Sociality-Aware Access Point Selection in Enterprise Wireless LANs

Guangtao Xue, *Member*, *IEEE*, Qi He, Hongzi Zhu, *Member*, *IEEE*, Tian He, *Senior Member*, *IEEE*, and Yunhuai Liu, *Member*, *IEEE*

Abstract—Well-balanced workload among wireless access points (APs) in a wireless local-area network (WLAN) can improve the user experience for accessing the Internet. Most load balancing solutions in WLANs focuses on the optimization of AP operations, assuming that the arrivals and departures of users are independent. However, through the analysis of AP usage based on a real WLAN trace of one-month collected at the Shanghai Jiao Tong University (SJTU), we find that such an assumption does not hold. In fact, due to users' social activities which is particularly time for enterprise environments, they tend to arrive or leave in unison, which would disruptively affect the load balance among APs. In this paper, we propose a novel AP allocation scheme to tackle the load balancing problem in WLANs, taking into account the social relationships of users. In this scheme, users with intense social relationships are assigned to different APs so that jointly departure of those users would have minor impact on the load balance of APs. Given that the problem of allocating an AP for each user so that the average of the sums of social relation intensity between any pair of users in each AP is *NP-complete*, we propose an online greedy algorithm. Extensive trace-driven simulations demonstrate the efficacy of our scheme. Comparing to the state-of-the-art method, we can achieve about 64.7 percent balancing performance gain on average during peak hours in workdays.

Index Terms—IEEE 802.11, Enterprise Wireless LANs, load balance, social relation analysis

1 Introduction

IEEE 802.11 (WiFi) wireless local area networks (WLANs) have become one major technology to provide ubiquitous wireless Internet access for enterprise. In an enterprise WLAN, one major issue is unbalanced workload distribution among access points (APs) [1], [2], [3], [4], where APs associated with a large number of users have much heavier workload burden than those which have less users. The consequence of unbalanced AP work load distribution is twofold. On the one hand, overloaded APs would cause constant packet loss, which dramatically degrades user experience. On the other hand, to improve the quality of network service, the IT department of an enterprise might have to take a cost-inefficient solution, for example, by deploying more APs.

In the literature, the AP load balancing problem in WLANs has been extensively studied. A large majority of proposed methods, called *arrival-based AP allocation*, consider the arrivals of new users and allocate an AP with light workload for each one of them. A critical assumption in these approaches is that users behave independently with regard to joining or leaving the WLAN. While it may be true in a public WLAN (e.g., in a popular cafe or an airport), where users have individual schedules, users of an enterprise WLAN often act collaboratively because of social activities in the enterprise. For example, in a university, students are likely to enter to and leave a classroom together for a class. Especially, events that users of an AP jointly leave the AP would disruptively affect the global load balance among APs. For example, as illustrated in Fig. 1, suppose there are two APs in an enterprise WLAN and initially they are balanced. At a certain time instance, a large group of users in AP1 leave together because of certain social activity. In this case, it will take a long time for the two APs to get load balanced again using arrival-based AP allocation algorithms. Another category of proposed methods, called *online adjustment-based AP allocation*, migrate users from heavy-duty APs to those with light workload until load balance among all APs is achieved. In a typical enterprise WLAN, the churn of users is highly dynamic, which would introduce huge computation cost and frequent connection disruptions. As a result, there is no successful solution, to the best of our knowledge, to addressing the AP load balancing problem in enterprise WLANs.

In this paper, we take a data-driven approach in designing and evaluating our AP load balancing algorithm in enterprise WLANs. We collect extensive real AP usage trace data of one month at Shanghai Jiao Tong University (SJTU), involving more than 14,669 users and 354 APs deployed in 22 buildings. Two users are said to have a *social relationship* if they share a common *social activity*. A social activity is defined as for a pair of users to join, access or

- G. Xue, Q. He, and H. Zhu are with the Department of Computer Science and Engineering, Shanghai Jiao Tong University (SJTU), 800 Dongchuan Road, Shanghai 200240, P.R. China. E-mail: {gt_xue, vanessaheq, hongzi}@sjtu.edu.cn.
- T. He is with Department of Computer Science and Engineering, University of Minnesota, 4-205 EE/CSci Building, 200 Union Street SE, Minneapolis, MN 55455. E-mail: tianhe@cs.umn.edu.
- Y. Liu is with Center of Internet of Things, The Third Research Institute of Ministry of Public Security, 339 Bisheng Road, Shanghai 201204, P.R. China. E-mail: yunhuai@siat.ac.cn.

Manuscript received 25 Feb. 2012; revised 24 Sept. 2012; accepted 2 Oct. 2012; published online 19 Oct. 2012.
Recommended for acceptance by W. Jia.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-2012-02-0145.
Digital Object Identifier no. 10.1109/TPDS.2012.301.

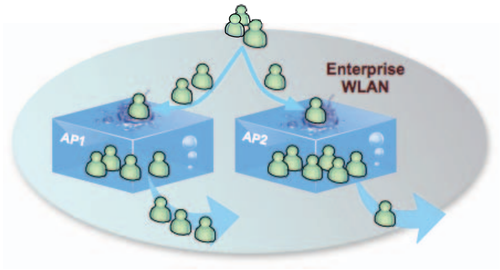


Fig. 1. Examples of AP selection for new users in an enterprise WLAN.

leave a common AP at the same time (within a short period of time). A social relationship is said to be more *intense* if the corresponding pair of users share more social activities. By mining the social relationships among users on the trace data, we find that a large fraction of users prefer to come and leave together, exhibiting strong sociality among them. For example, 60 percent users have been observed to leave a common AP together more than 10 times within one month. Motivated by this observation, we propose a novel AP allocation scheme to tackle the load balancing problem in WLANs, taking into account the social relationships of users. The core idea of this scheme is to assign users with intense social relationships to different APs so that jointly leaving of those users would have minor impact on the load balance of APs.

To implement this scheme, however, is very hard. We prove that the problem of allocating an AP for each user so that the average of the sums of social relation intensity between any pair of users in each AP is minimized is *NP-complete*. We implement an online greedy algorithm, where the pair of users with the greatest social relation intensity is dispersed. And one of them is allocated to the AP with minimum incremental sums of social relation intensity. We verify the performance of our algorithm through extensive trace-driven simulations. The results demonstrate the efficacy of our scheme. Compared with the state-of-the-art strategy in which a new user is assigned to the AP with the least workload, our algorithm always has better load balance performance and can dramatically achieve about 64.7 percent balancing performance gain during the peak hours in workdays.

