STEADFAST: A New Wireless Network at MIT

Itamar Belson, Madeleine Severance, Tzer Wong
{ibelson, madksev, tzer}@mit.edu

Instructor: Mark Day
Section: 1PM

May 6, 2016

1 Introduction

MIT is seeking a new, well-designed wireless network that enables users to efficiently and effectively connect to the Internet through access points (APs) distributed throughout campus. Offering a wired connection to the Internet, these APs provide clients with a dependable interface through which clients can provide individualized data, optimize personal connections, and transmit and receive information.

Utilizing existing infrastructure, Steadfast is able to scale to support the size of MIT's user population, which consists of up to 25,000 clients attempting to connect to the network through approximately 4,000 APs distributed throughout campus. The design utilizes the three main modules of the system –the client, APs, and the IS&T server– while prioritizing user happiness, network utilization, scalability, and fault tolerance. Through the consideration of various use cases that may arise, the system is able to support a wide variety of MIT's needs while abiding by the design goals and specifications.

The first half of this report outlines the design by specifying and assessing the impact of each component of the system. The second half of the report comprises of computational analysis and evaluation of the proposed design as well as discussion of tradeoffs that motivated the underlying design decisions.

2 System Overview

The system can be split into three main modules: the client, APs, and the IS&T server. These modules work together to provide users access and continued connection to the Internet. Detailed below are several specifications and assumptions for each module that are critical to the overall design.

2.1 Clients

The client is the module that represents a user who is attempting to connect to the Internet. It has two main components: the monitor and the controller. The monitor has knowledge of the client's performance, specified by the required throughput, which remains constant over time, and actual achieved throughput, which may fluctuate over time. The monitor also receives input from the user when he or she is unhappy through a graphical user interface. The controller is the client's main module that communicates with the APs. The controller polls the monitor every second to determine the current level of performance, which dictates the client's connection requirements. Clients fall into two categories: large and small. A large client, such as a user attempting to stream video, has a large required throughput and consistently achieves close to that determined amount. A small client, such as a user attempting to send and receive emails, has a smaller required throughput that may fluctuate. Steadfast must be flexible in order to handle both types of users with their different requirements.

2.2 Access Points

APs belong to the MIT network and have been placed throughout campus in predetermined locations. A single AP may support up to 128 clients at a time and has a wireless connection range of approximately 125 feet. APs broadcast heartbeats every 30 ms so that clients may detect which APs are within range. They broadcast their data on one of eleven predetermined channels and have a maximum bandwidth of either 54 Mbits/sec for normal APs or 96 Mbits/sec for newer, high-capacity APs. Once a client connects to an AP that can handle its load, it is free to transmit and receive data from the network. In order to effectively connect clients to the rest of the network, APs provide a wired, reliable connection to the Internet and to the IS&T server. IS&T has indicated that a reliable transport protocol already exists and data delivery is guaranteed between an AP and its connected clients.

2.3 IS&T Server

MIT IS&T has indicated that it would like to accumulate specific data in order to monitor and assess network performance. As such, it has provided a server with 10 TB of disk to store amassed data. APs communicate with the server through a wired connection, while clients can communicate with the server through the wireless connection of an AP. The server contains a prepopulated table with the unique MAC address, geolocation, and bandwidth capacity of every AP in the network. In addition, under the updated design specifications, while the IS&T server stores its data reliably, it is susceptible to failures that may last up to two minutes at a time.

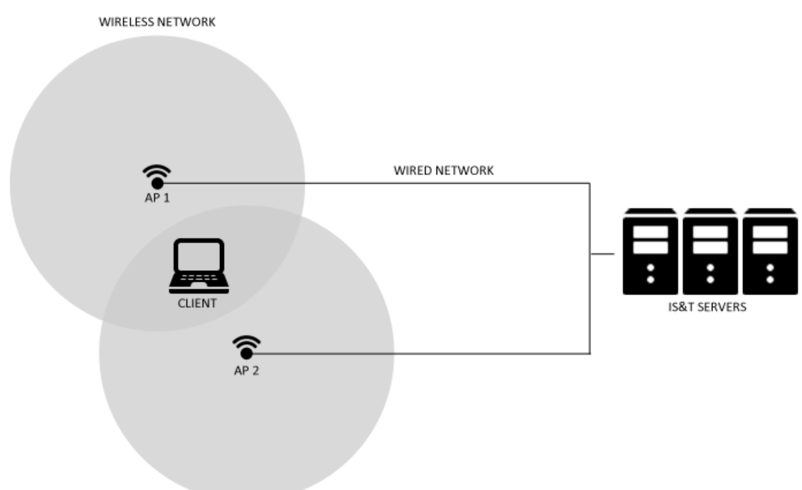


Figure 1. System Overview Diagram. A diagram of the three system modules –the client, APs, and the IS&T server– and their interaction within the system.

