SSC18-PII-19

An Open Source Radio for Low Cost Small Satellite Ranging

Grant Iraci* and Chris Gnam†
University at Buffalo Nanosatellite Laboratory
240 Bell Hall, Buffalo, NY 14260

Faculty Advisor: Dr. John Crassidis
University at Buffalo, State University of New York

ABSTRACT

Precise orbit knowledge is crucial for many satellites and their missions. While GPS is capable of providing the accuracy required by nearly all small satellite missions, many CubeSats forgo the use of GPS modules due to various constraints. Because of this, CubeSats often make use of Two Line Elements sets (TLEs) provided by the Joint Space Operations Center. However due to the potential for infrequent updates, and the possibility of "cross-tagging", the use of TLEs presents issues for many CubeSat missions. A low cost (<$50) method of obtaining range measurements using a satellite's communications radio is presented. The ranging precision is found to be 0.1552 km under strong signal conditions and .3038 km under realistic worst case conditions. These results were then incorporated into an orbital simulation to test TLE identification and orbit determination. It is demonstrated via simulation that these range estimates can be used to reliably identify the correct TLE in the case where TLEs may be mismatched or "cross-tagged" after initial deployment using only the satellite's existing radio hardware. The proposed communications radio and ping measurement design are open source, and the complete source code , hardware design, and supporting documentation can be found at https://github.com/UBNanosatLab

INTRODUCTION

Orbit knowledge is a key part to many small satellite missions, especially when collected data is linked to ground location. With the advent and widespread availability of GPS, precise orbit knowledge may appear to be a solved problem. However, GPS receivers suitable for satellite use have a number of drawbacks that make them impractical for many small satellites. GPS receivers are power consuming, take up valuable real estate inside the satellite, and require electrical and software interfaces with the rest of the system. Additionally, in the U.S. and many other countries, GPS modules suitable for use in satellites are subject to a number of regulatory hurdles that make procurement difficult, time consuming and expensive.

This typically results in many small satellite missions using Two-Line Element sets (TLEs) provided by the Joint Space Operations Center (JSpOC). These TLEs can be propagated via use of the Standard General Perturbations 4 (SGP4) model, however like all propagation methods, propagation via SGP4 is subject to error over time, and TLEs are not necessarily updated regularly. Previous studies of the Planet Labs Flock 1B satellites have shown a 1-sigma error of 20 - 70 km after 2 days.[1] Data provided by Planet Labs also shows that TLEs can be subject to cross-tagging shortly after initial deployment, where the TLEs are mismatched between two or more satellites. This results in errors that can grow large enough to disrupt communication with a ground station, which would be tracking the wrong object.

For these reasons, it is advantageous to repurpose existing hardware in the satellite to obtain sufficient orbit knowledge. The proposed design accomplishes this via a software modification to a general purpose CubeSat communications radio currently in development. This modification allows for the acquisition of reasonably precise range measurements without additional satellite hardware.

With only a single ground station these range measurements can be used to reliably identify TLE cross-tagging via a maximum likelihood approach. With

* Undergraduate Student, Department of Computer Science: grantira@buffalo.edu
† Undergraduate Student, Department of Aerospace Engineering: crgnam@buffalo.edu

the introduction of two or more ground stations, these measurements can obtain a full orbital state estimate.

This paper presents a low cost design and prototype implementation of a space-based transceiver capable of obtaining range estimates for small satellites. Based on the performance data collected from this prototype implementation, a method for identifying TLE cross-tagging is developed and simulated. A second ground station is then added to the simulation to demonstrate that this ranging method can be used to obtain a full orbital state estimate.

