NC STATE UNIVERSITY

College of Engineering

Department of Mechanical and Aerospace Engineering

MAE 480 - Aerospace Vehicle Design

Validation, Verification, and Testing Proposal

Timothy Japit

Communications, Ground Station, and Data Handling Lead

Rover to Define Terrain
(RTDT)

Constituent	Name	Affiliation	Approval	Date
Customer	Dr. Steven Berg	NCSU		
Space Section Instructor	Michael Hughes	NCSU		
Course Advisor	Dr. Felix Ewere	NCSU		

Table 1-1. Approvals

1 Introduction

1.1 Purpose

To verify that the RTDT CubeRover and its subsystems meet the design requirements that was specified, as well as testing to validate the CubeRover under the conditions that were specified by the project definition. Verification tests will be done by component before the system integration tests to simplify failure analysis and validation tests will be done after the completion of the verification tests.

1.2 Scope

The scope of this document will cover the Communications subsystem, testing the performance of the 2.4 GHz 802.11n communications protocol with TCP/IP socket. The performance tests include a data transfer rate or bitrate test and a test on the range limit of the communications protocol. These are the verification tests that will confirm the functionality of the built-in Wi-Fi capabilities within the Raspberry Pi 3 Model B+ processor onboard the rover.

1.3 Subsystem Overview

The Communications subsystem is responsible for communicating payload and telemetry data from the rover to the ground station and commands from the ground station to the rover. The rover is equipped with Raspberry Pi cameras for detecting obstacles, temperature sensors for internal temperature readings, accelerometers to keep track of its orientation, and many other data acquiring tools that feed their information to the onboard computer. The information collected will then be transmitted via the 2.4 GHz frequency to the lander that will act as a relay to communicate with NASA's ground station. Conversely, the commands for the rover will be sent to NASA ground station, transmitted to the lander through the S-band, and forwarded to the rover via the 2.4 GHz frequency.

1.4 Driving Requirements

The functional requirement driving the verification tests on this document is **FR 8**. This functional requirement states that the CubeRover shall communicate with the ground station through the lander. The implication is that the rover would have to use a communications protocol that is supported by the lander. The communications protocol is specified in **DR 8.1**, declaring that the rover shall use 2.4 GHz 802.11n Wi-Fi radio using TCP/IP to communicate with the lander. This protocol was chosen because of its prevalence and its success history by Astrobotic who used this protocol for their rover with the Peregrine lander [1]. Derived from this requirement are design requirements **DR 8.1.1** and **8.1.2**, which correspond to the specifications of the data rate and communications range, respectively. The data rate shall be greater than or equal to 40 kilobits per second (kbps) to enable near instantaneous commands to be sent in case of errors or navigational problems where immediate action is required, and the communications protocol shall have a strong enough signal to maintain connection within a 100-meter radius of the lander. By verifying these design requirements, the Communications subsystem of the CubeRover shall communicate with the ground station through the lander.

2 Literature Review

2.1 Data Rate/Bitrate

The data transfer rate or bitrate of a radio communications protocol of a certain frequency are limited by a number of factors such as, the bandwidth that is available for that frequency band, the number of levels in the digital signal, and the level of noise or disturbance that is on a channel [2][3]. The Nyquist Bitrate defines the theoretical maximum bitrate in a noiseless channel,

$$C_{noiseless} = 2B\log_2 M \tag{1}$$

where $C_{noiseless}$ is the capacity or the bitrate of the noiseless channel in bits per second (bps), B is the bandwidth of the channel in hertz (Hz), and M is a dimensionless unit that represents the number of discrete levels in the digital signal used to represent the data being communicated [2][3]. This implies that the theoretical maximum bitrate of a noiseless channel will increase proportionally as the bandwidth of the channel increases or increase logarithmically as the number of levels in the signal increase. However, using the Nyquist theorem to calculate the theoretical maximum in a noisy channel and increasing the levels in the signal in hopes of increasing the capacity could cause problems.

