GPS/WiFi real-time positioning device: An initial outcome.

Author/Contributor:

WAYN, C.J; LI, B; DEMPSTER, A.G; RIZOS, C

Publication details:

Lecture Notes in Geoinformation & Cartography

pp. 439-456

978-3-540-87392-1 (ISBN)

1863-2246 (ISSN)

Event details:

5th International Conference on Location Based Services and TeleCartography
Salzburg

Publication Date:

2008

Publisher DOI:

http://dx.doi.org/10.1007/978-3-540-87393-8_26

License:

<https://creativecommons.org/licenses/by-nc-nd/3.0/au/>

Link to license to see what you are allowed to do with this resource.

Downloaded from <http://hdl.handle.net/1959.4/44305> in <https://unsworks.unsw.edu.au> on 2023-04-26

GPS/WiFi Real-Time Positioning Device: An Initial Outcome

Cheong Joon Wayn

School of Surveying and Spatial information Systems, UNSW, Australia

Binghao Li

School of Surveying and Spatial information Systems, UNSW, Australia

Andrew G. Dempster

School of Surveying and Spatial information Systems, UNSW, Australia

Chris Rizos

School of Surveying and Spatial information Systems, UNSW, Australia

Abstract: The Global Positioning System (GPS) has a typical outdoor positioning accuracy of up to 15m for civilian users. Thus, it has become a viable method for civilian to carry out coarse positioning. However, it has its shortcomings; it is available only in indoors with a clear view of the sky. Since WiFi has become another proven positioning technology that is capable of performing positioning in indoor environments and urban canyons, it is desirable to combine both of these technologies for ubiquitous positioning. Therefore, by means of integrating GPS positioning with a WiFi positioning system, indoor and outdoor positioning may be performed using only one device. The device can be implemented using FPGA embedded systems technology that allows easy reconfiguration of the device. Such a combination allows GPS and WiFi positioning technology to transition seamlessly.

Keywords: GPS/WiFi integration, GPS/WiFi positioning, localisation, orientation, Namuru.

1. Introduction

Since the rapid growth of the user community of IEEE 802.11a/b/g wireless networks (WiFi), WiFi has become another viable positioning method (P. Bahl and V. N. Padmanabhan, 2000) (Li, et al., 2006). Depending on the method of positioning used, the WiFi positioning system may not require the MU to have a Line of Sight (LOS) path to the WiFi Access Points (AP) and is least affected by multipath. Implicitly, WiFi is capable of performing positioning in indoor environments and urban canyons. Therefore, by means of integrating GPS positioning with WiFi positioning system, indoor and outdoor real-time positioning may be performed using only one device. Such a combination allows GPS and WiFi positioning technology to transition seamlessly, such that WiFi pos. may compensate for areas where GPS unavailable, and vice-versa.

The application of such an independent device may be in the field of navigation for the Blind and Vision Impaired (BVI). Such an application allows the BVI to navigate not only in outdoor but also indoor environments. Another application of such a device will be in the field of location based services (LBS). The rapid growth of technology has allowed many useful technologies to be integrated together into mobile personal handheld devices such as handheld gaming devices, Personal Digital Assistant (PDA), mobile phones, and laptops. The integration of GPS and WiFi into these devices may allow mobile users to easily find the location of shops or businesses in both indoor and outdoor environments.

Preliminary investigation found little theory or implementation of GPS-WiFi integration. The only technique for such integration can only be found in (Singh, et al., 2004). However, the WiFi positioning technique uses trilateration rather than fingerprinting. Hence, this research attempts to use the fingerprinting method for WiFi positioning and changed the structure of the GPS-WiFi integration mechanism as proposed in (Singh, et al., 2004).

Taking advantage of the reconfigurable nature of the newly developed GPS receiver, the

Namuru II (Mumford, et al., 2006) will be adopted for use to provide GPS coordinates, and the soft-core NIOS II processor within the Namuru II will be used to process a fully tested WiFi positioning algorithm. Effort is also put into redeveloping the WiFi positioning algorithm by using MATLAB programming that resembles the C/C++ programming style. The successful algorithm is then appended into the Namuru II. The reason to use MATLAB is to ease the debugging process and verification of the developed system. The reason to implement the integration of the two positioning technologies rather than simply simulate it is to be able to demonstrate the practicality of such integration and to ensure real-time positioning is realizable.