DESIGN OVERVIEW

Satellite Hardware

The end of the system in the satellite is based around an open-source CubeSat data radio currently in development at the University at Buffalo. The radio centers on a Silicon Labs Si4464 Radio Frequency Integrated Circuit (RFIC) using Gaussian Frequency Shift Keying (GFSK) modulation. This was chosen primarily for its wide frequency coverage, allowing radios based on this design to operate in both VHF and UHF satellite communication bands. It also supports routing many signals normally internal to the radio to external pins, allowing precise measurement of timing events. Paired with the RFIC is a power amplifier and an MSP430 microcontroller (MCU), chosen for its naturally radiation-resistant Ferroelectric Random Access Memory (FRAM). Also present on the board are a Temperature Compensated Crystal Oscillator (TCXO) to provide an accurate frequency reference for the transmitter module, an RF switch for switching between transmit and receive signal paths, and a digital to analog converter which controls amplifier power. The prototype radio and supporting electronics are assembled into an add-on board for a commercial off-the-shelf MSP430 development board.

Ground Station

The ground station used in this effort was a repurposed GPS tracker board developed for high altitude balloon flights. It utilizes the HopeRF RFM26W breakout module, based around the Si4463, another RFIC from the same family as the one used in the data radio. The ground station is controlled by a Teensy LC ARM microcontroller breakout board. The Teensy LC has a high resolution timer with an input capture feature, allowing the precise timing of external events. With minor modifications to software, the ground station could be replaced with an identical copy of the hardware used on the satellite in future experiments.

Firmware Design

Range measurements are obtained by measuring the time required for a transmission to be sent by the ground station, received by the satellite, replied to, and received by the ground. Keeping processing delay in the satellite radio consistent is essential to minimize error in measured position. The firmware controlling both radios is designed from the ground up with this use case in mind. The firmware is written in C and makes use of interrupts for precise timing.

Also critical to keeping delays consistent is having precise knowledge of when packets are transmitted and received. Packets consist of five major parts: a preamble to train the receiver's clock recovery, a sync word to distinguish packets from noise and provide phase information, a length byte, payload data, and a checksum to verify data integrity. When the RFIC receives and recognizes the sync word, it is configured to drive an external pin high, triggering an interrupt inside the MCU which starts a hardware delay timer. The sync word is used for this because it immediately follows the transition-dense training sequence, meaning the receiver has good knowledge of the bit timing of the transmitter, leading to a more precise landmark within the packet.

After the entire packet has been received, the RFIC notifies the MCU that a packet is available, which then copies the packet into an internal buffer and checks its contents. If the packet is not a ping packet, the timer is stopped and the packet is processed to be handed over to a flight computer. If the packet is a ping, the MCU prepares a response, loads it into the internal buffer of the RFIC, and waits. Once the delay timer expires, it triggers an interrupt in the MCU, which then commands the RFIC to transmit the enqueued response packet.

The ground station times this entire process by transmitting a ping packet. The RFIC on the ground is configured to drive a pin high when it starts transmitting the packet, which triggers a capture of the current timer counter value in the Teensy LC. When receiving the reply packet, the RFIC drives another pin high upon detecting the sync word, which again captures the current value from the timer counter. The firmware in the ground station then computes the time difference between the start of transmission and receiving a response and relays this information back to the ground station computer to be logged.

Computing the Range

To obtain time-of-flight measurements, the system is first calibrated at zero distance. In the case of a satellite this would be done upon integration of the flight unit. Calibration requires sending a large number of pings to the radio and timing the response time. Taking a large number of measurements and averaging them together establishes a baseline of the delays introduced by the radio. Using this baseline, the time of flight can be computed by:

$$t_{tof} = \frac{t_{measured} - t_{baseline}}{2} \tag{1}$$

The factor of two comes from the fact the measurement is a round trip measurement and therefore contains two one way distances. From the time of flight, the range can be obtained by the equation:

$$\rho = t_{tof}\, c \tag{2}$$

where c is the speed of light.

HARDWARE TESTING

Methodology

To evaluate the performance of the radio design, prototypes of the satellite and ground station hardware have been constructed. These prototypes are then used to measure the round-trip ping time, including time of flight as well as implementation delays, under a variety of signal conditions.

The first scenario considered is nearly ideal conditions. The radios were placed in close proximity so as to obtain a very high signal to noise ratio. Then the ground station is commanded to send a volley of 10,000 pings to the satellite and record the time until it heard a response to each. These times are then logged on a computer for analysis.

