A Novel On-Line Association Algorithm in Multiple-AP Wireless LAN

Liang Sun, Lei Wang, Zhenquan Qin(✉), Zehao Ma, and Zhuxiu Yuan

School of Software, Dalian University of Technology, Dalian, China
{liang.sun,lei.wang,qzq}@dlut.edu.cn, johnmazehao@gmail.com,
zhuxiu.yuan@gmail.com

Abstract. Nowadays, wireless LAN has become the most widely deployed technology in mobile devices for providing Internet access. As a result, WLAN users usually find themselves covered by multiple access points and have to decide which one to associate with. In traditional implementations, most wireless stations would select the access point with the strongest signal, regardless of traffic load on that access point, which might result in heavy congestion and unfair load. In this paper, we propose a novel on-line association algorithm to deal with any sequence of STAs during a long-term time such as one day.

1 Introduction

Wireless local area networks (WLANs) have become a popular technology for access to the Internet and enterprise networks. In conventional implementations of WLANs, each station (STA) scans multiple wireless channels to detect the APs within the communication range, and chooses an AP that has the strongest received signal strength indicator (RSSI). The most apparent disadvantage of the RSSI-based STA association approach is that RSSI does not provide any information about the current traffic load of the AP [5]. Thus how to select an AP in a WLAN to guarantee high throughput and balance load for each STA is a challenging issue [6].

Obviously, the association algorithm can be used to achieve different objectives. For instance, it can be used to maximize the overall throughput of a system [9], achieve the network-wide bandwidth allocation fairness among STAs [4], and balance the load among APs. These plausible objectives can be obtained by one or two of the following parameters: the bit rate served by APs and the utilization of APs. Though these plausible objectives can be obtained by the two parameters by the periodical off-line optimal solutions, these are not desired feasible association algorithm which could be implemented in real-world situations. Consequently, more feasible association algorithms could deal with a sequence of STAs and maximize the overall traffic of a system.

The rest of the paper is organized as follows. In Sect. 2, we discuss the related work. In Sect. 3, we describe the system model considered in the paper. Section 4 gives the problem formulation of the AP association in WLANs and presents the

L. Ma et al. (Eds.): WASA 2017, LNCS 10251, pp. 890–902, 2017.
DOI: 10.1007/978-3-319-60033-8_78

algorithm, followed by Sect. 5 which analyzes the proposed algorithm. We report the simulation and experimental results in Sect. 6. Finally, we conclude the paper in Sect. 7.

2 Related Work

Association algorithms for WLANs have been intensely studied by both the research community and the industry. Fairness and load balancing are two interrelated dimensions of the AP association problem.

None of the works mentioned above jointly considered the two interrelated dimensions in wireless LANs. Le *et al.* [7] proposed a distributed algorithm where each STA selected an appropriate contention window size to fairly share the channel occupancy time while maximizing the aggregated throughput of the network. Throughput based max-min fairness suffered from low network throughput in multi-rate wireless LANs [3].

Here we discuss a completely on-line algorithm with no need to run again when an STA is arriving or leaving. Actually, it is difficult to give consideration to the two interrelated dimensions at the same time. And there are many mechanisms to guarantee the quality of bandwidth allocated to the STAs [10]. Therefore, when we design an on-line association algorithm which could guarantee the quality of bandwidth of the associated STAs, we just consider the AP selection and the load balancing issues.

Mehta *et al.* [8] provided a simple framework to design a trade-off function between two factors in the on-line algorithm in. The paper proposed an on-line algorithm with competitive ratio $1 - 1/e$ to solve the AdWords problem among the Internet search engine companies. Based on the solution of the paper, we find that the capacity of each AP is limited, and the arrival and demand of the STAs are arbitrary. Therefore, we will apply this method to deal with the on-line case of our problem.

3 Network and System Description

3.1 Network Model

We consider an IEEE 802.11e based WLAN that comprises a large number of APs. Let A denote the set of APs and let N denote their quantity, i.e., $N = |A|$. All APs are attached to a controller, which makes the decision of which AP an STA should associate with. Each AP $a \in A$ has a theoretical traffic of C_a. Each AP has a limited transmission range and it can only serve STAs that reside in its service range.