2. Developing and Testing the WiFi Positioning Algorithm

The development and testing done in this section ensures that the implementation of the WiFi positioning algorithm in hardware will be smooth and easy to realize in the next section. In addition, the testing of the algorithm with various parameters allows the best algorithm implementation, with minimum computational requirements and least error to be chosen.

The empirical method of “Location Fingerprinting” can be described by two stages, namely the online stage and the offline stage. In the offline stage (training phase), a fingerprint database containing Radio Signal Strength Indicator (RSSI) measurements and location of Reference Points (RP) must be created. To generate the database, RPs must be carefully selected so that the entire test-bed is covered (Lee, et al., 2005). Furthermore, the number of RPs to be used should be a balance between considerations of accuracy (the more RPs the higher the accuracy, because the granularity is lower) and the labour effort required to survey the patterns (or “fingerprints”) of RSSIs at the RPs for the database (Li, et al., 2007). In the online stage (positioning phase), the Mobile User (MU) measures the RSSIs whenever it requires its position to be determined. An appropriate algorithm is used to compare the measurements with the fingerprints in the database (Li, et al., 2007). The outcome is the likeliest location of the MU.

Understanding the significant effects of MU’s orientation on RSSI (Li, et al., 2007), a directional approach is used to collect fingerprints of RPs. The non-directional method takes the RSSI of 4 cardinal directions and averages the RSSI into a single entry per Access Point (AP) per RP. In comparison, the directional method takes the RSSI of the 4 cardinal directions and stores the RSSI collected from each direction as a separate entry, resulting in 4 entries per AP per RP. The RSSI vector is defined as a set of RSSI arranged with respect to its AP in vector form. Thus, the directional approach will result in 4 RSSI vectors assigned to one RP while the non-directional approach results in only one RSSI vector assigned to one RP (Haeberlen, et al., 2004). The block diagram below illustrates the proposed implementation of the WiFi positioning algorithm. It is worth noting that for the scope of this discussion, the Post-Processing stage is not implemented.

Figure 1 Block diagram of the overall WiFi positioning system

2.1. Missing AP resolution

Wireless technology has emerged to be an essential tool in communications. Almost every building in urban areas is equipped with Wireless Access Points (AP). For example, in many

developed countries (and especially in the U.S.), the Central Business District (CBD) area of cities, university campuses, public hospitals, shopping malls, and other public spaces, WiFi coverage is widespread. These APs are typically deployed for permanent use and located at fixed positions. It would be advantageous for a WiFi positioning system to exploit its surrounding APs for better accuracy. Therefore, all APs that can be detected are incorporated into the system.

Since we attempt to use APs that we have no control over in both the Reference Database and Test data, we thus cannot ensure that all APs will cover the entire test-bed. Hence, the issue of Test Points (TP) collecting RSSI from an AP that is not listed in some RPs, or vice-versa, has to be addressed. This situation is referred to as the “missing AP” case.

The “missing AP” cases can be due to one of the following reasons:

1. Case 1: The TP is far away from the RP and hence cannot acquire any signal from some APs in the list of APs for that RP or vice-versa.
2. Case 2: The TP near a RP detects a newly deployed (turned on) AP near the test-bed that the RP did not detect or vice-versa. (Cheng, et al., 2005)
3. Case 3: The TP near a RP is unable to detect an AP the RP had detected earlier or vice-versa because the AP was turned off. (Cheng, et al., 2005)

To accommodate these situations, the APs that cannot be detected either by the Reference Point or Test Point is replaced with an RSSI of -100dBm (Binghao Li, 2006). This seems to be an intuitive solution for case 1 because the weakest RSSI that we can practically acquire is approximately -90dBm while -100dBm appears to be an interpolated value to indicate that the AP is far away from the TP or RP. This method may also well accommodate for cases 2 and 3, where no other method can be used to predict whether an AP is turned off or newly deployed, the respective APs should not be penalised too much.

