

# Feasibility Study of Mobile Phone WiFi Detection in Aerial Search and Rescue Operations

*Wei Wang, Raj Joshi, Aditya Kulkarni, Wai Kay Leong and Ben Leong*  
National University of Singapore

## Abstract

In this paper, we show that it is feasible for a mobile phone to be used as an SOS beacon in an aerial search and rescue operation. We show with various experiments that we can reliably detect WiFi-enabled mobile phones from the air at distances up to 200 m. By using a custom mobile application that triggers WiFi scanning with the display off, we can simultaneously extend battery life and increase WiFi scanning frequency, compared to keeping the phone in the default scanning mode. Even if an application is not installed or used, our measurement study suggests that it may be possible to detect mobile devices from their background WiFi emissions alone.

## 1 Introduction

It is not uncommon for hikers to become lost while trekking in the wilderness. With the proliferation of Unmanned Aerial Vehicle (UAV) technology, it has become feasible to employ UAVs in aerial search and rescue operations as they are not affected by difficult terrain.

Some researchers have proposed that image processing be used in UAV-based search and rescue operations [1, 9]. However, image-processing-based approaches have many drawbacks. Specifically, they work only when lighting conditions are favorable and there is line-of-sight between the UAV and the lost persons. Also, the probability of successful detection depends on the quality of the images. Thus, the UAVs would need

to carry larger and heavier optical and infrared cameras, thereby reducing the effective flight time.

We believe that even though lost hikers typically have no access to communications, it is not unlikely that many would be carrying 802.11-compliant (or WiFi-enabled) mobile phones with them. In this paper, we explore the feasibility of locating such lost persons from the air by detecting the WiFi signals from their mobile devices.

We argue that our proposed approach is feasible by showing that (i) we can reliably detect WiFi-enabled mobile phones from relatively long ranges from the air; and (ii) we can significantly extend battery life using a custom application that transmits WiFi frames with the phone display turned off. In fact, we show that battery life can be increased by up to 4.2 times with our custom application compared to the default scanning mode.

Compared to previous approaches [1, 9], our WiFi-based detection technique has several advantages: (i) the detection range is longer ( $> 200$  m) and our approach does not require line-of-sight; (ii) there are no false positives in the detection; and (iii) the payload of the UAV is lighter as we only need a WiFi-enabled embedded board, and this improves the flight time.

Our approach works best if we know in advance the MAC address of the lost person's mobile device. This is easily achieved if hikers install our custom "SOS Beacon" application on their mobile devices before they go hiking. But even if the application is not installed or used, our measurement study suggests that we might still be able to detect mobile devices from their background WiFi emissions alone.

## 2 Related Work

There are some previous proposals that use image processing to detect lost persons from the air. Doherty and Rudol proposed a technique that employs both infrared and conventional imaging to detect the human body [5, 9]. However, their technique works only if most

of the body is exposed and lies flat on the ground. An-drilika et al. proposed a way to overcome this limitation by detecting human body parts [1]. Success rates are reported to be up to 66% for UAV flight altitudes between 1.5 m to 2.5 m above ground. In general, image-processing-based approaches perform poorly when lighting conditions are unfavorable, and they also require line-of-sight.

Zorn et al. showed that the GSM signals from a mobile phone can be used to locate lost persons in a search and rescue mission [14]. Similarly, Zimmermann et al. proposed that both the time difference of arrival (TDOA) and the angle of arrival (AOA) of a GSM signal be used for the same purpose [13]. While these techniques are supposedly effective, they require the availability of cellular network coverage, which is highly unlikely in remote wilderness.

Liu et al. had earlier proposed that UAVs can attempt to locate access points on the ground during warflying by measuring RSSI [6]. However, they made certain assumptions and did not investigate the link characteristics and WiFi scanning with actual hardware. While there is previous work on characterizing aerial-terrain WiFi links [11, 12], that characterization was done between a powerful ground station access point with multiple antennas and a WiFi-enabled UAV. Before our measurement study described in Section 3.3, it was not clear whether WiFi signals from mobile devices can be detected reliably from the air and what is the effective detection range.

### 3 Measurement Study

In this section, we describe experiments to demonstrate the feasibility of our technique. Specifically, we investigate (i) the WiFi scanning characteristics of common mobile devices, (ii) their power consumption, and (iii) the effective range of WiFi signal detection from the air.