We use S to denote the set of mobile STAs that have resided in the network range during a long-term time T. Our association algorithm is designed for the network in which STAs could arrive or depart freely. To the best of our knowledge, the network is regarded as stable when the time is measured in terms of tens of seconds. Therefore, we focus on a long-term time such as one day as the running time of the algorithm. Each STA is associated with a single AP to obtain service over a wireless channel. Because we don't take infrastructure into consideration, for STA $s \in S$ and AP $a \in A$, we use the maximal bit rate between STA and AP as the total bandwidth of the AP. In this paper, we first consider STAs with specified required bandwidth b_s and time t_s. APs will try to allocate the demanded bandwidth to its associated STAs, and STAs consume all bandwidth allocated to them and always have traffic to send or receive in their demanded time. Furthermore, we consider a general case where STAs can leave before their demanded time is used up.

3.2 System Description

We develop a centralized on-line association mechanism that determines the appropriate STA-AP associations to maximize the network traffic in a long-term time. We now discuss the main implementation aspects of an association control system. First, the system requires the relevant information of the session on each STA, such as the bandwidth and time demand of the session, the maximal bit rate that it experiences from each AP. Second, it needs an algorithm to determine the appropriate STA-AP association. Third, it needs a mechanism to enforce these decisions, including association, handover without user interference, and denial of service. We assume that such a mechanism is deployed at each AP, for instance, by using the emerging IEEE 802.11e extension [10] or any fair bandwidth allocation, and we build our association algorithm on top of it.

4 Algorithm Design

This section, we focus on the network scenario that each STA runs only one session which specifies a demanded bandwidth in its demanded time at one time.

4.1 Problem Definition

The AP association problem is the following: There are N APs, each with theoretical traffic C_a. S is a set of mobile STAs. Each AP a has the same total bandwidth b_a for STA $s \in S$ without concerning about the relative position between them and the interference. A sequence $s_1, s_2, \ldots, s_n$ of STAs $s_i \in S$ arrive on-line during T, and each STA s_i must specify required bandwidth b_{s_i} and time demand t_{s_i} according to its current session. The objective is to maximize the overall traffic of system at the end of T while respecting the quality of bandwidth of STAs. The notations and definitions to be used as summarized in Table 1. When a new STA s arrives at time t_s, we can divide the theoretical traffic into three parts in our association algorithm, the unused traffic before t_s, the allocated traffic and the remaining traffic after t_s as shown in Fig. 1. So the theoretical analysis based on the AdWords model will be incorrect. In our opinion, letting the unused part be small enough can be a reasonable solution to

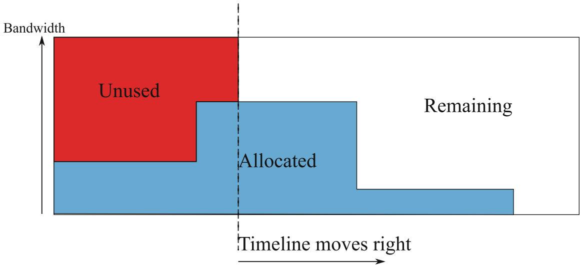


Fig. 1. The three parts of the theoretical traffic

Table 1. Notations

Symbol	Semantics
A	The set of all access points
S_a	The set of STAs which are associated with AP s
S	The set of all STAs
A_s	The set of candidate access points of STA s
C_a	The theoretical traffic of AP a
b_s	The demanded bandwidth of STA s
b'_s	The actual allocated bandwidth of STA s
t_s	The demanded time of STA s
b_a	The total bandwidth of AP a
$\psi(a)$	The tradeoff function on AP s
N	The number of APs in the network
n	The number of STAs which have associated with APs
t_s	The time a new STA s arrives
T	The running time of the association algorithm

adapt to the AdWords model. In order to achieve this, we define the "budget" for an AP as the theoretical traffic during a specified amount of time ("budget time"). We also suppose the amount of budget is the same for every AP, and call this a *budget window*. As our algorithm can only be valid during a short period of time at the beginning of budget window, we make the algorithm restart whenever the time-line reaches a specified short period (say, $1/k^2$ of "budget time") within budget window.

4.2 Algorithm

Now we will present our association algorithm when an STA s arrives in Algorithm 1. First, we define a trade-off function $\psi(x)$ of an AP as following:

$$\psi(x) = 1 - e^{-(1-x)}.$$

where x will be substituted with the ratio of allocated traffic within the total traffic of the AP's budget window. To make it clear, the allocated traffic is the traffic that this AP has used to communicate with its associated STAs, plus that which it has planned to use with its associated STAs in the future. And the total traffic of the AP's budget window is the size of the budget window. So on the other side, the ratio is related to the *free* traffic which this AP can allocate to upcoming STAs.

