Improving the latency of 802.11 Hand-offs using Sentinel based Architecture

Lenin Ravindranath, Fredrick Prashanth, Leo Prasath, Praveen Durairaj, Arul Siromoney
Department of Computer Science and Engineering, Anna University, Chennai, India

Abstract

The IEEE 802.11 based wireless networks have seen rapid growth and deployment in the recent years. WLAN based networks have been deployed in many enterprises and for public Internet access services. Critical to the 802.11 MAC operation, is the *hand-off* function which occurs when a mobile node moves its *association* from one *access point* to another. The small cell size of WLAN creates frequent hand-offs for mobile users. Multimedia and Voice applications require hand-offs between access points to be fast to maintain the quality of the connections. Previous studies have shown that the latency of these hand-offs are high which is dominated by the scanning phase where the client scans to discover the candidate set of next access points.

In this paper, we present a *sentinel* based approach to reduce the scanning latency. Sentinels are stationary nodes in the network which monitors the WLAN continuously. During hand-off the client probes the sentinel to get the minimal set of channels to scan. The sentinels provide the client with a list of channels which is sorted on the best AP the client can associate with at that point of time. The APs are sorted based on the neighbor graph, the client's current location, the nearest hand-off points and the AP load which is continuously monitored by the sentinels. Our simulation results show that, about 90% of the time, the client associates with the first AP on the sorted list provided by the sentinels.

Keywords

Wireless LAN, Fast Hand-off, Scanning, Sentinel, WLAN monitoring, Hand-off points, Location Determination

1. Introduction

The popularity of IEEE 802.11 networks [1] is rapidly increasing and is been deployed in many organizations and enterprises. 802.11 permits speeds up to 11Mbps in 802.11b and up to 54 Mbps for 802.11g/a [2]. The high speed of WLAN networks provides transparent connectivity to the wired Internet and enables mobile users to use multimedia and voice applications.

The main issue of 802.11 WLAN networks is the Hand-off between APs. In WLAN, a hand-off can be defined as the process of leaving a basic service set of an access point to enter a new one. A hand-off is triggered by the degradation of the signal quality which falls below a predefined threshold. Especially, for real time multimedia service such as VoIP, the hand-off between access points should be fast to maintain the quality of the connections.

At MAC layer, the hand-off process as defined in the IEEE 802.11 Standard [1] can be decomposed into three phases: scanning, authentication and reassociation. In the first phase, a client scans for APs by either sending ProbeRequest messages (Active Scanning) or by listening for Beacon messages (Passive Scanning) in all the channels. After scanning all channels, an AP is selected by the client using the Received Signal Strength Indication (RSSI), link quality, etc. In the authentication phase the selected AP exchanges IEEE 802.11 authentication messages with the client. Finally, if the AP authenticates the client, the client sends ReassociationRequest message to the new AP.

Previous studies have shown that the latency of the hand-offs are high and is dominated by the scanning phase [3]. Thus, reducing the scanning latency reduces the hand-off latency and helps smooth hand-off between Access Points.

In this paper, we present the sentinel based architecture to reduce the scanning latency. Sentinels are stationary nodes which cooperate to monitor the wireless network continuously. Sentinels are deployed in such a way that it covers the entire WLAN network. Sentinels help a client by providing it with a minimal set of channels during hand-off. The list of channels is sorted on the best AP the client can associate with at that point of time. The APs are sorted based on the neighbor graph [5], the client's current location, the nearest hand-off points and the AP load which is continuously monitored by the Sentinels.

The rest of the paper is organized as follows. In Section 2 we discuss the related work. We present the Sentinel based Architecture in Section 3 and the Hand-off Scheme in Section 4. In Section 5 we describe the Hand-off engine which keeps track of the set of sorted APs for all the clients continuously. In Section 6 we show the effectiveness of the approach using results of the simulation. Section 7 concludes our paper with the proposed future work to improve the Hand-off scheme further.

2. Related Work

Several schemes have been proposed [4] - [9] to reduce the latency of the scanning phase, the authentication phase during hand-off and thus reducing the overall latency.

SyncScan mechanism was proposed by Ishwar et al. [4] in which clients keeps track of the signal strengths of all the APs at any point of time. This method proved to avoid the entire scanning latency but increased the overhead of the clients and was heavily dependent on the synchronization of different APs in sending Beacon packets.

