Phantom: Physical Layer Cooperation for Location Privacy Protection

Sangho Oh*, Tam Vu*, Marco Gruteser*, Suman Banerjee†

*WINLAB, Rutgers University, Rt 1 Tech Center, North Brunswick, NJ 08902-3390, USA
Email: {sangho, tamvu, gruteser}@winlab.rutgers.edu
† 1210 West Dayton St. Department of Computer Sciences, University of Wisconsin, Madison
Email: suman@cs.wisc.edu

***Abstract*—Localization techniques that allow inferring the location of wireless devices directly from received signals have exposed mobile users to new threats. Adversaries can easily collect required information (such as signal strength) from target users, however, techniques securing location information at the physical layer of the wireless communication systems have not received much attention. In this paper, we propose Phantom , a novel approach to allow mobile devices thwart unauthorized adversary's location tracking by creating forged locations. In particular, Phantom leverages cooperation among multiple mobile devices in close vicinity and utilizes synchronized transmissions among those nodes to obfuscate localization efforts of adversary systems. Through an implementation on software-defined radios (GNU Radios) and extensive simulation with real location traces, we see that Phantom can improve location privacy.**

I. Introduction

Technology trends are leading to an increasing number of wireless transmitters that move around with us as we go about our daily lives. Many of these transmitters are virtually always on—they send messages for push email, handoffs, or sensor status updates even without any explicit user action. At the same time, widely available radio hardware is becoming increasingly flexible and openly programmable. Such hardware significantly lowers the bar for an adversary to intercept and decode wireless signals.

While message content can usually be protected through encryption, any transmitted signal will expose information about the location of the transmitter. Even without decoding any of the transmitted bits, adversaries can use a variety of well-known localization techniques to determine the position and track the movements of a user. Examples of such techniques are Received Signal Strength (RSS) Fingerprinting [1], [2], Time-Of-Arrival (TDOA) [3], or Angle-Of-Arrival (AOA) [4] localization. Thus, emitting wireless signals can be misused to cause significant threats to people, property, or might be a nuisance to individuals in form of unwanted and bothersome activities.[1] In such cases, even a relatively small amount of confusion (tens of meters) about a position can sometimes lead to significant privacy gains—it would hide which store a person entered, or which room a VIP is located in, for example.

However, existing techniques can only provide very limited protection against such attacks on location privacy at the physical layer. Transmit power randomization [6] can throw off standard localization systems, but localization algorithms could easily filter out such changes by applying differential RSS techniques [7]. Although using directional antenna can improve user location privacy by changing RSS information on adversary sensors [8], its physical size and the requirement for antenna steerability pose design problems in portable mobile devices.

In this paper, we design, implement and evaluate Phantom, which provides physical layer location privacy protection by creating a number of fake ghost locations around the true locations of users. The key insight behind Phantom is that a group of actual collaborating nodes mislead a location inferencing system. It achieves this by having the cooperating nodes transmit the same signal simultaneously that make the inferencing system believe that the actual nodes are located in certain ghost locations. While their synchronized transmissions arrive at receivers within normal multipath delay spreads are indistinguishable from regular multipath components. Thus, intuitively, Phantom creates stronger multipath effects that affect the accuracy of localizations techniques. We demonstrate that multi-transmitter cooperative transmission is possible using software-defined radios within the 802.11g radio standard that is currently widely used. Using an indoor test-bed, we show how ghost nodes are created in adversary localization systems and evaluate the location privacy gain that can be achieved depending on the selection of transmission power levels from two cooperative transmitters.

The paper is organized as follows. In Section II, we introduce Phantom and its collaboration protocol. We then explain our experiments in creating ghost locations in Section III. The performance of the proposed system is measured in a indoor test-bed in Section IV. Finally, we draw conclusions in Section V.

II. Phantom : PHY-Location Privacy Protection System

Phantom protects the location privacy of wireless users through their cooperative transmission‘s from, which create confusion in the adversarys localization process. We show

[1] FootPath system reportedly allows tracking the movement of cell phones in shopping malls [5].

in Fig.1 an example scenario where *Alice* is accessing the Internet through a wireless access point (AP) and the adversary *Eve* is trying to determine her location using a RSS-based fingerprinting technique [1]. We focus on RSS-based localization systems in this paper since they are easily implemented and outperform TOA- or AOA-based techniques in multi-path environments [9].[2]

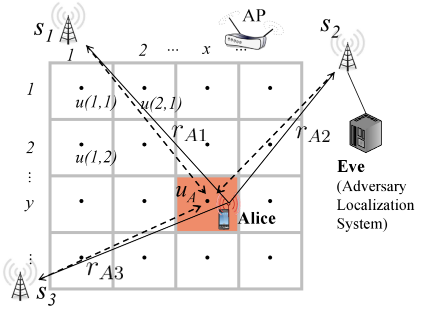


Fig. 1. Adversary localization systems tracking users.

