# Fault Detection and Mitigation in WLAN RSS Fingerprint-based Positioning

Christos Laoudias, Michalis P. Michaelides and Christos G. Panayiotou
KIOS Research Center for Intelligent Systems and Networks
Department of Electrical and Computer Engineering, University of Cyprus
Email: laoudias@ucy.ac.cy, michalism@ucy.ac.cy, christosp@ucy.ac.cy

***Abstract*—The provision of reliable location estimates in case of unpredicted failures or malicious attacks, which inject faults and compromise the performance of the positioning system, is very important. Thus, our main interest is on the fault tolerance of WLAN fingerprint-based methods, rather than the absolute positioning error in the fault-free case. We study the Nearest Neighbor method and as a first step we develop a robust detection scheme to accurately detect faults. We incorporate this into a hybrid positioning method that switches to a modified distance metric, instead of the Euclidean, if faults are present. Experimental results indicate that the proposed approach exhibits higher resilience to faults compared to other positioning methods.**

*Keywords:* Wireless networks; Positioning methods; Signal strength fingerprints; Fault detection; Fault tolerance.

## I. Introduction

The increasing demand for indoor location-based services, including asset tracking or in-building guidance and navigation, has motivated the development of positioning methods that exploit the existing wireless networks. Several methods rely on WLANs and use Received Signal Strength (RSS) fingerprints to determine location, owing to the wide availability of WLAN Access Points (AP) and the ease of collecting RSS samples without specialized equipment; see [1] for an overview and taxonomy of fingerprint-based methods.

Accuracy is an important requirement and has been the main interest of researchers so far. However, fault tolerance is also highly desirable because the RSS values in the fingerprints are corrupted in the presence of faults, thus leading to accuracy degradation. For instance, some APs may become unavailable during positioning either due to unexpected failures, such as power outages, or as the result of an attack. Detecting faults and providing adequate accuracy under faults has surprisingly received little attention; in [2] statistical significance testing is used to detect signal attenuation or amplification attacks that disturb the RSS values. The communication capabilities among MicaZ nodes are used in a Wireless Sensor Network (WSN) setup to collaboratively detect node failures in the *MoteTrack* system [3]. However, this approach cannot be directly applied to methods that rely on WLAN APs. Regarding fault tolerant positioning, authors in [4] improve the robustness to RSS attacks by deploying redundant nodes or APs and using a median-based distance metric, instead of the Euclidean, in the underlying positioning algorithm. Similarly, an adaptive distance metric is used in [3] to cater for faulty nodes.

Robust fault detection is important in order to switch to a fault tolerant positioning method if required. In our approach we address both issues and examine the well known Nearest Neighbor (NN) positioning method [5] as a case study. First, we present a fault detection scheme that is applicable to the NN method and proves to be very effective when APs fail accidentally or maliciously. Then, we combine it with a modified distance metric, which is employed in case of fault detection, in order to build a hybrid positioning method that greatly improves the fault tolerance of the original method.

We present our fault model and the measurement setup in Section II. In Section III we focus on the NN positioning method and provide the details of our fault detection scheme. The proposed hybrid method is described in Section IV, followed by an experimental evaluation with respect to its fault tolerance. Finally, Section V provides concluding remarks.

## II. Fault Model

### A. Model Description

In this work, we use a set of predefined reference locations $\{L : \ell_i = (x_i, y_i),\ i = 1, \ldots, l\}$ to collect RSS values from $n$ APs deployed in the area of interest and build the radio map (offline phase). A reference fingerprint $r_i = [r_{i1}, \ldots, r_{in}]^T$ associated with location $\ell_i$, is a vector of RSS samples and $r_{ij}$ denotes the RSS value related to the $j$-th AP. Usually, $r_i$ is averaged over multiple fingerprints collected at $\ell_i$ to alleviate the effect of noise in RSS measurements and outlier values. During positioning (online phase), we exploit the reference data to obtain a location estimate $\widehat{\ell}$, given a new fingerprint $s = [s_1, \ldots, s_n]^T$ measured at the unknown location $\ell$.

In our fault model we consider the case where several APs used in the offline phase are not available during positioning. This can be caused by unpredicted AP failures due to power outages or WLAN system maintenance. When an attack is assumed, an adversary can easily cut off the power supply of some APs or use specialized equipment to severely jam the communication channels to make the attacked APs unavailable. We simulate this fault model by removing the RSS values of the faulty APs in the original test fingerprints.

### B. Measurement Setup

We collected our reference data in a typical modern office environment on the second floor of a three storey building.

There are 31 WLAN APs installed in the building and on average 9.7 APs are detected per location. We used a smartphone to collect 30 RSS fingerprints at 107 distinct reference locations on the second floor for a total of 3210 reference fingerprints and the RSS values range from $-101$ dBm to $-34$ dBm. We collected additional testing data by walking over a path that consists of 192 locations. One fingerprint is recorded at each location, and the same path is sampled 3 times.