It's convenient to discretize the traffic into k equal parts, and we call each part a *slab*, so each slab contains $\frac{1}{k}$ of the total traffic in budget window. The AP allocates from lower slabs first, and when lower slabs are emptied, higher slabs will be used. The slab which the AP is currently allocating traffic from is called an *active* slab. We number each slab from 1 (lowest slab) to k (highest slab), and the active slab is $slab(i)$. Then we will get another form of the ψ function:

$$\psi_k(i) = 1 - e^{-(1-i/k)}.$$

When an STA s arrives, the STA s must notify the centralized AP controller its demanded bandwidth and time. Then each AP makes a bid to this STA. The bid is in the unit of traffic, that is, the bid equals the amount of traffic the AP can allocate. Of course, not all AP can satisfy the bandwidth demand of s, as the available bandwidth of an AP might be smaller than it (i.e., $b_a - \sum_{s\in S_a} b'_s < b_s$). In this case, the AP bid with traffic smaller than $b_s \times t_s$. If this AP gets associated, from the next moment until an STA leaves, its bandwidth will be fully utilized. Formally, this would be: (b'_s is the actual bandwidth allocated to the STA.)

$$b'_s = \begin{cases} b_s & b_a - \sum_{s\in S_a} b'_s \geq b_s; \\ b_a - \sum_{s\in S_a} b'_s & \text{otherwise.} \end{cases}$$

Algorithm 1. The AP-association algorithm

Require: bandwidth and time demand of STA s, represented by b_s and t_s respectively.
Ensure: *max_priority_ap* of STA s

```
1: max_priority ← −1
2: for a in all APs do
3:    bid ← b'_s × t_s
4:    if max_priority < bid × ψ_k(slab(i)) then
5:       max_priority ← bid × ψ_k(slab(i))
6:       max_priority_ap ← a
7:    end if
8: end for
9: max_priority_ap is the chosen AP for STA s.
```

5 Theoretical Analysis

In this section we analyze the performance of our algorithm in the special case when all bids made by the candidate APs are equal. Then the association algorithm can be simplified to a new algorithm which is just based on the available traffic of the APs. For convenience, we call the new simplified algorithm as SIMPLIFIED-ASSOCIATION algorithm as shown in Algorithm 2.

Algorithm 2. THE SIMPLIFIED-ASSOCIATION ALGORITHM

Require: bandwidth and time demand of STA s, represented by b_s and t_s respectively.
Ensure: $max\_priority\_ap$ of STA s

$max\_priority \leftarrow -1$
for a in all APs **do**
  if $max\_priority < \psi_k(slab(i))$ **then**
    $max\_priority \leftarrow \psi_k(slab(i))$
    $max\_priority\_ap \leftarrow a$
  end if
end for
$max\_priority\_ap$ is the chosen AP for STA s.

We wish to give a lower bound of the total traffic achieved by SIMPLIFIED-ASSOCIATION ALGORITHM. Let us define the *type* of AP in a period according to the fraction of traffic served by that AP at the end of the algorithm SIMPLIFIED-ASSOCIATION ALGORITHM: say that the AP in some period is of type j if the fraction of its theoretical traffic spent at the end of the algorithm lies in the range $((j-1)/k, j/k)$, By convention an AP in some period who spends none of his budget is assigned type 1.

Lemma 1. *In some period, if OPT associates STA s with AP a of type $j \leq k-1$, then* SIMPLIFIED-ASSOCIATION ALGORITHM *lays all demanded traffic of s in some slab i such that $i \leq j$.*

The lemma follows immediately from the criterion used by SIMPLIFIED-ASSOCIATION ALGORITHM for associating STAs with APs: A has type $j \leq k-1$ and therefore transmits at most j/k fraction of his theoretical traffic at the end of SIMPLIFIED-ASSOCIATION ALGORITHM. It follows that when STA s arrives at the beginning of some period, A is available to SIMPLIFIED-ASSOCIATION ALGORITHM for associating with s, and therefore A must associate s with some AP who has transmitted at most j/k fraction of his total traffic in this period s.

For simplicity we will assume that the AP in each period of type i transmit exactly i/k fraction of their theoretical traffic, that the amount of traffic pieces of each STA do not straddle slabs. The second one is justified by fact the traffic piece is small compared to the traffic in each period (e.g. taking the piece to be smaller than $\frac{1}{k^2}$ of the traffic of each period). The total error resulting from this simplification is at most $\frac{(2n-1)N}{k}$ and is negligible, once we take k to be large enough. Now, for $i = 1, 2, \ldots, k-1$, let x_i be the number of periods for all APs of type(i). Let β_i denote the total traffic transmitted by the AP in each period from slab i in BALANCE. It is easy to see (Fig. 2) that $\beta_1 = (2n-1)N/k$, and for $2 \leq i \leq k$, $\beta_i = \frac{(2n-1)N}{k} - (x_1 + \ldots + x_{i_1})/k$.