As typical in RSS fingerprinting techniques, we assume that the adversary Eve has obtained a RSS signature database. This database contains the RSS values, $R_u = \{r_1, r_2, r_3\}$, received at Eve's radio signal sensors, $S = \{s_1, s_2, s_3\}$, for target transmissions from each of the reference locations $u(x_i, y_i), i \in C$, where C is reference location set. During actual localization, Eve measures the signal strength from Alice's radio transmission at each of the sensors in S, yielding $R_A = \{r_{A1}, r_{A2}, r_{A3}\}$. She then estimates the location of Alice, (u_A), as the reference location that minimize the mean square error $(arg\ min_i||R_A - R_u||^2, i \in C)$. The accuracy of Eve's location estimation is affected by the granularity of the grid reference points and RSS variations due to shadow and small scale fading, among other factors. Note, that Alice could simply change her transmission power to cause errors in the mean square error estimation, but Eve can easily compensate by searching for a best match over several possible scaled value of R_A. We now describe a countermeasure that cannot be circumvented with this simple scaling technique, since it does not uniformly affect the signal at all sensors.

A. Location Privacy Protection through Ghosts Creation

Phantom protects the location privacy of wireless users by jointly transmitting signals from multiple cooperating nodes, say *Alice and Bob*. The signal received at an adversary sensor is then the convolution of Alice's and Bob's signals, and this convolution creates a different shift in the RSS at each sensor. Thus, the adversary cannot simply rescale all sensor values by a common factor—the best match is likely to somewhat randomly fall on a different reference location. This is further illustrated in Figure 2, where the joint transmission leads the adversary to measure a different signal vector R_G, and the location that minimizes $||R_G - R_u||^2$ is the ghost location u_G. This creates the appearance that a transmitter is located at this location, which we refer to as a ghost location. By modulating their transmit powers Alice and Bob can create different ghost locations and thus cause confusion about the number of real transmitters and their locations. Note that compared to other anonymization techniques using cooperators (e.g., MAC masquerading in 802.11 networks), the performance of Phantom is not limited by the number and the mobility of cooperators.

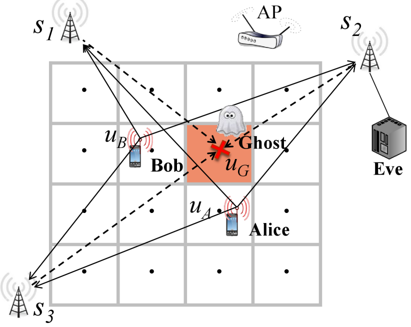


Fig. 2. Adversary localization system tracking ghost instead of real users.

This approach works as long as it is difficult for the adversary to distinguish the transmission from Alice and Bob. This requires that Alice and Bob transmit the same bit-string. It also requires that the transmissions are sufficiently synchronized so that they arrive at each sensor within a delay spread that is indistinguishable from the naturally occurring multipath delay spreads. It furthermore requires that the transmitter hardware is precise enough so that there are no noticeable differences in center frequency or other radiometric features that can be used to distinguish the transmitters. Because the ghosts are created by simultaneously transmitting *indistinguishable signals at variable power levels* from more than two nodes, the cooperating nodes need to agree on a set of transmission parameters, such as frequency, time slots, power levels, and bitstrings. We will further discuss or evaluate these requirements in the following sections.

B. Sketch of Protocol

Realizing a practical system for ghost creation poses several challenges. First, the cooperating nodes must be located at (slightly) different locations but transmit the same bitstrings using the same transmission parameters. This requirement is difficult to meet if the nodes do not have access to an out-of-band communication channel and cannot agree on the content of all future messages before they are separated. We therefore explore a protocol where not all messages are protected by the phantom system. Instead the protocol seeks to intersperse additional phantom messages into the regular communications. The phantom messages are in effect dummy messages whose sole purpose is to create confusion about the true number and locations of transmitters. We assume that the frequency and content of dummy messages can be chosen so that it is difficult for the adversary to distinguish these dummy message from those used to transmit real messages.