The main contributions of this paper are highlighted as follows:

- We collect a large trace of AP usage in SJTU and analyze the user AP access behaviors. We find that a large fraction of users prefer to come and leave together, exhibiting strong sociality among users.
- We comprehensively analyze the traces and study impacts of the user social relations on the AP load balance. We focus on the user leaving behaviors and show that the load balancing performance of APs can be degraded by 42.6 percent.
- To keep the APs constantly balanced, we distribute users with distant social relations to APs. We formulate it as an optimization problem and prove its NP-completeness.
- We design an online greedy algorithm to tackle the problem and prove the computational cost is $O(n^2 \log(n))$, where n is the number of users. We also conduct extensive trace driven simulations. Experimental results show that, compared with the traditional selection algorithms, the balancing performance can be improved by up to 64.7 percent.

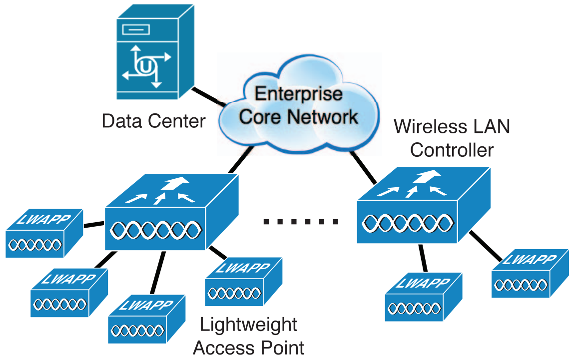


Fig. 2. The structure of WLAN in SJTU.

The remainder of this paper is organized as follows: In Section 2, we exhibit the social relations through the analysis of a real trace collect from SJTU campus and then study impacts of the social relations on the AP performance, focusing more on the user co-leaving events. In Section 3, we formally define social relations and provide the system model. We prove that the AP load balancing with social relations is a problem of NP-complete and propose an online algorithm to deal with it. Section 4 describes the methodology to evaluate the performance our AP selection algorithm and presents the results. Section 5 presents related work. Finally, we present concluding remarks and outline the directions for future work in Section 6.

2 Enterprise WLAN Trace Data Analysis

In this section, we will first introduce the AP usage trace that we collected in SJTU, and then thoroughly study the pairwise social relationships between each pair of WLAN users based on the trace. We will further investigate the impact of social behavior of users to the AP load balance performance in an enterprise WLAN at the end of this section.

2.1 Collecting Trace Data

We collected the usage log records of all WiFi APs deployed in Minhang Campus, SJTU from May 20 to June 21, 2011. The trace involves 14,669 users and 354 APs installed in 22 buildings. Fig. 2 illustrates the structure of WLAN in SJTU, which consists of WiFi APs, WLAN controllers, and a back-end data center. In the WLAN of SJTU, a set of neighboring APs are connected to a WLAN controller, which is responsible for assigning users to specific APs in its domain. In addition, all AP usage information is logged in the data center. The information contained in the trace includes *user ID* (i.e., MAC address of wireless card), connected time stamp (referred to as the time instance when a user successfully connected to an AP), disconnected time stamp (referred to as the time instance when a user disconnected from an AP), and *served traffic* (referred to as the accumulative traffic a user sent to or received from an

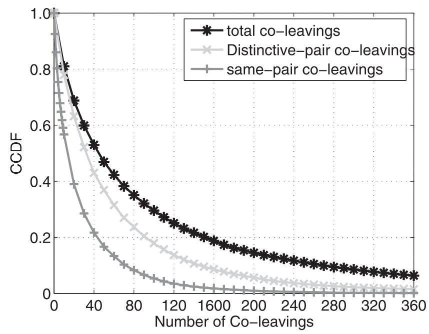


Fig. 3. The CCDF of co-leaving events.

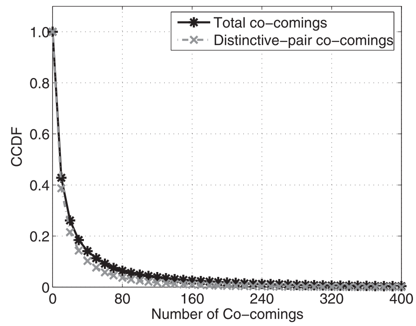


Fig. 5. The CCDF of co-coming events.

AP during a connection). Currently, a controller adopts a simple load balancing strategy based on the number of users on each AP to assign new users. As a result, a new user is always assigned to the AP with the least number of users among all the APs in the domain of this controller.

2.2 Capturing Pairwise Social Relationships

Users in an enterprise domain often have routine activities, such as classes in schools and department meetings in corporations. These social activities may have great influence on the way users use WLAN. From the perspective of AP access behavior, we study three main events in the trace data that may reflect those social activities as follows:

- *Co-coming* is referred to as the event that a pair of new users try to connect the same AP in the domain of a WLAN controller at the same time or within a short period of time. A group of new users who are more than two can be treated as different pairs in combination.
- *Encountering* is referred to as the event that a pair of users keep the connections with the same AP for a certain period of time. Notice that a co-coming does not necessarily lead to an encountering as one of two users may leave sooner than the given period of time.
- *Co-leaving* is referred to as the event that a pair of users leave the same AP at the same time or within a short period of time.

Two users are said to have a *social relationship* if they share common aforementioned events. Indeed, it is likely for two users to have common events by chance instead of attending the same social activities, especially when a relatively long time period of time is used to extract co-coming and co-leaving events. Such *fake* social relationships are random and have no capability to predict future AP access behavior of users. We consider fake social relationships as noise and diminish its effect by carefully choosing appropriate time periods for event extraction and aggregating multiple common events between the same pair of users for a single social relationship. Intuitively, more common events observed between a pair of users indicate a stronger social relationship between the two users.

