Proceeding of the IEEE 28th
Canadian Conference on Electrical and Computer Engineering
Halifax, Canada, May 3-6, 2015

A Sociality-Aware Online AP Association Algorithm Based on Spectral Clustering

Haibo Ni, Lei Wang, Zhenjie Yang
Ming Zhu, Feng Ding, Zhenquan Qin
School of Software
Dalian University of Technology, China
aaron.nihaibo@gmail.com, lei.wang@dlut.edu.cn, yangzhenjie.dlut@gmail.com
zhuming@dlut.edu.cn, dingfeng@dlut.edu.cn, qzq@dlut.edu.cn

***Abstract*— Imbalanced load distribution among multiple APs (access points) has become a serious problem in enterprise WLANs. Load imbalance causes inefficiency of network usage and unreasonable resource allocation. We find that users tend to come and leave together, which can lead to fluctuation of network load. We propose a novel online AP allocation scheme (ASC) to keep load balanced when users come and leave together. We study the *homophily* of users in social network, where users with tight social relationships will have more similar network usage and more opportunity to leave in union. When users come, we collect user data first, then we adopt the spectral clustering to classify users according to the collected user data. The controller distribute users from the same cluster to different APs. Through the simulation, we find that our scheme improve the overall balancing performance by at least 22% compared with the least load first selection algorithm (LLF) and 58% compared with random selection algorithm (RS).**

I. Introduction

Wireless local area networks (WLANs) has become the most common way for enterprise and educational institution to access to the Internet. Along with the increase of APs, users have more chances to receive more than one AP signal. Imbalanced load distribution among different APs has become a serious problem in enterprise WLANs. Uneven distribution network load can cause very serious problems.

In most cases, users choose an AP that has the strongest received signal strength indicator (RSSI), but this AP may associate with a number of users and has much heavier network load. This association approach can lead to inefficient use of the wireless network resources. With the development of study on load balancing, more and more AP selection schemes are raised to keep the network load balanced, such as the number of users connected to the AP and the bandwidth that users could get from the AP.

All these schemes do not consider how user's coming and leaving affect the network load. Through our analysis based on the user data we collected, we find that users tend to come and leave together in union in campus and enterprise WLANs. As shown in Fig. 1, there are two APs in the WLAN and the network load is balanced now. Supposing that a number of users leave AP1 at some time because they have strong social relationship, it will take a long time to balance the network. So if we distribute these users to AP1 and AP2 separately, this situation would not happen. Thus how to assign different users to proper APs is a challenging issue.

We propose an online AP selection scheme based on the spectral clustering, which can take users' comings and leavings into account. The main contributions of this work are summarized as follows:

1) *Take users' social relationships into account*: we propose an approach to classify users according to their network usage. We can get the bandwidth usage curve based on the records. Then we can construct matrix whose row represents every user's bandwidth usage over time and column represents the number of users. The spectral clustering is used to classify users into different clusters. So users that come and leave in union are more likely classified to the same cluster.
2) *When users disconnect the AP, the time it takes to reach load balancing is much shorter than other AP association schemes*: Because we take network stability into consideration when we allocate the users, so when they leave, the network can keep load balanced.
3) *Keep the network balanced*: we propose the AP selection scheme based on spectral clustering. Users with similar network usage will be allocated to different APs. So different APs have similar throughput so that the network is kept balanced.
4) *Use the data we collect as simulation input*: The simulation input is the data we collect in our teaching building. We collect the data for more than one month and 2000 users are recorded.
5) *Design an online algorithm*: We first present the online AP selection algorithm taking social relationships into account in enterprise WLANs. All the existing AP association schemes based on social relationships are based on training, but they could not provide online association algorithm. This is also the most prominent contribution of this paper.