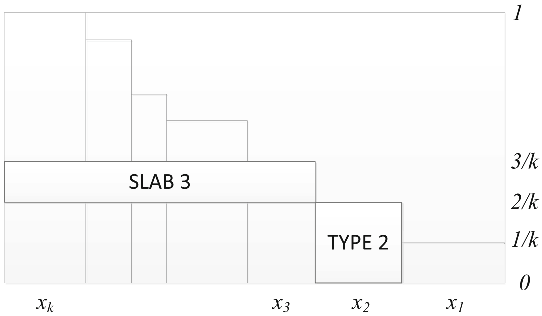


Fig. 2. The APs in each period are ordered from right to left in order of increasing type. We have labeled here the APs in each period of type 2 and the traffic in slab 3.

Lemma 2.

$$\forall\, i, 1 \leq i \leq k-1: \quad \sum_{j=1}^{i}(1+\frac{i-j}{k})x_j \leq \frac{i}{k}(2n-1)N.$$

From Lemma 1:

$$\begin{aligned}\sum_{j=1}^{i} x_j \leq \sum_{j=1}^{i} \beta_j &= \beta_1 + \sum_{j=2}^{i} \beta_j \\ &= \frac{i}{k}(2n-1)N - \sum_{j=1}^{i} \frac{i-j}{k} x_j.\end{aligned}$$

The traffic of Simplified-Association Algorithm is

$$\begin{aligned}\alpha - \mathrm{AS} &\geq \sum_{i=1}^{k-1} \frac{i}{k} x_i + [(2n-1)N - \sum_{i=1}^{k-1} x_i] - \frac{(2n-1)N}{k} \\ &= (2n-1)N - \sum_{i=1}^{k-1} \frac{k-i}{k} x_i - \frac{(2n-1)N}{k}.\end{aligned}$$

This gives the following LP, which we call L. In both the constraints below, i ranges from 1 to $k-1$.

$$\begin{aligned} \max \quad & \Phi = \sum_{i=1}^{k-1} \frac{k-i}{k} x_i, \\ \text{subject to} \quad & \forall i\colon \ \sum_{j=1}^{i}(1+\frac{i-j}{k})x_j \leq \frac{i}{k}(2n-1)N \\ & \forall i\colon \ x_i \geq 0. \end{aligned}$$

And the dual LP, D, will be used in the case of arbitrary demand bandwidth.

$$\begin{aligned} \min \quad & \Phi = \sum_{i=1}^{k-1} \frac{i}{k}(2n-1)Ny_i, \\ \text{subject to} \quad & \forall i\colon\ \sum_{j=i}^{k-1}(1+\frac{j-i}{k})y_j \geq \frac{k-i}{k} \\ & \forall i\colon\ y_i \geq 0. \end{aligned}$$

According to the analysis in [8], the value of Φ of the programs L and D goes to $(2n-1)N/e$.

Lemma 3. *The competitive ratio of* Simplified-Association Algorithm *is at least* $1-\frac{1}{e}$.

Recall the traffic of Simplified-Association Algorithm is at least $(2n-1)N - \Phi - \frac{(2n-1)N}{k}$, hence it tends to $(2n-1)N(1-\frac{1}{e})$. Since OPT is $(2n-1)N$, the competitive ratio is at least $1-\frac{1}{e}$.

6 Evaluation

In this section, we test our association algorithm on Matlab [1] and a real TestBed respectively. We compare the performance of our algorithm with that of the following ones:

- Strongest Signal First (SSF): The default STA-AP association mechanism in the 802.11 standard.
- Largest Available Bandwidth (LAB): The STAs will associate with the APs which has the largest available bandwidth compared with the demanded bandwidth of the STAs based on 802.11e.

In order to compare our algorithm with other association algorithms in variety of settings, we select several different bandwidth allocation algorithms. The various cases of test algorithms are labeled in Table 2.

Table 2. Algorithm combinations

Label	Bandwidth allocation	Association algorithm
QoS_F/ALG	Fixed bandwidth	ALG
QoS_F/SSF	Fixed bandwidth	SSF
QoS_F/LAB	Fixed bandwidth	LAB

This set of algorithms are examined carefully according to the performance metrics listed in the following:

– Per-STA bandwidth in Mbps.
– The overall traffic of the network.
– The traffic on the APs in Mbps.
– The competitive ratio.