Fig. 3 shows the complementary cumulative distribution function (CCDF) of total co-leaving events, co-leaving events between the same pair of users, called *same-pair co-leavings*, and co-leaving events between distinctive pairs of users, called *distinctive-pair co-leaving events* over all pairs of users, using 10 minutes as the time period to extract co-leaving events in the trace data. We observe an obvious gap between the number of total co-leaving events and that of distinctive-pair co-leaving events, which presents that a large number of same-pair co-leaving events. For example, about one-fifth of all pairs have more than 130 co-leaving events but only about 90 co-leaving events are distinctive pair, with about 40 co-leaving events are same pair. Furthermore, to investigate the probability of co-leaving events, we further plot the CCDF of ratio of the number of co-leaving events to the total number of leaving events over all users in Fig. 4, using three different time periods of time for event extraction, namely, one minute, ten minutes and half an hour. It can be seen that about 60 percent users have the probability of 0.25 to leave an AP together with other users when using ten minutes to extract co-leaving events. Analysis results on co-coming and encountering events (see Figs. 5 and 6) are similar except that new WLAN users

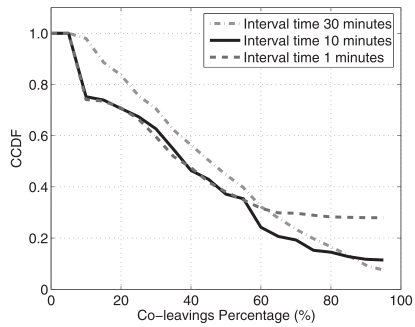


Fig. 4. The CCDF of ratio of the number of co-leaving events to the total number of leaving events.

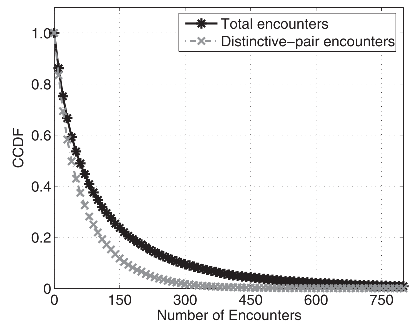


Fig. 6. The CCDF of encountering events.

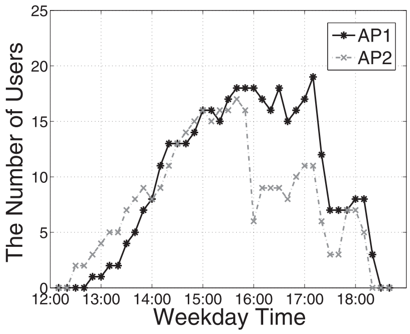


Fig. 7. The dynamic of the number of users on APs.

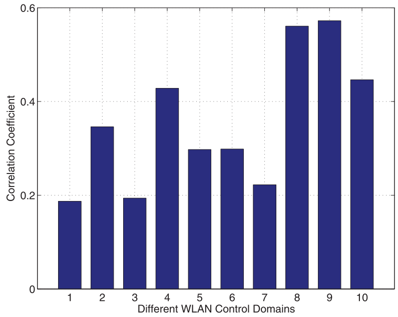


Fig. 9. The mean Pearson correlation coefficient between any pair of WLAN controllers.

demonstrate rather independent behavior (indicated by the slim gap between two CCDF plots of total co-coming events and distinctive-pair co-coming events).

In summary, all previous observations imply that most users do not leave or stay within an AP independently and show strong sociality in their AP access behavior.

2.3 Impact of Sociality on AP Load Balance

In this section, we investigate impacts of pairwise social relationships on the AP selection and load balance.

We first study co-leaving events as such events can dramatically change both the distributions of the number of users and workload among APs within a WLAN controller domain. For example, Figs. 7 and 8 show the number of users and throughput of two close APs under the same controller, respectively, from 12:00 to 18:00 on May 21. On the one hand, we find that the current AP selection strategy adopted by the controller is quite effective during the time period from 12:00 to 15:30 when both users and the traffic load are rather evenly distributed. On the other hand, when a sudden drop of the number of users from 17 to six on AP1 around 16:00 happened, we find the throughput of AP1 also dropped about 65 percent, which leads to a big throughput difference of about 20 Mbps between AP1 and AP2. It took about 2.5 hours for the current AP selection strategy to rebalance the workload between AP1 and AP2. To quantitatively study the load balance among a set of APs, we define *balancing index* as follows:

Definition 1. *Given n APs, let T_i denote the throughput of the ith AP, $i = 1, \dots, n$, the balancing index [5] is defined as*

$$\beta = \frac{(\sum T_i)^2}{n \cdot \sum {T_i}^2}.$$

We further define the normalized balancing index as

$$\overline{\beta} = \frac{\beta - \frac{1}{n}}{1 - \frac{1}{n}}.$$

This index has been widely used in literature to assess the balancing performance. Other fairness metrics, such as max-min and proportional fairness [6], [7] may be applicable and left for future work. With different load-balancing strategies, the balancing index ranges from $\frac{1}{n}$ to 1 with larger index value indicating better balance. For instance, the average balance index drops about 42 percent during the hour from 17:00 to 18:00 in the above example.

We further examine whether different WLAN controller domains experience similar co-leaving event patterns. We count the number of co-leaving events on an AP in every 10 minutes and get a vector of the number of co-leaving events for each AP. For each WLAN controller, we compute the mean Pearson correlation coefficient between any pair of APs under the same controller, using their vectors of the number of co-leaving events. Fig. 9 shows that the mean correlation coefficients over top-10 WLAN controllers with most users. It can be seen that the means of coefficients vary dramatically from 0.12 to 0.58. For a majority of controllers, the coefficients are small, which indicate highly diverse co-leaving event patterns.

We then study the impact of *net* leavings in the domain of a WLAN controller, which refer to the total number of leaving users minus that of new coming users in a WLAN controller area in a short period of time. As we observed that co-coming events are more independent, the number of net leavings heavily depends on co-leaving events. Fig. 10 shows an example of the net leavings and the

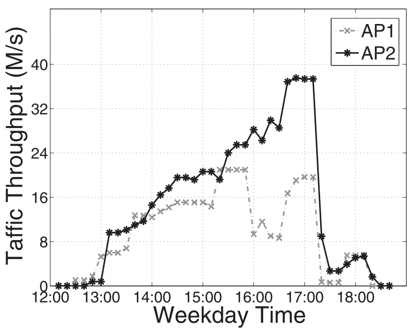


Fig. 8. The dynamic of throughput on APs.

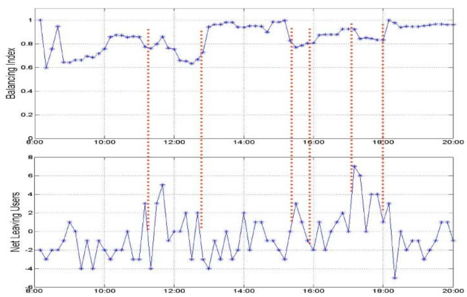


Fig. 10. An example of the net leavings and the corresponding balancing index.