However, it would be almost impossible for a TP to be near a RP to have many “missing AP” cases occurring at once. Hence, the case where there are too many “missing AP” should be heavily penalised. There are 2 methods of detecting such cases:

1. Method 1: If “missing AP” cases are more than a pre-determined threshold. (Cheng, et al., 2005)
2. Method 2: If “missing AP” cases are more than a pre-determined percentage of APs being compared.

Since the number of APs being compared between a TP and a particular RP has a huge variation from one RP to another, method 2 should be used. It is experimentally found through testing herein that allowing 75% of APs being compared to have “missing AP” cases is optimal (Li, et al., 2007).

2.2. Inverse Distance Weightage (IDW) Interpolation

In the directional Reference Database, each RP has four RSSI vectors with one RSSI vector representing each direction of the same RP, as opposed to the non-directional database which has only one RSSI vector per RP. Since the four NNs are chosen from a list of “4 x number_of_RP” (4 refers to the four cardinal directions, North, East South, West (NESW)), it is very possible that the case where “2 or 3 of the 4 NNs are from the same RP” may arise. In this case, the possible area that the calculated position could lie in is smaller than the normal case. (The normal case assumes the 4 NNs are from different RPs.) This case is illustrated in Figure 2.

investigate the effects of changing those parameters. There are 91 Reference Points and 32 Test Points used in the analyses. Each Reference Point has 4 sets of RSSI vectors, each corresponding to one of the NESW directions, and similarly in the case of the TPs. Hence there are 4x32 tests to calculate instead of 32, since the directional method is used. The Cumulative Distribution Function (CDF) plot is used to analyse the magnitude of errors caused by inaccurate positioning. The use of both Directional and Non-Directional (conventional) methods are compared.

Figure 4 CDF plot of the magnitude of error for all test points using (a)directional method (left) and (b)non-directional method (right). In the highlighted points for both plots, $F(x)$ of 0.9 corresponds to the 90th percentile error, while $F(x)$ of 0.5 corresponds to the median error.

The detailed horizontal error statistics generated based on the directional method shows 1-sigma error to be 3.25m, median error to be 2.07m and 90th percentile error is found to be approximately 5m. As for the non-directional case, the 1-sigma error is found to be 3.98m with a median error of 2.75m and 90th percentile error of 7m. Hence, the directional method shows significant improvement over the non-directional method, as indicated by the CDF graphs. Compared to the non-directional method, which has a 90th percentile error of about 7m, the directional database gives a significant improvement of 2m.

In the Nearest Neighbour algorithm, the RSSI used for each AP at each reference point in the reference database is the mean of the RSSI signals for each AP over the collection time. The principle of taking the mean RSSI instead of an instantaneous RSSI is justified because the RSSI is corrupted by fast fading, resulting in the RSSI varying quickly as a function of time. However, the corrupted RSSI typically varies about the mean RSSI, resembling a Gaussian distribution (Haeberlen, et al., 2004) (Cheng, et al., 2005). Therefore taking the mean RSSI can mitigate the problem of collecting a biased or corrupt RSSI. Many literatures have discussed the long term variation or distribution of the RSSI signal (Wang, et al., 2003) (Cheng, et al., 2005). However, the optimal duration of sampling of the RSSI signals such that the actual mean RSSI is captured has not yet been discussed. To examine this, a simple experiment consisting of a WiFi receiver/logger (Sony VAIO VGN-U70P) and 3 APs (SparkLAN 802.11b Wireless AP) in a corridor is conducted with all devices remaining in a static position. It is worth noting that the WiFi receiver is set to use passive scanning. AP1 is set to have approximately equal distance from the receiver with AP3, while AP2 is set to have the nearest distance to the receiver amongst all APs. Figure 5 is a plot of the RSSI mean for each of the 3 APs over time.

Figure 5 Plot of variation of “RSSI (-dBm) mean over time (seconds)” (circle markers) and the “mean RSSI (-dBm) of 100 seconds” (red line) for (a)AP1, (b)AP2, and (c)AP3

It can be seen that the mean for AP1 and AP3 converges to the 100-second-mean almost immediately to within $\pm 1\%$. The plot of AP2 shows large variations of mean RSSI over time. The plots of RSSI over time are shown in Figure 6.