The second scenario approximates realistic worst-case signal to noise ratios. The radios are isolated from one another within a building until they can no longer communicate. They are then connected by a coaxial cable with attenuation in line. The attenuators are adjusted until the proportion of packets received indicates a 10^{-4} bit error rate (BER). This BER is chosen as it is the highest BER that is considered while performing link budget analysis for the communications system under development.

The third scenario explores the effects of unrealistically low signal to noise ratios. The same coaxial cable test setup is used for this test, but with enough attenuation to bring the bit error rate above two orders of magnitude worse than what is considered the worst case (10^{-2}).

The fourth and final scenario considered is a short range field test. This test spans approximately 1.1 km where there is direct line-of-sight. A line-of-sight path is used as it is realistic to what the system will experience in orbit and reduces the potential for reflections and attenuation from buildings affecting the measurements. The actual distance is confirmed with GPS receivers at both ends. This test is only run for 1,000 pings.

Results

Table 1 shows the standard deviation in the timing signals, and the associated standard deviation in the range measurements, for the different test cases.

Table 1 - Standard Deviation of ½ Round Trip

	Standard Deviation (μs)	**Standard Deviation (km)**
Near Ideal	0.5174	0.1552
High BER	1.0128	0.3038
Extreme BER	1.6050	0.4815
Short Range	0.6554	0.1965

Overall, as can be seen in Figure 1 and Figure 2 the distribution of ping response times closely follows a Gaussian distribution, illustrated in red. The response times are still normally distributed even under the extreme BER case. This suggests a Gaussian distribution is an appropriate model for the error and justifies its use in simulation. Note that the plotted data is the error with respect to the sample mean and measures round trip time and therefore has twice the standard deviation of the range measurements.

Bench testing results show high precision measurements are possible. Under nearly ideal circumstances, the standard deviation of round trip ping response times is only 1 microsecond. Dividing this by a factor of two obtains the standard deviation of one of the two times of flight. Multiplying this by the speed of light yields the standard deviation of range measurements based off of the time of flight.

Results for the realistic worst case are similarly impressive. The standard deviation is only about two times worse than under the nearly ideal conditions. This indicates that while the standard deviation of ping measurements, and by extension the precision of range measurements, does get worse as the signal to noise ratio decreases, it remains reasonable under the conditions to be expected in orbit.

In the case of unrealistically high bit error rates, where the bit error rate is two orders of magnitude beyond the worst case, the radio still performs well. At a 10^{-2} bit error rate, on average, 1 out of every 100 bits is corrupt. Sending a typical 255 byte data packet (262.5 bytes including framing using this design) has approximately a 1 in 10^{9} chance of succeeding under these conditions. Under these conditions, the standard deviation of ping measurements is still only a factor of three worse than ideal. This means that any time ping measurements can be reliably heard through the noise, useful measurements can be obtained.

The short range test shows that this method is actually capable of measuring distances. Taking the first 100 ping measurements, approximately 3 seconds worth, the measured distance is calculated to be 1222 meters. A 95% confidence interval for this measurement is (1182, 1262) meters. The actual distance as measured using GPS receivers located at each end is 1149 meters, an error of +73 meters in the measured value. This does fall outside the 95% confidence interval. However, when viewed as a timing error this is only an error of 0.165 microseconds. Over the course of the entire 23000 microsecond ping process, this is equivalent to an error of 7 parts per million. Due to the ground station's use of a simple crystal oscillator lacking temperature compensation, error of this magnitude is to be expected under varying temperature conditions.

The measurements given were conducted with the ground station end outside on a cold day while the calibration was done inside. Thermal drift of the crystal oscillators driving the timers in the microcontroller is expected to be approximately 10 parts per million fast at the roughly 25 ºC temperature difference experienced. This results in the ground station overestimating the ping time, and fully accounts for the error in timing. This type of error is proportional not to the time of flight, but to the entire ping time, which is dominated by fixed delays. Therefore, an error of this magnitude at 1000 km would be only 80m. Future versions of the hardware will address this issue by using temperature compensated oscillators in the ground station.