[2] Although we do not discuss the details on TOA and AOA-based localization techniques in this paper, Phantom can also create ghosts against those adversary techniques by obscuring their TOA and/or AOA measurements.

For example, this could be achieved by encrypting both types of messages and sending one dummy message for every real message. Figure 3 illustrates this overall approach, wherein a simultaneously transmitted dummy packet is transmitted between the regular messages from source A and B. Note how dummy packet is transmitted synchronized at τ_j^i by both A and B but with identical header and payload information, which requires coordination between the transmitters A and B.

The transmission of dummy packets can be coordinated, for example, by a *Back-end Coordinator (BC)*, which could be set up for the devices of VIPs and their entourage. The BC provides coordination messages that contain information about the user's transmission power and transmission time for dummy packets, which allows the nodes, Alice and Bob in our example, to synchronize their transmissions. These coordination messages must be encrypted and authenticated to protect their confidentiality and authenticity.

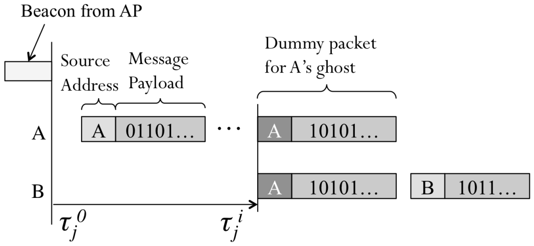


Fig. 3. Synchronized transmission of dummy packets for ghost creation.

Dummy packet creation: To render simultaneously transmitted packets indistinguishable for Alice, every bit of the dummy packets should be identical. This includes all header information such as source and destination addresses as well as the payload. The complete packet information could be provided by the BC to all cooperators. To reduce the amount of information that the BC has to sent, the payload can also by a pseudorandom number generator and then encrypted with the same encryption scheme that is used for regular messages, to mask the randomness of the dummy messages. In this case, the BC only needs to provide the header information, the seed, and the payload length, for the cooperators to agree on the message content.

Overhead: Since dummy packets cannot be used for communications, they are pure protocol overhead to achieve location privacy. The number of dummy packets transmitted to create the ghost location of A should be similar to the number of regular packet transmitted from A. Note, however, that this overhead only applies to uplink transmissions. In applications, where downlink traffic dominates the total network overhead of these dummy packets will be lower.

III. Experimentation Using GNU Radios

In this section, we experiment with multi-node cooperative transmissions using software defined radio systems, and demonstrate their combined transmissions are demodulated like regular packets by 802.11g network cards. Although multi-transmitter signal combining techniques have already been developed as a concept of Single Frequency Networks (SFN) in OFDM networks, we are not aware of prior experiments using software defined radios for actual OFDM packets. We investigate the technical feasibility of Phantom through proof-of-concept experiments using GNU software defined radios (GNU Radio) [10], which are used to have better control over timing and frequency than in commodity radio devices such as Wi-Fi.[3] We chose Orthogonal Frequency Division Multiplexing (OFDM)-based 802.11g Wi-Fi protocols for Phantom implementation to demonstrate an implementation with a real-world, popular protocol. The Cyclic Prefix (CP) [12] of the OFDM symbols also alleviates the level of time synchronization required for dummy packets.

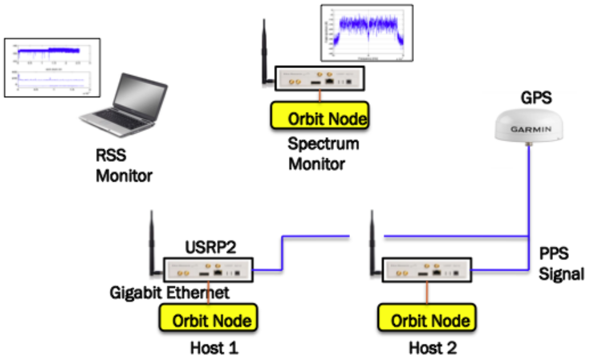


Fig. 4. Synchronization test on GNURadios.