#### 3.1 WiFi Scanning Behavior

A mobile device with its WiFi adapter enabled will periodically scan for WiFi access points (WiFi scan) by switching to the various 802.11 channels and sending a small number of Probe Request frames. We can passively detect the device by listening to the Probe Request frames. In addition, we can also actively probe for the device by sending an RTS frame, to which it will respond with a CTS frame. This latter approach increases the likelihood of detection but requires the MAC address of the device to be known a priori. Both these approaches only work when the device switches to the monitored 802.11 channel.

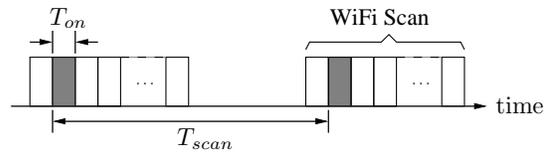


Figure 1: Definition of  $T_{on}$  and  $T_{scan}$ . Each box represents the time spent in a particular 802.11 channel. Shaded box is the monitored channel.

We investigate the WiFi scanning behavior with a representative set of mobile devices. Using a 500 MHz x86 ALIX system board [8], we actively probe the devices with RTS frames (of zero duration field) at a rate of 500 frames per second on a fixed 802.11 channel. A mobile device will respond with a CTS frame if it happens to be tuned to the chosen channel. By setting up several ALIX boards to monitor different 802.11 channels, we observed that the mobile devices we tested switched through all possible 802.11 channels in sequence during each WiFi scan.

We estimate the duration for which the device stays on the monitored channel ( $T_{on}$ ) by measuring the average time span of each batch of CTS frames received. Similarly, we estimate the period  $T_{scan}$  by measuring the time interval between the first CTS frames of consecutive batches (see Figure 1). Our experiments show that the value of  $T_{scan}$  depends on the state of the device, of which we investigated three common scenarios: (i) display is on with the WiFi Settings screen in the foreground; (ii) display is on with some other application in the foreground; and (iii) display is off regardless of what application was in the foreground. The results are summarized in Table 1.

We make several observations: (i) different mobile devices have different  $T_{on}$  and  $T_{scan}$  intervals, even if they might be running the same Android OS version (c.f. HTC Rhyme, HTC Desire and Motorola Electrify). (ii) In some cases, the  $T_{scan}$  intervals are not fixed, but follow a certain pattern. (iii) In all cases, the  $T_{scan}$  interval is minimum when the user is at the WiFi Settings screen, i.e. the phone scans for WiFi access points most aggressively. On the other hand,  $T_{scan}$  increases exponentially or the phone will stop scanning completely (indicated with ‘-’) when the display is off. In order to detect a mobile device, it is necessary that the UAV flies below a certain speed which allows it to remain within the detection range of a mobile device for at least  $T_{scan}$ , and that the interval between two consecutive RTS probes from the UAV be smaller than  $T_{on}$ .

We also investigated the average number of Probe Request frames sent by the device during the interval  $T_{on}$ , when it scans a given channel, and found that fewer Probe Request frames are sent when we are actively probing with RTS frames. As shown in Table 1, the number of Probe Request frames sent when we are ac-

Table 1: WiFi scan intervals for different mobile devices under different user scenarios.

| Device Model              | OS            | $N_a$ | $N_p$ | $T_{on}$ (ms) | $T_{scan}$ (s)                                 |                                    |   |
|---------------------------|---------------|-------|-------|---------------|--|------------------------------------|---|
|                           |               |       |       |               | WiFi Settings Screen                           | Other Screens                      | Display Off                             |
| HTC Rhyme                 | Android 2.3.5 | 3.0   | 5.4   | 256           | 7  | 16                                 | –                                       |
| HTC Desire                | Android 2.3.5 | 2.3   | 5.0   | 66            | 6  | 15.5                               | –                                       |
| Samsung Galaxy Y          | Android 2.3.6 | 1.1   | 5.0   | 127           | 5.9  | 16,32,128,128,128,...              | –                                       |
| Samsung Galaxy SII        | Android 2.3.6 | 2.6   | 4.7   | 115           | 11.9   | 17,33,65,129,129,129,...           | –                                       |
| Acer Liquid               | Android 2.1   | 2.9   | 4.9   | 83            | 5.5  | 5.5                                | 5.5                                     |
| Motorola Electrify        | Android 2.3.5 | 2.4   | 5.1   | 69            | 9.9  | 45.5                               | 30,30,15,45,45,15,30,30,15,45,45,15,... |
| Apple iPad 3rd Generation | iOS 5.5       | 3.0   | 5.4   | 72            | 4,6,11,11,4,6,11,11,...                        | 18,18,33,33,63,106,243,483,483,... | 44.9                                    |
| Apple iPhone5             | iOS 6.0       | 1.8   | 5.4   | 57            | 8.7,10.7,10.7,10.7,10.7,8.7,10.7,10.7,10.7,... | 31,31,62,62,31,31,62,62,...        | 3,7,12,17,62,62,62,...                  |