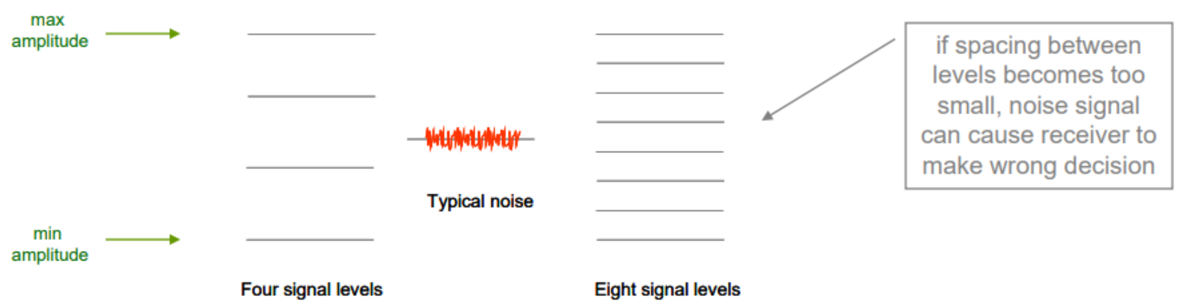

Figure 2-1 Discrete Signal Level Spacing [2]

Calculations with the Nyquist theorem are usually paired with the Shannon Law which defines the maximum theoretical bitrate limit over a channel with Gaussian distributed noise [2].

$$C_{noisy} = B\log_2(1 + SNR) \tag{2}$$

where C_{noisy} is the capacity or bitrate of the noisy channel in bits per second (bps), B is the bandwidth of the channel in Hz, and SNR is the unitless signal to noise ratio compared in watts. The number of discrete levels in the message signal is not included in Eq. (2) implying no direct correlation between the levels and capacity. For a channel of a given bandwidth, the Nyquist theorem is used to determine the number of signal levels that should be used for the channel and Shannon Law is used to determine the maximum data rate [2][3][4]. The conversion between the signal and noise measured in dBm to milliwatts can be shown from the equation below,

$$S_{mW} = 10^{\frac{S_{dBm}}{10}} \tag{3}$$

where S_{mW} is the signal/noise strength in milliwatts and S_{dBm} is the signal/noise strength in dBm. Then once the conversion to milliwatts has been done, the signal to noise ratio can be calculated [4][5]. Otherwise, a simple method to calculate the bitrate is to determine a data size to send and then by using the *ping* command in the command line of a computer to obtain the latency of a round trip communication,

$$C = \frac{2D}{L * 10^{-3}} \tag{4}$$

where *C* is the data rate in bytes per second (Bps), *D* is the size of the data sent each trip in bytes, and *L* is the average round trip latency reported from the *ping* command [6].

2.2 Range

The range for a communications system depends on the frequency band, the protocol that is used, as well as the obstructions that is in between the two devices or an access point (AP) and a client device. Received Signal Strength Indicator (RSSI) is the measure of how strong the signal received by the client device is at some distance from the AP [7]. The Path Loss or Path Attenuation equation is used to calculate the RSSI of a device with an unobstructed line of sight to the AP,

$$RSSI(d) = RSSI(d_0) - 10n_p \log_{10}\left(\frac{d}{d_0}\right) \tag{5}$$

where *RSSI* is measured in decibel-milliwatts (dBm), *d* is the distance between the device to the AP in meters, and n_p is the attenuation factor in dBm. The calculation is done by assuming a baseline value or measuring the signal strength from a reference distance d_0 [8][9][10]. However, the purpose of the verification test is to determine the maximum range that the rover can drive away from the lander while maintaining a strong enough signal strength to communicate data. Therefore, a range of acceptable signal strengths that are required for usable connectivity is required and presented as follows,

