Thread Network In Clyde Building Report

Devon Ward, Daniel Harman

April 12, 2024

Introduction

Thread is a mesh networking protocol which has been developed mainly for smart home devices [Ste]. The main idea is to remove a single point of failure for the networking smart home devices, allowing them to communicate with each other and avoid having a potential problem where one device fails, taking down the entire network. Another beneficial aspect of thread is that it is designed to be low power [RR19], which is also a really good thing for IoT devices, which are often power constrained.

One problem that Thread networks have currently is that they are not very well tested. There are many sources that have successfully tested Thread networks and proven that they can operate in a noisy environment but most publications are focused on throughput achievements, rather than distance and reliability.

Our project seeks to attempt to classify the performance of Thread in the Clyde engineering building at Brigham Young University, and determine if a Thread network is capable of allowing two users to communicate from the graduate research lab in CB 480 to anywhere in the Clyde building. In doing so, we also seek to understand how many nodes are required to allow for uninterrupted communication in the Clyde building.

Our experiments seek to quantify the range of the Thread devices using ESPRESSIF ESP32-H2 Dev boards [Sys] to get an estimate of how many routers we will require for the Thread network. We then plan to strategically place the routers throughout the Clyde building, and attempt to walk through the building without losing connection to the parent node in the graduate lab. We will quantify exactly how many router Thread nodes we will require, and attempt to use the minimum number possible.

We found that we only required one Thread router node, one parent node, and one child node to allow the child node to go all around the basement of the Clyde, and through the rest of the building without losing connection. We were also able to briefly go outside the Clyde building and still communicate with the graduate lab.

Related Work

There are many sources that have successfully tested Thread networks and proven that they can operate in a noisy environment [RIR18] but very little to classify their performance in indoor urban environments. Work has previously been done to quantify the throughput of Thread networks in an indoor area [Jai+23] [Sis+19] but most publications are focused on throughput achievements, rather than distance and reliability.

Many tutorials exist for building a Thread network with an ESP32 [Ope] as well as details on how to implement the design and program the microcontroller [ESP] to run the thread network.

There are papers that discuss distance as a metric for the Thread network operating [Kha+23] that talk about the jitter in the network. However, they do not discuss the maximum distance for transmission when the Thread mesh breaks down. For our project we wanted to investigate the maximum transmission distance for an indoor environment containing lots of metal doors over multiple floors, something that is not present in the current literature.

1

Methodology

We approached our problem in the following steps:

1. Program the ESP32-h to run a thread network

2. Set up a TCP thread connection built on top of UDP to allow secure reliable communication over the thread network to measure performance

3. Measure distance leader and child thread devices can communicate

4. Add router thread nodes until we can map the entire Clyde building

In order to program the ESP32-h devices, we used ESPRESSIF's tutorials which allow us to flash the ESP32 devices with an already functional thread network. This significantly reduced the time for development of the thread network.

Second, we wanted to set up a TCP connection. This is unintuitive because thread by design uses UDP, and there is no real sense of acknowledgements, since the mesh network is utilizing router devices to send UDP packets from the leader to the child nodes. However, we want to see when we lose connection with the child nodes as we walk around in the Clyde building. Therefore, we use a TCP client built on top of the UDP client to see when we stop receiving acknowledgement packets from the child to the leader node.

Once the TCP mesh network is set up and running, we then walk around the Clyde building with only the leader and child, and categorize the number of walls we can transmit through, and the standard distance we can communicate. By doing this we can find the blind spots of the Clyde building that we cannot reach without router nodes in the mesh network. Then we add single router nodes one at a time and see how many router nodes we need to cover all of our previously identified blind spots and have reliable communication throughout the entire Clyde building.

Implementation/Experimentation

Describe your experiments and your results. Discuss your results.

Conclusion

Summarize your work and your results. Indicated any future directions.

References

References

[RIR18]   Wojciech Rzepecki, Łukasz Iwanecki, and Piotr Ryba. "IEEE 802.15.4 Thread Mesh Network – Data Transmission in Harsh Environment". In: *2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*. 2018, pp. 42–47. DOI: `10.1109/W-FiCloud.2018.00013`.

[RR19]   Wojciech Rzepecki and Piotr Ryba. "IoTSP: Thread Mesh vs Other Widely used Wireless Protocols – Comparison and use Cases Study". In: *2019 7th International Conference on Future Internet of Things and Cloud (FiCloud)*. 2019, pp. 291–295. DOI: `10.1109/FiCloud.2019.00048`.

[Sis+19]   Srikanth Sistu et al. "Performance Evaluation of Thread Protocol based Wireless Mesh Networks for Lighting Systems". In: *2019 International Symposium on Networks, Computers and Communications (ISNCC)*. 2019, pp. 1–8. DOI: `10.1109/ISNCC.2019.8909109`.

[Jai+23]   Dheryta Jaisinghani et al. "IoT in the Air: Thread-Enabled Flying IoT Network for Indoor Environments". In: *2023 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*. 2023, pp. 142–147. DOI: `10.1109/PerComWorkshops56833.2023.10150308`.

[Kha+23]   Sohaib Bin Altaf Khattak et al. "Enhancing Wheelchair Communications Utilizing Thread Protocol for Improved Patient Safety". In: *2023 IEEE International Smart Cities Conference (ISC2)*. 2023, pp. 1–6. DOI: `10.1109/ISC257844.2023.10293573`.

[ESP]   ESP32. *Networking API - Thread*. URL: `https : / / docs . espressif . com / projects / esp - idf / en / stable / esp32 / api - reference / network / esp _ openthread . html`. (accessed: 04.12.2024).

[Ope]   OpenThread. *Build a Thread Network with the ESP32H2 and ESP Thread Border Router Board*. URL: `https : / / openthread . io / codelabs / esp - openthread - hardware # 1`. (accessed: 04.12.2024).

[Ste]   Steve. *Everything you need to know about the Thread network*. URL: `https : / / nuventureconnect . com / blog / 2021 / 07 / 12 / everything - you - need - to - know - about - thread/`. (accessed: 04.12.2024).

[Sys]   Espressif Systems. *ESP32-H2-DevKitM-1*. URL: `https : / / docs . espressif . com / projects / espressif - esp - dev - kits / en / latest / esp32h2 / esp32 - h2 - devkitm - 1 / index . html`. (accessed: 04.12.2024).