tively probing with RTS frames ( $N_a$ ) is smaller than that when we are only listening passively ( $N_p$ ) for all devices tested. We suspect that this is because according to the 802.11 standard, the device will always respond with a CTS frame at the MAC layer when an RTS frame is received. Hence, the RTS/CTS exchange from the active probing would likely interrupt or delay the periodic Probe Request transmissions. In spite of this, the probability of detection with active probing is still much higher than that with passive listening, because in the former, the device responds to every RTS frame it successfully receives.

### 3.2 Battery Lifetime Analysis

It is well known that WiFi can be a significant drain on battery life. For a search and rescue scenario, it is even more critical that we try to prolong the battery life of the mobile device as much as possible. In this light, it is important to understand the battery consumption when a mobile device is kept in the WiFi scanning mode. Also, we see from Table 1 that the default WiFi scanning frequency for most devices is relatively low, even in the WiFi Settings screen scenario. It is likely that if a phone is to be used as an SOS beacon, we would want a higher scanning frequency and longer battery life.

To control the scanning frequency of the mobile device, we developed a custom Android application that uses the *WiFiManager* [2] API of Android to trigger periodic WiFi scans at a user defined frequency, even when the display is off. We hooked each mobile device to the *Monsoon Power Monitor* [7] as shown in Figure 2, to measure the current drawn. The Monsoon Power Monitor directly supplies current to the mobile device and is thus able to precisely measure the current supplied by sampling every 200  $\mu$ s. 600,000 samples were taken



Figure 2: Setup of the Monsoon Power Monitor to monitor current consumption of the device.

over a two-minute duration for three different scenarios: (i) Everything Off, where the display is off and WiFi is disabled; (ii) WiFi Settings Screen, which has the highest default scan rate and where the display is on with WiFi enabled; and (iii) Custom Application, where the display is off and WiFi scan is triggered by the custom application. The expected battery lifetime ( $T_{alive}$ ) was estimated with the following relation:

$$T_{alive} (h) = \frac{\text{Battery Capacity (mAh)}}{\text{Average Current (mA)}}$$

During our measurements, the mobile devices were not connected to any WiFi access point and all other background applications were closed. Airplane mode was also turned on to minimize extraneous power consumption. For our custom scanning application, we also tried setting the phones to different scanning rates. Our results for three different mobile phone models are shown in Table 2.

We found that different phone models have different maximum WiFi scanning frequencies. We suspect that this might be due to the difference in the underly-

Table 2: Estimated battery lifetime  $T_{alive}$  under different scenarios.

| Device Model       | Battery Capacity (mAh) | Everything Off |                 | WiFi Settings Screen |              |                 | Custom Application |              |                 |             |
|--------------------|------------------------|----------------|-----------------|----------------------|--------------|-----------------|--------------------|--------------|-----------------|-------------|
|                    |                        | Avg Cur (mA)   | $T_{alive}$ (h) | Freq (Hz)            | Avg Cur (mA) | $T_{alive}$ (h) | Freq (Hz)          | Avg Cur (mA) | $T_{alive}$ (h) | $T_{ratio}$ |
| HTC Desire         | 1400                   | 2.4            | 583.3           | 0.17 <sup>†</sup>    | 170.1        | 8.2             | 0.17 <sup>†</sup>  | 32.6         | 42.9            | 5.23        |
|                    |                        |                |                 | 1.36 <sup>‡</sup>    |              |                 | 115.5              | 12.1         | 1.47            |             |
| Motorola Electrify | 1700                   | 3.4            | 500.0           | 0.10 <sup>†</sup>    | 145.6        | 11.7            | 0.10 <sup>†</sup>  | 33.0         | 51.5            | 4.40        |
|                    |                        |                |                 | 1.12 <sup>‡</sup>    |              |                 | 80.1               | 21.2         | 1.81            |             |
| Samsung Galaxy SII | 1650                   | 1.9            | 868.4           | 0.09 <sup>†</sup>    | 139.3        | 11.8            | 0.09 <sup>†</sup>  | 92.4         | 17.9            | 1.52        |
|                    |                        |                |                 | 0.57 <sup>‡</sup>    |              |                 | 134.5              | 12.3         | 1.04            |             |

<sup>†</sup>Default scanning frequency

<sup>‡</sup>Maximum scanning frequency

ing firmware or driver. The maximum scan frequency achievable was 1.36 Hz, 1.12 Hz and 0.57 Hz for the three phone models tested.

