

SIMULATION OF HANDOFF IN WIFI WIRELESS NETWORKS

by

CHRISTOPHER J. FRIESEN

Adviser
HALA ELAARAG

A senior research paper submitted in partial fulfillment of the requirements
for the degree of Bachelor of Science
in the Department of Mathematics and Computer Science
in the College of Arts and Science
at Stetson University
DeLand, Florida

Spring Term
2004

TABLE OF CONTENTS

TABLE OF CONTENTS.....	ii
LIST OF FIGURES.....	iii
ABSTRACT.....	1
1. INTRODUCTION.....	2
2. BACKGROUND.....	3
3. RELATED WORK.....	5
4. IMPLEMENTATION.....	7
5. RESULTS.....	12
6. CONCLUSION.....	13
7. ACKNOWLEDGMENTS.....	14
REFERENCES.....	15

LIST OF FIGURES

Figure 1: List of the codes that can be sent to a client awaiting authentication.....	10
Figure 2: Pseudo code of the handoff process from the originating Access Point.....	12
Figure 3: Pseudo code of the handoff process on the destination Access Point.....	12

ABSTRACT

Since its debut in the late 1990s, wireless technology has become as much a necessity as the wired technology. Each new iteration of wireless technology brings with it new features like speed increases and security enhancements. The technology is at the point where it is being used for real time applications, like video streaming and voice chat. As these applications develop, a problem has risen. On a wireless network, it is not acceptable to require a roaming client to stay within the limited range of a single access point (AP) and the delays incurred from jumping between APs causes issues when the client is partaking in a real time conversation or chat. In this paper, a way of improving mobility of wireless clients will be researched. The focus will be on the building of a wireless simulator with special attention on the connection of a client to an access point. The process by which clients communicate to the AP will also be stressed. Next, an explanation of how clients can hop between APs will be explained along with an implementation of how to keep this hop as clean and errorless as possible, on a TCP network. The simulation will be written in a Linux environment using the standard C++ networking libraries.

1. INTRODUCTION

Wireless technology has grown in popularity exponentially. The development of faster wireless technology is empowering users to do the things they were only permitted to while sitting at a terminal on a wired network. Today, the technology is good, but it is not perfect. There are still many issues facing the wireless community that need to be worked out, like better security and handling of error prone networks. Another larger problem is the use of TCP to communicate between the clients and their access points (AP). TCP was designed with specific requirements to prevent faults in the connecting, sending, receiving, and disconnecting processes of a client. There is a lot of overhead in these requirements which can be a burden to a wireless client who might be on a slow link, is trying to move between different APs, or trying to utilize the network for real time applications. In order to use real time applications like voice communication or data streaming, the rate of error must be kept at a minimum, but there must be a steady rate of information sent at all times. On a wired network, this is not a large issue, because the line speed is generally high. With a wireless network, the speed of the medium is dependent on factors such as signal strength and the number and type of clients trying to use the medium. This can cause the line speed to fluctuate or just be slow. Also, the more clients that are connected to a given AP, the harder it is for the medium to keep up with all of the clients and for each client to get fair use of the medium.

2. BACKGROUND

One of the main issues with wireless mobility is the latency involved when a client moves out of the range of one AP and into the range of another, called a handoff [MIS02]. Handoff can cause jittering in the connection which interrupts the stream of data being sent to the client.

A. Process

The handoff process involves two steps:

1. Discovery - this is when the client scans the network by looking for the beacon messages that each access point broadcasts. Also, the client can build a priority list of APs, depending on its signal strength.
2. Reauthentication - Using the priority list built during discovery, the client is synchronizing itself with the best AP in its list. It involves having authorization credentials and state information sent from the original AP to the new one.

B. Delays

The two steps required for a successful handoff introduce latency issues. These issues are as follows:

1. Probe Delay - this is the amount of time it takes the client to complete a scan of available networks and to build its priority list. It is required to send somewhere between 3 to 11 messages in order to complete this task.

2. Authentication Delay - this is the amount of time it takes for the client to reauthenticate to the AP it chose from its priority list. Depending on the type of authentication, either 2 or 4 packets need to be exchanged.
3. Reassociation Delay - this is the amount of time it takes for the client to signal the AP that the handoff is complete. It is required that a minimum of 2 packets be exchanged.

3. RELATED WORK

Much research has been done in attempts to improving the handoff procedure for wireless Asynchronous Transfer Mode Networks (WATM-N) [AKY96]. ATM networks are designed to provide high speed communication for roaming clients, which makes it a good candidate to use when real time applications are involved. Current ATM handoff optimization techniques are broken up into four categories [MED02]:

1. Connection Reestablishment - requires that the client establishes a new connection each time a handoff is required.
2. Route Argumentation - extends the clients current connection to the new AP.
3. Partial Rerouting - searches for the shortest route between APs.
4. Connection Prediction - attempt to predict where the client will go.