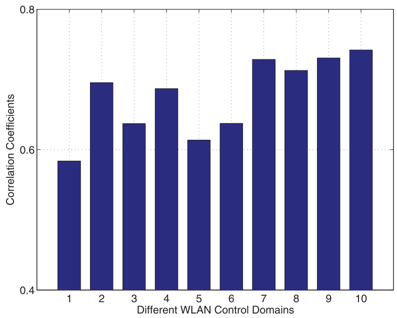


Fig. 11. The Pearson correlation coefficient between net leavings and balancing index on each WLAN controller.

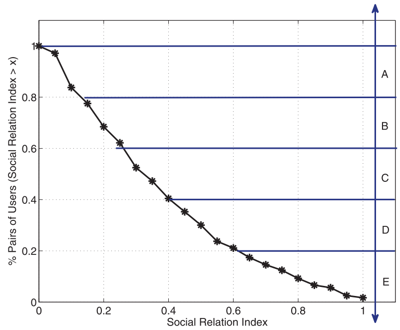


Fig. 12. The CCDF of user percentage of social relation index.

corresponding balancing index from 8:00 AM to 20:00 PM on May 21 in a WLAN controller, using 10 minutes to count net leavings. It can be seen that the net leavings have direct influence on the balancing index. As indicated by vertical lines in Fig. 10, every time a large number of net leavings will incur a drastic degradation of the balancing index (e.g., around 8:30 AM and 11:30 AM).

To further study the tight connections between the balancing index and the net leavings, we compute their Pearson correlation coefficient. Fig. 11 draws the correlation coefficients for top-10 WLAN controllers with most users. Strong correlations have been observed in all these controllers (e.g., the minimal correlation coefficient can reach 0.59), which indicates that the AP load unbalance is mainly affected by co-leaving events in social relationships.

2.4 Summary

In summary, we have the following key observations about the social relations and co-leaving events.

First of all, the co-leaving events are universal and they are one of the main causes of the unbalanced APs. In particular, the net leavings have the directed consequence on the AP unbalance.

Second, the unbalancing introduced by co-leavings can somehow be repaired by the new coming users, while the effectiveness depends on the new coming rate. When it is not sufficient, the unbalance can last long.

Third, to minimize impacts of social relations in AP balancing, users of strong social relations should be dispatched to different APs so that even they leave together, less impact is expected.

3 Social Relation-Based AP Selection

In this section, we deal with load unbalanced APs issues due to users' social activities by present a new social relation-based AP selection algorithm for WLANs. We first introduce the social relation index, a key concept in the algorithm design. We then formally define the AP selection problem with social relations and prove its NP-completeness. In the last, we propose a simple yet effective algorithm and prove that the algorithm has the complexity of $O(n^2 log(n))$, where n is number of total users.

In the following, we start with discussing social relation index.

3.1 Strength of Social Relationships

As co-leaving events are the major factor that dramatically affects the AP load balance, we define social relation index to quantify the strength of a social relationship as follows:

Definition 2. *Given two users u and v, the social relation index between them is defined as*

$$\delta(u,v) = P(L(u,v) \mid E(u,v)),$$

where $L(u,v)$ and $E(u,v)$ denote the co-leaving and encountering events between u and v, respectively.

In other word, $\delta(u,v)$ is the conditional probability that u and v encounter at the same AP and then leave the AP in unison. Thus, a high social relation index implies a stronger relation between users and vice versa. We consider the conditional probability of co-leaving events is because co-leaving events are the main reason that causes dramatic unbalanced workload among APs in the same controller domain.

Fig. 12 plots the CCDF of the social relation index δ for every pair of users with co-leavings. A straightforward finding is that the relations between users are quite tight. For some pairs of nodes, nearly 35 percent encounters have 50 percent probability to incur a later co-leaving, and 20 percent encounters have the probability over 60 percent.

To investigate the fundamental traffic characteristics of users with different social relations, we partition users into five groups. Each group accounts about 20 percent of users and is labeled from A to E. The ranges of the social relation index for different groups are also depicted in Fig. 12. We compute the traffic volume Pearson correlation coefficients between users in the same group and draw the mean of the coefficient for each group in Fig. 13. We find that, in general, higher social relation users exhibit higher coefficients (group D and E), implying that users with tighter relations have more similar traffic demands than strangers. One of the possible reasons is that users with tighter social relations may subscribe the AP for similar targets and do similar jobs through WLANs, say, classmates in a class. They should be distributed to different APs not only because of their co-leaving properties, but also the similar traffic demand patterns. Users of looser social relations have no such properties.

3.2 Problem Statement

In this section, we formally define the AP selection problem with social relation consideration. The original

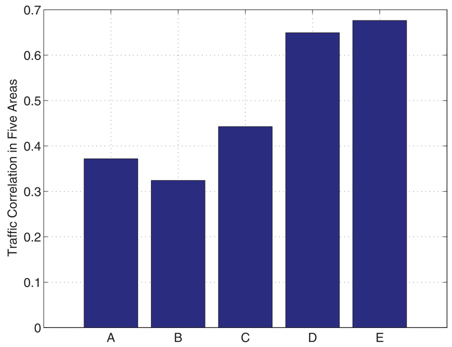


Fig. 13. The average Pearson correlation coefficient between user traffic with different social relation index in five areas.

AP balancing problem is to distribute the subscripting users to different APs so that all the APs are kept balanced in all the time, i.e., $\min(\sum \beta)$. In practice, this problem has no optimal solution because the optimal solution requires the exact leaving time for each user. Such information is of the future and can never be obtained. Fortunately, we recall that the main cause of the AP unbalance is the user co-leaving in social relations is (Section 4.2). Alternatively, we take another approach toward the optimal AP balance scheme. We try to distribute user pairs of tighter social relations to different APs so that users in the same APs are diversified. In other words, they have fewer social relations and will be unlikely to present similar access behaviors. The problem is formally defined as follows:

Social relation AP selection problem (SORAP): Given a number n of users willing to subscribe to M APs from AP_1 to AP_M, suppose each user $u \in [1, n]$ has a demanded throughput $w(u)$ and there is a social relation index between any pair of users $\delta(u, v), u, v \in [1, n]$. Assume the bandwidth of the APs are $W(i), i = [1, M]$, the problem is to find an allocation for each user to an AP such that

$$\min : \sum_{i=1}^{M} \sum_{\forall u,v \in A_i} \delta(u, v)$$

$$subject : \sum_{u \in AP_i} w(u) < W(i), i \in [1, M].$$

Here, the constraint $\sum_{u \in AP_i} w(u) < W(i)$ is due to the fact that the aggregated throughput demands cannot exceed the provided AP bandwidth.