The last column in Table 2 ( $T_{ratio}$ ) shows the ratio of  $T_{alive}$  when using our custom application to that in the WiFi Settings screen scenario. At the default scanning frequency of each phone, our custom application significantly improves  $T_{alive}$  for all three phones. This is because our custom application can trigger WiFi scans with the display turned off. At the maximum scanning frequency, we still obtain considerable improvements to  $T_{alive}$  for the HTC Desire and Motorola Electrify, but not with the Samsung Galaxy SII. However, we can still considerably increase  $T_{alive}$  for this phone by setting a lower scanning frequency. We tested the Galaxy SII with a scanning frequency of 0.22 Hz (2.4 times of the default 0.09 Hz) and found that the  $T_{alive}$  increased by 21%.

Note that the maximum scanning frequency is an important parameter that will eventually determine how fast a UAV can fly while scanning an area. In summary, our application allows us to scan at a higher frequency than the default scanning frequency, which would increase the probability of detection at higher UAV speeds. It also extends battery life compared to keeping the device on the WiFi Settings screen.

### 3.3 Can We Reliably Detect Mobile Devices from the Air?

Finally, we attempt to answer the following question: *can we reliably detect WiFi signals from a mobile device from the air?* To do so, we conducted a series of measurements with a mobile phone and a WiFi-equipped quadrotor UAV.

In our experiments, we adopted the active approach, i.e. the quadrotor keeps sending RTS frames to the mobile device at the rate of 100 frames per second and records any received CTS frames. The reason for this is that if we use a passive approach, there are only a limited number of opportunities to detect the mobile device since

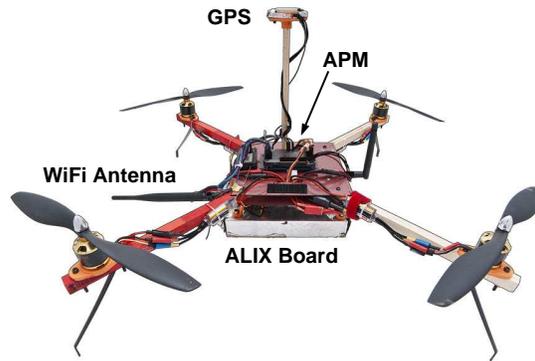


Figure 3: WiFi-enabled quadrotor.

it only send a small number of Probe Request frames while scanning a channel (see Table 1). On the other hand, if we use the active approach, we will be able to detect the mobile device for the entire duration that it is on the monitored 802.11 channel.

**Hardware.** Our self-assembled quadrotor is shown in Figure 3. The central plate is made of carbon fiber, and the arms are made of wood. The GPS receiver [10] is mounted about 15 cm above the central plate so as to minimize the electromagnetic interference from on-board electronics. We use the ArduPilot Mega 2.5 (APM) [4] as the flight controller.

The quadrotor carries an ALIX system board which communicates with the APM over a serial interface. The ALIX board is equipped with an Atheros AR5414 802.11a/b/g wireless card [3] (with 23 dBm transmission power) which is connected to a Motorola ML-2452-APA2-01 dipole antenna. The antenna has a horizontal beamwidth of 360° and vertical beamwidth of 35°. We use the 802.11g mode in our experiments. The weight of the whole system is about 1.4 kg.

**Experimental Setup.** In addition to the quadrotor and mobile phone, we also have an ALIX board on the ground adjacent to the phone that acts as an access point

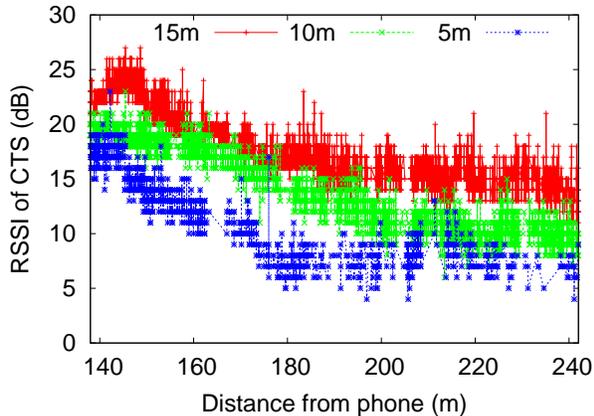


Figure 4: Plot of RSSI of the CTS frames from Samsung Galaxy SII to quadrotor.