The remainder of this paper is organized as follows. In Section II, we discuss the related work. Section III describes the social relationships and spectral clustering, followed by Section IV which proposes the AP selection scheme based on

978-1-4799-5829-0/15/$31.00 ©2015 IEEE

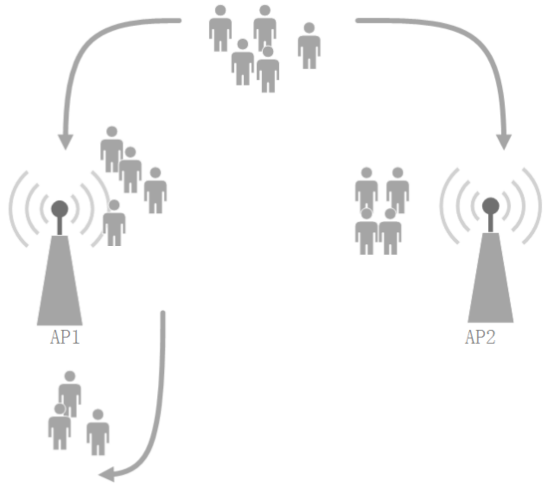

Fig. 1. Users' coming and leaving in WLANs.

pre-classified clusters. In section V, we give the simulation results and have a discussion about the results. Finally, conclusion is presented in Section VI.

II. Related Work

Association algorithms have been intensely studied in the literature. And several association metrics have been proposed instead of RSSI. Balachandran *et al*. [1] proposed that user connected with the AP that can provide a minimal bandwidth required by the user. Nicholson *et al*. [2] designed the AP selection scheme based on real-time monitoring. In this paper, a tentative node is associated with each AP to get useful information used to choose an AP, such as bandwidth and delay. Obviously the AP that can provide better service will be selected. Bejerano *et al*. [3] presented an efficient solution to determine the STA-AP association for max-min fair bandwidth allocation, by leveraging the strong correlation between fairness and load balancing.

Bejerano *et al*. [4] presented a new load balancing technique by controlling AP's coverage range, which is similar to cell breathing in cellular networks. Shakkottai *et al*. [5] considered the scenario where users could multihome, i.e., split their traffic amongst all the available APs. In [6], a load-balancing scheme was proposed for overlapping wireless LAN cells. Overload APs force the handoff of some users to balance the load. Yen *et al*. [7] modeled AP selection under the framework of game theory, where the goal of each STA is to maximize the achievable throughput, by considering both the number of STAs which associate with the same AP and the set of link rates these STAs posses.

All schemes mentioned above don't consider users' coming and leaving. Recently there are more studies in this field. In [8], the authors implemented an OFDM-based WLANs called HJam, which fully explored the physical layer features of the OFDM modulation method and combined data packets and a number of control messages to be transmitted together. Chen [9] proposed a unique solution called SAP. SAP utilized the OAMI architecture to provide seamless handoff experience for users, which could balance the network load.

The solutions existed just consider how users affect the network, but they don't take users's social relationships into consideration. Guangtao Xue *et al*. [10] proposed a novel AP allocation scheme to tackle the load balancing problem in WLANs, taking into account the social relationships of users. But they must know every two users' social relationship index. Chaoqun Yue *et al*. [11] mentioned that users had more similar application and network usage should have stronger social relationships. Of course, the conclusion is based on a mass of measured data.

Muhammad U. Ilyas *et al*. [12] proposed a wavelet-based scheme, called COmmunication Link De-anonymization (COLD) to accurately identify a candidate set of top-k users with whom a given user possibly talked to. Wavelet transform had been used to process the packets and relevance of users had been calculated. Here we use the bandwidth information for clustering. The method is similar to that of [12]. Therefore we want to design an AP selection algorithm considering both the users' social relationships and network traffic information.

We present an online AP association algorithm based on clustering. Social relationships are also considered in our system.

III. Social Relationships and Spectral Clustering

In this section, we first introduce how we collected the data in the teaching building. More details about social relationships and spectral clustering are also discussed. Moreover, we give a introduction about how we apply spectral clustering to classify user data for our AP selection algorithm.