Figure 6 Plot of variation of RSSI (-dBm) over time (seconds) for (a)AP1, (b)AP2, and (c)AP3

The deviation of RSSI over time from the mean is approximately ± 3 dBm for both AP1 and AP3 cases. However, AP2 shows peaks (-90dBm) at irregular intervals, while the typical RSSI ranges in -40dBm to -50dBm. It has been found through experiments that cases like those indicated for AP2 are not uncommon in practical situations. This case may be explained by the presence of strong multipath signals or the behaviour of the passive scanning software/hardware, however the exact cause of this issue has not been exactly identified yet. Statistically, this suggests outliers that corrupt the estimation of the mean RSSI.

However, the Nearest Neighbour algorithm is trying to match the difference between the user collected mean RSSI and the RP's mean RSSI. This means that as long as the reference point captures the outlier that the user might capture whilst standing at the reference point, the difference in mean will be in the range of ± 3 dBm. This difference is relatively small compared to the possible range of mean RSSI from -35 dBm to -90 dBm, thus not affecting the final result. The final positioning accuracy will undoubtedly improve if this effect can be mitigated. Also from the RSSI plots of AP1 and AP3, it can be suggested that a RSSI collection time of less than 5 seconds is sufficient to approximate the RSSI mean from one AP.

3. GPS-WiFi Integration

The integration of GPS with WiFi for positioning is a useful yet immature domain. Preliminary literature studies have shown only one example of such integration (Singh, et al., 2004). Although such a system is desirable, it is still at the state of development. As being discussed earlier, the unavailability of GPS in indoors and the availability of WiFi positioning in indoors are able to complement each other, thus providing a practical approach to realise a true indoor and outdoor ubiquitous positioning system.

The integration strategy originates from the method (Singh, et al., 2004) used to overcome multipath in GPS by extracting information from WiFi RSSI readings and the local map. However, this method uses the trilateration-based WiFi positioning, which has been proven to be less accurate and less reliable than the fingerprinting method in heavy multipath environments, such as indoors (Cheng, et al., 2005).

To improve the accuracy of indoor positioning, the fingerprinting method is proposed as a replacement for the trilateration method. Modifications are also made to the flowchart as proposed in (Singh, et al., 2004) because it is highly dependent on the GPS parameters. It is possible to acquire GPS multipath signals in indoor environments, which may lead to a wrong determination of MU's position. Although the GPS signals are acquired indoors, they are most probably multipath signals that are reflected by nearby walls and buildings. Therefore, we suggest using both WiFi and GPS parameters to allow better detection of the user's location.

The implemented algorithm will require the entire Reference Database to be separated into sections/buildings/vicinities (referred to as Blocks in this paper). This method is more feasible if the system is implemented across a large area (e.g. a metropolitan area). Hence, the system will not search the entire database to calculate the user's position, but the vicinity of the user is first identified, then a Block of Reference Database (or two) is selected for subsequent computations. The benefits of using such an implementation are:

1. Huge reduction in computational burden.
2. Allows nearby Blocks to be loaded or unloaded from the system when necessary.
3. Allows different coordinate systems to be used in different situations (e.g. x,y,z for indoors, and LLA for outdoors).

The last benefit listed concerns the coordinate system of GPS which uses LLA. This is not an intuitive coordinate system for indoors and most maps of building have their own origin, and use the xyz Cartesian coordinate system. It would be more practical and less computationally demanding if the output coordinate system can be set to local x,y,level coordinates (NED) when the user is indoors, and Latitude, Longitude, Altitude (LLA) in the WGS84 datum when the user is outdoors. The reason being that outdoor users typically would acquire maps from sources (i.e. Google Earth) that use LLA coordinates, while indoor users typically use a map of a building which has its own origin of x,y,z coordinate. Consider if the output coordinate of the system is to be matched with a number of indoor building maps and a large road map, the output coordinates can easily adapt to these maps without any conversion. If the system only outputs LLA coordinate, a conversion process (LLA to xyz) will have to take place in order to match the user to the indoor maps when the MU is indoors, hence increasing computational burden. Hence,

it is proposed to have a ‘block’ of reference database for each indoor building and each area of pre-surveyed outdoor vicinities.