Recall that both time and frequency synchronization are critical to make dummy transmissions indistinguishable from regular ones (and even to just pass a regular CRC packet integrity check). The timing offset (t_{off}) between two radios should be smaller than the CP of 802.11g symbols. Hence, nodes need to synchronize their local clocks to a common reference clocks, which can be achieved by exploiting Pulse Per Second (PPS) signal from GPS (Global Positioning System) receivers or beacons from APs. A frequency offset between two OFDM transmissions can induce severe inter-carrier interference and disturbs packet demodulation. Typically, 10% of sub-carrier internal is allowed for frequency offsets [13]. The precision of typical oscillators used in commodity radios is $20 - 50\,\mathrm{ppm}$, which can produce up to $50 - 100\,\mathrm{kHz}$ frequency offset in $2.5\,\mathrm{Ghz}$ bands. We use these values as a guide for required frequency synchronization.

A. Experiment Setup

Figure 4 shows the layout of the experiments using three GNU Radios. We use two of them for cooperative transmission, which transmit regular 802.11g OFDM packets created by MATLAB codes developed in [14]. We use standard 802.11g packets, rather than creating custom OFDM symbols with extended CP size (which would simplify implementation of multi-node synchronization). This is to demonstrate that our scheme can be effective with Wi-Fi protocols and actually changes the RSS values on off-the-shelf Wi-Fi receivers. Hence, we use laptops with commodity 802.11g network cards

[3] Specifically, we use the Universal Hardware Driver (UHD) for Ettus Research products instead of the standard GNU Radio software to enable sub-microsecond transmission time control [11]

as adversary sensors, specifically cards from two different vendors (Atheros and Broadcom).

B. Time and Frequency Synchronization

Assuming GPS PPS is available[4], we demonstrate synchronizing two independent GNU Radios using a low-cost GPS module (Garmin 18V) [15], [16][5]. Although the two radios synchronize their clocks to the PPS signal every second, differences in clock drifts still create time offsets between them (this difference also grows during the synchronization interval). Figure 5(a) shows the correlation value between the received signal and a 802.11g preamble. Both transmitters transmit a stream of 2000 packets (duration of $230\,\mu$s) over $1\,$s. We can find that the initial time offset of $100\,\mu$s linearly increases up to $4\,\mu$s, as shown in Figure 5(b). Considering the size of CP of 802.11g symbols, which is $0.8\,\mu$s, this amount of clock offset should not be ignored, but if the drift is known the radios can be calibrated. We solved this fine time synchronization problem by adding more baseband samples to the packet data of the radio running a faster clock. With this approach, a maximum $100\,$ns time offset is achieved over the entire $1\,$s intervals, as shown in the bottom graph in the same figure.

The center frequencies of two radios also have offsets due to their oscillators' difference, but can be overcome through similar calibration efforts. Figure 5(c) shows the measured spectrum of two radios signals before calibration, which are then calibrated to within $3\,$KHz (less than 1% of inter-carrier space) in Fig. 5(d).

C. Effect of Synchronized Transmissions

Figure 6 shows the RSS measured from the combined signal while we gradually increased the transmission time offset between two radios by $10\,$ns. The result shows that the combined signal is demodulated at the receiver nodes when they are time synchronized within 2μs. The measured RSS is increased by $2-3\,$dB. Using a packet monitoring application, we also verified that no biterrors occured at the receiver node. Surprisingly, the measured synchronization margin of $2\,\mu$s is much larger than the $0.8\,\mu$s CP size of 802.11g radios. We assume that error correction codes in the 802.11g system help recover from inter-symbol interference errors due to the imperfect time synchronization.

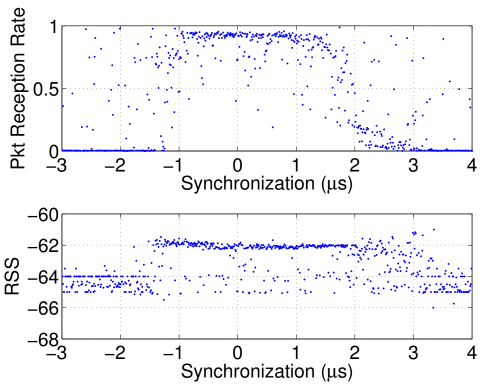
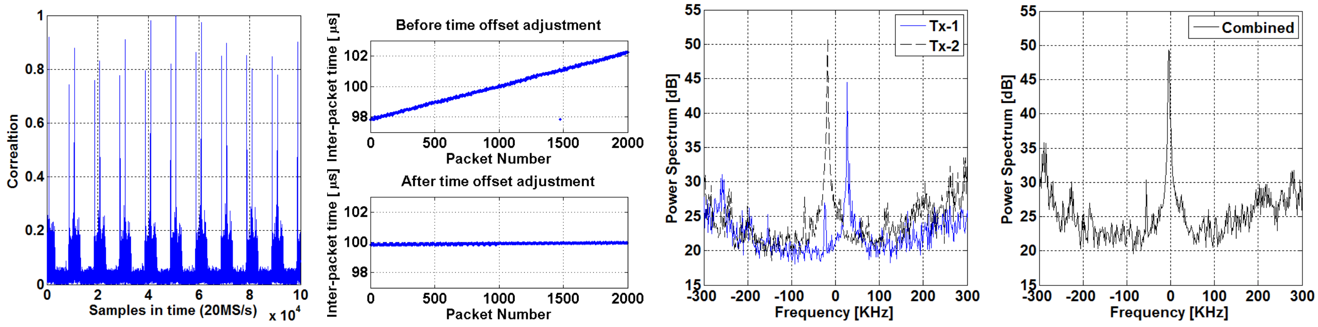