6.1 Simulations

We first consider a static network, which involves 2 fixed APs in the network, and each has a capacity of 4 Mbps. There are 4 STAs accessing the network successively. The parameters of the APs and the STAs are shown in Table 3. The algorithms, SSF and LAB, make greedy association decisions to satisfy the current STAs. The bandwidth allocated to the STAs is so unreasonable that the APs cannot provide the enough bandwidth for the coming STAs such as STA 4 as shown in Table 4. And ALG allows STA 4 associate with AP a to achieve more throughput in the system.

Table 3. Parameters of the APs and the STAs in scenario 2

STAs	Start(s)	End(s)	Demanded bandwidth (Mbps)	Max bit rate (Mbps)	
				AP a	AP b
1	0	200	1	4	2
2	50	200	1	2.7	2
3	100	200	2	2	4
4	150	200	2	3	4

Table 4. Scenario 1: dynamic STAs. The STA 4 could not associate with any AP in the network in the association algorithms, SSF and LAB

Metrics	Algorithms	APs							
		a (4 Mbps)				b (4 Mbps)			
		STAs				STAs			
		1	2	3	4	1	2	3	4
Association	QoS_F/ALG	√			√		√	√	
	QoS_F/SSF	√	√		⊗			√	⊗
	QoS_F/LAB	√	√		⊗			√	⊗
Traffic ($\times 10^3$ Mb)	QoS_F/ALG	0.2	0	0	0.1	0	0.15	0.2	0
	QoS_F/SSF	0.2	0.15	0	0	0	0	0.2	0
	QoS_F/LAB	0.2	0.15	0	0	0	0	0.2	0

Scenario 2 involves 9 APs and many mobile STAs in a $100 \times 100\,\mathrm{m}^2$ network. The locations of the APs are fixed as shown in Fig. 3. Each AP has a capacity of 54 Mbps. The arrival and departure of the STAs are subject to the normal distribution. There are two peak hours, 15 and 21, during a day (24 h) as shown in Fig. 4(a). The locations of the STAs are random in the network. We plot the network traffic under ALG and LAB as shown in Fig. 4(b). The network traffic in ALG improves about 37% compared with the LAB algorithm. And it is obvious that the traffic on each AP under ALG is more balancing compared with LAB as shown in Fig. 4(c) and (d).

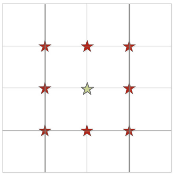
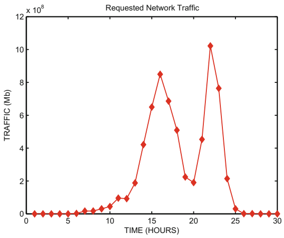
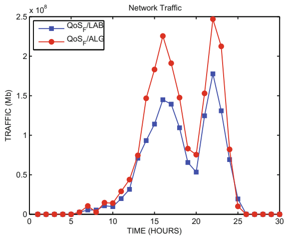


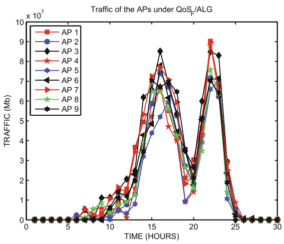
Fig. 3. The topology of the static network



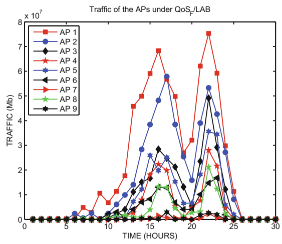
(a) Network traffic request of the system



(b) Network traffic of the system



(c) Traffic on the APs under ALG



(d) Traffic on the APs under LAB

Fig. 4. Scenario 3: dynamic network.

Scenario 3 involves 3 fixed APs in the network, each has a capacity of 4 Mbps as shown in Fig. 5. And the maximal bit rates between the APs and the STAs are the capacity of the APs. Then we can get the competitive ratio compared with SSF as shown in Fig. 6.