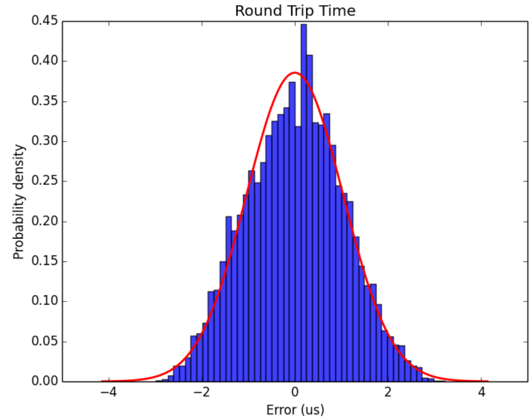

Figure 1 - Near Ideal Round Trip Error

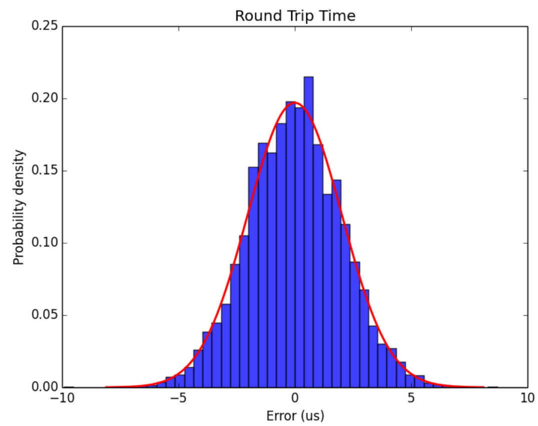

Figure 2 - Extreme BER Round Trip Error

ORBITAL SIMULATION

This section takes the hardware performance determined from the previously explained testing procedures, and applies it to an orbital simulation. This is used to evaluate the orbit determination accuracy using the proposed method. This enabled an exploration of potential applications while on orbit.

Simulation Design

In order to adequately assess the ability to both estimate the orbital state and to identify cross-tagging, realistic truth data for a CubeSat, and its corresponding TLEs, were required. Planet Labs publishes all of their orbital state data (acquired via a combination of GPS and ranging from multiple ground stations), along with the accompanying TLE history for each of their satellites, updated daily. The choice is then made to use data from their FLOCK-2P constellation, a set of 12 satellites deployed on June 22nd, 2016. These satellites are an optimal choice, as they were deployed together. Thus it enabled easy study of potentially cross-tagged objects.

For the purposes of this simulation the published state vectors produced by GPS measurements are assumed to be truth. The given state was numerically propagated considering J2 and atmospheric drag effects. This was done for each satellite in the constellation.

Each ping event is modeled in three subsections, broken up by four key events:

1. Ground station begins ping
2. Sync word received by satellite
3. Ping reply sent by satellite
4. Sync word received by ground station

When the ground station begins the ping, there is a delay of about 8ms until the sync word is sent. Both the position of the satellite and the ground station are then propagated forward by this 8ms delay to obtain their locations when the sync word transmission begins. Once the sync word is sent, the transmission then begins flight towards the satellite. For simplicity, the time of flight of the transmission is calculated based on the position of the satellite propagated immediately after the sync word is sent. This assumption is reasonable, as even under worst case scenarios, the satellite will have only moved less than 40 meters, thus the added complexity of finding the point at which the transmission and satellite meet can be neglected.

Once the sync word is received by the satellite, there is a delay of about 5ms until the satellite begins sending the return transmission. In addition, there is again the 8ms delay after the return transmission starts and when the return sync word is sent. The position of both the satellite and the ground station is then propagated forward by this amount of time, and then the time of flight of the sync word is found using the new position of the satellite and the ground station. Once the return sync word is received, there is a 10ms delay between the end of this ping event, and the beginning of the next. The satellite's orbit is then propagated forward by this new amount of time, and a new ping event can begin. A record of each time step used throughout this entire process is kept and added together, providing a time measurement of how long the ping event took.