Fig. 6. Demodulation of the synchronized packets.

IV. Performance Evaluation on Indoor Test-bed

We conducted an indoor experiment with a number of Wi-Fi adversary sensors to show how the dummy packets in Phantom can induce ghost locations against adversaries using RSS fingerprinting techniques for localization. We used an isolated indoor test-bed, ORBIT [17], to exclude variables from external sources, e.g., interference, noise sources, signal scatterers. The test-bed, shown in Fig. 7(a), is a grid of 400 (20×20) wireless nodes in a $3600\,$sq.ft. area. Nodes, separated by approximately 1m spacing, are equipped with Atheons 5212 Wi-Fi network cards.

The adversary is assumed to build a RSS signature database at 400 reference points. We measure the performance of Phantom against various numbers and locations of adversary sensor nodes. We initially use 5 adversary sensors (A-Sensors) shown in Fig. 7(b), which is normally a sufficient number to precisely locate transmitters within 1m accuracy on the test-bed. We implement the transmitter portion of Phantom using two GNU Radios, which are fixed at grid coordinates $(3,8)$ and $(8,3)$. The adversary system localizes the target node A by comparing the measured RSS with the radio fingerprints



(a) Correlation with 802.11g preamble.

(b) Fine time synchronization.

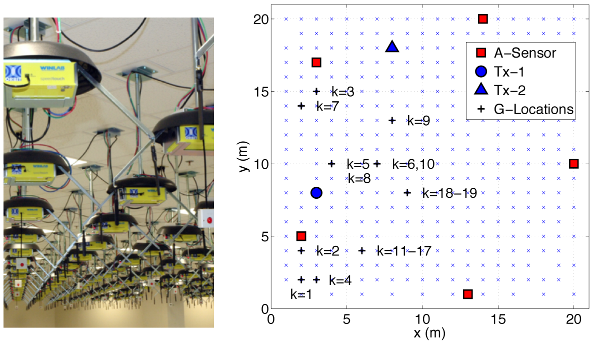
(c) Before frequency synchronization.

(d) After frequency synchronization.

Fig. 5. Time and frequency synchronization test using GNU Radios.

[4] Even indoors, a common PPS can be provided by devices such as GPS repeaters or Pseudolites

[5] We split the signal from a single GPS clock for use in both radios due to restrictions in our test-bed environment. However, we found the maximum time offset among 6 GPS modules was less than $500\,$ns, which is sufficient to synchronize multi-radios within the size of OFDM symbol CP duration.



(a) ORBIT Grid. (b) Ghosts created by Phantom .

Fig. 7. Privacy performance test in ORBIT test-bed.

database, as described earlier. The dummy packets that they simultaneously transmitted from A and B induce the adversary to measure different signal vector R_G, which will lead the adversary to a ghost locations u_G

A. Creation of a Cluster of Ghosts

Since the RSS measured from adversary sensors for dummy packets are mainly affected by the transmission powers of the two transmitters, the locations of ghost nodes can be dynamically changed according to the power configuration of the transmitter node pair. We refer to the transmission power configuration through a power index $k \in \{1, 2, \cdots, 19\}$, which defines a power level of k for the first transmitter and $20 - k$ for the second transmitter. By transmitting dummy packet with various power index values Phantom can create a cluster of ghosts, albeit at increased overhead. Fig. 7(b) shows the cluster of ghosts generated in our example configuration.