Signal Strength (dBm)	Description	Required for
-30	Maximum achievable, client is a few feet from the access point	N/A
-67	Minimum for applications requiring reliable and timely packet delivery	VoIP/VoWi-Fi, video streaming
-70	Minimum for reliable packet delivery	Email, web browsing
-80	Minimum for basic connectivity	N/A
-90	Unusable	N/A

Table 2-1 Acceptable Signal Strengths [9]

Eq. (5) was then rearranged to obtain a distance measure from a reference distance and RSSI, and acceptable RSSI,

$$d = d_0 * 10^{[RSSI(d_0) - RSSI(d)]/10n_p} \tag{6}$$

by measuring a reference RSSI with a distance of 1 meter away from an access point or signal source, referencing Table 2-1 for signal strengths, and applying Eq. (6), it is possible to calculate the theoretical distance that the rover has to travel away from the lander to maintain the desired signal strength.

To properly verify the performance of the communications protocol, some baseline numbers were obtained through research to serve as reference. The Institute of Electrical and Electronics Engineers (IEEE) specifies that the 2.4 GHz 802.11n should have a bandwidth of 20 MHz for each channel. The theoretical maximum data rate for this frequency band is up to 288.8 Megabits per second (Mbps), it has experimental indoor maximum range of around 70-150 meters and maximum outdoor range of up to 250 meters [11][12][13][14].

3 Verification Tests

3.1 Overview

There are two experimental tests that will be conducted to verify the performance of the Communications subsystem of the RTDT CubeRover. These tests are the data transfer rate or bitrate test and the communications range limit test. The bitrate test will measure how many bits of data can be transferred between the rover and lander every second, while the range limit test will confirm how far the rover can drive away from the lander before it loses connection. The set-up for both tests involve connecting the Raspberry Pi onboard the rover to a desktop computer that serves as a mock lander.

3.2 Data Rate/Bitrate Test

3.2.1 Test Design

A desktop computer or a laptop with a known will be set as a stationary mock lander. It should have a wireless local area network capability to communicate at 2.4 GHz 802.11n with TCP/IP socket. The CubeRover will be drive to a fixed distance of 100 meters from the mock lander and test 5 points in the perimeter of that distance. The mock lander shall ping the rover 3 times at each point from the command line, each ping consisting of 4 packets of 20,000-byte (20 kilobytes) data. Once entered, the mock lander will be sending 20 kilobytes and receive 20 kilobytes worth of data. The minimum, maximum, and average round trip times will be recorded in milliseconds and averaged over the 5 points in which the data is taken. There will be a total of 60 data points averaged of lander-rover-lander transmission over this experiment.

3.2.2 Procedure

1. Power on the computer/laptop that will serve as the mock lander
2. Confirm networking capability (2.4 GHz 802.11n) by checking in Device Manager > Network Adapter
3. Drive the rover to 100 meters away from the mock lander
4. Open command line
5. Type **ping -l 20000 <CubeRover IP address>**
6. Repeat step 5, 3 times
7. Record the minimum, maximum, and average round trip times
8. Repeat steps 4-7 on 4 different locations along the perimeter
9. Average the round-trip times from all 15 data points
10. Calculate the data rate using Eq. (4) and convert it to kilobytes per second

3.2.3 Required Resources

- Windows computer/laptop with 2.4 GHz 802.11n
- Monitor and HDMI/Display Port cables
- Mouse and keyboard
- Power outlet
- Microsoft Excel or Matlab for data recording
- Open field for the rover to distance itself from the mock lander
- Table and chair for test administrator

3.2.4 Expected Results

The resulting average round trip times is expected to be less than or equal to 1000 milliseconds. A latency equal to that number will return a data rate of 40 kilobytes per second as specified in **DR 8.1.1**.

3.2.5 Success Criteria

Three levels of success and the failure condition are defined as follows,

- **Exceptional:** Average round trip times is less than or equal to 100 milliseconds, implying a data rate of greater than 400 kilobytes per second.
- **Extensive:** Average round trip times is less than or equal to 500 milliseconds, implying a data rate of greater than or equal to 80 kilobytes per second.
- **Acceptable:** Average round trip times is less than or equal to 1000 milliseconds, implying a data rate of greater than or equal to 40 kilobytes per second.
- **Failure:** Average round trip times is greater than 1000 milliseconds, implying a data rate of less than 40 kilobytes per second.