Mishra et al. proposed a mechanism of selective scanning using the construction of Neighbor Graph [5]. The neighbor graph is constructed dynamically which specifies the APs to which hand-off was done from the current AP. The neighbor graph is distributed to the clients during association to an AP. The client then scans only the APs in the neighbor graph thus decreasing the latency of scanning.

Kevin et al. proposed a selective scanning mechanism using Sensor Networks [6] where a client probes a sensor for the list of channels to scan during hand-off. The sensor constructs the neighbor graph and sends the list of adjacent APs to scan.

On the other hand various methods based on PreAuthentication have been proposed to decrease the Authentication and Association latency during hand-off [7] - [9].

3. Design and Architecture

In this section, we discuss our Sentinel based Architecture for helping clients during hand-off. Fig. 1 shows our basic Architecture. It has the following components.

3.1 Sentinels

Sentinels in literature are monitors or watchdogs which cooperate to accomplish a job. In our model, Sentinels do a similar work to help accomplish efficient hand-offs. Sentinels here are stationary nodes which continuously monitor the entire WLAN network and help the clients during hand-off.

In an enterprise which has both wired and wireless LAN networks, there are plenty of desktop systems connected through the wired network. As proposed by Alec et al. in their latest research [10], the stationary desktop machines attached with inexpensive wireless cards can be used as Sentinels to monitor the wireless network and help the mobile client during hand-off. Also, the desktop machines have high processor speed, more idle time, fast wired connectivity and the dense deployment of desktops in an enterprise or a university enables us to cover the whole WLAN network for monitoring and helping.

Sentinels do the following two functions:

1. **Monitoring:** It actively monitors the WLAN network by sniffing all the packets seen in the air in promiscuous mode. It summarizes the packets sniffed and dumps the data into various tables in the Data Collection Server (DCS). All sentinels cooperate to monitor the network by dumping data into the same Data Collection Server.

2. **Helping:** It helps the client during hand-off by providing it with a minimal set of channels to scan. The sentinels contact the Hand-off engine to get the minimal sorted set of channels to scan to give it to the client.

Sentinels are deployed in such a way that it covers the entire network. We assume that, there is atleast one sentinel which sniffs in the channel of an AP and thus a client will always find a sentinel in the current AP channel to get the minimal sorted set of channels to scan. The density of desktop systems in an enterprise or a university enables us to easily deploy such architecture.

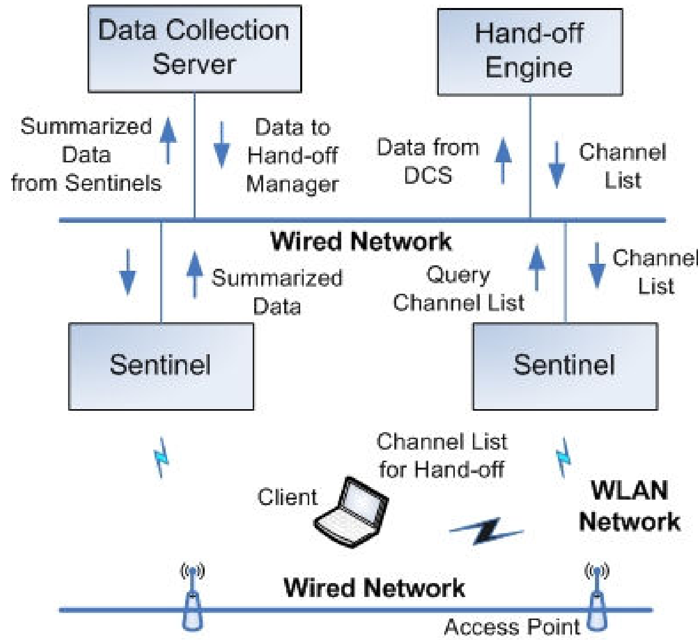

Figure 1: Sentinel based Architecture for efficient Hand-off

3.2 Data Collection Server (DCS)

The sentinels summarize the sniffed packets for a period of time and dump the data into the tables in the Data Collection Server. The following are the tables maintained at the DCS which the sentinels update:

1. **WLAN Traffic Table:** The MAC addresses of the sender and the receiver, the packet count and the number of bytes seen are recorded in the WLAN Traffic table. This table is used to infer the AP load (the number of clients associated with an AP and the number of bytes sent from and to an AP)

2. **Signal Strength Table:** The signal strength of the clients as monitored by the sentinels and the signal strength of the associated AP as monitored by the client is recorded in this table. This table is used to find the clients current location. Thus the client helps the sentinels to find its location by sending the signal strength of the associated AP to the DCS through the AP[1].