A. Collecting Trace Data

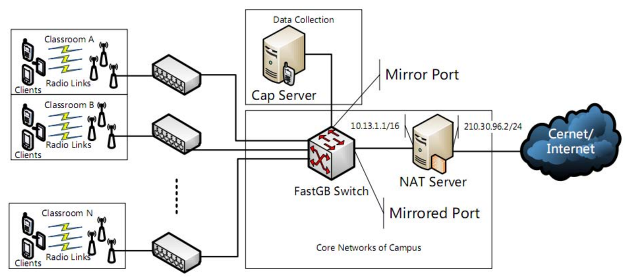

Fig. 2. The network architecture

We collected data from the AP sets in the zone B of teaching buildings in School of Software, Dalian University of Technology. The teaching building consists of 2 floors and each floor has 7 classrooms. We use *tcpdump* to collect data and use *python* to analyze the collected data. Fig. 2 is the network architecture to collect the data. We have collected data for 37 days and the overall data size is 1.5TB. More than 2000 users are recorded.

By using and analyzing these data, we can simulate the algorithm we present. Because we have relatively fixed network users every day, we can calculate the mean traffic of every user in a day. The mean traffic is used as simulation input. Python is utilized to process the collected user data. The people who connect with the network is relatively fixed, so we collect user data every and take an average, respectively. After processing the data, we get the data we want. The data is last about 6 hours.

B. Spectral Clustering

Clustering has become one of the most popular techniques for data analysis. The application field is very wide, like computer science, biology, geography and statistics. In recent years, the spectral clustering has become one of the most popular clustering algorithms [13]. It is easy to implement with standard linear algebra methods, and can achieve better results than other clustering algorithms such as k-means algorithm.

Then the spectral clustering is briefly introduced below. Spectral clustering algorithm regards each object of the data set as a vertex V of figure, the degree of similarity between the vertices is quantified. And the degree is the corresponding weights of vertices connected edge E. Then we can get the undirected weighted graph $G(V,E)$ based on the similarity. The clustering problem can be converted into graph partition problem. Based on the rule of optimal partition in graph theory, that is, maximizing internal similarity in subgraph, minimizing the similarity between subgraphs.

Although according to different criterion function and spectral mapping method, spectral clustering algorithm can be implemented with different specific implementation methods, the implementation method can be concluded into the following three main steps:

1) Build similarity matrix W of the object data set;
2) By calculating the preceding k eigenvalues and eigenvectors of the similarity matrix or Laplacian matrix, building characteristic vector space;
3) Apply the k-means or other classical clustering algorithm to complete the clustering of characteristic vector in characteristic vector space.

C. Social Relationships

Social relationships have been studied for tens of years. The combination of AP association and social relationships is an interesting research field. And there are some well-established theory about social relationships and network. Homophily is one of the most familiar one. Homophily is the tendency of individuals to associate and bond with similar others. The presence of homophily was discovered in [14]. Users in homophilic relationships will share common characteristics [15]. When in the WLANs, the theory turns into that users with strong social relationships have higher chance to have similar network usage. On the other hand, Chaoqun Yue *et al.* [11] also mentioned that users had more similar application usage should have stronger social relationships.

TABLE I
NOTATIONS

Symbol	Semantics
A	The set of APs
a	The AP
U	The set of users
u	The user
t	The time
$t\_c(u)$	The time user u comes
$type(u)$	Cluster number of user u
$ap\_bd[a][t]$	The traffic of AP a in time t
$ap\_bd[b][t]$	The traffic of AP b in time t
$state(u)$	The state of user u, online or offline
$ap\_usr[a][1][t]$	The number of users with cluster number "1"of AP a in time t
$ap\_usr[a][2][t]$	The number of users with cluster number "2" of AP a in time t
$ap\_usr[a][3][t]$	The number of users with cluster number "3" of AP a in time t
$usr\_bd[u][t]$	The bandwidth user u occupies in time t
α	The balancing index
β	The sum of absolute difference

IV. AP Selection Algorithm based on Spectral Clustering

In this section, We will give the details about our online AP selection algorithm 'ASC' based on spectral clustering. We focus on the scenario that there are n access points in the enterprise WLAN. When users come, they will be allocated to one of the n APs. Then we will collect their network usage for five minutes. Spectral clustering is used to classify users according to cluster number.