3.1 Integration Hardware

The implementation stage uses two main hardware components. The WiFi signals are acquired using the G2 Microsystems’s G2C501-HDK Hardware Development Kit, while the GPS signal acquisition, correlation, and receiver processing are executed on the UNSW SNAP lab’s Namuru II GPS receiver board. The WiFi positioning algorithm and the WiFi/GPS integration algorithm are implemented on the Namuru II processor as well.

Firstly, the G2C501-HDK Hardware Development Kit is a development board produced by G2 Microsystems to support 2.4GHz WiFi communication and serial UART communication, along with many other functions. Thus, we exploit the 2.4GHz WiFi communication capability of the device to acquire RSSI readings and send it to the Namuru II through the UART for further processing. This device will be simply referred to as the “G2 receiver”.

The Namuru II is an open-source GPS receiver research platform based on an L1 GPS RF frontend and is implemented on an Altera® Cyclone II™ Field Programmable Gate Array (FPGA) chip. The FPGA is composed of an array of configurable logic cells, and each cell can be programmed to perform one of a variety of simple digital logic functions. This is used to implement the correlators and a soft-core processor known as NIOS II™. In addition, the Namuru II incorporates a dual UART, 3 axis accelerometer, 2 axis gyro, and ample of memory. Table 1 summarises the technologies available and the technologies utilised for this implementation.

	<i>G2 Development Kit</i>	<i>Namuru II Development Kit</i>
<i>Microprocessor</i>	<i>LEON 2 SPARC V8 32-bit CPU, clocked at 44 MHz</i>	<i>Altera Cyclone II, up to 260Mhz</i>
<i>Wireless Com.</i>	<i>External 2.4 GHz and EPC antennas</i>	<i>N/A</i>
<i>Short Range Comm.</i>	<i>3 axis 125 kHz magnetic interface</i>	<i>N/A</i>
<i>Wired Communication</i>	<i>Dual RS-232, SPI</i>	<i>Dual UART, USB 2.0</i>
<i>Positioning</i>	<i>N/A</i>	<i>2x Zarlink GP2015 GPS L1/L2 front ends</i>
<i>Sensors</i>	<i>temperature, security seal, motion, 3 axis shock</i>	<i>3 axis accelerometers, 2 axis gyro</i>
<i>Memory</i>	<i>320kb ROM 80kb RAM (2kb NVM)</i>	<i>32MB ROM, 64MB SDRAM, 256K SRAM</i>

Table 1 Comparison of capabilities between the G2 Development Kit (used as a WiFi receiver) and Namuru II Development Kit (used as a GPS receiver and GPS/WiFi integration module). Shaded boxes show the technologies used.

Altera has provided a PC-based development environment known as Quartus II, for configuring the logic elements using Verilog and VHDL. The NIOS II C/C++ IDE is used to develop, compile and load the compiled binary onto Namuru II. The NIOS II IDE (on the PC) also has a console that maintains connection with the NIOS II CPU (on the Namuru II) after the binary is loaded onto it. This is known as the NIOS II console which is used to display debug information during runtime.

The Namuru II acquires the RSSI readings through the UART and performs the WiFi positioning and GPS/WiFi integration utilising the NIOS II soft-core processor. GPS coordinates are acquired from the GPS Architect software. All software processes are developed

in NIOS II C/C++ Integrated Development Environment (IDE).

Figure 7 Block Diagram of overall integrated system ported into the NIOS II soft-core processor

Figure 7 illustrates the block diagram of the overall system. The GPS Architect block (Figure 6.1) is the Mitel® firmware which is already ported to the Altera® NIOS II soft-core processor for the required processing of the GPS correlators and timing signals. These functions are implemented as real-time interrupts that runs as a higher priority task .

The blocks of WiFi positioning algorithms are similar to those described earlier. However, additional functions were added to accommodate the separated blocks of reference database. (Section 2 implemented only a single reference database for all reference points.)

All of the implemented functions runs in/from the main() task, which is the lowest priority task (i.e. last executed when the processor is freed by other Interrupt Service Routines (ISR)). However, the UART receive function is an exception, which uses the Altera ISR to retrieve data from the Namuru II UART.