The results from the hardware testing outlined above also characterized the statistics of the radio's performance, providing a standard deviation for the error in the ping process. This is added in as a Gaussian noise to the time measurement obtained previously, thus fully simulating an expected measurement.

In order to extract a range measurement, the calculations outlined in equations (1) and (2) are used, where the baseline is assumed to be known to arbitrarily good precision. This was done with some sample data to demonstrate the accuracy of range estimation via this method, and the results and shown in Figure 3.

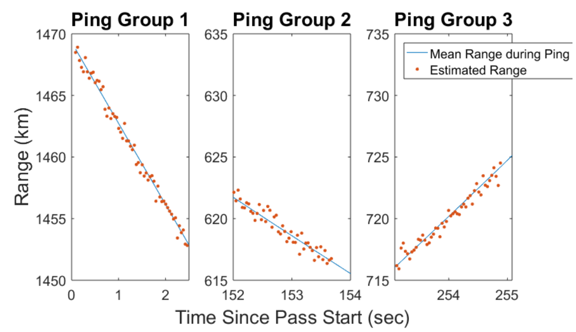

Figure 3 - Example of Simulated Ping Events

TLE IDENTIFICATION

The simulation was run for each of the FLOCK 2P satellites throughout their first week in orbit. Simulated range measurements were generated for each satellite using the GPS state data and assuming a single ground station. Then, each TLE in FLOCK 2P was propagated using SGP4, and expected range measurements for each TLE assumption were generated. Next, the probability of each range measurement given each TLE assumption is computed. These probabilities are used to calculate weights, which are used to identify the most likely assumption.[2] By observing the calculated weights for each TLE, cross-tagging can be potentially identified. As ping measurements are collected, it is expected that the weight of each incorrect TLE will decrease to 0, while the weight of the correct TLE will approach 1.

For the purposes of this estimation, we assume that the propagated states from each TLE are representative of a truth value for that particular TLE. Then the probability of range measurements $\tilde{\rho}$ given the state estimate x_k from TLE k is given by

$$p(\tilde{\rho} \mid x_k) = \frac{1}{[2\,\pi\sigma_\rho^2]^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\tilde{\rho} - \hat{\rho}_k)^2 \sigma_\rho^2\} \qquad (3)$$

Where $\tilde{\rho}$ is the measured range, x_k is the state of the current TLE, σ_ρ^2 is the variance of the estimated range, and $\hat{\rho}_k$ is the predicted range based on the current TLE.

The weights are initialized to be equal and sum to one, and are updated via a multiple-hypothesis approach [2]

$$w_k^i = w_k^{i-1} p_i(\tilde{\rho} | \boldsymbol{x_k}) \qquad (4a)$$

$$w_k^i \leftarrow \frac{w_k^i}{\sum_k w_k^i} \quad (4b)$$

where w_k^i is the weight of the k^{th} TLE at the i^{th} time step and the "←" symbol denotes replacement.

Results

The simulation is run for each satellite, until 2 ground station passes occur. This was repeated for each day for a week after the initial deployment. Each pass over the ground station produced 90 range estimates, which only took up approximately 4 seconds of each pass time. This ensures that each pass is still useful for completing nominal communication with the satellite.

This process is proven able to identify the correct TLE in 80% of cases by the end of the first week. These results are obtained using range-only measurements from a single ground station. This is shown below in Figure 4. In the remaining cases where the correct TLE was not properly identified, there were two or more satellites that remained very close together, making them difficult to distinguish.

Figure 4 shows the performance of this method for the FLOCK 2P-9 satellite on the third day after initial deployment.