for the mobile phone to associate with. This ensures that the phone stays on a fixed WiFi channel and can be continuously probed by the quadrotor, as opposed to it responding intermittently if it switches between channels in a WiFi scan. This ALIX board also overhears the RTS/CTS frame exchange between the quadrotor and the phone, so that we can accurately determine the frame delivery ratio.

Our experiments were performed on a sports field which has a maximum span of 250 m. This limits the range of our experiments because we cannot fly the quadrotor beyond the field due to safety reasons. The phone was placed on a foam support about 20 cm above the ground at one corner of the field while the quadrotor was flown from the middle of the field to the opposite corner, at a constant speed of 1 m/s.

**Effective Range.** In Figure 4, we plot the RSSI of the received CTS frames for the Samsung Galaxy SII mobile phone at various ground distances, for three different flight altitudes — 5 m, 10 m and 15 m. We also plot the corresponding CTS frame delivery ratio against RSSI in Figure 5. What our results show is that when the RSSI is above a threshold, 15 dB in this experiment, the probability of the quadrotor successfully receiving the CTS response from the mobile phone is very high. Our plots suggest that the effective detection range (for free space) is greater than 200 m at an altitude of 15 m.

**Effect of Flight Altitude.** To investigate the effect of quadrotor flight altitude, we fixed the positions of the mobile phone relative to the quadrotor at three different distances (90 m, 150 m and 210 m) and measured the RSSI value of the received CTS frames as the quadrotor ascends up to 30 m in height. We did not go higher for safety reasons. We plot the results in Figure 6. While the RSSI generally seems to improve with altitude, the maximum signal strength is achieved at about 15 m and

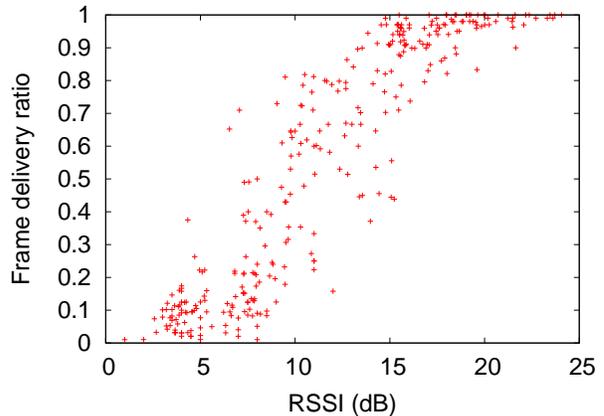


Figure 5: Plot of frame delivery ratio against RSSI.

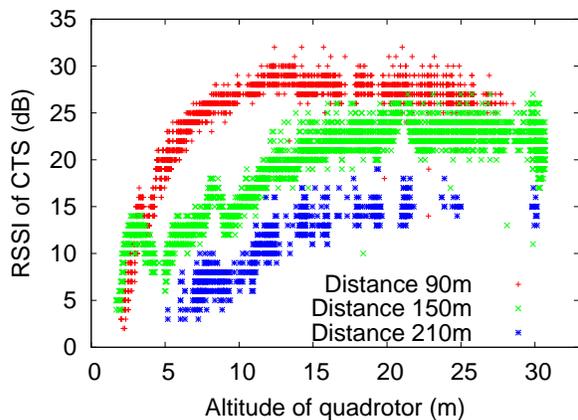


Figure 6: Effect of flight altitude on RSSI.

flying above this height does not improve signal reception. Also, as expected, the maximum achievable RSSI decreases with increased distance.

**Effect of Phone Orientation.** It turns out that the orientation of the phone also affects the received signal at the quadrotor. In Figure 7, we plot the RSSI of the received CTS frames for both the vertical and flat orientations for the HTC Desire phone. Reception is clearly better when the phone is in the vertical orientation. This suggests that if a lost hiker intends to use a mobile phone as a WiFi beacon, instead of leaving the phone flat on the ground, it would be helpful to set the phone in a vertically-upright orientation.