## III. Nearest Neighbor Positioning Method

Nearest Neighbor method estimates location by minimizing the Euclidean distance $D_i$, between the observed fingerprint during positioning $s$ and the reference fingerprints $r_i$

$$\widehat{\ell}(s) = \arg\min_{\ell_i} D_i, \quad D_i = \sqrt{\sum_{j=1}^{n} \big(r_{ij} - s_j\big)^2}. \qquad (1)$$

Essentially, all reference locations are ordered according to $D_i$ and the location $\ell_i$ with the shortest distance between $r_i$ and $s$ in the $n$-dimensional RSS space is returned as the location estimate. The $K$ Nearest Neighbors (KNN) method estimates location as the mean of $K$ reference locations with the shortest distances and has been reported to provide higher level of accuracy compared to NN [5].

In practical implementations, WLAN APs provide only partial coverage in the area of interest and it is not expected that the sets of APs in fingerprints $r_i$ and $s$ will be identical. Assuming fault-free positioning, a specific AP found in a fingerprint $r_i$ and not in $s$ can be due to the fact that $s$ is recorded in a location $\ell$ that is far from $\ell_i$. Even if $\ell$ and $\ell_i$ are spatially correlated, $s$ may not contain a RSS reading from that AP because of a transient effect in the WLAN adapter of the mobile device. Alternatively, if faults are also considered, then the missing AP can be the result of a random failure or a malicious attack during positioning. Thus, it is important to use a robust detection scheme to signify that there are faults in the currently observed fingerprint $s$.

To this end, we define $R_i$ and $S$ as the sets of APs that are present in fingerprints $r_i$ and $s$, respectively. Using these definitions, $D_i$ in (1) can be viewed as

$$D_i = \sqrt{\sum_{j \in R_i \cap S} d_{ij} + \sum_{j \in R_i \setminus S} d_{ij} + \sum_{j \in S \setminus R_i} d_{ij}} \qquad (2)$$

where $d_{ij} = \big(r_{ij} - s_j\big)^2$. The first term refers to the intersection of $R_i$ and $S$ and represents the distance with respect to those APs that are common in fingerprints $r_i$ and $s$. The second term employs those APs that are detected in $r_i$ and not in $s$, while the last term increases the distance $D_i$ further by considering those APs that are found in $s$ and not in $r_i$. Note that in the standard KNN method a small constant is used ($-105$ dBm in our setup) to replace the missing RSS values $s_j$ and $r_{ij}$ in the second and third term of (2), respectively.

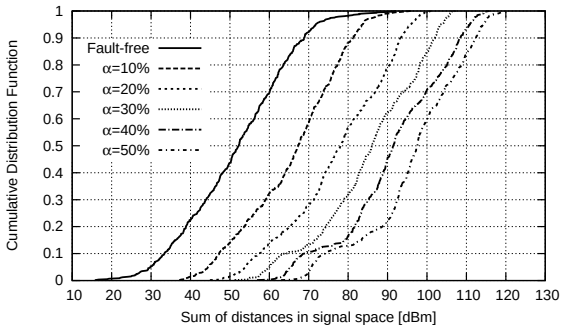



Fig. 1. CDF of the fault indicator $D_{sum}^{(2)}$.

### A. Fault Detection

In the presence of faults, fingerprint $s$ is expected to be dissimilar to the fingerprints in the radio map and is thus considered anomalous (outlier). Several techniques have been studied to detect outliers in diverse research areas and application domains; see [6] for a survey on anomaly detection techniques that are applicable to fault detection in WSNs.

In the context of KNN method we can exploit the distances $D_i$ that are already computed with (2), to decide whether fingerprint $s$ is corrupt or not. We may use a fault indicator based on these distances, e.g. distance from the $K$-th nearest neighbor and the intuition is that under faults the value of the indicator will violate a certain threshold. For instance, authors in [2] use the distance from the nearest neighbor (i.e. $K = 1$), denoted as $D_s$, to detect signal attenuation or amplification attacks. In this work we use the sum of distances to the $K$ nearest neighbors, denoted as $D_{sum}^{(K)}$, as a fault indicator which was proved experimentally to be more robust compared to $D_s$, under the fault model described in Section II-A. Note that in case $K = 1$ these two fault indicators are equivalent. As a first step in our fault detection scheme, we select an appropriate threshold $\gamma$ based on the distribution of $D_{sum}^{(K)}$ for the fingerprints contained in the original test set. Subsequently, a fault is detected in $s$ during positioning if $D_{sum}^{(K)} > \gamma$.