A. Problem Definition

The AP selection problem is as follows: There are n APs in the WLAN, user $u \in U$ comes and leaves as time passes. When users come, we first allocate users to AP $a \in A$ according to AP's load. That is, users will be allocated to the AP with less load. Then we collect the network usage of users for 5 minutes. Based on the trace data, we will collect the users that are online for the last 5 minutes. Next we classify these users by using spectral clustering. After that every user will get a cluster number after the clustering. The simulation will get good results no matter how many clusters our simulation adopts, but 3 clusters will get better results than the others according to our verification. We classify them into 3 clusters. Users with similar network usage will be assigned with the same cluster number. Then we will assign users to one of the n APs $a \in A$. This operation is feasible according to existing methods as described in [16], they proposed a fast handoff algorithm, called HaND. HaND employs a novel zero-channel-dwell-time architecture to reduce the handoff delay. The controller will exchange the users' information before the handoff through the wired network. When it triggers handoff, the user u will connect with the next AP a directly. So adjusting users' connected AP is applicable without interrupting the users. Balancing index is introduced to evaluate the performance of the algorithm. The notations and definitions used are listed in Table I.

B. Algorithm

Our algorithm takes not only the users' social relationships but also the number of users connected with each AP into consideration. In our algorithm, we assume that each AP has enough available bandwidth for allocating. Also this is an online AP association algorithm.

In our algorithm, we apply two ways to measure the results, the balancing index and Sum of Absolute Difference. We define the balancing index α [17] and Sum of Absolute Difference β as follows:

$$\alpha = \sum_t \frac{(\sum_{a\in A} ap\_bd[a][t])^2}{n\sum_a ap\_bd[a][t]^2}$$

$$\beta = \sum_t \sum_{a\in A, b\in A} |ap\_bd[a][t] - ap\_bd[b][t]|$$

From the equation, we can see that when the throughput of different APs become closer, the balancing index will become approach to 1. For example, when there are 2 APs, AP1 and AP2 with the throughput of 3Mbps and 5Mbps, the balancing index is 0.94. But if AP1 and AP2 have the throughput of 3Mbps and 10Mbps, the balancing index is 0.546. Apparently, when AP1 and AP2 have the same throughput, the balancing index is 1. The balancing index has been widely used in the literature [6]. The Sum of Absolute Difference is linear function, so it will become bigger if the load is not kept balanced. So the two measurements can tell us which algorithm is better. Bigger α and smaller β mean that our algorithm can keep network balanced when many users leave together.

Algorithm 1's specific steps are as follows:

1) Collect all online users' network usage data when an user comes.
2) Get the matrix whose row represents the every user's bandwidth usage over time and column represents the number of users.
3) Manipulate the matrix to adapt spectral cluster's requirements.
4) Use spectral cluster to classify users, in theory, users with strong social relationships will be classified to the same cluster. Users in the same cluster will be allocated to different APs.
5) Apply AP selection algorithm to complete the distribution.

Algorithm 1: ASC

Require: $u \in U$, $type(u)$
Ensure: higher α and minimum β of the network.

for t passes **do**
  if $t\_c(u)$ == t **then**
    u connects with $a \in A$ first.
  end if
  $state(u)$ is online.
  After 5 minutes
  For all users online, spectral clustering is used to classify users.
  Every u gets its $type(u)$.
  for u online **do**
    if $type(u)$ == 1 **then**
      Find the AP a with the least users with cluster number 1.
      u connects with a.
    else
      if $type(u)$ == 2 **then**
        Find the AP a with the least users with cluster number 2.
        u connects with a.
      else
        if $type(u)$ == 3 **then**
          Find the AP a with the least users with cluster number 3.
          u connects with a.
        end if
      end if
    end if
  end for
end for
for t **do**
  Calculate α and β.
end for
α and β are the values we need.

V. SIMULATION

In this section, we will test our AP selection algorithm and compare the performance with the random and least load algorithm. We use two measurements to test if ASC can keep the network balanced when there are many together-comings and together-leavings.

- Random Selection (RS): Users will select APs in a random way without considering the network load and signal strength.
- Least Load first (LLF): The users will connect with APs which have least network load by that time. To some extent, this approach will help to achieve load balancing.