Theorem 1. *SORAP is NP-complete.*

Proof. We prove the theorem by a reduction from the weighted maximum cut problem, a well-known NP-complete problem. Given a weighted undirected graph $G(V, E, W)$, where V is the set of nodes, E is the set of edges, and W is the set of weights, the maximum cut is to partition the nodes into a number N of disjointed sets so that the sum of link weights in each set is minimal.

The reduction is as follows: Given the set of users, we construct a graph $G'(V', E', W')$, where V' is the set of users, E' is the set of edges, and W' is the set of weights. If user u and user v have a positive relation index $\delta(u, v)$, then there is an edge $e_{uv} \in E'$ and the weight on this edge is $\delta(u, v)$. We first assume that each AP has infinite bandwidth. The SORAP of assigning all users to M APs is to minimize $\sum_{i=1}^{M} \sum_{\forall u,v \in AP_i} \delta(u, v)$, which is equal to find the maximum cut in the graph G' so that the sum of relation index of users in each AP is minimal. Therefore, the SORAP is NP-hard even with unlimited bandwidth of APs.

For the verification of the problem, given an allocation and a threshold, we can easily compute the $\sum_{i=1}^{M} \sum_{\forall u,v \in AP_i} \delta(u, v)$ in polynomial time and check whether it is below the threshold or not. Therefore, the SORAP can be verified in polynomial time.

Combining these two statements, SORAP is NP-complete. This completes the proof. □

AP SELECTION ALGORITHM

$Input$: $\delta(u, v)$ for all users u and v
$Output$: $S(AP_i)$

initial $\Delta = \{\delta(u, v) | \forall i, u \notin S(AP_i) \cup v \notin S(AP_i)\}$
initial $T(AP_i) : \sum_{u \in AP_i} w(u)$
update Δ upon receiving new users
while $(\Delta \neq \emptyset)$
  $\{u, v\} = \arg\max_{u,v}(\delta(u, v) \in \Delta)$
  if $\forall i$, $u \notin S(AP_i)$ do
    for each $AP_i, i = 1, ..., m$
      $C_i(u) = \sum_{\forall w \in S(AP_i)} \delta(u, w)$
      if $T(AP_i) + w(u) \leq W(i)$ then $C_i(u) = \infty$
    $\kappa = \arg\min_i C_i(u)$
    $S(AP_\kappa) = S(AP_\kappa) + u$
    $T(AP_\kappa) = T(AP_\kappa) + w(u)$
    if $\exists S(AP_i)$, $v \in S(AP_i)$, then $\Delta = \Delta - \delta(u, v)$
end while

Fig. 14. The pseudocode of a AP selection algorithm.

3.3 Online Greedy Algorithm

In this section, we propose an online algorithm called Social Relation Distribution (SRD) to solve the SORAP problem. We start from the design principles, and then present the detailed algorithm. At last, we analyze the performance of the algorithm and show that it has a conditional approximation factor.

3.3.1 Design Principles

In general, the SORAP problem is to distribute users in different APs so that the social relation index δ between users within the same AP is minimized. Toward this goal, user pairs with tight δ should be dispersed in different APs. We assign users to APs so that the minimum increment on total δ is incurred.

3.3.2 Online Algorithm

For ease of presentation, we use the following notations. Let $S(AP_i), i = 1, \ldots, M$ be the set of users associated in the ith AP. For the AP selection algorithm, $S(AP_i)$ will be the final output from which users can determine the AP it associates with. Fig. 14 shows the pseudocode of the AP selection algorithm. Notice that this is an online algorithm, and therefore $S(AP_i)$ will not be empty initially. Let $T(AP_i) = \sum_{u \in S(AP_i)} w(u)$ be the traffic at the APs, and Δ be the set of social relation index between users that have no associated APs, i.e.,

$$\Delta = \{\delta(u,v) \mid \forall AP_i, u \notin S(AP_i) \cup v \notin S(AP_i)\}.$$

The algorithm starts by initializing the social relation index set Δ. Upon receiving the subscription requests from new users, the corresponding social relation index will be added to its set Δ. The algorithm then enters the main loop from lines 4 to line 14. It will continue the loop until the set Δ is empty. In the loop, at line 5, we retrieve the maximal δ from Δ, i.e.,

$$\{u,v\} = \arg\max_{u,v}(\delta(u,v) \in \Delta).$$

The next following operations will then to distribute u or v when it has no association. Without loss of generality, suppose u has no associations (line 6). Then, for each AP_i, we compute a cost function $C_i(u)$ as the estimated increment of the total social relation index when u is added to AP_i, i.e.,

$$I_i(u) = \sum_{\forall w \in S(AP_i)} \delta(u,w).$$

Notice that the cost will be set as infinite if the bandwidth constraint cannot be satisfied (lines 8 and 9). The demand of each user bandwidth $W(u)$ can be estimated by user's history trace as studied in work [8]. Among all $S(AP_i)$, we find the one with the minimal cost C_i and distribute the user u in it (lines 10-11). We will then update the traffic of the AP accordingly. In the last line, we update the social index set Δ(line 8). When both u and v have AP associations, we remove $\delta(u,v)$ from Δ. the algorithm terminates when Δ is empty.

We then analyze the total time complexity of this online algorithm. Indeed, the most costly operation is the retrieval of the max social relation index $\delta(u,v)$, which needs the sort operation (line 5). It takes $O(|\Delta| \log |\Delta|)$ time where is the number of social relation indexes. It can be up to n^2, where n is the number of users. Therefore, the cost of the sort is $O(n^2 \log n)$. From lines 7 to 9, we need compute the cost function. As in these operations, each $\delta(u,w)$ can be calculated at most once. Therefore, the total cost is no more $O(|\Delta| + M)$, where m is the number of APs. The other operations from 9 to 13 are all single operations and take constant time. With all these analysis, the worst running time for the AP selection algorithm is $O(n^2 \log n + M)$.

3.3.3 Algorithm Conditional Approximation

In this part, we provide a conditional approximation factor of our AP selection algorithm. When the constraint on the AP bandwidth is always satisfied, we have the following theorem. More general performance guarantees are left for future work.