3. **Hand-off Info Table:** The ReAssociation and ReAuthentication packets seen are dumped into this table by the sentinels. A hand-off is identified by these packets. This table is used to construct the neighbor graph [5]. Signal Strength Table and the Hand-off Info table are used together to find the Hand-off points.

All the packets are recorded with the timestamp and we assume that the timing is synchronized with respect to all sentinels and clients. The duplicate packets sniffed by different sentinels on a single AP channel are removed in the DCS using the sequence number of the packets.

[1]We assume that the wired network is connected to the wireless network

3.3 Hand-off Engine

The Hand-off engine works with the data collected in the DCS to find the client's location, neighbor graph, AP load and to track hand-off points. Using these data the hand-off engine continuously creates a sorted list of best APs for each client to scan. During hand-off, the hand-off engine provides the client (through the sentinel) with the current list of best APs (channels) to scan. Thus the hand-off engine is the heart of our architecture and we elaborate the design of the hand-off engine in Section 5.

The design of our architecture is such that it can support all the future work mentioned in Section 7 and thus proves to be scalable.

4. Hand-off Scheme

This section explains the Hand-off scheme in our architecture. We modify the client such that the following hand-off mechanism happens. The Hand-off scheme is depicted in Fig. 2

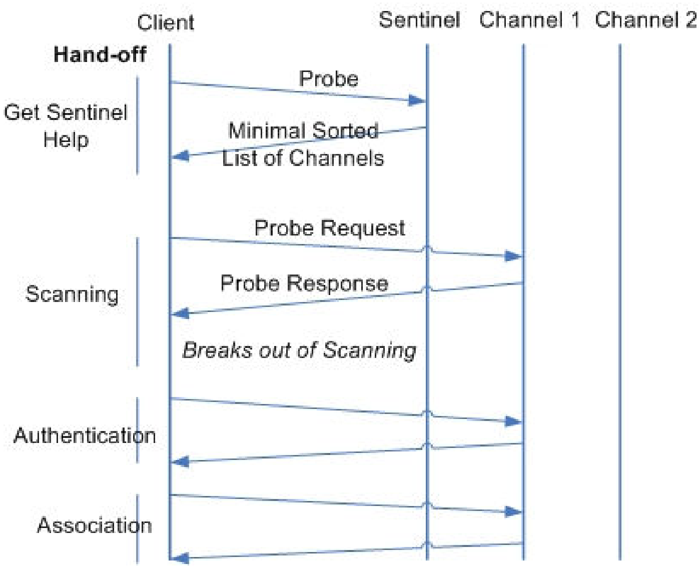

Figure 2: Hand-off Scheme. The sentinel sends a set of channels sorted on the best AP it can associate with to the client. In this diagram the sorted order is Channel 1 followed by Channel 2. The client breaks out of scanning once probe response is received from Channel 1.

- A Hand-off is triggered in the client by the degradation of signal quality with the associated AP.
- The client then sends a probe request in the current AP channel.
- A sentinel on seeing the probe request gets the sorted list of channels for this client from the hand-off engine and sends this list as response to the client.
- The client then scans the channels in the given order.

And the following is the algorithm followed by the client during the scanning of the given set of channels:

- The client sends a probe request to the channels in order and waits for the probe response. Once a probe response is obtained, it breaks out of scanning and proceeds with the Authentication phase. This proves to be an effective method as the channels are already sorted based on the best APs for that client.

Our results show that, about 90% of the time, the scanning breaks after probing the first channel in the list and thus reducing the scanning latency drastically.

5. Hand-off Engine Design

In this section, we explain the design of the Hand-off Engine which continuously keeps track of the sorted list of best AP channels to scan during hand-off for each client. The design is shown in Fig. 3.

Apart from the components shown, the Hand-off Engine has a static Radio Map table, an initial Neighbor graph and a set of Hand-off Points to start with, all obtained during a *training phase* using our architecture.

Constructing the Radio Map table is similar to the approach followed by Bahl et al. [11] in RADAR. The mapping of <x, y> location to "the signal strength of client seen by all the sentinels and the signal strength of the associated AP seen by the client" is stored in the Radio Map table. This data is collected during the training phase.