We test our algorithm in the scenario where contains one controller and five APs, naming AP1, AP2, AP3, AP4 and AP5, respectively. There are users coming and leaving as time goes by, and the coming, leaving and network usage is based on the data we collected in section III.A. We will compare the performance with RS and LLF mentioned above. We don't consider the capacity of APs, in other words, each AP can provide enough bandwidth. The network topology of simulation is shown in Fig. 3. The bandwidth users occupy will change with time passing. We will allocate users to one of the n APs on the basis of RS, LLF and ASC.

In our simulation, according to the data we collected, there are 236 users coming and leaving as time passes. When user u comes, we first connect it with any one of APs according to the LLF algorithm. Then we use 5 minutes to collect its network usage and also collect other APs' network usage that have already connected to the network. After collecting users' data, we use spectral clustering mentioned

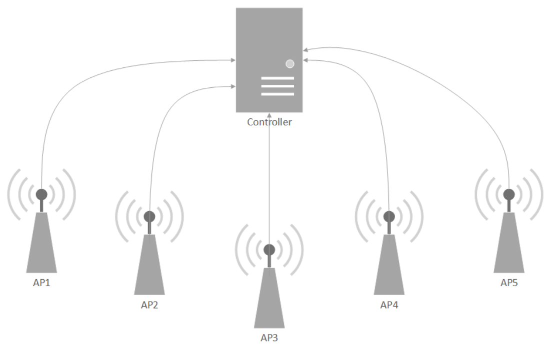

Fig. 3. The network topology.

above to classify users. Then we complete the AP association according to the scheme mentioned in ASC. Finally we get results of more than 6 hours. To explain the results clearly, we calculate the two measurements every 10 minutes. First we use the balancing index as the measurement. Fig. 4 and Fig. 5 is the results of balancing index. The horizontal axis is the time and vertical axis is the balancing index.

Fig. 4 shows that ASC is better than RS algorithm in almost all of the time. We can see that balancing index of our algorithm is bigger than that of random algorithm from 9:00am to 11:30am. And in the afternoon, ASC also performs better than RS from 13:00pm to 14:50pm. But we also see that our algorithm performs worse than RS from 11:30am to 13:00pm. We will discuss later why this happened.

Fig. 5 shows that our algorithm is better than LLF. The graph is like Fig. 4 that ASC performs better most of the time, especially in the morning and in the afternoon, timing that our algorithm is worse than LLF in the noon. The phenomenon is like what we find in Fig. 4.

Now let us talk about the phenomenon arises in Fig. 4 and Fig. 5. We can see that this appears in the noon, Because we collected the data in the teaching building, students tend to sleep in the dormitory, so there are few students studying in the classrooms. The spectral clustering is not less efficient when there are fewer students. So the three algorithms performs almost the same in the noon. But our algorithm ASC performs better than the other algorithms in the morning and in the afternoon.

We can see that the balancing index of the three algorithms in Table II. From the table, we can draw the conclusion that our algorithm performs 22% better than LLF and 58% better than RS, the value is even bigger in the morning and in the afternoon.

TABLE II
BALANCING INDEX OF THREE ALGORITHMS

Label	balancing index
ASC	20.919
LLF	17.157
RS	13.157

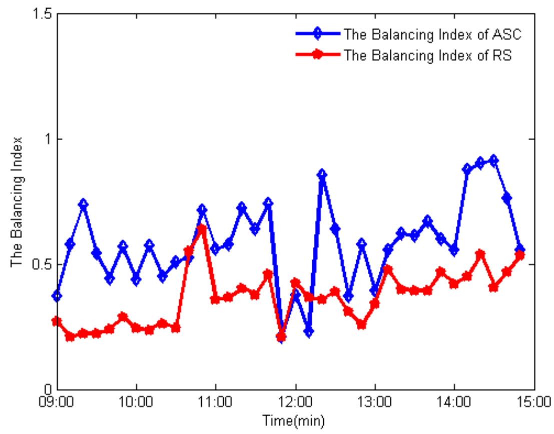

Fig. 4. The results of ACS and RS in Balancing Index.

