Location Tracking and Location Based Service Using IEEE 802.11 WLAN Infrastructure*

Can Komar and Cem Ersoy
Computer Engineering Department , Bogazici University, Istanbul, Turkey
e-mail: {komarcan, ersoy}@boun.edu.tr

Abstract. **Location tracking systems including the GPS, Active Badge, Bat, SmartFloor and Cricket systems attracted a significant group of researchers during the recent decades. Whatever the coverage and accuracy of these systems are, they all depend on customized hardware and/or trained personnel for implementation. Our aim in this study was to develop a solution that will not be hardware dependent and easy to deploy. In this paper, we propose a pure software based solution, namely WLAN Tracker, which is based on the signal strength measurements from different wireless access points (WAPs). Unlike similar systems, WLAN Tracker is designed to operate in multistory buildings. Moreover, as a demonstrative example, a simple location based service (LBS) is added to WLAN Tracker in order to provide text-based services to its clients. We investigate the performance of our system by comparing its location estimations with real values.**

1. Introduction

The wireless local area networks (WLANs) are increasing their coverage area and communication speeds, where 11Mbps IEEE 802.11b WLANs are very common now. The challenge in the wireless networks has shifted from speed to services. Now, roaming across WLAN areas while browsing the Internet is not enough for the end-users. The WLAN service providers are trying to increase their market share by offering different and exciting services to their subscribers. Most of the hot services offered or proposed are the ones that depend on the user's location.

Location tracking in IEEE 802.11 WLANs has been a hot topic in the recent years. RADAR from Microsoft [1], PhD from Carnegie Melon [2], Active Campus from UCSD [3] and Ekahau Positioning Engine™ from Ekahau, Inc [4] are examples of such location tracking systems. In this paper, the implementation and performance evaluation of a similar system, WLAN Tracker, which is optimized to operate in multistory buildings, is investigated. In Section 2, implementation and function of our system, WLAN Tracker, are explained and a future framework is proposed. In Section 3, the experiments conducted and the performance evaluation results are given. In Section 4, the conclusion and possible future work are summarized.

* This work is supported by the State Planning Organization of Turkey under the grant number DPT 98K120890.

2. WLAN Tracker

WLAN Tracker is a location tracking and location based services system developed within Computer Engineering Department, Bogaziçi University. The aim of the system is to track the users with IEEE 802.11 supported devices across the coverage area of a WLAN. WLAN Tracker system consists of server and client components. The client communicates with the server and sends the necessary data to it where the server performs the location tracking operation. WLAN Tracker operates in two phases, the data collection phase and the execution phase. In the data collection phase, the signal map that will be used in the execution phase is constructed. During this phase, the user walks around the map area and takes samples with the help of the GUI. These data are stored in a database. In the execution phase, the client periodically sends the signal levels detected to the server for location estimation. In Figure 1, the layout of WLAN Tracker system during the execution phase is given. $Client_1$ can detect signals of WAP_A, WAP_B, and WAP_C whereas $Client_2$ can detect signals of WAP_A and WAP_B. Both clients send the detected signal strength of each WAP and its MAC address to the WLAN Tracker Server periodically. WLAN Tracker Server evaluates the received data and each independent thread tries to estimate the location of the client. If the client's location can be estimated, its position is updated on the WLAN Tracker GUI. The location estimation threads are independent so they can be configured independently and they can operate on maps that have common regions.

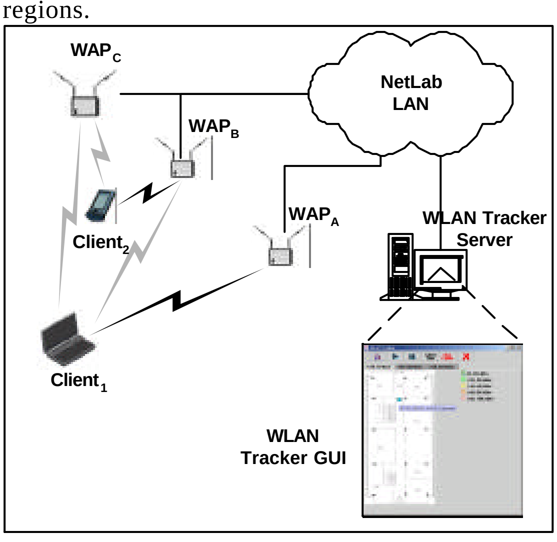

Figure 1. WLAN Tracker system in execution.