3 System Design

The general functions and assumptions of each of the existing modules provide a foundation upon which Steadfast is designed. Through the consideration of their qualities and interactions and the analysis of tradeoffs inherent to each module, the overall system is designed to ensure user satisfaction and network optimization even in the most strenuous of network conditions.

3.1 Precomputation

As described above, the IS&T server is prepopulated with the MAC address, geolocation, and capacity of each AP in the network. However, before Steadfast begins to connect clients, the server must accumulate additional information in a designated preprocessing period. This process will be useful not only to better inform clients of optimal AP connections but also to enable adequate load balancing across the overall system.

3.1.1 Server Data Table

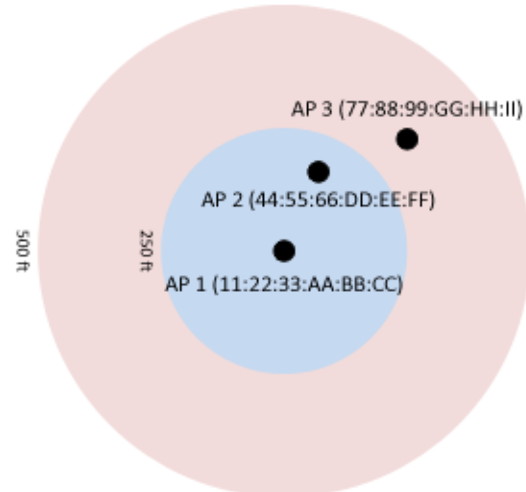
Upon initial connection to the network, each AP uses its wired connection to the IS&T server to send its public IP address and its MAC address, as well as its operating channel. This information is appended to the existing table in the row specified by the particular AP's MAC address. In addition to accumulating this information, the server runs geographical algorithms using the table's location data to determine which APs are within a 250-foot range and which APs are within a 500-foot range of each particular AP. Each AP in the table now has an exhaustive list of all of the APs that are within 250-feet and all of the APs that are within 500-feet of its position.

AP Number	MAC Address	Geolocation	IP Address	Operating Channel	Available Bandwidth	Range (0-250)	Range (250-500)
1	11:22:33:AA:BB:CC	41'24"12.3"	18.111.14.15	2	96 Mbits/sec	2	3
2	44:55:66:DD:EE:FF	41'24"12.4"	18.111.14.12	5	54 Mbits/sec	1, 3	-
3	77:88:99:GG:HH:II	41'24"12.6"	18.111.12.11	8	54 Mbits/sec	2	1

Figure 2. Server Data Table. An example of the server data table following all precomputation processes. The first three columns represent prepopulated data, the next three columns represent data that has been received from each AP, and the final two columns represent data that has been computed on the server.

3.1.2 AP Data Table

After the server has completed its precomputations, it sends the relevant data to each AP. Namely, the server will send to each AP a list containing the MAC address, geolocation, IP address, channel, and initial bandwidth capacity of every AP within both the 250-foot range as well as the 500-foot range. This table will take a form similar to the server AP table but divided into two parts according to distance range: 0-250 feet and 250-500 feet. In addition to the existing data columns, the table will have an additional column that will be used to monitor the current count of connected users for each AP in the table. Once initialized, this table will allow each AP to monitor the resources of all of its nearby APs as described in the following section.



MAC Address	Geolocation	IP Address	Operating Channel	Available Bandwidth	User Count
11:22:33:AA:BB:CC	41'24"12.3"	18.111.14.15	2	96 Mbits/sec	0
44:55:66:DD:EE:FF	41'24"12.4"	18.111.14.12	5	54 Mbits/sec	0
77:88:99:GG:HH:II	41'24"12.6"	18.111.12.11	8	54 Mbits/sec	0

Figure 3. AP Map and Corresponding Data Table. An example of a potential AP network configuration and the corresponding data table that would result on AP 1 (11:22:33:AA:BB:CC). In that map, the inner region indicates a range of 0-250 feet while the outter region indicates a range of 250-500 feet. The corresponding data table on AP 1 is comprised of two parts, one for each range.