3.3 Range Limit Test

3.3.1 Test Design

The test design for the range limit test is similar to the bitrate test. A desktop computer or a laptop with a known will be set as a stationary mock lander. It should have a wireless local area network capability to communicate at 2.4 GHz 802.11n with TCP/IP socket. The difference is that the CubeRover will have to drive to varying distances away from the mock lander. The rover will function as a server in this test such that the mock lander can pose as the client to connect and request data. Once connection is established through TCP/IP with a secret port number, the rover can drive away from the mock lander. The test administrator will request the dBm from the rover through the command line periodically until the rover is 100 meters away from the mock lander to monitor that the RSSI is less than or equal to -67 dBm. A number of 5 readings will be taken along the 100-meter perimeter, the minimum, maximum, and average of the data points will be reported.

3.3.2 Procedure

1. Power on the computer/laptop that will serve as the mock lander
2. Confirm networking capability (2.4 GHz 802.11n) by checking in Device Manager > Network Adapter
3. Start the program that will report the measured dBm on the rover and client program on the lander through the command line
4. Connect the lander to the rover that will function as a server
5. Record the RSSI reported at 1 meter away from the mock lander as a reference
6. Drive the rover to 100 meters away from the mock lander while requesting the RSSI at the 25, 50, and 75-meter checkpoints
7. Request the RSSI 5 times at different spots along the 100-meter perimeter
8. Record the minimum, maximum, and average dBm
9. If the average RSSI reported is less than or equal to -67 dBm, drive further away from the lander while requesting RSSI values at 25-meter intervals
10. Record the distance once the reported RSSI is consistently at -90 dBm or greater

3.3.3 Required Resources

- Windows computer/laptop with 2.4 GHz 802.11n
- Monitor and HDMI/Display Port cables
- Mouse and keyboard
- Power outlet
- GCC (C compiler) and GDB (GNU debugging tools for C)
- Cygwin for compiling C programs in Windows
- Microsoft Excel or Matlab for data recording
- Open field for the rover to distance itself from the mock lander
- Table and chair for test administrator

3.3.4 Expected Results

The theoretical expectations from a measured -40 dBm at 1 meter away from the lander calculated using Eq. (6) are tabulated as follows,

Range (m)	Signal Strength (dBm)	Minimum Requirement for
1	-40	N/A (baseline)
100	-60	N/A
500	-67	VoIP/VoWiFi, video streaming
1000	-70	Email, web browsing
10,000	-80	Basic connectivity
100,000	-90	N/A (unusable)

Table 3-1 Theoretical Signal Strength at Varying Range

The experimentally measured signal strength is expected to be around -67 dBm with the CubeRover at a distance of 100 meters from the mock lander, adhering to design requirement **DR 8.1.2**.

3.3.5 Success Criteria

Three levels of success and the failure condition are defined as follows,

- **Exceptional:** Signal strength is indicated to be less than or equal to -67 dBm with the rover at a distance of less than or equal to 500 meters away from the mock lander.
- **Extensive:** Signal strength is indicated to be less than or equal to -67 dBm with the rover at a distance of less than or equal to 250 meters away from the mock lander.
- **Acceptable:** Signal strength is indicated to be less than or equal to -67 dBm with the rover at a distance of less than or equal to 100 meters away from the mock lander.
- **Failure:** Signal strength is indicated to be greater than -67 dBm with the rover at a distance of less than or equal to 100 meters away from the mock lander.