3.2 Separated Reference Database

As explained earlier, the entire reference database has to be separated into several smaller Blocks, which is clearly visible from Figure 7. To determine which Block to be used, a search for the best matching Block(s) of Reference Database has to be done. This is called the FindBlock function. This technique uses the Media Access Control (MAC) address of the APs that the user acquires and attempts to match it with the list of MAC addresses of the APs in each Block of Reference Database. A non-matching AP, where the user acquires a AP that is not in a particular Block of Reference Database, is a case of “missing AP”. The number of “missing AP” cases in each Reference Database is recorded, and the Block with the least “missing AP” cases will be chosen. This is implemented in findBlock() as follows:

1. For each Block “k”, attempt to match the user’s AP MACs with the list of MAC addresses of the AP’s in each Block of Reference Database. The counts of “missing AP” cases will be stored in missing_APs[k].
2. Sort the missing_APs[] array in ascending order.
3. The k-th Block with the least “missing AP” cases is chosen.
4. The “wifi_mode” is set to INDOOR or OUTDOOR depending on the environment (outdoor/indoor) of the chosen Block of Reference Database.

FindBlock may produce the same number of “missing AP” for two or more Blocks. This condition may occur when the user is at the border between one Block of Reference Database and another (i.e. between Indoor and Outdoor). Hence, if more than one Block of Reference Database suggested by FindBlock to use, then Database Matching is executed for the two best

possible Blocks found instead of one. The Database Matching algorithm produces the residual of k-Nearest-Neighbours matching by means of its Manhattan distance. Based on this metric, the most possible Block is chosen and its set of Nearest-Neighbours are used to perform interpolation (a.k.a. Location Calculation) using the IDW method as described and tested in section 2.

3.3 GPS/WiFi Integration Mechanism

Figure 8 Flowchart of GPS/WiFi Integration Mechanism

The implemented integration mechanism is shown in the form of a flowchart in Figure 8. The flowchart is implemented in the form of if-else statements. “wifi_mode” is a variable determined in the online phase by a function that calculates the user’s vicinity (“INDOOR” / ”OUTDOOR”). If the user’s vicinity is undetermined due to APs not matching those contained in the Reference Database (i.e. user in an area not mapped by the reference database), “INVALID” will be assigned to “wifi_mode”. Horizontal Dilution of Precision (HDOP) is a function of how good the geometry of the tracked satellite signals is. The lower the HDOP, the more precise the MU’s calculated horizontal position is. More details are explained in (Kaplan, et al., 2006). GPS_FIX is a Boolean variable determined by the Geometric Dilution of Precision (GDOP) and the number of satellite signals acquired and the Code to Noise Ratio. This variable is acquired from GPS Architect. N is the number of satellite signals tracked. $P_{GPS}(t)$ is the GPS calculated position in the Latitude, Longitude, Altitude (LLA) system. $P_{WO}(t)$ is the outdoor WiFi calculated position in the LLA system. $P_{Wi}(t)$ is the indoor WiFi calculated position in the x,y,z system, with reference to the origin of the local building map. $P_F(t)$ is the final calculated

position of the overall system.

The basic concept of mapping the flowchart is to first take the WiFi parameters as a priority to determine the indoor/outdoor status of the user, then using the validity of GPS parameters to determine which technology (GPS or WiFi) to perform positioning for the user.

3.4 Initial Evaluation

Figure 9 Map of test-bed with reference points (stars) and Test Points (circles) embedded (each tick in x and y axis indicates a meter)

The location chosen as the test-bed (Figure 9) is level G of the EE building, UNSW, where a rear entrance allows the test-device to move from indoors to outdoors, and vice-versa. This location is chosen because there are fewer passersby at non-peak hours. There are three steps to set up the system and the test-bed. The first is to ensure sufficient APs are available in the vicinity of the test-bed. Then the fingerprints of each Reference Point are collected. Finally, the test-device is used to collect test data for analysis. In the first step, a WiFi pre-survey in the vicinity of the test-bed is carried out to identify areas where insufficient APs are detected. Then, an adequate number of APs are deployed in/near the test-bed. APs are also deployed outdoors to ensure WiFi positioning also is possible outdoors.