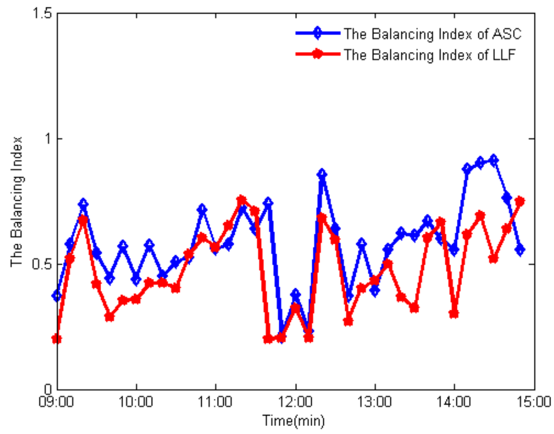

Fig. 5. The results of ASC and LLF in Balancing Index.

Fig. 6 and Fig. 7 are the results of the Cumulative Sum of Absolute Difference. We use cumulative function to represent the Sum of Absolute Difference. Fig. 6 and Fig. 7 show that our algorithm is better than RS and LLF. The Sum of Absolute Difference is intuitive to describe the balancing index. As we can see, the gap of two curves become bigger almost all the time. So we can say that through the measurement our algorithm is better than the other two algorithms. Also we can see the Cumulative Sum of Absolute Difference of the three algorithms in Table III. From the table, we can draw a conclusion that our algorithm ASC performs 38% better than LLF and doubles efficiency compared with RS.

TABLE III
THE SUM OF ABSOLUTE DIFFERENCE OF THREE ALGORITHMS

Label	The Sum of Absolute Difference (Mbps)
ASC	151.57
LLF	208.27
RS	317.03

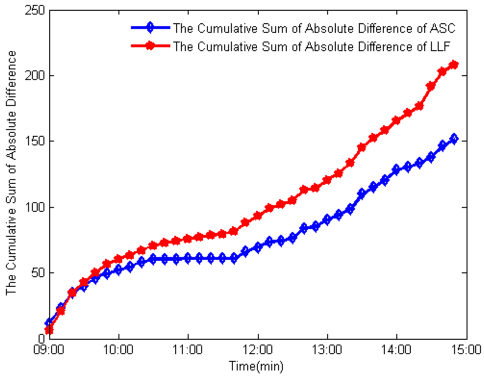

Fig. 6. The results of ASC and LLF in Sum of Absolute Difference.

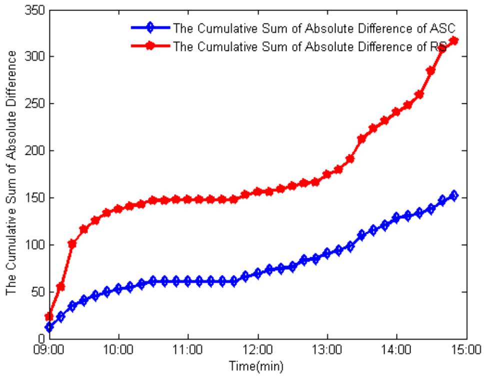

Fig. 7. The results of ASC and RS in Sum of Absolute Difference.

VI. Conclusions

In this paper, we propose an online AP selection algorithm ASC based on spectral clustering. There are five apparent advantages of our algorithm. First, we take users' social relationships into account, which are not considered in most of existing AP selection schemes. This phenomenon happens quite often in WLANs of enterprises and educational institutions. Load balancing in this kind of WLAN is more likely influenced by users' together-comings and together-leavings. Second, When users disconnect the AP, the time it takes to reach load balancing is much shorter than traditional AP association schemes. Third, we can keep the network balanced. Fourth, we use the data we collected in the teaching buildings as simulation input. We collected data for about 37 days and more than 2000 users were recorded. Last but not least, our AP selection algorithm is an online AP association algorithm, which is not mentioned in most existing AP selection algorithms. This is also the most important contribution of this paper. We evaluate the performance of our algorithm by simulation. And balancing index and the Cumulative Sum of Absolute Difference are adopted to test our algorithm. Simulation results show that our algorithm can improve the load balancing by at least 22% when compared with the LLF and 58% compared with RS.