Theorem 2. *SRD has the approximation factor of $\frac{M}{M-1}$, where M is the number of APs.*

Proof. For a new user u, if u is associated with ith AP, we define the internal social relations (ISW) with respect to ith AP as

$$I_i(u) = \sum_{\forall w \in S(AP_i)} \delta(u,w), \tag{1}$$

and the external social relations (ESW) as follows:

$$E_i(u) = \sum_{\forall w \in S(AP_j), i \neq j} \delta(u,w). \tag{2}$$

Notice that the ISW is just the cost function we defined in the algorithm design (line 8). Observing (2) and (1), we have the connections between $I_i(u)$ and $E_i(u)$ as

$$E_i(u) = \sum_{j \neq i} I_j(u). \tag{3}$$

Defining $U(u) = \sum_{\forall v} \delta(u,v)$, we have

$$U(u) = E_i(u) + I_i(u).$$

In line 10 in Fig. 14, we select the AP with the minimal cost function $I_i(u)$ to associate. As there are M APs to join, we have

$$I_i(u) <= \sum_{j=1}^{M} \mathrm{S}(\mathrm{AP}_j)/M. \tag{4}$$

Because of the (3), when the total internal social relations increases by $I_i(u)$, the total external social relations will increase by at least $(M-1)I_i(u)$. When the algorithm terminates and all users have been associated, we have the relations between the total internal social relations and total external social relations as (4). Thus, consequently, at termination, $\sum_{\forall u} I_i(u) \le \sum_{\forall u} E_i(u)/(M-1)$. Notice that the optimization goal in the problem statement is just to minimize the total internal social relations $\sum_{\forall u} I_i(u)$, or equivalently speaking, to maximize the total external social relations $\sum_{\forall u} E_i(u)$. Let F^* denotes the optimal solution. It has the maximum total external social relations being less than $\sum_{\forall u} I_i(u) + E_i(u)$. In other words,

$$\left| \frac{F^* - \sum_{\forall u} E_i(u)}{F^*} \right| <= 1/M.$$

Thus, the approximation factor of SRD is $\frac{1}{1-\frac{1}{M}} = \frac{M}{M-1}$. This completes the proof. □

3.4 Discussion

The online algorithm tries to associate new users to the AP so that the increment of the internal social relations is minimized. The rationale behind this strategy is to assign users with strong social relationship to different APs because they tend to come or/and leave together, which may cause unbalanced workload of APs. Our algorithm can be implemented on WLAN controllers, where most recent historical login records of users can be collected and used for extracting social relation indexes between users. We will further investigate how much historical data should be used for our online algorithm in the following section via extensive trace-driven simulations.

Generally speaking, our algorithms can achieve much performance gain in scenarios where users show obvious social relationships. In public WLANs such as cafes and airports, however, users tend to be more random compared with the case in enterprises. It is not clear whether users in those WLANs also present strong sociality. The study is beyond the scope of this work and we will leave the investigation of relationships among users in public WLANs for future work.

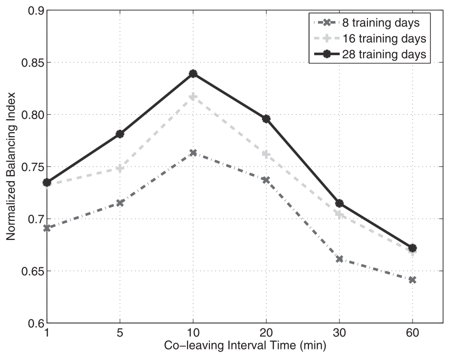


Fig. 15. The balancing index versus the length of co-leaving extraction time interval.

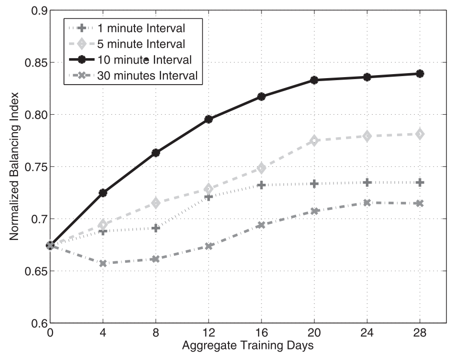


Fig. 16. The balancing index versus the age of historical data.

4 Performance Evaluation

In this section, we evaluate users' sociality based AP selection algorithm. We first introduce the evaluation methodology, and then present the experimental results.

4.1 Methodology

We evaluate our SRD AP-selection algorithm based on trace-driven simulations. We use the same trace as described in Section 2 and use four-week trace data from May 20 to June 17, 2011 as the learning stage for establishing social relationships between users, leaving the trace data from June 18 to June 21 for AP selection experiments.

We compare SRD AP-selection algorithm with the state-of-art arrival-based algorithm, called Least Loaded First (LLF) [1], where a new coming user is allocated to the AP with the least workload. We consider the balance index of throughput among all APs in WLAN controller domains to evaluate the performance of our SRD and the LLF algorithms.

4.2 Effect of Extraction Time Interval Length

We first examine the effect of the length of the time intervals used to extract events on the load balancing performance. As described in Section 2, an appropriate time interval helps reduce fake social relationships. We vary the length of time interval for event extraction from 1 minute to 1 hour. For each time interval, we extract co-leaving events using different amount of historical data and assign all users to APs using the experimental data.

Fig. 15 plots the average normalized balancing index over all WLAN controllers. It can be seen that, as the length of extraction time interval gets larger, the normalized balancing index first increases to a maximum when ten minutes are used to extract events and then decreases. The reason is that an inappropriately large time interval will generate more co-leaving events, which facilitates the calculation of social relationship strength between users. On the other hand, it will also introduce more fake social relationships as the chance for two non-related users to share a common co-leaving event increases, which results inaccurate social relationships. In contrast, a smaller time interval can result more accurate social relationships but the number of such relationships, which is important to SRD algorithm, is less. In addition, we can see that more historical data result better load balance performance.

4.3 Effect of Training Stage

In this experiment, we examine the effect of the amount of historical data for establishing social relationships to the load balance performance. We choose different the time intervals to extract co-leaving events and increase the time for training social relationship index from last one day (i.e., June 17), last two days (i.e., June 16-June 17) till last 28 days (i.e., May 20-June 17).

Fig. 16 shows the average normalized balancing index over all WLAN controllers as a function of historical data. For all time intervals, as more historical data are available, the normalized balancing index increases and stabilizes when the length of training stage reaches about four weeks. This implies that information older than four weeks does not help in capturing social relationships but does not hurt either.

4.4 Comparison with LLF

In this simulation scenario, we use 10 minutes as the time interval to extract co-leaving events and take all training data for establishing pairwise social relationships. We use the same experimental data for SRD and LLF to assign users to APs.