The WLAN Tracker system uses the infrastructure developed within the Mobile Application Services and Testbed (MAST) system [5]. The tests are done on the first three floors of the Computer Engineering building excluding the ground floor. The dimensions of each floor are 29.9 m x 16.2 m. The testbed used during the experiments consists of five WAPs, two laptop computers and one desktop computer. The server software runs on a Windows 2000 Professional desktop and the clients are run on laptops with RedHat Linux 7.1. Java and C are used during the implementation.

2.1. Data Collection Phase

In this phase, a signal map of the experiment area is constructed. The person who takes the sample data stops at predetermined locations and clicks the current location on the map. After the map is clicked, the signals' strengths received from WAPs are measured. This operation is repeated for *n* times with a period of *t* milliseconds intervals in order to avoid the temporary fluctuation periods in the received signal strength. In our experiments, *n* was chosen as 100 and *t* was chosen as 10 making each data collection step about 1 second long. After iterations are completed, the mean of the collected signal strength data is calculated for each WAP and recorded to the database along with the coordinates. These collected data are called as the Signal Map.

2.2. Execution Phase

In the execution phase, the signal strength data are sent to the WLAN Tracker server by the client for location estimation. The server uses the signal map constructed in the previous phase during the execution. The WLAN Tracker server is implemented in a multithreaded manner. There is a main thread that listens to the socket. Each map is managed by a separate thread, which evaluates the data and performs an estimation operation. When the server receives the signal data, the main thread authenticates the packet content. The MD5 hash within the packet is investigated. This hash is constructed by the client with a combination of user name, password and signal strength data strings. For authentication, the main thread recomputes the hash and compares it with the one received. If they match, the packet is authenticated. After the packet is authenticated, it is passed to the map threads. Location of the client is estimated by each thread separately with the analysis of the packet. If the necessary option is enabled, the client's location history is used during this process. If the estimation can be completed successfully, then the location of the spot that is indicating the client is updated on the floor map. If the client is a new one, then a new spot representing it is added to the floor map. If a client does not send signal strength data or its location is not estimated in a map during a period of time, it is assumed that the client has been shut down or it is out of the respective map. In this case, the client information and its spot are deleted. In our experiments this timeout was chosen as 15 seconds.

2.3. Location Estimation Process

During the location estimation process, each map thread works independently without communicating with other map threads. Thus, a map may be added or removed without affecting the system. For example, some maps in the system may have common areas and a client within this area is displayed on both maps. In the location estimation, first of all, the map thread tries to determine if the received packet belongs to its responsible map area or not. If there is a common WAP between its signal map data and the received signal data, then it is concluded that the client is in its map area. After that, an elimination filter is applied on the signal map and the best 20 matches with the received data according to the distance between them are selected. This initial filtering is applied to decrease the size of the search set that will be used in the further steps of the estimation.

The distance between each data set can be computed according to different metrics, Euclidian distance or Manhattan distance. For a point (x, y) in the signal map, say there are k measurements in the signal map, and there are n measurements in the signal strength data sent by the client. If each measurement in the signal map is represented with $(WAP_i, Signal\_Strength_i)$ tuple, and if each measurement in the received data is represented with $(WAP_j, Signal\_Strength_j)$ tuple, we can find c tuples, which are measurements belonging to the same WAP, i.e. $WAP_i = WAP_j$. In that case, the signal strength data can be represented in c tuples, each tuple represented as $(Signal\_Strength_i, Signal\_Strength_j)$ or as (m_n, a_n) where $n = 0, 1, \ldots, c$. The distance of two measurement sets, $D(m, a)$ can be calculated either as in Equation 1 or as in Equation 2. The first equation gives the Euclidian distance and the second one gives the Manhattan distance between two measurement sets.

$$D(m, a) = \sqrt{\sum_{k=1}^{c} (m_k - a_k)^2} \quad (1)$$

$$D(m, a) = \frac{\sum_{k=1}^{c} |m_k - a_k|}{c} \quad (2)$$

The initial reduced set may be passed through the history filter and the signal filter, if they are enabled in the configuration file. After all enabled filters are applied, the size of the search set is inspected. There are three possible cases:

0: The location of the client cannot be detected within the current map. The timeout value of the client is checked. If it has exceeded the predetermined threshold, it is concluded that the client has left the map area and its spot is disabled.

