set C_i with any other STA, because corresponding nodes in Q forming a clique means each STA interferes with every other STA. Thus, at least $(K+1)M$ sets C_i's are needed, but at most MK sets are available. A contradiction.

Consider now the general case where the number of nodes in Q is K^v and the chromatic number (minimum number of colors to color a graph) of Q is K^c, where $K^v \geq K^c \geq K+1$. We can transpose the original graph Q to a new graph Q^c by mapping all same-color nodes in Q to a single node of the same color in Q^c; all edges shared by the same-color nodes in Q will now be shared by a single node in Q^c. Note that Q^c has K^c nodes, each of a different color, and forms a clique.

Similarly, we can map STAs in G corresponding to these same-color nodes in Q to a single STA in G^c. Schedule π for G can be transposed to a new schedule π^c for G^c, where it contains the same number B of sets C_i's, but the previous schedule for STAs corresponding to the same color in Q are now for a single STA. Clearly, the minimum number of times any STA appears in set C_i's is now $M^c \geq M$. Now we see the same contradiction we countered in the special case: each of $K^c \geq K+1$ STAs must appear at least M times alone in schedule π^c (given corresponding nodes in Q^c form a clique), but there are only at most MK STA sets. Having shown there is a contradiction in the general case, we have shown that if the max-min throughput of the corresponding instance in `MaxMin` is at least $1/K$, then original Q of `COL` is K-colorable.

Since both directions are proven, solving `MaxMin` is equivalent to solving `COL`, hence `MaxMin` is NP-hard. □

V. Practical Implications

We discuss the implications of having shown `MaxSum` and `MaxMin` are both NP-hard. First, it has been shown that both `IS` [9] and `COL` [10] are both inapproximable in polynomial time to a constant factor. Given solving `MaxSum` and `MaxMin` is equivalent to solving `IS` and `COL`, we conclude that `MaxSum` and `MaxMin` are inapproximable in polynomial time to a constant factor. Notice that we formulated our optimization problems `MaxSum` and `MaxMin` for the special case when there is only one channel and each STA can only associate to one particular AP. Given deriving the system throughput for the special case is already NP-hard and inapproximable, deriving the system throughput for the general case when there are more than one channel and each STA can associate to one of several APs within range (which must be no easier) is also NP-hard and inapproximable. For future work, we are designing heuristics exploiting clues provided by the constructed NP-hardness proofs that indicate the roots of difficulty of the sought optimization problem.

References

[1] A. Mishra et al., "A Client-driven Approach for Channel Management in Wireless LANs," in *IEEE INFOCOM*, Barcelona, Spain, Apr. 2006.

[2] Y. Bejerano, S.J. Han, and L. Li, "Fairness and Load Balancing in Wireless LANs using Association Control," in *ACM MobiCom*, Portland, OR, Aug. 2004.

[3] W. Wang et al., "Efficient Interference-Aware TDMA Link Scheduling for Static Wireless Networks," in *ACM MobiCom*, Los Angeles, CA, Sept. 2006.

[4] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," in *IEEE Journal on Selected Areas in Communications*, Mar. 2000, vol. 18, no.3.

[5] P. Gupta and P. R. Kumar, "The capacity of wireless networks," in *IEEE Transactions on Information Theory*, Mar. 2000, vol. 46, no.2.

[6] M. Kodialam and T. Nandagopal, "Characterizing Achievable Rates in Multi-hop Wireless Networks: The Joint Routing and Scheduling Problem," in *IEEE INFOCOM*, San Diego, CA, Sept. 2003.

[7] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *ACM MobiCom*, New York, NY, Sept. 2003.

[8] J. Lee et al., "RSS-based Carrier Sensing and Interference Estimation in 802.11 Wireless Networks," in *IEEE SECON*, San Diego, CA, June 2007.

[9] J. Hastad, "Clique is hard to approximate within $n^{1-\epsilon}$," in *Acta Mathematica*, 1999, vol. 182, pp. 105–142.

[10] U. Feige and J. Kilian, "Zero Knowledge and the Chromatic Number," in *Journal of Computer and System Sciences*, 1998, vol. 57, no.2, pp. 187–199.