## 4 Conclusion

In this paper, we have demonstrated with a comprehensive measurement study that it is patently feasible for a mobile phone to be used as an SOS beacon in an aerial search and rescue scenario. Based on the conservative estimate of a 200 m detection range, a single UAV flying

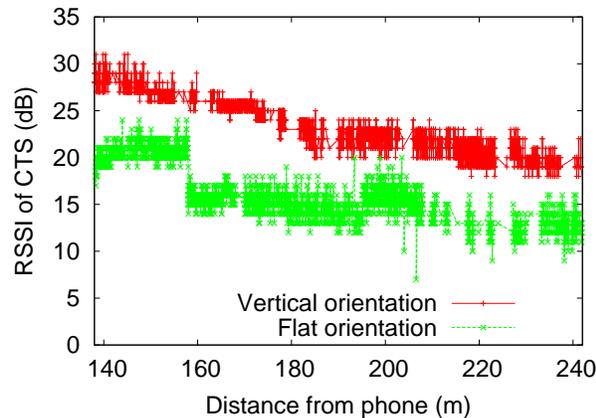


Figure 7: RSSI of the CTS frames from HTC Desire to quadrotor.

at a cruise speed of 5 m/s would be able to scan the 3,000-hectare nature reserve of Singapore in about 4 hours.

What remains to be done is to assess the effect of foliage on the attenuation of the received signal because such attenuation will reduce the effective range of the sensing by the UAV and affect the resulting flight paths. We believe however that our approach will still likely be more effective than previously proposed image-processing-based techniques [1, 9].

Furthermore, since most mobile phones are also equipped with GPS, it should be relatively straightforward to program the SOS beacon application to send the GPS location of the mobile phone directly to the quadrotor together with the CTS response. This means that when a lost hiker is found, we will also be able to know his exact GPS location.

## Acknowledgments

This work was supported by the Singapore Ministry of Education grant T1 251RES1204.

## References

- [1] ANDRILUKA, M., SCHNITZSPAN, P., MEYER, J., KOHLBRECHER, S., PETERSEN, K., VON STRYK, O., ROTH, S., AND SCHIELE, B. Vision based victim detection from unmanned aerial vehicles. In *Proceedings of IROS '10* (Oct. 2010).
- [2] ANDROID. WiFiManager API. <http://developer.android.com/reference/android/net/wifi/WifiManager.html>.
- [3] COMPEX SYSTEMS. miniPCI Wireless Adapters (WLM54AG-23). <http://www.compex.com.sg>.
- [4] DIY DRONES. ArduPilot Mega 2.5. <http://code.google.com/p/ardupilot-mega/wiki/APM25board>.
- [5] DOHERTY, P., AND RUDOL, P. A UAV search and rescue scenario with human body detection and geolocalization. In *Proceedings of AUS-AI '07* (Dec. 2007).
- [6] LIU, Z., CHEN, Y., LIU, B., CAO, C., AND FU, X. Hawk: An unmanned mini helicopter-based aerial wireless kit for localization. In *Proceedings of INFOCOM '12* (Apr. 2012).
- [7] MONSOON SOLUTION. Monsoon power monitor. <http://www.msoon.com>.
- [8] PC ENGINES. ALIX3d2 system board. <http://www.pcengines.ch/alix3d2.htm>.
- [9] RUDOL, P., AND DOHERTY, P. Human body detection and geolocalization for UAV search and rescue missions using color and thermal imagery. In *Proceedings of Aerospace Conference* (Mar. 2008).
- [10] U-BLOX. LEA-6 GPS Receiver. <http://www.u-blox.com>.
- [11] YANMAZ, E., KUSCHNIG, R., AND BETTSTETTER, C. Channel measurements over 802.11a-based UAV-to-ground links. In *Proceedings of Wi-UAV '11* (Dec. 2011).
- [12] YANMAZ, E., KUSCHNIG, R., AND BETTSTETTER, C. Achieving air-ground communications in 802.11 networks with three-dimensional aerial mobility. In *Proceedings of INFOCOM '13* (Apr. 2013).
- [13] ZIMMERMANN, L., GOETZ, A., FISCHER, G., AND WEIGEL, R. GSM mobile phone localization using time difference of arrival and angle of arrival estimation. In *Proceedings of SSD '12* (Mar. 2012).
- [14] ZORN, S., ROSE, R., GOETZ, A., AND WEIGEL, R. A novel technique for mobile phone localization for search and rescue applications. In *Proceedings of IPIN '10* (Sep. 2010).