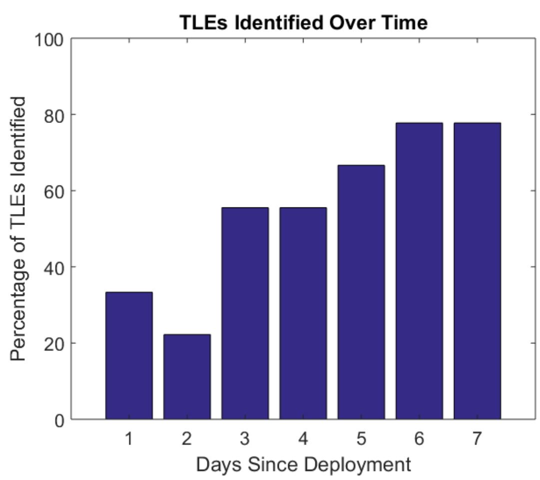

Figure 4 - TLEs Correctly Identified Over the First Week

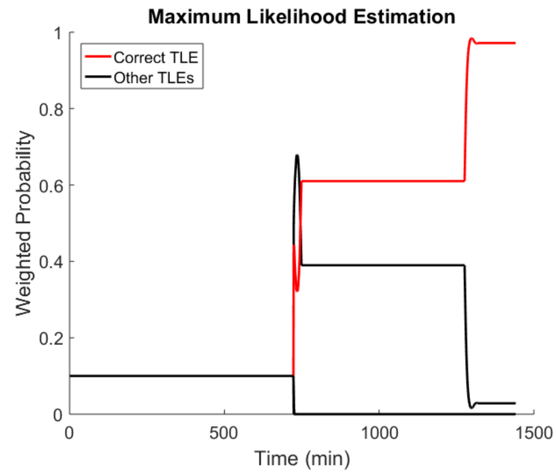

Figure 5 – Maximum likelihood solution for FLOCK 2P-9 after third day

Orbit Determination from Two Ground Stations

As demonstrated in the previous section, a single ground station can be used to correctly identify the proper TLE using only range measurements. However, due to the lack of observability of the orbital plane, it is not possible obtain a full orbit estimate from range only measurements from a single ground station.[3] If a second ground station is available however, a full orbit estimate can be obtained.

For this, the same simulation architecture outlined previously is used, however a second ground station receiver is added. When a ping event is simulated from the primary ground station, both the primary and secondary ground stations listen for the satellite's response.

For this method, both ground stations require extremely well-coordinated time. Fortunately, this can be accomplished by syncing both ground stations to GPS time. The error due to imperfect timing synchronization was added as Gaussian white noise to the second ground station's time measurement.

A standard Extended Kalman Fitler (EKF) is used to perform the state estimation. The same simulated truth model outlined before is used here, and the accompanying TLE is used as the initial estimate. The results for position error are shown in Figure 6, and velocity error in Figure 7. All estimates are within their respective 3σ bounds, and the performance characteristics are seen.

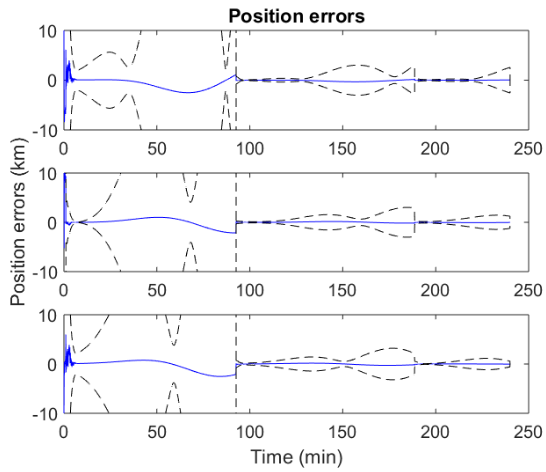

Figure 6 - Position Errors from Orbital EKF and range measurements from two ground stations

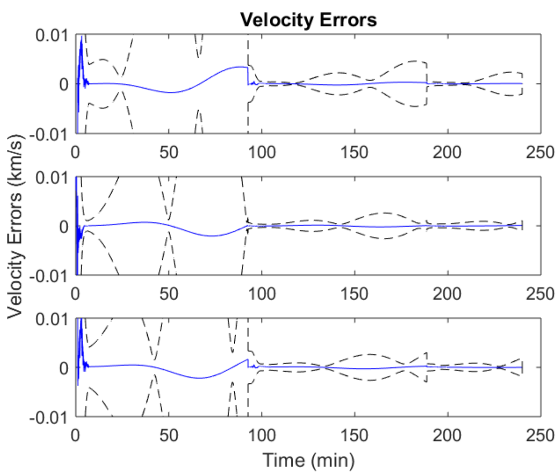

Figure 7 - Velocity Errors from Orbital EKF and range measurements from two ground stations

CONCLUSIONS

In this paper, a design of low cost radio hardware and corresponding software capable of obtaining time of flight based range measurements was introduced. A prototype of its implementation was demonstrated to have excellent performance even under low signal strength conditions.