For the second step, Kismet (a device customised for WiFi fingerprint database collection) is used to collect RSSIs for the WiFi Fingerprint database. Again, the directional method is used for data collection. After the collection, the data is transferred to a PC and pre-processed by a MATLAB program into C/C++ initializations before it is loaded into the Namuru II.

Figure 10 (a)Equipments set up for testing (left), (b)A laptop connected to the Namuru II, the UART cable connecting Namuru II to G2 Dev Board, and inverter connected to a battery to power up the boards(right).

In the third step, which is the testing stage, the implemented system as described in section 3.2

is brought up to test. The equipment set-up is shown in Figure 10. The implemented system collects WiFi signals through the integrated hardware and all subsequent data processing is carried out in the Namuru II, which includes WiFi positioning, GPS/WiFi integration, etc. Then, the Namuru II sends the output and debug information to a PC, and displays it on the NIOS II console.

Analysis of the results is done only for 2D. All altitude values are assumed to be 0. 6 Test Points are strategically chosen in the test-bed (Figure 9), in both indoor and outdoor environments. A number of observations (ranging from 10 to 20, over a 2 minute interval) are collected to analyse the statistical positioning accuracy of each point.

Figure 11 CDF plot for all test points (x in meters)

Combining all the error vectors for each test point, the overall accuracy behaviour of the system can be summarised by the CDF plot in Figure 11. 60% of the points can be positioned within 4.5m of accuracy, while a large probability of misdetection (approximately 40%) due to the inaccurate indoor/outdoor detection of the user causes an error of more than 10m.

The test-bed has been intentionally chosen to be in the vicinity of tall buildings, which resemble urban canyons, so that the importance of WiFi positioning can be demonstrated. To analyse the availability of GPS, an investigation of the possible satellites visible during the test period was conducted using SkyplotTM.

Figure 12 Possible Satellites that can be seen. Shaded region shows the part of sky blocked by tall buildings

Figure 12 shows the possible satellites that are visible above the horizon on the night of the experiment. The EE building in the test-bed is aligned North in the downward direction in the map (Figure 9) Thus, only the Northwest direction of the sky is clear, while the other sky sectors are mostly blocked by the building. The shaded region in Figure 12 indicates the possibly blocked area of the sky. Thus, only satellites 27, 7, 25, 28 and 20 are likely to be detected. Figure 12(right) shows the satellites that can be detected at one time (satellites 25 and 20), and also suggests most probably only two satellites can be detected at any one time from

the test-bed.

Relating this back to the experiment, this situation compromises the availability of GPS during testing, and thus the system is forced to use WiFi for positioning even in outdoors since only 2-3 GPS satellites could be tracked. However, when the system moves further away from the test-bed, a WIFI INVALID case is observed, and the system is forced to use GPS for positioning whenever the conditions mentioned in section 3.3 are satisfied (i.e. there is a good view of the sky). A POSITION UNAVAILABLE case is observed when a user is indoors of an unknown building.

3.5 Further Work

The initial evaluation has shown several issues to improve in this implementation. The different hardware used for RSSI collection has identified an issue in the transformation of RSSI signals from one device to another. (Haeberlen., et al., 2004) and (Cheng, et al., 2005) has suggested the mapping of received RSSI from one device to another is a linear transformation and will be implemented in the future.

Also, the WiFi/GPS Integration Algorithm is not sufficiently tested yet as GPS was unavailable on the test-bed. Another test-bed with clear view of the sky should be tested with the system. Also, the positioning accuracy of a moving test device has not been addressed due to the fragile prototype, which is susceptible to physical movements and jitter during operation.

4. Conclusion

In short, the research has undertaken a simulate-develop-test approach to achieve the outlined objectives. The simulate stage includes the development and testing of the k-wNN fingerprinting algorithm for the WiFi Positioning, in which the parameters chosen attest to the expected accuracy of the system. In the development stage, the system is modified to accommodate for practical considerations, and successfully ported into Namuru II to be integrated with the already available GPS system. In the testing stage, the integrated system is put to test in a test-bed resembling challenged indoor environments and urban canyons, such that WiFi is capable of assisting GPS during its unavailability. The implementation has also suggested a solution to search a large-sized WiFi Reference Database efficiently. The separation of the entire Reference Database into several Blocks is an effective method to reduce computational burden and positioning delay.