4 Schedule

4.1 Timeline

2021	January			
	1-8	9-15	16-22	23-31
Data Rate Test				
Range Limit Test				

Table 4-1 Gantt Chart of Relevant Tests

4.2 Tasks

The tasks for the foreseeable future include:

- Procuring the Raspberry Pi 3 Model B+
- Creating programs for the range limit testing
- Testing the programs on the client and server side individually and as a system
- Finding an open field that will be large enough for both tests
- Procuring the computer/laptop that will serve as the mock lander
- Conducting verification tests

5 Conclusion

Once the tests are conducted, the limits of the data transfer rate and range for the specific chip onboard the Raspberry 3 Model B+ will be revealed. If any one of the success criteria in both tests are achieved, then the Communications subsystem team shall move on to integration testing with the GNC subsystem for the communication of rover commands and the Payload subsystem for the communication of payload data. Otherwise, the Communications subsystem team shall procure an antenna with a strong enough gain that should be able to increase the throughput of the data rate and further the range of the 2.4 GHz band onboard the rover and re-test the subsystem with the same test design and procedure.

References

[1] "CubeRover Payload User Guide," *Astrobotic*, 2020, www.astrobotic.com/cuberover-payload-user-guide.
[2] "Data Rate Limits in Digital Transmission," *York University*, 2019, https://www.eecs.yorku.ca/course_archive/2015-16/F/3213/CSE3213_06_AnalogDigitalTransmission_F2015_posted_part3.pdf
[3] Goyal, A., "Maximum Data Rate (Channel Capacity) for Noiseless and Noisy Channels," *GeeksforGeeks*, 2019, https://www.geeksforgeeks.org/maximum-data-rate-channel-capacity-for-noiseless-and-noisy-channels/
[4] Pierce, J. R., "An introduction to information theory: Symbols, signals & noise," *Dover Publications*, 1980, 2nd edition, New York.
[5] Ian, "Difference Between dB and dBm," *DifferenceBetween.net*, 2011, http://www.differencebetween.net/science/difference-between-db-and-dbm/
[6] zx8754, "Calculate Upload/Download Speed by Ping," *stackoverflow*, 2019, https://stackoverflow.com/questions/4575537/calculate-upload-download-speed-by-ping
[7] "RSSI," *speedcheck*, retrieved Oct. 9, 2020, Available: https://www.speedcheck.org/wiki/rssi/#signal-strength
[8] Premachandran, R., "SNR, RSSI, EIRP and Free Space Path Loss," *Cisco*, retrieved Oct. 9, 2020, Available: https://community.cisco.com/t5/wireless-mobility-documents/snr-rssi-eirp-and-free-space-path-loss/ta-p/3128478
[9] "Understanding RSSI," *metageek*, retrieved Oct. 9, 2020, Available: https://www.metageek.com/training/resources/understanding-rssi.html
[10] Chruszczyk, L., Zajac, A., & Grzechca, D., "Comparison of 2.4 and 5 GHz WLAN network for purpose of indoor and outdoor location," *International Journal of Electronics and Telecommunications*, Vol. 62, No. 1, 2016, pp. 71-79. http://dx.doi.org.prox.lib.ncsu.edu/10.1515/eletel-2016-0010
[11] "IEEE 802.11," *Wikipedia*, https://en.wikipedia.org/wiki/IEEE_802.11#Protocol, retrieved Oct. 29, 2020
[12] "IEEE 802.11ac: What Does it Mean for Test?," *Litepoint*, https://web.archive.org/web/20140816132722/http://litepoint.com/whitepaper/80211ac_Whitepaper.pdf, retrieved Oct. 29, 2020
[13] IEEE, "802.11-2016 - IEEE Standard for Information technology—Telecommunications and information exchange between systems Local and metropolitan area networks—Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE*, 2016, retrieved Oct. 8, 2020, Available: https://ieeexplore.ieee.org/document/7786995
[14] Villegas, E. G., López-Aguilera, E., Vidal, R., & Paradells, J., "Effect of Adjacent-channel Interference in IEEE 802.11 WLANs," *2007 2nd International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, pp. 118-125, 2007. doi:10.1109/crowncom.2007.4549783