1: The location of the client is successfully determined. If the client already exists on the map, its spot location is updated else a new spot is added to the map.

> 1: The location of the client is ambiguous. A final selection is applied and the location that gives

the minimum distance with the current data is returned as the result. Then, update operations are performed as in the previous case.

2.4. Proposed Framework

Although WLAN Tracker is implemented as a stand-alone application at the beginning, it is planned to integrate it as a module in a larger communication framework. The whole framework will consist of different modules. One of these modules will be a session manager of the PERA system [6]. This module will conduct all registration and session management operations. In addition to these, it will perform presence and instant managing related functions. SIP [7] is selected as the signaling protocol between modules, because of its simple and extendable structure. Presence and IM are two of the extensions supported by SIP [8][9]. Standard SIP clients will be able to register with the session manager and will receive and send presence and IM packets. In addition to these services, standard XML syntax will be extended to support location based presence and messaging.

It is proposed to include a streaming server within the system. For this purpose, it is being planned to use WIDE system, which is based on the Infostation concept [10][11]. The WIDE system will be modified to support SIP and will be integrated into the framework. The streaming data can be sent with RTP. LBS proposed in current WLAN Tracker system will be modified and improved in the framework accordingly. The Directory Services Manager (DSM) will be used to provide LBS to users. The DSM will work in coordination Directory Services Database (DSDB) and User Profile Database (UPDB) to manage the LBS. Some of the services provided will be voice or video streaming, notifications such as flash news and commercial advertisements.

3. Experiments and Performance Evaluation

Current WLAN Tracker system is implemented in Java and during all experiments laptops are used. It is planned to port the client to PDA environment in the future framework. The OS specific layer of the client is needed to be adapted to Windows CE or embedded Linux environment depending on the PDA. In addition to this, the client may be ported to C++, if new environment does not support Java. Several experiments are conducted before and during the implementation of the WLAN Tracker. Some of these experiments are performed to explore the properties of the wireless medium and its affects on the system design. Another set of experiments is conducted to measure the performance of the WLAN Tracker.

3.1. Effects of Environmental Conditions on Wireless Communication

The wireless communication is not reliable due to the propagation problems of the transmission medium. The signal quality can drop or rise suddenly due to attenuation and multipath fading properties of the environment. To analyze the stability of the environment and improve the algorithms used in the WLAN Tracker, the effects of the environment and floor attenuation on signal strength are inspected.

Effect of Different Environments on Signal Strength These tests are performed for two different environments, empty-office and busy-office. The client is replaced on the line of sight of the WAP, oriented such that the antenna of the PCMCIA card is looking towards the WAP antenna. Under this orientation, the signal power is sampled for 1000 times with five-second intervals, for a period of about 1.5 hours. In the empty-office environment, no obstacles existed between the wireless client and the WAP during the tests. In the busy-office environment, a busy hour office is simulated, with people blocking the line of sight or other wireless devices passing through the area where the test is conducted. The measurements are plotted on the time-based graph as in Figure 2 and Figure 3. The variance of empty-office environment is 3.986 where as it is 7.878 in busy-office environment. These large variance values make the sampling and location tracking operations rather difficult.

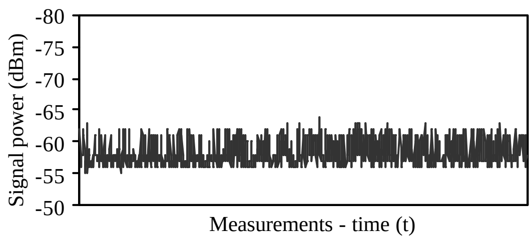

Figure 2. Empty-office measurements.

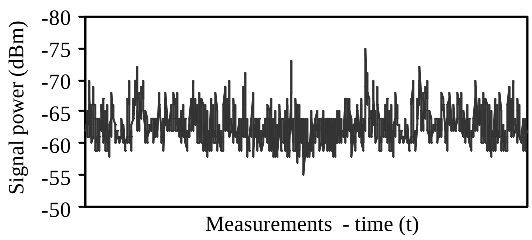

Figure 3. Busy-office measurements.