During the training phase an initial neighbor graph is constructed which depicts the neighbor APs to which hand-offs has been made from an AP. The construction of the neighbor graph is similar to the approach proposed by Mishra et al. [5]. We also update the neighbor graph dynamically after the training phase.

An initial set of Hand-off points are identified during the training phase and stored. The hand-off points are identified as points (location of the client) where a client ReAssociates with another AP. This table is updated with the new hand-off point each time a hand-off takes place.

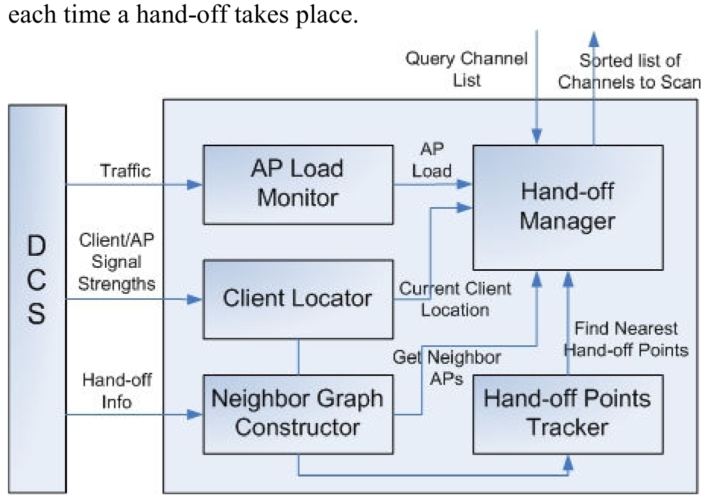

Figure 3: Hand-off Engine Design

5.1 AP Load Monitor

The AP Load Monitor gets data continuously from the WLAN Traffic table in the DCS and maintains a table of current load of all APs. It identifies the number of unique clients of an AP and the number of bytes sent from and to an AP using the data from the DCS. Considering the AP Load during Hand-off, helps us to achieve *Load Balancing* between APs.

5.2 Client Locator

The Client Locator uses the Signal Strength table in the DCS and the static Radio Map table, identified during the training phase, to find each client's current location. This approach is similar to that proposed by Bahl et al. [11] in RADAR.

During the training phase, the user specifies his current location in the geographic map of the network area; the signal strength of the client observed by the sentinels and the signal strength of the associated AP observed by the client is recorded. The mapping of the <x, y> location and the signal strengths are stored in the Radio Map table.

During the real-time phase, the DCS records the signals strength of the "client as perceived by the sentinels" and that of the "associated AP as perceived by the client" in the Signal Strength Table. The Client Locator then applies *nearest-neighbor* technique on the data in the DCS and Radio Map Table to identify the client's current location. This approach is similar to the approach used in RADAR. Thus the Client Locator keeps track of all the clients at any point of time.

5.3 Neighbor Graph Constructor

The Neighbor Graph Constructor continuously gets data from the Hand-off Info table in the DCS and constructs the Neighbor Graph. The construction of the neighbor graph is similar to the approach proposed by Mishra et al [5]. The neighbor graph depicts the previous hand-offs taken place in the WLAN network in a period of time. APs are nodes and a directed edge between $<AP_i, AP_j>$ exists in the graph if a hand-off between AP_i and AP_j has taken place. Thus the neighbor graph gives the neighbor APs to which hand-off has been done before from the current AP. A hand-off is identified by monitoring the ReAssociation packets. We maintain a dynamic neighbor graph, where we add a directed edge $<AP_i, AP_j>$ if hand-off from APi and APj has taken place and there is no edge $<AP_i, AP_j>$ before. An edge is deleted if a hand-off doesn't take place between them for a period of time.

5.4 Hand-off Points Tracker

A hand-off point is a location where a hand-off takes place and is uniquely identified by the location and the APs involved. The Hand-off Points Tracker uses the Hand-off Info table to track hand-offs and uses the Client Locator to find the location of the client and thus the hand-off point. During each unique hand-off in the WLAN, a new record is added to the Hand-off points table. The Hand-off Manager uses the Hand-off Points table to find the nearest Hand-off Points to sort the channels. Our simulation results show that, as the number of hand-off point records increases, the efficiency of channel sorting also increases. Thus the effectiveness of the system increases as it learns new Hand-off points.