3.2 Data Table Processes

With the relevant data table on each AP, the system now has the necessary components to enable inter-AP communication via the wired network, which will be used to facilitate optimal client connection and overall load balancing. While the details of how these processes influence client connection protocols will be described in the next section, this section provides information as to how an AP's data table is maintained and updated to provide useful information to the system.

3.2.1 Data Table Processes on Client Connection

As previously described, each client has relevant information about its size and required throughput before engaging with the system. When a client successfully connects to the network, its corresponding AP is able to utilize this information to inform the greater system of the added load. Namely, when a client successfully connects to the network, it immediately transmits its required throughput value to the AP, which the AP uses to compute its new available throughput value within the table. In addition, the AP increments its user count to reflect the new client.

After updating these values, the AP informs all of its nearby APs of the changes. Using the IP address for each nearby AP in its table, it sends the updated values to each of the nearby APs via the wired network. In this way, it has updated its own values of available throughput and user count while informing each of its neighbors of the change. As such, whenever a new client successfully connects to the network, the client's AP as well as its nearby APs will be up-to-date with the added load.

3.2.2 Data Table Processes on Client Disconnection

A similar process occurs when a client disconnects from the network. Prior to disconnecting from its host AP, the client sends a disconnection signal, which includes its unchanged required throughput. The host AP uses this notification to increase its available throughput by the specified amount and to decrement the active user count within its table to reflect the newly available resources. After updating these values within its own table, the AP uses the IP address stored for each AP in its data table to inform all nearby APs of the new values. Similar to the process of client connection, this process ensures up-to-date data tables on the client's AP and on its nearby APs following a client disconnection.

3.3 Client Connection

While the previous sections provided the foundation for the system, this section will detail the process by which a new client is able to identify, locate, and connect to an optimal AP within range. The optimality of a client-AP connection depends on several factors including the load on the particular AP in question as well as the overall load across all of the APs within a client's detectable range. In addition to these metrics, the method by which clients connect to the network takes into account the different client types as well as the network conditions that may arise.

3.3.1 Initial Contact

A new client first establishes contact with the network through one of the available APs within its range. In order to identify such an AP, the client listens for a detectable heartbeat on a channel chosen at random between one and eleven. If no heartbeat is discovered on this channel, the client progresses by randomly picking a different channel and again listening for a heartbeat. This process continues until a heartbeat is detected, whereby the client connects to the AP on the detected channel, or until all channels have been exhausted, whereby the client is notified that it is not within the network's range. The randomness in this initial process is intended to spread a potentially unbearable instantaneous load on any single AP. In the case where the client has successfully identified and connected to a preliminary AP within range, the client will now be able to communicate with the system in an effort to better optimize its connection.

3.3.2 Preliminary Connection

The preliminary connection serves as the client's first established contact with the system. Through this connection, the client receives necessary information that will allow it to more intelligently identify, locate, and connect to an optimal AP. Namely, upon a client's successful connection with the preliminary AP, the client obtains the data stored within the first part of the AP's data table. As described in Section 3.1.2, this portion of the table contains identifying information and current metrics for all

APs within a 250-foot radius of the preliminary AP. The client uses this data to generate a prioritized list of potentially optimal APs that satisfy its personal connection requirements. This process is summarized below:

1. The client removes any APs from the table that are currently serving the maximum 128 clients or whose current available throughput cannot support its required throughput. If no APs remain in the list, the client disconnects from the preliminary AP and attempts to connect to a different preliminary AP on a different channel.
2. The client sorts the remaining list of APs within the table in order of decreasing available bandwidth. Ties are broken according to the number of clients that the AP is currently serving such that APs with fewer clients are listed first.
3. If the client self-identifies as small, the order of the list is reversed. This reordering ensures that small clients fill in small vacancies and do not fill up relatively open APs where a large client could fit in at a later time.