Effect of Multi-Floor Environment on Signal Strength The concrete walls have high attenuation affect on signal strength. Computer Engineering building in which our tests are conducted consists of such walls. According to the experiments conducted, one floor can reduce the signal power approximately from 15 dBm to 35 dBm. This attenuation affect has been the basic concept for the design of the signal filter. To measure the floor attenuation, two WAPs are placed in vertically symmetric positions on different floors, WAP_1 on the second floor and WAP_2 on the first floor. The signal strengths of WAPs detected by the client on both floors are recorded. For the sake of simplicity, only the first 100 measurements are shown in Figure 4 from a total of 500 samples. Signal strength distributions on different floors are almost symmetric as expected, and there are two horizontal regions that are separated from each other with at least 15 dBm difference.

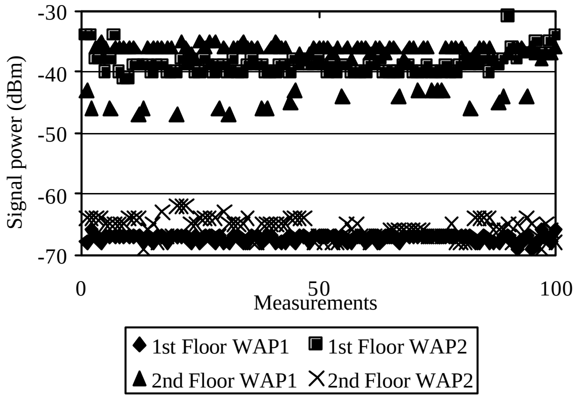

Figure 4. Distribution of signal strengths in multi-floored environment.

Table 1. Effect of floor attenuation on signal strength (in dBm).

	1st Floor WAP_1	1st Floor WAP_2	2nd Floor WAP_1	2nd Floor WAP_2
Min	-69	-41	-47	-69
Max	-62	-31	-35	-62
Mean	-67.35	-38.97	-38.34	-65.47

Table 1 shows that, there is an average of 27.7 dBm attenuation in the signal received from a WAP on another floor. The minimum average floor attenuation is 22 dBm and the maximum average floor attenuation is 31 dBm. In wireless networks, the signal strength decreases with distance due to fading effect. In an ideal environment, the points that receive signals in the same strength froms perfect circles surrounding the WAP. For a plane, on which sampling points are evenly distributed, the location estimation accuracy is inversly proportional with the numbber of sampling points in that signal strength region. The number of sampling points is proportional with the area in an evenly distributed case. So, area of two regions can be compared instead of calculating the number of sampling points. Equation 3 gives the area for $-x$ dBm region whereas Equation 4 gives the area for $-nx$ dBm region. When the areas are compared, second one is $2n$-1 times larger than the first one. It can be concluded that, the accuracy for the $-nx$ dBm region is $2n$-1 times less than the accuracy for the $-x$ dBm region. Thus, weak signals do not improve the accuracy of the estimations dramatically.

$$A_x = pr^2 \quad (3)$$

$$A_{nx} = p(nr)^2 - p((n-1)r)^2 \quad (4)$$

Signal Filter In WLAN Tracker system, a weak signal can be either from a WAP on a different floor or from a point away from the WAP. In the first case, this signal should not be evaluated by the WLAN Tracker, because it belongs to another floor and taking it into consideration may result in wrong floor estimations. In the second case, this signal probably will not improve the accuracy of the estimation. In both cases, the elimination of the weak signal measurements do not decrease the accuracy drastically and it eliminates possible miscalculations. A signal filter mechanism is developed in WLAN Tracker. The signal filter basically eliminates all the measurements below a threshold. Its effects on the results are analyzed in the following section.

3.2. Performance of WLAN Tracker

Indoor and outdoor experiments are conducted to measure the performance of the WLAN Tracker. The indoor experiments are conducted on a single floor and multi floor basis. A set of experiments is conducted to measure this factor in our system. Keeping all other factors constant, the number of WAPs is changed and estimations are performed. The distribution of error distances is given in Figure 5. The average error distance is 8.144 m for one WAP case, 4.767 m for two WAPs case and 2.244 m for three WAPs case.