Each of the above techniques has its disadvantages. In connection reestablishment, the amount of time it takes for a client to recreate a connection is too high and becomes a waste of network resources. Sangheon Pack and Yanghee Choi [PAC02] found a possible solution by requiring a roaming client to be authenticated to multiple APs around it, based on patterns in its movement. Route argumentation has the potential to result in long looping routes. One way to overcome this is to use a partial rerouting algorithm to compute the optimal route between two APs. Although it can provide better performance, it requires resources to compute the route. One such algorithm that has

been researched is called Nearest Common Node Rerouting (NCNR) [AKY96]. It attempts to reroute a client using the least amount of bandwidth by eliminating connections that are not necessary. This is calculated in as efficient manner as possible, saving time and resources. Li-Yun Chiang [CHI02] has also solved the routing issue by using vectors to represent the shortest path between APs. The final technique, connection prediction, requires that a tree of the network be maintained and searched when a handoff is required. This creates much more overhead than is acceptable [MED02].

4. IMPLEMENTATION

There are many aspects of the 802.11 specification that are required in order to create a realistic simulation. Fortunately, not everything needs to be simulated to achieve a handoff. Only the scanning or connection procedure, authentication, and the process by which the clients communicate with their corresponding access points, called Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA).

A. Scanning

The scanning procedure is the means by which the mobile STA is able to find an AP to connect to. In our project, the information sent to the AP during the stations first connection is as follows:

1. Type – This is a flag showing that the connection is from an STA or possibly a STA in a handoff state.
2. Position – This is the position, an X and a Y coordinate, at which the STA is located, sent to the AP so it can determine if the STA should be connected to it.

The AP also stores this position in order to determine if the STA has left its range while it is moving.

Once the AP has received a message from the STA, it processes the information and sends a response back to the STA.

1. Position – This is the position of the AP, sent to the STA so it can determine when it has left the APs range.
2. State – The state at which the AP has calculated the STA to be in, either it is not connected or not authenticated.
3. Distance – The distance the STA is from the center of the AP. This value is used by the STA to pick the closest AP in its field of scanning.
4. Socket ID – This is a unique identifier that the AP recognizes the STA as. It is used by the STA during handoff to identify itself.

B. Authentication

In order for a client to connect to an AP, a simple handshaking process must be completed. To initiate the connection, the client wishing to connect sends a request to the AP with the following information [LAN99]:

1. Authentication Algorithm Identification – this is used to determine if both parts of the network are using the same algorithm to process their information. The two types are *Shared Key* and *Open System*. Shared Key uses Wired Equivalent Security (WEP) and will not be used in this project. Open system is transmitting on an insecure line without the worry of the data being intercepted.
2. Station Identity Assertion (SSID) – this is the unique identification of the access point being connected to. This must be present or authentication will fail.
3. Authentication Transaction Sequence Number – at this time, this field will be set to 1 to signal the beginning of the process.

4. Authentication Algorithm Dependent Information – this section is left blank if Open System authentication is used.

When the AP receives a request, it processes it and sends a message back to the client with the state of its authentication. This message contains the following information:

1. Authentication Algorithm Identification – this field has the same function as the first step of authentication process.
2. Authentication Transaction Sequence Number – this is set to 2 to signal to the client that it is the second phase of the authentication process.
3. Authentication Algorithm Dependent Information – is left blank for the same reasons as above.
4. Result of Authentication – this is a code sent back to the client specifying the state of its authentication as shown in *Figure 1*.

Figure 1: List of the codes that can be sent to a client awaiting authentication

Status Code	Meaning
0	Successful
1	Unspecified Failure
2 – 9	Reserved
10	Cannot support all requested capabilities in the Capability Information field
11	Reassociation denied due to inability to confirm that association exists
12	Association denied due to reason outside the scope of this standard
13	Responding station does not support the specified authentication algorithm
14	Received an Authentication frame with authentication transaction sequence number out of expected sequence
15	Authentication rejected because of challenge failure
16	Authentication rejected due to timeout waiting for next frame in sequence.
17	Association denied because AP is unable to handle additional associated stations
18	Association denied due to requesting station not supporting all of the data rates in the BSSBasicRateSet parameter
19 – 65 535	Reserved

C. Carrier Sense Multiple Access with Collision Avoidance

CSMA/CA is a distributed coordination function (DCF) that is used to check to see if a medium is being used, because a client can not transmit if the medium is being used by another. This process is achieved in two different ways:

1. Physical Carrier Sense – this information is provided by the physical layer (PHY). The PHY sends messages to the MAC layer with details about the state of the medium.
2. Virtual Carrier Sense – this uses a network allocation vector (NAV) to maintain predictions of future traffic, based on the duration information transmitted by a client before it begins exchanging data with the AP. This sense was not implemented.