3.3.3 Optimized Connection

After retrieving the necessary information from the preliminary AP and generating the prioritized list of potentially optimal connections, the client is ready to identify and connect to a more optimal AP. The client disconnects from the preliminary AP and begins to traverse through the generated prioritized list, listening in sequence for a heartbeat on each APs corresponding channel. Upon discovering an AP, the client first attempts to verify that the discovered AP is indeed the AP referenced in the prioritized list. To do so, the client connects to the AP and compares its MAC address with the MAC address in the prioritized list. If the addresses match, the client polls the AP's data table to ensure that it can still support its required throughput and that it is serving less than the maximum 128 clients.

If, however, the addresses do not match, the client polls the AP for its identifying information and current metrics and inserts it into the corresponding position within the prioritized list to be potentially used later. In the cases where no heartbeat is detected, the client realizes that a sought after AP can no longer support its required throughput, or an incorrect AP is found, the client moves onto the next AP in the prioritized list.

This process continues until either the client successfully identifies and connects to an AP that can still support it, in which case it sends its relevant information to the AP as described in Section 3.2.1 and can begin sending application data, or the client reaches the end of the prioritized list and runs out of suggestions from that particular preliminary AP. In the latter case, the client attempts to identify a new preliminary AP on a channel that it has not yet listened to and restarts the optimization process with the new AP's data.

3.3.4 Connection Failure

Throughout the connection process, the client maintains a list of the eleven channels, marking a particular channel as invalid if it has tried and failed to connect to the network through that channel. As such, in any of the client connection protocols listed above, the client only listens to a channel a maximum of one time. In the case where the entire list of eleven potential channels has been exhausted, the system has failed to connect the client, receives an unhappy input from the user, and must suggest an AP in a nearby location on campus. To do so, the client reconnects to one of its previously identified preliminary APs and indicates that it is searching for suggestions within an extended range of 500 feet.

In this case, the preliminary AP sends the client the second portion of its data table, which contains data corresponding to neighboring APs within a 250-foot to 500-foot range. The client uses this data to construct a prioritized list of suggested APs within an extended range in a process similar to that outlined in Section 3.3.2. Through this new list, the system is able to recommend nearby locations that may offer better connection support.

3.4 Historical Data

The final component of the system is the process by which the IS&T server aggregates the requested data from each AP. IS&T would like the server to collect data about the number of bytes transferred by each AP as well as the number of connected clients to each AP every second. In order to handle this data set, Steadfast allots a sizeable portion of the server in the form of a data table that will be used solely for data aggregation.

3.4.1 Transport of Historical Data

Since the server will be receiving and compiling data from the 4,000 APs distributed throughout campus, the manner by which each AP sends the data to the server must be structured as to minimize the overall instantaneous load on the machines. In addition, since the new specifications for the system indicate that the server may fail for periods of up to two minutes at a time, the manner by which the system transfers the data from APs to the server must ensure reliable transport even in the presence of failures. As such, Steadfast relies on each AP's internal memory capabilities to lighten the load on the servers and to allow the server to better recover from outages. More specifically, since each AP has information about its number of transferred bytes and connected user count, it polls itself for this data every second and appends the information to a growing data chunk. In addition to the byte count and the user count, each element in the chunk is tagged with its corresponding timestamp. This data chunk amasses thirty seconds worth of data after which the AP sends it to the server via the wired network. The time interval of thirty seconds was chosen as to minimize the number of packets processed by the server while not abusing the expensive resources of the AP.

3.4.2 Historical Data Fault Tolerance

If the IS&T server is running and successfully receives the data, then the server sends an acknowledgement back to the AP for the corresponding packet. The server then splits the received data into partitions for each second and uses the timestamp tag to place the data into the corresponding AP row and time column for each partition within its growing data table. If however, the server has crashed and the AP has not received an acknowledgment, then the AP will try to send the data again. This process continues until the server has successfully received the message and the AP has received an acknowledgment. As such, Steadfast follows a retry protocol to ensure successful data transmission in the presence of server failures or packet drops. The timestamp tags on the data chunk elements are in place to ensure that when the server eventually receives the message, it will be able to correctly sort the data into its corresponding position within the data table.

4 System Evaluation

Steadfast is designed with the intent of achieving the desired specifications while maximizing user happiness and optimizing network utilization. Taking into consideration the restrictions imposed by the system infrastructure and the requirements of IS&T, the system design can translate into a real-world solution that is capable of efficiently and effectively tackling the various situations that may arise across campus.