In our experiment, the privacy gains, measured by the distance between ghost locations and true locations, are limited by the size of the test-bed. Therefore, considering that signal attenuation is a log function of distance, the privacy gains in real networks can be expected to grow with increased separation of the nodes to several tens of meters for moderate numbers of adversaries.

B. Discussion and Future Work

We leave a number of experimental issues as further research topics. First, multi-technique attacks that utilize combination of different localization techniques to identify dummy packets should be further evaluated. It is not obvious that the adversary's TOA-based algorithms and AOA-based algorithms will point to similar ghost locations as the RSS-based algorithms. If those locations are very different from each other, then the adversary can identify dummy packets simply though the increased variance across localization techniques. Second, their exist a number of possible attacks with more sophisticated measurement equipment that needs more exploration—for example, directional antenna attacks with array antenna systems. Although the precision and accuracy of AOA measurements degrades easily in multi-path environments, attacks that threaten the privacy provided by Phantom may be possible with a large enough number of sensors and sufficient angular resolution.

V. Conclusion

The protection of user location privacy in the PHY layer is a fundamental problem for secure communications. We proposed Phantom to protect wireless users from adversaries who try to localize users without their permission. Phantom enables users to dynamically create confusion about their location by creating additional ghost transmission from different locations with the same identity. We introduced protocols for generating such ghost nodes through simultaneous transmissions from multiple nodes. We also implemented a proof of concept using software defined radios as transmitters and explored issues related to frequency and time synchronization of such transmitters. Through indoor test-bed experiments, we demonstrated the feasibility of inducing localization errors through cooperative transmissions.

Acknowledgment

This study was supported in part by the US National Science Foundation grant CNS-0845896.

References

[1] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building rf-based user location and tracking system," in *Proc. of IEEE Computer and Communications Societies (INFOCOM)*, (Tel Aviv, Israel), 2000.

[2] P. Castro, P. Chiu, T. Kremenek, and R. R. Muntz, "A probabilistic room location service for wireless networked environments," in *Procc of Ubiquitous Computing (UbiCom)*, pp. 18–34, 2001.

[3] P. Chen, "A cellular based mobile location tracking system," in *Proc.of Vehicular Technology Conference*, vol. 3, pp. 1979–1983, jul 1999.

[4] S. Sakagami and et al., "Vehicle position estimates by multibeam antennas in multipath environments," *IEEE Transactions on Vehicular Technology*, vol. 41, pp. 63 –68, Feb. 1992.

[5] FootPath system: http://www.pathintelligence.com/en/products/footpath/footpath-technolog%y.

[6] T. Jiang, H. J. Wang, and Y.-C. Hu, "Preserving location privacy in wireless LANs," in *Proc. of the conference on Mobile systems, applications and services*, pp. 246–257, 2007.

[7] J. H. Lee and R. M. Buehrer, "Location estimation using differential rss with spatially correlated shadowing," in *Proc. of IEEE conference on Global telecommunications (GLOBECOM)*, pp. 4613–4618, 2009.

[8] K. Bauer and et al., "The directional attack on wireless localization: how to spoof your location with a tin can," in *Proc. of IEEE conference on Global telecommunications*, pp. 4125–4130, 2009.

[9] A. Hatami and et al., "On RSS and TOA based indoor geolocation," in *IEEE Wireless Communications and Networking Conference*, vol. 4, pp. 2267 –2272, april 2006.

[10] http://gnuradio.org/redmine/wiki/gnuradio.

[11] http://ettus-apps.sourcerepo.com/redmine/ettus/projects/uhd/wiki.

[12] R. V. Ne, *OFDM for Wireless Multimedia Communications*. Artech House Universal Personal Communications, 1999.

[13] G. Malmgren, "Impact of carrier frequency offset, doppler spread and time synchronization errors in OFDM based single frequency networks," in *Global Telecommunications Conference*, vol. 1, pp. 729–733, Nov. 1996.

[14] https://www.cgran.org/wiki/ftw80211ofdmtx.

[15] H. Toyoizumi and M. Genda, "Precise 1PPS signal by GPS," *IEEJ Transaction on Electronics Information and Systems*, vol. 125, no. 8, pp. 1217–1222, 2005.

[16] P. Vyskocil and J. Sebesta, "Relative timing characteristics of GPS timing modules for time synchronization application," in *Proc. of Satellite and Space Communications*, pp. 230 –234, Sep. 2009.

[17] D. Raychaudhuri and et al., "Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2005.