It takes only one of these mechanisms to set the medium as being busy. When they both suggest that the line is free, any connected client is free to attempt to transmit. It is at this point where the most number of collisions will occur, so upon the completion of transmission by one client, each of the other clients choose a random backoff time before attempting to use the medium.

C. Handoff

Handoff, as defined previously, is the procedure of predicting and rerouting packets for a wireless client as it hops between different APs [AKY96]. The originating AP will be labeled **A** and the candidate AP will be labeled **B**. In our simulation, all of the APs are physically connected to each other and constantly scan for new APs. The first step in a handoff is for **A** to determine which AP the STA will jump to, **B**. It does this by sending the calculated new location of the STA to each AP within its range and picks the closest to the STA and sends all of that STAs information to it. Upon receiving handoff information, **B**, sends returns a status message to **A** based on the success or failure of the handoff attempt.

```

loop indefinitely
  find access points
    if AP is already in list
      continue
    else
      add access point to list
  check for STA marked for handoff
    if STA in handoff state
      find closest AP to STA
        while handoff not complete
          perform handoff
          receive handoff state
        done
      remove STA
    else
      let STA go
  else
    continue
done

```

Figure 2: Pseudo code of the handoff process from the originating Access Point

```

check all connections for message
  check if message is from an AP
    check if AP is sending handoff information
      process handoff information
      send handoff response
    else
      process new AP connection attempt
  else
    process STA message
done

```

Figure 3: Pseudo code of the handoff process on the destination Access Point

5. RESULTS

While facing many issues in the creation of this simulation, a lot of valuable information has been learned. The number of factors required to efficiently perform a handoff in a real-time environment demand a considerable amount of calculations on a large pool of data. In creation of this simulation, a realistic representation of the current Wi-Fi hardware was created as accurately as possible. During this process, we began to discover the issues that would be faced when implementing handoff for real world applications. Issues such as being able to determine if a STA disconnected or is going to need to undergo the handoff process. Another observation is that accurate tracking of an STA is difficult. Some sort of data would need to be stored on the AP, requiring it have the hardware to store STA specific information, such as location and direction.

6. CONCLUSION

Wireless technology today is still in a very infant state. It still has many issues that still need to be overcome, such as better security, better connectivity, and better handling of mobile clients. Through my experiences, the ability to perform a handoff as smoothly as would be required for real-time applications would be difficult to implement. The major barrier that will need to be overcome is the accurate and quick prediction of a STA's future position outside the range of an AP, so the STA's authentication and state data can be transferred and accepted before the STA needs to be reconnected.

A. Future Work

There are many aspects of wireless technology that were not implemented, because they were not required for a basic simulation. Some things that could be implemented for future tests are as follows:

1. Active Scanning – this is the process by which a STA scans in the background for the appearance and disappearance of other APs. This would eliminate the time required to scan for a new AP when it is realized that a handoff is required.
2. Wireless Security – There are many new schemes of security being developed to protect those using wireless technology.

7. ACKNOWLEDGMENTS

Funding for this project made possible by acceptance of the Deans Fund at Stetson University.

REFERENCES

- [AKY96] Bora A. Akyol and Donald C. Cox. Rerouting for Handoff in a Wireless ATM Network. Cambridge, MA. 1996. <http://wireless.stanford.edu/~akyol/icupc.ps>
- [CHI02] Li-Yun Chiang. An Efficient Handoff Algorithm in Wireless ATM Networks. 2002. <http://athena.cs.ccu.edu.tw/advisees/thesis/ms89/jly89.pdf>
- [LAN99] LAN MAN Standards Committee of the IEEE Computer Society. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. 1999. <http://standards.ieee.org/getieee802/download/802.11-1999.pdf>
- [MED02] Sirisha R. Medidi and Forouzan Golshani. Handoff In Mobile ATM Networks: Optimized Rerouting. 2002. <http://wireless.netlab.uky.edu/SeminarArchive2001-2002/HandoffMobileATMNetworks.pdf>
- [MIS02] Arunesh Mishra, Minho Shin, and William Arbaugh. An Empirical Analysis of the IEEE 802.11 MAC Layer Process. University of Maryland. 2002. <http://www.cs.umd.edu/%7Ewaa/pubs/handoff-lat-acm.pdf>
- [PAC02] Sangheon Pack and Yanghee Choi. Pre-Authenticated Fast Handoff in a Public Wireless LAN Based on IEEE 802.1x Model. 2002. http://mmlab.snu.ac.kr/research/publication/docs/pwc2002_shpack.pdf