Acknowledgments

This work is supported by Natural Science Foundation of China under Grants No. 61070181, No. 61272524 and No. 61202442 and the Fundamental Research Funds for the Central Universities. This work is also supported by the Fundamental Research Funds for the Central Universities under Grants No. DUT14ZD218 and No. DUT14QY29.

References

[1] A. Balachandran, P. Bahl, and G. M. Voelker, "Hot-spot congestion relief in public-area wireless networks," in *Mobile Computing Systems and Applications, 2002. Proceedings Fourth IEEE Workshop on*. IEEE, 2002, pp. 70–80.

[2] A. J. Nicholson, Y. Chawathe, M. Y. Chen, B. D. Noble, and D. Wetherall, "Improved access point selection," in *Proceedings of the 4th international conference on Mobile systems, applications and services*. ACM, 2006, pp. 233–245.

[3] Y. Bejerano, S.-J. Han, and L. E. Li, "Fairness and load balancing in wireless lans using association control," in *Proceedings of the 10th annual international conference on Mobile computing and networking*. ACM, 2004, pp. 315–329.

[4] Y. Bejerano and S.-J. Han, "Cell breathing techniques for load balancing in wireless lans," *Mobile Computing, IEEE Transactions on*, vol. 8, no. 6, pp. 735–749, 2009.

[5] S. Shakkottai, E. Altman, and A. Kumar, "The case for non-cooperative multihoming of users to access points in ieee 802.11 wlans." in *INFOCOM*, 2006.

[6] H. Velayos, V. Aleo, and G. Karlsson, "Load balancing in overlapping wireless lan cells," in *Communications, 2004 IEEE International Conference on*, vol. 7. IEEE, 2004, pp. 3833–3836.

[7] L.-H. Yen, J.-J. Li, and C.-M. Lin, "Stability and fairness of ap selection games in ieee 802.11 access networks," *Vehicular Technology, IEEE Transactions on*, vol. 60, no. 3, pp. 1150–1160, 2011.

[8] K. Wu, H. Li, L. Wang, Y. Yi, Y. Liu, D. Chen, X. Luo, Q. Zhang, and L. M. Ni, "hjam: Attachment transmission in wlans," *Mobile Computing, IEEE Transactions on*, vol. 12, no. 12, pp. 2334–2345, 2013.

[9] X. Chen, Y. Zhao, B. Peck, and D. Qiao, "Sap: Smart access point with seamless load balancing multiple interfaces," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 1458–1466.

[10] G. Xue, Q. He, H. Zhu, T. He, and Y. Liu, "Sociality-aware access point selection in enterprise wireless lans," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 24, no. 10, pp. 2069–2078, 2013.

[11] C. Yue, G. Xue, H. Zhu, J. Yu, and M. Li, "S3: Characterizing sociality for user-friendly steady load balancing in enterprise wlans," in *Distributed Computing Systems (ICDCS), 2013 IEEE 33rd International Conference on*. IEEE, 2013, pp. 491–499.

[12] M. U. Ilyas, M. Z. Shafiq, A. X. Liu, and H. Radha, "Who are you talking to? breaching privacy in encrypted im networks." in *ICNP*, 2013, pp. 1–10.

[13] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[14] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual review of sociology*, pp. 415–444, 2001.

[15] C. R. Shalizi and A. C. Thomas, "Homophily and contagion are generically confounded in observational social network studies," *Sociological Methods & Research*, vol. 40, no. 2, pp. 211–239, 2011.

[16] X. Chen and D. Qiao, "Hand: Fast handoff with null dwell time for ieee 802.11 networks," in *INFOCOM, 2010 Proceedings IEEE*. IEEE, 2010, pp. 1–9.

[17] R. Jain, D.-M. Chiu, and W. R. Hawe, *A quantitative measure of fairness and discrimination for resource allocation in shared computer system*. Eastern Research Laboratory, Digital Equipment Corporation Hudson, MA, 1984.