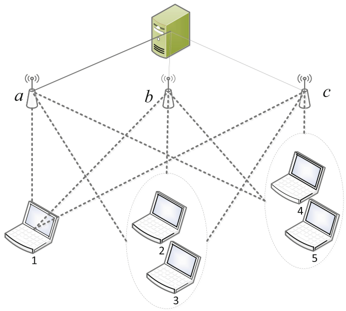


Fig. 5. The topology of the static network

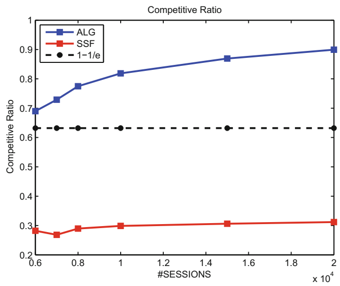


Fig. 6. Competitive ratio

6.2 Experiments

In this section, we will report our results of experiments which help us understand the performance of our association algorithm. Experiments are conducted with Thinkpad R61e laptops equipped with Atheros AR2425 802.11g wireless cards. Each laptop is loaded with the modified Madwifi driver v0.9.4 [2] to collect experimental data. The topology of the network is shown in Fig. 8.

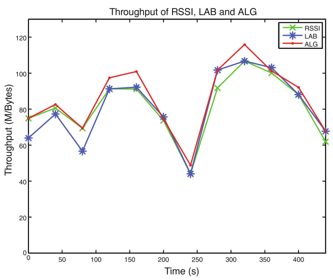


Fig. 7. Scenario 4: the network traffic of the testbed.

Fig. 8. The topology of testbed

We compare the network traffic of our algorithm with that of LAB and SSF, and present that our algorithm has better performance than the other two as shown in Fig. 7. Actually, the network in the testbed is a worse case for our algorithm, because our association algorithm performs better in congested networks. Though this network is light-loaded because of the limited equipment, our algorithm still performs better than the other two.

7 Conclusions

In this paper, we propose a novel on-line association algorithm to deal with any sequence of STAs during a long-term period such as one day. One important advantage of our algorithm is that it does not need any periodical off-line optimal solutions. We give a strict proof that the competitive ratio of the algorithm is $1 - 1/e$ when APs allocate the demanded bandwidth of their associated STAs. Simulation results show that the proposed association algorithm can improve the network traffic by more than 37% when compared with conventional association algorithms. Our algorithm also performs better than SSF and LAB in the experiments even in a less congested network, which may improve more performance for other association algorithms than that of ours.

We plan to test and verify the performance of our association algorithm with more APs and more STAs in our testbed. In this condition, we try to find the difference between this experiment and the previous one to optimize our association algorithm. Meanwhile, interference which will influence the performance in a congested network will be taken into consideration in our model.

References

1. Matrix laboratory. http://www.mathworks.com
2. Multiband atheros driver for wifi. http://madwifi-project.org
3. Bejerano, Y., Han, S.J., Li, L.E.: Fairness and load balancing in wireless LANs using association control. In: Proceedings of the 10th Annual International Conference on Mobile Computing and Networking, MobiCom 2004, pp. 315–329. ACM, New York (2004). http://doi.acm.org/10.1145/1023720.1023751
4. Bredel, M., Fidler, M.: Understanding fairness and its impact on quality of service in IEEE 802.11. In: IEEE INFOCOM 2009, pp. 1098–1106. IEEE (2009)
5. He, Y., Perkins, D., Velaga, S.: Design and implementation of class: a cross-layer association scheme for wireless mesh networks. In: INFOCOM IEEE Conference on Computer Communications Workshops, pp. 1–6. IEEE (2010)
6. Kim, H., De Veciana, G., Yang, X., Venkatachalam, M.: Alpha-optimal user association and cell load balancing in wireless networks. In: 2010 Proceedings of IEEE INFOCOM, pp. 1–5. IEEE (2010)
7. Le, Y., Ma, L., Cheng, W., Cheng, X., Chen, B.: Maximizing throughput when achieving time fairness in multi-rate wireless LANs. In: 2012 Proceedings of IEEE INFOCOM, pp. 2911–2915. IEEE (2012)
8. Mehta, A., Saberi, A., Vazirani, U., Vazirani, V.: Adwords and generalized online matching. J. ACM **54**(5). http://doi.acm.org/10.1145/1284320.1284321

9. Nassiri, M., Heusse, M., Duda, A.: A novel access method for supporting absolute and proportional priorities in 802.11 WLANs. In: The 27th Conference on Computer Communications, INFOCOM 2008, pp. 709–717. IEEE (2008)
10. Ni, Q., Romdhani, L., Turletti, T.: A survey of QoS enhancements for IEEE 802.11 wireless LAN: research articles. Wirel. Commun. Mob. Comput. **4**(5), 547–566 (2004). http://dx.doi.org/10.1002/wcm.v4:5