The final result has achieved its main objective, which is to be able to seamlessly transition between indoor and outdoor environments by using the newly proposed algorithm. However, more testing are still required to prove the transition between the 2 different technologies, namely the GPS and WiFi positioning system, is feasible. A few simple static testing has suggested improvements to be made to the current real-time positioning system implementation.

For a feasible realisation of the system, the entire Reference Database (e.g. the Reference Database of a metropolitan city) may not be ported into the system due to memory limitations. Hence, Blocks of Reference Database in the user's surrounding can be acquired through an internet server and 'cached'. Opposing to the method of sending MU's RSSI vectors to the internet server where the server then executes the positioning process and returns the position to the MU (e.g. Skyhook Wireless <http://www.skyhook.com>), this method reduces the computational burden of the internet server. Also, this method allows the MU to perform positioning without having continuous internet connectivity, since intermittent internet connectivity is sufficient for the MU to download the necessary Reference Database. Hence, this method is proposed to be implemented in later stage to support positioning while internet connectivity is unavailable. The probabilistic algorithm for WiFi positioning as introduced in (Roos, et al., 2002) is a more accurate method due to its ability to account for the probability

distribution of the RSSI signals. However, in depth analysis on the computational capability of the embedded system should be performed before undertaking this task. The system should also be extended to accommodate building level or altitude information, such that a fully operational 3D positioning device is achieved.

References

- Binghao Li. (2006). *'Terrestrial mobile user positioning using TDOA and Fingerprinting techniques'*, PhD thesis, University of New South Wales.
- Cheng, Y., Chawathe, Y., LaMarca, A., & J. Krumm. (2005). *Accuracy Characterization for Metropolitan-scale Wi-Fi Localization*. Proceedings MobiSys 2005.
- Guenther Retscher, E. M. (2007). *Performance and accuracy test of a WiFi indoor positioning system*. Journal of Applied Geodesy 2007.
- Haeberlen., A., Flannery., E., Ladd., A., & Rudys., A. (2004). *Practical Robust Localization over Large-Scale 802.11 Wireless Networks*. Philadelphia, PA: Proc. of the 10th ACM International Conference on Mobile Computing and Networking (MOBICOM).
- Kaplan, E. D., & Hegarty, C. J. (2006). *Understanding GPS : principles and applications*. Boston : Artech House.
- Lee, H. K., Li, B., & Rizos, C. (2005). *Implementation procedure of wireless signal map matching for location-based services*. Proc. IEEE Int. Symposium Signal Processing and Information Tech.
- Li, B., Kam, J., Lui, J., & Dempster, A. (2007). *Use of Directional Information in Wireless LAN based indoor positioning*. Sydney, Australia: Symp. on GPS/GNSS (IGNSS2007).
- Li, B., Salter, J., Dempster, A., & Rizos, C. (2006). *Indoor Positioning Techniques Based on Wireless LAN*. Sydney, Australia: First IEEE International Conference on Wireless Broadband and Ultra Wideband Communications.
- Mumford, P., Parkinson, K., & Dempster, A. (2006). *An Open GNSS Receiver*. Surfers Paradise, Australia: Symp. on GPS/GNSS 2006.
- P. Bahl and V. N. Padmanabhan. (2000). *RADAR: An in-building RF-based user location and tracking system* (Vol. 2). Tel-Aviv, Israel: Proc. IEEE INFOCOM.
- R. Singh, M. Guainazzo, C. S. Regazzoni. (2004). *Location determination using WLAN in conjunction with GPS Network*. Proc. IEEE 59th VTC—Spring.
- Roos, T., Myllymaki, P., Tirri, H., Misikangas, P., & Sievanen, J. (2002). *A Probabilistic Approach to WLAN User Location Estimation*. International Journal of Wireless Information Networks 2002.
- Wang, Y., Jia, X., & Lee, H. (2003). *An Indoors Wireless Positioning System Based on Wireless Local Area Network Infrastructure*. Proceedings of the 6th International Symposium on Satellite Navigation Technology Including Mobile Positioning & Location Services.