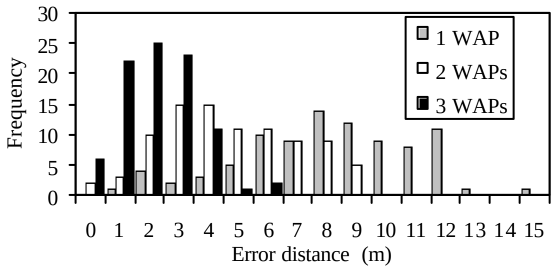

Figure 5. Error distribution vs. number of WAPs.

Indoor Multi Floor Experiments WLAN Tracker system is designed to function in multi-floor environments. For this purpose, a signal level filter is designed and implemented in the location estimation algorithm. To measure the effects of the signal filter on the results, two sets of experiments are performed. One with filter enabled and another with filter disabled. In both cases, 100 estimations are done for 30 sample points. The number of miscalculations of the floor for both cases is recorded. If a client is shown on more than one map, even including the correct one, it is counted as a miscalculation. These tests are repeated for all three floors in order to reflect different environments on each floor.

Table 2. Effect of signal level filter.

	Filter OFF		Filter ON	
	TRUE	FALSE	TRUE	FALSE
1st Fl.	96	4	99	1
2nd Fl.	92	8	100	0
3rd Fl.	90	10	98	2
Total	278	22	297	3

Results of the experiments are shown in Table 2. The failure rate has dropped from 7.33 per cent to 1 per cent on average. It can be concluded that, the signal level filter is very successful for eliminating floor estimation errors. On the other hand, the effects of the filter on the estimation accuracy must be considered. Signal level filter runs on the principle of eliminating the weak signals and ignoring them in the calculations. This elimination may affect the accuracy negatively, if there are not many WAPs in the environment and their signals are very weak. Signal level filter can be safely turned off in the single floor environments. In multi-floor environments, the number of WAPs and their locations in each floor must be considered. Although signal level filter's effect on the accuracy was not very much in our experiments, as shown in the analysis section, this statement may not be always valid and the accuracy of estimation depends on the configuration of the WLAN.

Outdoor Experiments The performance of the WLAN Tracker is measured in the outdoor environment, too. For this purpose, the rectangular space, which is about 240 m^2 in area, in front of the Computer Engineering building,is divided into grids each with a side about 2 m long. The map of the space that consists of 15 x 4 squares is constructed. The data collection phase is performed for the center of each square and a map consisting of 60 distinct points are prepared. During experiments, 10 estimations are done for each square for a total of 600 estimations. The error distances are measured according to Manhattan formula. The average error is 6.537 m with a variance of 11.671.

The average error in the outdoor experiments is larger than the average error obtained in the indoor experiments. One of the reasons is the size of sampling points. In the indoor experiments, the distance between each sampling point is 1.2 m where as in the outdoor experiments it is 2 m. Taking samples at larger intervals is advantageous in the data collection phase but decreases the accuracy. Another reason for larger error is the number of WAPs signal received from. During outdoor experiments, we could detect only the two WAPs that are on the right wing of the building. There were cases, in which only one WAP's signal could be detected.

3.3. Performance Factor Sensitivity Analysis

In our performance analysis, the affect of internal parameters of the WLAN Tracker algorithm on the results is analyzed. The indoor environment experiment results are used since they are conducted in detail and the sample rate is larger than the outdoor environment tests. Three factors are changed and investigated. These factors are the equation used in distance computation, signal filter and history filter. Our metric is the average error made in the location estimations. We tried to figure out the importance of the factors investigated. The methods used during analysis are 2^k Factorial Design and Signal Table Method [12]. The factors are labeled as *A*, *B*, and *C* and their levels are assigned as –1 and +1 as shown in Table 3. The average error distances measured with different options are shown in Table 4. According to Table 3, we define three variables X_A, X_B, and X_C as in Equation 5, 6 and 7. The performance y, which is average error in *meters*, can be regressed on X_A, X_B, and X_C using a regression model of the form in Equation 8. Substituting the eight observations in the model, we get the eight equations from Equation 9 to Equation 16. These eight equations can be solved uniquely for the eight unknowns. However, the Sign Table Method has a simple solution, which can be seen from Table 5.

Table 3. Factors A, B, C with their levels.

Factor	Level –1	Level +1
Distance Metric (*A*)	Euclidian	Manhattan
Signal Filter (*B*)	Off	On
History Filter (*C*)	Off	On

Table 4. Average errors at each level of factors (in meters).