5.5 Hand-off Manager

The Hand-off Manager keeps track of the minimal sorted set of channels to scan for each client during hand-off at any point of time. The sentinel contacts the hand-off manager to get the set of channels to scan for the client during hand-off.

The Hand-off Manager uses the above four components to find the minimal set of channels to scan and to sort the channels. It does the following for each client every period of time

- It finds the client's currently associated AP.
- It uses the Neighbor Graph Constructor to find the neighbor APs and their channels. Finding the neighbor APs reduces the number of channels to scan.
- It contacts the Client Locator to find the current location of the client.
- Using the Hand-off points table from the Hand-off Points Tracker, it finds the nearest hand-off points (in terms of distance) from the current client location. Only the points which involve the current AP are considered and the points greater than a threshold distance are eliminated. Thus the set of channels to scan is further minimized.
- The channels are sorted based on the nearest hand-off points.
- Finally, it uses the AP Load Monitor to find the load of the APs in the sorted list and sorts the list further if load balancing is required.

Thus the Hand-off manager uses the neighbor graph, the client's location, the previous hand-off points and AP load to prepare a list of channels for each client to scan during hand-off. Our simulation results show that the above channel sorting algorithm proves to be effective.

6. Simulation and Results

In this section, we substantiate our analytic proposal through results of simulation of the Sentinel based Architecture. We used simulation models with up to 10 access points, 15 sentinels and 4 channels. We studied the effectiveness of the channel sorting technique by increasing hand-off points and by increasing the number of records in the static Radio Map database. We measure the effectiveness of the channel sorting using the ratio of the number of times the first probed channel succeeded to total number of hand-offs.

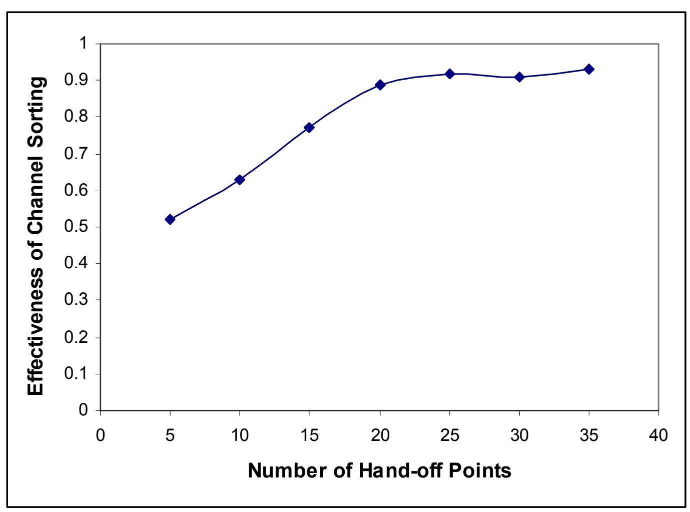

Figure 4: As the number of Hand-off Point increases, the effectiveness of channel sorting also increases.

Fig. 4 shows the increase in effectiveness of the channel sorting technique when the number of hand-off points increased from 5 to 35. The efficiency was about 90% i.e. 90% of the time, the client associates with the first AP on the sorted list provided by the sentinels. Our Radio Map table had 30 records during this experiment.

Fig. 5 shows the increase in the effectiveness of the channel sorting by increasing the known Locations in Radio Map. As we increase the number of records in the Radio Map table, the accuracy of location determination increases and thus the overall efficiency.

Our results also show that the scanning latency reduces drastically as the number of channels probed is on an average 2 (probing the sentinel and then the first channel in the list).

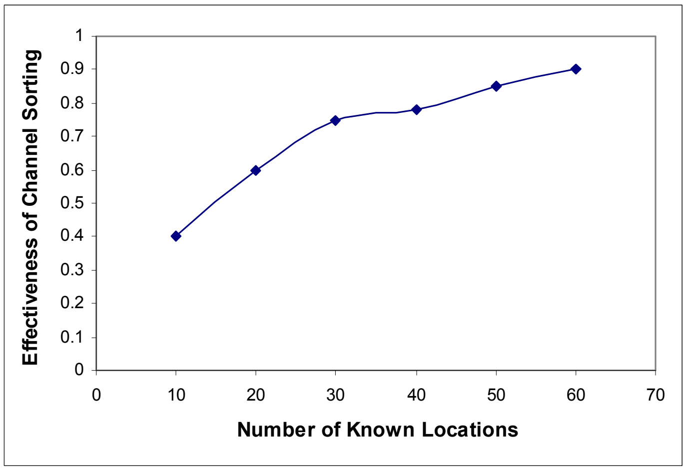

Figure 5: As we increase the number of records in the Radio Map table during the training phase, the accuracy of location determination increases and hence the channel sorting efficiency.