4.1 User Happiness

Steadfast prioritizes user happiness over general network utilization. According to the design, once a client is successfully connected to the network, he or she will never be disconnected in favor of another user and will never experience a deficiency in network bandwidth. Along with the absence of disconnection mechanisms, the AP prioritization and selection algorithm, as detailed in Section 3.3.2, further enforces this promise of the system. According to the algorithm, a successful connection indicates that, once connected, the identified AP will always be able to accommodate the necessary bandwidth of the client. Although the actual throughput of the client might be smaller than the required throughput, the achieved throughput will never surpass the required throughput. While this decision may result in unconnected clients and unused AP bandwidth, the system actively avoids unexpectedly and inexplicably disconnecting users from the network to accommodate later users, which would result in a poor user experience. As a result of this design choice, a user would only potentially be unhappy if he or she were unable to connect, and were recommended locations of nearby APs instead.

4.2 Optimizing Network Utilization

While user happiness is a primary objective of the system, the ability to optimize network utilization is also crucial to the overall success of the system, both in tackling specific network circumstances and in achieving certain design goals. While the latter will be detailed in the following sections, this section will focus on specific scenarios that the system is expected to support on an active campus.

4.2.1 Single Client Connection in Under-Utilized Part of the Network

When a single client attempts to connect to the system in a largely under-utilized part of the network, the connection protocol allows the client to connect efficiently given that it is in range of a viable AP. Regardless of whether the client is small or large, this connection scenario is the best case for a client as many potential APs would arise on its prioritized list and the first one on this list would often be capable of supporting its connection requirements. As a result, the client would not need to scan all of the possible channels and could bypass the potentially longer connection time offered by an alternative, brute force connection protocol.

4.2.2 Large Number of Clients in One Location

The case of a large number of clients attempting to rapidly connect to the system via a specific set of APs in a single location often arises in lecture halls and auditoriums. While many of the effects of such a scenario cannot be entirely eliminated, Steadfast attempts to alleviate the sudden load as best as possible. Through incorporating randomization into the preliminary AP selection process, the clients distribute the initial load across the available APs in range and, as such, would not overload any particular channel. In addition, the AP data tables are constantly updated to reflect the current network status as outlined in Section 3.2. This relatively up-to-date snapshot of the network allows a client to best prioritize available APs and optimal connection options in range. While the lack of rebalancing mechanisms may see some clients without a connectable AP, the system values the promise that any successfully connected client is able to operate without bandwidth disruptions. Although out of the system's control, Steadfast's design anticipates that regions of campus particularly vulnerable to such overloading scenarios are fashioned with a larger number of APs capable of supporting potentially large numbers of users to further alleviate the tradeoffs inherent to the system's design.

4.2.3 Areas of High Churn

In certain areas of high mobility on campus, such as hallways, clients will likely connect to and disconnect from the network frequently. In order to assess the performance of Steadfast in this scenario, it is useful to first estimate the time spent for a client to connect to an AP in the best case and in the worst case. The best case occurs when the first randomly selected channel results in a viable preliminary AP that generates a prioritized list where the first valid option results in a successful connection. In this case, the client will spend approximately 30 ms listening for the initial AP heartbeat, 5 ms for the controller to change channels, 30 ms listening for the heartbeat of the first option in the prioritized list, and a relatively small amount of additional time sending and receiving control data and constructing the prioritized list. As such, the best case would take approximately 65 ms plus the time necessary for the additional processes.

The worst case occurs when the client traverses through each of the eleven possible channels before discovering that no AP is within range. In this case, the client will spend approximately 30 ms listening for a heartbeat on each channel and 5 ms changing between each of the channels. As such, the worst case is similar to a brute force protocol and would take approximately 385 ms while resulting in no successful connection. Estimating that users move through the hallway at approximately six feet per second, the user will move approximately 2.3 feet during the required connection time in the worst case. Given these estimations and the fact that users in this scenario are expected to connect and disconnect from the network every 5-10 seconds, it is safe to assume that the system will be able to adequately connect, process, and disconnect clients in areas susceptible to high churn.

4.3 Scalability

Scalability is one of the primary design goals considered by the system's design. As described in the specifications, the system should be able to efficiently and effectively scale up to 25,000 simultaneous clients. As such, Steadfast emphasizes scalability throughout its various processes and protocols. The server is the central component that handles the preprocessing work and data aggregation processes. To support scale on the server module, the system ensures that following the preprocessing procedures, the server's sole duty is to collect and aggregate the desired data. As such, the system is designed such that the server is not involved in any client connection processes and can focus all of its resources on its primary goal.