	Euclidian		Manhattan	
	Hist. F. Off	Hist. F. On	Hist. F. Off	Hist. F. On
Signal F. Off	5.304	4.035	5.406	4.230
Signal F. On	5.208	3.975	5.190	4.105

$$X_A = \begin{cases} -1, if\_Euclidian \\ 1, if\_Manhattn \end{cases} \quad (5)$$

$$X_B = \begin{cases} -1, if\_Signal\_filter\_is\_OFF \\ 1, if\_Signal\_filter\_is\_ON \end{cases} \quad (6)$$

$$X_C = \begin{cases} -1, if\_History\_filter\_is\_OFF \\ 1, if\_History\_filter\_is\_ON \end{cases} \quad (7)$$

$$y=q_0+q_Ax_A+q_Bx_B+q_Cx_C+q_{AB}x_Ax_B+q_{AC}x_Ax_C+q_{BC}x_Bx_C+q_{ABC}x_Ax_Bx_C \quad (8)$$

$$5.304 = q_0 - q_A - q_B - q_C + q_{AB} + q_{AC} + q_{BC} - q_{ABC} \quad (9)$$

$$5.406 = q_0 + q_A - q_B - q_C + q_{AB} + q_{AC} + q_{BC} + q_{ABC} \quad (10)$$

$$5.208 = q_0 - q_A + q_B - q_C - q_{AB} + q_{AC} - q_{BC} + q_{ABC} \quad (11)$$

$$5.190 = q_0 + q_A + q_B - q_C + q_{AB} - q_{AC} - q_{BC} - q_{ABC} \quad (12)$$

$$4.035 = q_0 - q_A - q_B + q_C + q_{AB} - q_{AC} - q_{BC} + q_{ABC} \quad (13)$$

$$4.230 = q_0 + q_A - q_B + q_C - q_{AB} + q_{AC} - q_{BC} - q_{ABC} \quad (14)$$

$$3.975 = q_0 - q_A + q_B + q_C - q_{AB} - q_{AC} + q_{BC} - q_{ABC} \quad (15)$$

$$4.105 = q_0 + q_A + q_B + q_C + q_{AB} + q_{AC} + q_{BC} + q_{ABC} \quad (16)$$

Table 5. Matrix Calculation in 2^k Factorial Design.

I	A	B	C	AB	AC	BC	ABC	Y
+	-	-	-	+	+	+	-	5.304
+	+	-	-	-	-	+	+	5.406
+	-	+	-	-	+	-	+	5.208
+	+	+	-	+	-	-	-	5.190
+	-	-	+	+	-	-	+	4.035
+	+	-	+	-	+	-	-	4.230
+	-	+	+	-	-	+	-	3.975
+	+	+	+	+	+	+	+	4.105
37.45	0.41	-0.50	-4.76	-0.19	0.24	0.13	0.06	Total
4.68	0.05	-0.06	-0.60	-0.02	0.03	0.02	0.01	Tot/8
q_0	q_A	q_B	q_C	q_{AB}	q_{AC}	q_{BC}	q_{ABC}	

Sample variance of $$y = s_y^2 = \frac{\sum_{i=1}^{2^3} (y_i - \bar{y})^2}{2^3 - 1} \qquad (17)$$

In Equation (17), $\bar{y}$ denotes the mean of throughputs from all eight experiments. The numerator of Equation (17) is called the total variation of y or Sum of Squares Total (SST) and is expressed as in (18). For a 2^3 factorial design, the variation can be divided into seven parts. The result is (19).

$$y = SST = \sum_{i=1}^{2^3} (y_i - \bar{y})^2 \qquad (18)$$

$$SST = \sum_{i=1}^{2^3} (y_i - \bar{y})^2 = 2^3 (q_A{}^2 + q_B{}^2 + q_C{}^2 + q_{AB}{}^2 + q_{AC}{}^2 + q_{BC}{}^2 + q_{ABC}{}^2) \qquad (19)$$

The first three parts in parentheses in Equation (19) represent the portion of the total variation explained by the effect of A (the distance metric used), B (the signal filter), C (the history filter) and the remaining parts represent interactions among them, AB, AC, BC and ABC.

$$SST = 8(0.003+0.004+0.354+0.00+0.001+0+0) = 2.90 \qquad (20)$$

Table 6. Fraction of variation explained by factors.