Fig. 17 shows the average normalized balancing index over all WLAN controllers and all experimental data as a function of time in daytime. There are two main observations found in Fig. 17. First, it can be seen that SRD outperform LLF over most time. On average, SRC can achieve about 24.2 percent balancing index gain compared with LLF. Second, the performance of SRD is more stable and robust against user behavior than that of LLF. Especially, SRD performs well when suffering co-leaving

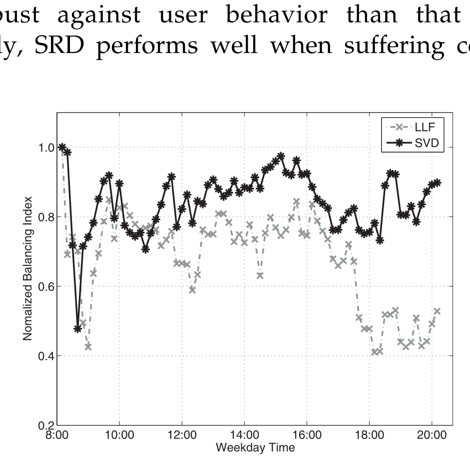


Fig. 17. The balancing index varying under two methods in simulation.

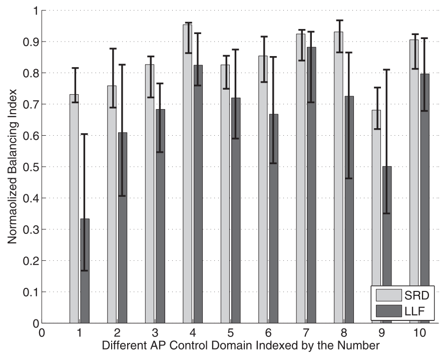


Fig. 18. The average balancing index.

events. For example, in SJTU, time periods from 12:00 to 16:00 and from 17:00 to 19:30 are peak hours for classes, against which SRD can achieve about 64.7 percent balancing index gain against LLF. The reason is that SRD can effectively cancel the negative impacts of social relations on AP load balance.

To further demonstrate the effectiveness of SRD on all sites, Fig. 18 shows the average normalized balancing index on all sites with 95 percent percentage confidence error bar. We find that SRD shows much higher stability than the LLF. The error bar can be reduced by 87.6 percent overall. The average normalized balancing index can be improved by 24.2 percent overall.

5 Related Work

In this section, we introduce some related works. The existing works can be classified into two classes.

The first class of works focuses on the AP balancing in WLAN. The existing algorithms can be further divided into two subclasses. One attempt to adjust the users during the runtime [9], i.e., moving users from one AP to another. These approaches have been widely challenged due to the large computational overhead. In addition, users will experience disruptions during the immigration, which is undesired or even unacceptable. Another majority of approaches distribute user when they arrive [10]. In the work [11], an AP-selection strategy is proposed that takes two major factors into account. One is the individual user throughput, and the other is the impact on the performance of other users who are already associated with the AP. Nicholson et al. [3], [12] design the AP selection algorithm based on the assessment of the AP. In its design, a node tentatively associates with each of the AP and evaluates the bandwidth and delay of the AP toward the Internet. The AP with a better service quality will be selected. A load-balance scheme [13] provides inter-AP fairness by adjusting the transmission power of the AP beacons while minimizing the load of the congested APs in the network. In [5], a distributed load balancing system is introduced that takes into account the AP loads defined as the aggregated uplink and downlink traffic through the AP. The author in [14] proposes to balance the load on the basis of the entire network. In the design, demand clusters will select the AP that is able to service the demand clusters and provide ample bandwidth, rather than the closest AP that often has the largest signal level. The author in the paper [4] proposes an online AP association strategy that maximizes the minimum throughput among all users at the cost of an acceptable overhead. A distributed algorithm [15] is proposed where the APs can tune their cell size according to their load and their neighbor's loads in a way transparent to the end users and thus improve the fairness and performance. In [16], the authors proposed a unique solution called Smart Access Point (SAP), which smartly balancing the network load across multiple interfaces based on users' time-varying traffic load conditions. In [17], 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. All these approaches, however, only consider the arrival of the users, and none of them have taken the social relations between users into account.

The other category studies the social relations in users [18]. Hsu et al. [19] studied the user behaviors in AP access and explicit patterns are observed. In [20], the authors tried to understand the interuser interaction in wireless environments by investigating the internode encounters. The Small World concept is explored. Furthermore, the author displays the feasibility of an infrastructure-less network, where most of the nodes can be reached through utilizing internode connectivity and encounters. These works only consider the social relations among users and never study the impact of such relations to the AP selection and balance in WLANs [11].

6 Conclusion and Future Work

In this paper, we have studied the impact of sociality of users on the AP selection and load balance in enterprise WLANs. By systematically analyzing the social relation behaviors through investigations on a real collected trace, we have found that many users exhibit strong sociality in their AP access behavior, such as leaving in unison, leading to sudden unbalanced work load among APs. We have proved the problem of distributing users with social relationships to APs is *NP-complete* and proposed an online greedy AP selection algorithm. Trace-driven simulation results show that our algorithm can achieve about 64.7 percent balancing index gain on average during peak hours in workdays comparing with the state-of-the-art AP selection strategy.

In our future work, we will further study the social relations on the WLAN performance. We will investigate the user social behaviors in each site and in each AP, mine their social behavior patterns, and explore more implications in the AP user management design. AP selections based on other fairness metrics such as proportional fairness is also a good topic.

Acknowledgments

This research was supported by the State High-Tech Development Plan (2013AA01A601), NSFC Grants 61170237, 61202375, and 61170247, and STCSM (12ZR1414900).

References

[1] G. Judd and P. Steenkiste, "Fixing 802. 11 Access Point Selection," *ACM SIGCOMM Computer Comm. Rev.*, vol. 32, no. 3, pp. 31-31, 2002.

[2] A. Balachandran, P. Bahl, and G. Voelker, "Hot-Spot Congestion Relief in Public-Area Wireless Networks," *Proc. IEEE Fourth Workshop Mobile Computing Systems and Applications*, pp. 70-80, 2002.

[3] A. Nicholson, Y. Chawathe, M. Chen, B. Noble, and D. Wetherall, "Improved Access Point Selection," *Proc. ACM Fourth Int'l Conf. Mobile Systems, Applications and Services*, pp. 233-245, 2006.

[4] F. Xu, C. Tan, Q. Li, G. Yan, and J. Wu, "Designing a Practical Access Point Association Protocol," *Proc. IEEE INFOCOM*, pp. 1-9, 2010.