Based on the performance of the prototype, it was demonstrated that this ranging method could be used effectively to identify the correct Two Line Element (TLE) from a deployment group. In the cases where it failed to identify the proper TLE, the resulting error was within 10km, thus not yet negatively impacting the satellites ability to communicate with the ground station. It is only when the error between multiple TLEs and truth is sufficiently small that this method is no longer effective.

When multiple ground stations are available, these range measurements can be used for obtaining a complete orbital state estimate. These estimates were demonstrated via simulation to be accurate to within 200m after two sets of ground station passes.

RELATED WORK

The potential to obtain range measurements from orbit using a communications radio was previously explored by Foster, Hallam and Mason of Planet Labs.[4] Their solution also uses time of flight measurements obtained by timing pings from the ground station, however they make use of multiple ground stations and software defined GPS.

The results obtained by Foster, Hallam and Mason showed a 1σ error of 650m for each of their range measurements from orbit. These measurements are combined to form a single orbital state estimate.

While this method is capable of identifying cross-tagged TLEs and obtaining a full orbital estimate, full implementation requires additional hardware and additional ground stations.

FUTURE WORK

The logical next step will be testing the hardware over larger distances. A high altitude balloon would provide the perfect vehicle for this test, allowing range measurements to be taken at distances of roughly 200km.

On orbit calibration remains an unaddressed problem. While nothing in the proposed design would be expected to behave differently in orbit, confirmation of this is desirable. Because most aging related drift comes from the oscillators, a potential avenue for partial on orbit calibration is the transmission of timing beacons after deployment. These beacons would allow a precise frequency reference on the ground to be used to calibrate the radio in orbit. Exploring the viability of this idea will require further experimentation.

Implementing the multiple ground stations needed for full orbital state estimates requires solving the technical challenge of synchronizing the time of the receivers. While GPS disciplined oscillators make this challenge much easier, further prototyping is required to determine the precision that can be obtained with low cost components.

As previously discussed, a full orbit estimate cannot be obtained via range-only measurements from a single ground station because the orbital plane is not observable by range only measurements[3]. However, analyses of TLEs have found that the vast majority of error is in the in-track direction, with the radial and cross-track errors being substantially smaller[1]. This means that it may be possible to use a constrained filter with the range measurements used to directly estimate the in-track position. This could potentially extend the useful lifetime of TLEs beyond a few days, though more exploration of this idea is needed to prove whether or not it is viable.

ACKNOWLEDGMENTS

The authors would like to thank Cyrus Foster of Planet Labs for providing FLOCK 2P state data. The authors would also like to thank graduate student Andrew Dianetti, and industry mentor Brian Bezanson for technical guidance. Second author Chris Gnam would also like to acknowledge support from the NASA Space Grant Consortia at UB.

REFERENCES

1. Riesing, K., “Orbit Determination from Two Line Element Sets of ISS-Deployed CubeSats”, Proceedings of the 29th Annual AIAA/USU Conference on Small Satellites, Logan UT, 2015.
2. Crassidis, J.L and Junkins, J.L, Optimal Estimation of Dynamic Systems, CRC Press, 2012.
3. Tapley, B. D., Schutz, B. E., and Born, G. H. Statistical Orbit Determination. Amsterdam: Elsevier Academic Press. 2004.
4. Foster C., Hallam H., and Mason J., “Orbit Determination and Differential-Drag Control of Planet Labs CubeSat Constellations,” arXiv:1509.03270 [physics.space-ph], Sep. 2015.