To accommodate for scale in the number of APs, the AP-server protocol is designed using chunks that comprise thirty seconds of AP data as opposed to individual seconds. This allows new APs to be incorporated into the system without added concern about the new load on the server. In addition to the concern of server scale, adding APs itself requires minimal work. Since the relevant AP information is stored on the server and the neighboring APs' metrics are communicated directly with one another, adding a new AP would consist of simply updating the server's data table and communicating the changes to each of the new AP's neighbors.

To accommodate for an increase in clients, the system incorporates several measures in its AP selection process. For example, each client may develop a different prioritized AP ordering that depends on various factors such as randomization, the size of the client, the AP's available bandwidth and user count, and the client's required throughput. In addition, large clients prioritize APs that have more available bandwidth while small clients prioritize APs that have less available bandwidth to allow larger clients to carve out sizeable portions of an AP's resources while small clients can fill in available openings. The new addition of high-capacity APs add yet another level of client scalability as they are capable of supporting the throughput of a greater number of users than normal APs. All of these factors lead to better network utilization and serve as load balancing mechanisms to spread the weight of clients across APs. While these processes aim to help the system support a larger number of clients, the prioritization of connected user happiness presents a tradeoff that may affect client scalability. As described in Section 4.2.2, the scenario of many users in a concentrated area may pose a bottleneck for the system that can result in unconnected, unhappy users. However, this issue can be overcome through adding additional high-capacity APs in identified areas of vulnerability.

4.4 Performance

The system also prioritizes performance in many aspects of its design in order to ensure that the system meets all of its specifications and to ensure user happiness across the network. Performance is measured in terms of the time it takes for a client to connect, the amount of control traffic transmitted throughout the network, and the time it takes for an AP's data to reach the IS&T server.

4.4.1 Client Connection Time

As described in the analysis in Section 4.2.3, a client will ping at most each of the eleven possible channels once before connecting to a suitable AP or receiving a recommendation to relocate to a less utilized part of the network. As the analysis demonstrates, this worst-case scenario takes approximately 385 ms to complete and may result in no viable connection. However, assuming that APs are well distributed throughout campus, Steadfast's client connection protocol would reduce this worst-case time in most scenarios through prioritizing APs that are able to support the client and eliminating those that cannot. As such, in most scenarios, Steadfast's protocol performs more efficiently and effectively than the naïve, brute force method of pinging all possible channels.

As described in Section 4.1, the system prioritizes user happiness over additional complexities in the client connection protocol in order to ensure user happiness of connected clients while maintaining relative system simplicity. This can be seen through the algorithm's use of an AP's available bandwidth and number of connected users and through its consideration of a client's required throughput as opposed to its constantly changing achieved throughput. The system considers these metrics in the client connection process as opposed to ones that would require overly complex algorithms in an effort to minimize connection time and to reduce the overall system complexity.

4.4.2 Control Traffic

The communication overhead of Steadfast can be measured through analyzing the amount of control traffic that is sent between clients and APs prior to final connection. The heaviest traffic sent between an AP and a client is comprised of the data contained within the AP's data table. While the system could have determined the optimal connection on the AP to reduce this overhead, such a design would add unnecessary load on the system's resources and would take away the connection flexibility from the client. Taking into consideration the worst case in AP configuration, the first part of the AP data table can contain up to twenty-two rows each with six columns of data. Estimating that each row would require 149 bits (48 bits for the MAC address, 64 bits for the geolocation, 32 bits for the IP address, 4 bits for the channel, and 1 bit to indicate the type of AP to determine the initial available bandwidth), the entire table would require 3,278 bits in the worst case. While this could result in a sizeable data transfer, such a transfer would occur at most once per preliminary AP connection. In most cases, this overhead would be much less than the worst given that each client only attempts to connect to each channel a maximum of one time.

Other instances of control data overhead are much smaller relative to the data table transfer as they merely contain single data values to verify an AP's identity or to check whether an AP can still support a client prior to final connection. Such packets may contain the client's required throughput, the AP's available bandwidth, or the number of clients currently connected to an AP.