[5] H. Velayos, V. Aleo, and G. Karlsson, "Load Balancing in Overlapping Wireless Lan Cells," *Proc. IEEE Int'l Conf. Comm.*, vol. 7, pp. 3833-3836, 2004.

[6] Y. Bejerano, S. Han, and L. Li, "Fairness and Load Balancing in Wireless LANs Using Association Control," *Proc. ACM 10th Ann. Int'l Conf. Mobile Computing and Networking*, pp. 315-329, 2004.

[7] J. Kleinberg, E. Tardos, and Y. Rabani, "Fairness in Routing and Load Balancing," *Proc. 40th Ann. Symp. Foundation Computer Science (FOCS)*, pp. 568-578, 1999.

[8] Y. Qiao, J. Skicewicz, and P. Dinda, "An Empirical Study of the Multiscale Predictability of Network Traffic," *Proc. IEEE 13th Int'l Symp. High Performance Distributed Computing*, pp. 66-76, 2004.

[9] D. Rathunde, "Dynamic Load Balancing During Message Processing in a Wireless Communication Service Network," US Patent 6,574,477, June 2003.

[10] K. Mittal, E. Belding, and S. Suri, "A Game-Theoretic Analysis of Wireless Access Point Selection by Mobile Users," *Computer Comm.*, vol. 31, no. 10, pp. 2049-2062, 2008.

[11] X. Wan, X. Wang, U. Heo, and J. Choi, "A New Ap-Selection Strategy for High Density IEEE 802. 11 Wlans," *Proc. Int'l Conf. Cyber-Enabled Distributed Computing and Knowledge Discovery*, pp. 52-58, 2010.

[12] I. Jabri, N. Krommenacker, T. Divoux, and A. Soudani, "IEEE 802.11 Load Balancing: An Approach for Qos Enhancement," *Int'l J. Wireless Information Networks*, vol. 15, no. 1, pp. 16-30, 2008.

[13] Y. Bejerano and S. Han, "Cell Breathing Techniques for Load Balancing in Wireless LANs," *IEEE Trans. Mobile Computing*, vol. 8, no. 6, pp. 735-749, June 2009.

[14] R. Akl and S. Park, "Optimal Access Point Selection and Traffic Allocation in IEEE 802.11 Networks," *Proc. Ninth World Multiconf. Systemics, Cybernetics and Informatics (WMSCI '05): Comm. and Network Systems, Technologies and Applications*, vol. 8, pp. 75-79, 2005.

[15] E. Garcia, R. Vidal, and J. Paradells, "Cooperative Load Balancing in IEEE 802.11 Networks with Cell Breathing," *Proc. IEEE Symp. Computers and Comm. (ISCC '08)*, pp. 1133-1140, 2008.

[16] X. Chen, Y. Zhao, B. Peck, and D. Qiao, "Sap: Smart Access Point with Seamless Load Balancing Multiple Interfaces," *Proc. IEEE INFOCOM*, pp. 1458-1466, 2012.

[17] K. Wu, H. Li, L. Wang, Y. Yi, Y. Liu, Q. Zhang, and L.M. Ni, "HJam: Attachment Transmission in WLANs," *Proc. IEEE INFOCOM*, pp. 1449-1457, 2012.

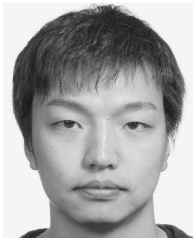
[18] M. Motani and V. Srinivasan, "Peoplenet: Engineering a Wireless Virtual Social Network," *Proc. ACM MobiCom*, pp. 243-257, 2005.

[19] W. Hsu, D. Dutta, and A. Helmy, "Mining Behavioral Groups in Large Wireless Lans," *Proc. 13th Ann. ACM Int'l Conf. Mobile Computing and Networking*, pp. 338-341, 2007.

[20] W. Hsu and A. Helmy, "On Nodal Encounter Patterns in Wireless LAN Traces," *Proc. Fourth Int'l Symp. Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, pp. 1-10, 2006.

Guangtao Xue received the PhD degree in computer science from Shanghai Jiao Tong University in 2004. He is currently an associate professor in the Department of Computer Science and Engineering at the Shanghai Jiao Tong University. His research interests include mobile networks, social networks, wireless sensor networks, vehicular networks, and distributed computing. He is currently the associate director of Network Computing Center, Shanghai Jiao Tong University. He is a member of the IEEE Computer Society and Communication Society, and the IEEE.

Qi He received the BS degree in computer science from ZheJiang University in 2009, and is currently working toward the master's degree at Shanghai Jiao Tong University under the supervision of Prof. Guangtao Xue. His research interests include data mining, social networks, delay-tolerant networks, and cloud computing. Currently, his research focuses on the improvement of wireless access point load balancing and data mining from user-based 3G mobile networks.



Hongzi Zhu received the PhD degree in computer science from Shanghai Jiao Tong University in 2009. He is an assistant professor in the Department of Computer Science and Engineering at the Shanghai Jiao Tong University. His research interests include mobile networks, vehicular networks and network security. He is a member of the IEEE, IEEE Computer Society, and Communication Society.

Tian He received the PhD degree under Professor John A. Stankovic from the University of Virginia, Virginia in 2004. He is currently an associate professor in the Department of Computer Science and Engineering at the University of Minnesota-Twin City. He is the author and coauthor of more than 100 papers in premier sensor network journals and conferences. He received the US National Science (NSF) CAREER Award 2009 and McKnight Land-Grant Professorship. He served a few program chair positions in international conferences and on many program committees, and also currently serves as an editorial board member for four international journals including ACM Transactions on Sensor Networks. His research includes wireless sensor networks, intelligent transportation systems, real-time embedded systems and distributed systems. He is a senior member of the IEEE.

Yunhuai Liu received the BE degree in computer science and technology from Tsinghua University, Beijing in 2000, and the PhD degree in computer science and engineering from the Hong Kong University of Science and Technology (HKUST) in 2008. From 2008 to 2010, he was at HKUST as a research assistant professor. From 2010 to 2011, he was at Shenzhen Institutes of Advanced Technology (SIAT), Chinese Academy of Sciences (CAS) as an associate professor. Currently, he is a researcher in Shenzhen Institutes of Advanced Technology (SIAT), Chinese Academy of Sciences (CAS). His research interests include wireless sensor networks, cognitive radio networks, and extreme-scale data center and data networks. He is a member of the IEEE.