Factor A (Distance metric)	0.72%
Factor B (Signal filter)	1.07%
Factor C (History filter)	97.73%
Interaction of A & B	0.15%
Interaction of A & C	0.25%
Interaction of B & C	0.07%
Interaction of A & B & C	0.01%

The calculations above and the results in Table 6 show that the history filter has a huge impact on the quality of the estimations compared to other factors used in the experiments. We can conclude that, the main concern for the accuracy of the tests is the history filter, which should be improved thoroughly. The signal filter has importance in the determination of the floor that the client is on, but does not have a high impact on the correctness of the estimation. The distance formula has the least effect, thus in our opinion, the selection criterion in the distance formula should not be the accuracy but the performance. In that case, the Manhattan distance formula can be preferred instead of Euclidian distance formula because of the extra floating-point arithmetic load that the latter one required.

4. Conclusions and Future Work

The WLAN Tracker is a location tracking and LBS system that is developed using IEEE 802.11 WLAN infrastructure. The system is developed in pure software without any specialized hardware. WLAN Tracker is developed in server-client architecture. The client collects the wireless signal strength and sends these data to server for evaluation. The server evaluates the sent data and estimates the user's location, which can be tracked via WLAN Tracker GUI. In addition to location tracking, some text based LBS services are defined. When a client enters a predefined LBS area, a text message is sent to the client. The history filter used is based on the assumption that the users of the system cannot exceed a predetermined speed. This method is used for filtering out instantaneous drops or rises in the measured signal power. During the periods with extreme movements, the statistical search methods may fail. In addition to history filter, signal level filter is defined for multi-floor effects. In a multi-floored environment, the wireless signal data may match with a point in signal map for a different floor or it may match with two distinct points in different signal maps. In that case, the client's location is shown on the wrong map. To solve this problem, signal level filter is developed.

A framework is designed in NetLab [6] in order to integrate several services. It is proposed to integrate WLAN Tracker system within this framework as a module. Some of other modules within the framework will perform functions such as presence, instant messaging and audio and video streaming.

References

[1] P. Bahl and V. N. Padmanabhan, "RADAR: An In-Building RF-Based User Location and Tracking System", *Proceedings of the IEEE Infocom 2000*, Vol. 2, pp. 775-784, Tel-Aviv, Israel, March 2000.

[2] A. Hills and D. Johnson, "A Wireless Data Network Infrastructure at Carnegie Mellon University", *IEEE Personal Communications*, Vol. 3, pp. 56–63, 1996.

[3] W.G. Griswold, R. Boyer, S.W. Brown and T, M. Truong, "A Component Architecture for an Extensible, Highly-Integrated Context-Aware Computing Infrastructure", *International Conference on Software Engineering 2003 (ISCE 2003)*, May 2003

[4] Ekahau, Inc., *Ekahau Positioning Engine™ 2.0 Specification*, http://www.ekahau.com/pdf/Ekahau_Positioning_Engine.pdf, 2003.

[5] NetLab, *Mobile Applications and Services Testbed*, http://netlab.boun.edu.tr/mast, June 2002.
[6] E. Deveci, *PERA: Location Based Multimedia Services Framework*, PERA System Technical Report TR-003, Bogaziçi University, June 2003.
[7] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler, "SIP: Session Initiation Protocol", *IEEE Network Working Group RFC 3261*, June 2002.
[8] B. Campbell, J. Rosenberg, H. Schulzrinne, C. Huitema and D. Gurle, "SIP Extension for Instant Messaging", *IEEE Network Working Group RFC 3428*, December 2002.
[9] Day M., J., J. Rosenberg, and Sugano H., "A Model for Presence and Instant Messaging", *IEEE Network Working Group RFC 2778*, February 2000.
[10] NetLab, *Wireless Information Delivery Environment*, http://netlab.boun.edu.tr/~wide, June 2003.
[11] F. Frenkiel, B. R. Badrinath, J. Borras, and R. Yates, "The Infostations Challenge: Balancing Cost and Ubiquity in Delivering Wireless Data", *IEEE Personal Communications*, pp. 66-71, April 2000.
[12] R. Jain., The Art of Computer Systems Performance Analysis, Wiley, 1992.