4.4.3 Data Transmission From AP to Server

Since IS&T would like data from each AP on the number of bytes transferred and the number of connected clients, the system implements a protocol to provide the desired information while minimizing the effects on the overall system. As described in Section 3.4.1, each AP aggregates thirty seconds of data before sending the chunk to the server. This protocol enables the server to collect its necessary data without being overloaded with packets from the APs. Since this data is sent from each AP via the wired network, the server will receive approximately 4,000 messages in each thirty-second interval. Taking into consideration the latency of 5-10 ms via the wired network and the processing and computation time required for each packet, the server should be able to successfully receive, process, and arrange the data with minimal packet drops.

In the case of a packet drop or a server outage, Steadfast relies on a retry mechanism and an acknowledgement protocol to ensure that every message is successfully delivered to the server. After a maximum failure time of two minutes, the amount of data that may have accumulated is four data packets per AP, which would result in approximately 40 ms of additional packet transmission time via the wired network. Although this worst-case scenario could result in up to 16,000 delayed packets, the design anticipates that the efficiency of the data packets, the low latency of the wired network, and the computational power of the server would result in little effect on the greater system.

4.5 Fault Tolerance

Since Steadfast will be expected to reliably serve thousands of clients, its ability to function when faced with system failures is vital to its success. Luckily, the specifications allow the design to depend on the reliability of many of the functions and processes of the system infrastructure without having to implement additional fault tolerance measures. As described in the system specifications, such protocols as the link-layer between clients and APs and the wired network between APs and the server are expected to function reliably. The single component of the system that is expected to experience significant outages is the IS&T server.

Steadfast tackles server failures through a series of measures that enable the system to continue functioning even when faced with an outage. The AP-server communication protocol implements a retry mechanism that incorporates acknowledgements upon successful data transport. As such, when the server eventually recovers after the maximum two minutes, it will receive the earlier dropped messages and will be able to utilize the accompanying timestamps to reliably file the data into its data table. Since the APs send their data to the server in thirty-second chunks, the server could expect to receive a maximum of four failed messages from each AP after recovering from a worst-case outage. These data chunks are not only designed to lighten the load on the APs, but also to allow the server to recover more elegantly from a failure state. If the server happens to fail when completing its preprocessing procedures described in Section 3.1, it is able to rapidly continue its efforts upon recovery without having to worry about significant data loss or its effect on the greater system. In addition, since the connection processes are completed on the client, a server outage would never affect the client's ability to connect to the greater network.

Although not described in the design specifications, Steadfast's design is also capable of recovering from failures in other system modules such as APs. Since nearby APs communicate between one another, a failing AP would be able to quickly recover through polling its neighboring APs and gathering the outdated metrics to update its data table.

4.6 Security

Security is not one of the primary design goals considered in Steadfast's design. Since the design specifications indicated no need for added layers of security, Steadfast elects to bypass excessive security protocols and to instead focus on the primary design goals of the system such as user happiness. Some ways that the lack of security may be exploited include manipulation of the server, interception of transmitted messages, and targeted attacks on the clients of the system. Although measures to prevent such attacks must be implemented before the system can be realized in the real world, the lack of security across the system is acceptable for this initial design proposal.

While not a requirement of the system's design, the updated design specifications offered a thought experiment that investigated the potential of an attacker to geographically track the system's users through their MAC addresses. While prior designs of the system would have been vulnerable to such an attack, Steadfast never receives nor stores individual client's MAC addresses essentially eliminating the threat of such a security breach.

5 Conclusion

Steadfast is a robust system that can effectively connect and retain users on the MIT network. The system is highly scalable, performs well in various use cases, and is able to recover effectively from outages. While the system goals and specifications guided design decisions, each decision was made with calculated consideration of user happiness and overall network utilization. Through various mechanisms focused on client connection and load balancing, Steadfast is able to function efficiently and effectively in the most strenuous network conditions. In addition, the system uses various communication channels between the server and the APs and takes into consideration various requirements of the client to intelligently identify, locate, and connect users to an AP that is optimal both for the client and for the system. Some tradeoffs were ultimately made in specific facets of the design, but were calculated with the primary design goals in mind. While some of these tradeoffs, such as the lack of security, may need to be amended before Steadfast can be realized, the system still successfully meets all of the design specifications while achieving all of its design goals.

6 Acknowledgements

We would like to thank our instructors Professor Mark Day and Dr. Amy Carleton for their helpful feedback both on the design of the system and the written report. We would also like to thank Professor Katrina LaCurts for providing helpful insight into the world of computer systems design.