7. Conclusion and Future Work

Thus the Sentinel based approach for monitoring the WLAN network and helping the client during hand-off reduces the scanning latency and involves less overhead at the client and no overhead at the AP. The method of finding the nearest hand-off points and giving the client a sorted list of minimum channels to scan proves to be an effective approach.

As we sniff almost all packets in the WLAN network, the sentinel based approach with client cooperation can be used for Network Performance Measurements, Fault Diagnosis [12] and various other Analyses.

Currently we use discrete deterministic model for location determination. The use of continuous probabilistic model [13] will increase the accuracy of location determination and hence hand-off points and hand-offs.

As the number of hand-off increases, the number of hand-off points also increases in the Hand-off Tracker table. Hence the computation for the nearest hand-off points decreases. We can use various *clustering* techniques to cluster the hand-off points to form a hand-off region.

Though finding the nearest hand-off point proves to be effective, it creates slight problems when two or more points are equidistant or no point is within threshold. In that case, predicting the client's movement (trajectory) seems to provide an effective solution. The nearest hand-off point in the client's predicted trajectory can be given more weight. This will increase the efficiency of channel sorting and thus will increase the probability of associating with the first AP probed during scanning.

References

[1] IEEE. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE Standard 802.11*, 1999.

[2] V. Nee, "New High-Rate Wireless LAN Standards," *IEEE Communications Magazine*, vol. 37, pp. 82–88, Dec. 1999.

[3] A. Mishra, M. H. Shin, and W. Albaugh, "An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process", *ACM SIGCOMM Computer Communication Review* vol. 3 pp. 93-102 Apr. 2003.

[4] Ishwar Ramani, Stefan Savage, "SyncScan: practical fast handoff for 802.11 infrastructure networks," Proceedings of IEEE Infocom 2005

[5] M. Shin, A. Mishra, and W. Arbaugh, "Improving the Latency of 802.11 Hand-offs using Neighbor Graphs," in *Proc. ACM Mobisys* September 2004.

[6] S. Waharte, Kevin. Ritzenthaler and R. Boutaba, "Selective Active Scanning for Fast Handoff in WLAN using Sensor Networks," MWCN 2004, Paris, France, October 2004.

[7] S. Pack and Y. Choi, "Fast Handoff Scheme based on Mobility Prediction in Public Wireless LAN Systems," *IEE Proceedings Communications*, Vol. 151, No. 05, October 2004.

[8] S. Pack and Y. Choi, "Fast Inter-AP Handoff Using Predictive Authentication Scheme in a Public Wireless LAN ",. in *Proceedings of IEEE Networks Conference*, Atlanta, GA, Aug. 2002.

[9] S. Pack and Y. Choi. "Pre-Authenticated Fast Handoff in a Public Wireless LAN based on IEEE 802.1x Model". *IFIP TC6 Personal Wireless Communications 2002*, October 2002.

[10] Paramvir Bahl, Jitendra Padhye, Lenin Ravindranath, Manpreet Sigh, Alec Wolman, Brian Zill, "DAIR: A Framework for Managing Enterprise Wireless Networks Using Desktop Infrastructure", *ACM Hotnets IV*, November 2005.

[11] P. Bahl and V. N. Padmanabhan, "RADAR: An In-Building RF-based User Location and Tracking System," in *IEEE Infocom 2000*, vol. 2, March 2000.

[12] A. Adya, P. Bahl, R. Chandra, L. Qiu, "Architecture and Techniques for Diagnosing Faults in IEEE 802.11 Infrastructure Networks," *Mobicom 2004.*

[13] T. Roos, P. Myllymaki, H. Tirri, P. Misikangas, and J. Sievanen, "A Probabilistic Approach to WLAN User Location Estimation," *International Journal of Wireless Information Networks*, vol. 9, no. 3, July 2002.

{leninravindranath, fredrickprashanth}@gmail.com
{leo.prasath, praveendurairaj}@gmail.com
asiro@vsnl.com