We examined several values for $K$ and the best performance, in terms of fault detection accuracy, was obtained for $K = 2$. The Cumulative Distribution Function (CDF) of $D_{sum}^{(2)}$ is plotted in Fig. 1 for the fault-free case (solid line) and when we inject faults in the test fingerprints. The first observation is that as the percentage of faulty APs ($\alpha$) is increased the CDF curve is shifted to the right. This fact indicates that we may choose the required threshold $\gamma$ to guarantee that the corrupt fingerprints will be detected with high probability. Note that according to the selection of $\gamma$ there is a trade off between the corrupt fingerprints that will go undetected when faults are present and the false detections in the fault free-case. We observe that in the fault-free case the $D_{sum}^{(2)}$ is below 72 dBm for 95% of the time and assuming that we can tolerate around 5% false detections, when no faults are present, we set $\gamma = 72$ dBm. This corresponds to the $65^{\text{th}}$, $32^{\text{th}}$, $15^{\text{th}}$, $11^{\text{th}}$ and $7^{\text{th}}$ percentile as $\alpha$ is increased from 10% to 50% and in other words our fault detection scheme is expected to detect

35%, 68%, 85%, 89% and 93% of the corrupt fingerprints, respectively. This level of performance may seem insufficient for $\alpha = 10\%$ or $20\%$, however as it will become evident later, the degradation in the positioning accuracy of KNN method is not significant for these values of $\alpha$.

### B. Experimental Evaluation

In order to assess the effectiveness of our fault detection scheme we adopt two performance metrics, namely the correct detections rate ($R_{cd}$) and the false detections rate ($R_{fd}$) which are defined as the ratio of the test fingerprints detected to be corrupt, either correctly or falsely, over all test fingerprints.

The $R_{cd}$ and $R_{fd}$ are plotted for several values of the detection threshold $\gamma$ in Fig. 2 and Fig. 3, respectively. When no faults are present (i.e. $\alpha = 0\%$), we can see in Fig. 3 that as the value of $\gamma$ is decreased the probability of false detections grows larger and vice versa. As expected, for $\gamma = 72$ dBm the $R_{fd}$ is around 5%. Moreover, when the percentage of faulty APs is low, i.e. $\alpha \leq 20\%$, the $R_{cd}$ remains below 0.75 for all $\gamma$ values; see Fig. 2. In this case it is hard to discern whether the missing values in the fingerprints are due to APs that have failed (accidentally or maliciously), because the APs do not provide ubiquitous coverage and there are missing values at different locations even under normal conditions. Thus, some corrupt fingerprints can be undetected. However, as it will be shown in Section IV-A, when few APs are faulty then the positioning accuracy is not severely degraded, thus failing to detect these faults is not crucial.

In addition false detections may still occur, when $\alpha \leq 20\%$. We can reduce them to some extend by selecting a higher value for $\gamma$; for instance when $\alpha = 10\%$, $R_{fd} = 4.3\%$ for $\gamma = 62$ dBm compared to 0.3% for $\gamma = 82$ dBm. On the other hand, using a higher threshold has a negative effect on the fault detection as $\alpha$ is further increased and leads to more corrupt fingerprints being undetected. Specifically, when $\alpha$ grows beyond 20% a higher $R_{cd}$ is desirable and this suggests using a lower $\gamma$ value in order to increase the sensitivity of our fault detection scheme. For instance, in case $\alpha = 40\%$ the $R_{cd}$ is 90% for $\gamma = 62$ dBm compared to 71% for $\gamma = 82$ dBm; see Fig. 2. Thus, using $\gamma = 72$ dBm provides a good compromise between low $R_{fd}$, especially in the fault-free case, and high $R_{cd}$ as the percentage of faulty APs is increased. Specifically, when $\alpha = 30\%$ the $R_{cd}$ is 0.76 for $\gamma = 72$ dBm, while there are almost no false detections. This means that for $\alpha \geq 30\%$, where high errors are introduced in the estimated locations, there is a high probability that a corrupt fingerprint will be detected. More importantly, in an attack scenario, there is low probability that an adversary will compromise the positioning system and heavily affect the accuracy without being detected.

## IV. Fault Tolerant Positioning

The standard KNN method can be modified appropriately in order to improve its fault tolerance. The distance metric in (2) is effective in the fault-free case, because it penalizes all APs that are not found in common between $r_i$ and $s$. However, in the presence of faults, it may not be able to guarantee the required fault tolerance and consequently an adequate level of accuracy. For instance, when AP failures occur during positioning, then a subset of the APs that would otherwise be sampled in fingerprint $s$, are no longer present. Thus, more APs fall into the $R_i \backslash S$ subset and errors in distances $D_i$ are increased due to the second term in (2). This leads to the wrong ordering of candidate locations and affects the positioning accuracy. In order to mitigate errors introduced by faulty APs, we may employ the following distance metric given by

$$D'_i = \sqrt{\sum_{j \in R_i \cap S} d_{ij} + \sum_{j \in S \backslash R_i} d_{ij}} \tag{3}$$

This metric ignores faulty APs in $R_i \backslash S$ and is expected to improve the fault tolerance of the standard KNN method, especially when a large number of APs are faulty. The proposed hybrid method incorporates the mechanism described in Section III-A, i.e. RSS distances $D_i$ are obtained with (2) to detect whether fingerprint $s$ is corrupt due to faulty APs. In case of fault detection, i.e. $D^{(K)}_{sum} > \gamma$, our method switches to $D'_i$ in (3) to calculate the RSS distances from all candidate locations and then determine location; otherwise location is estimated based on the already available distances $D_i$.

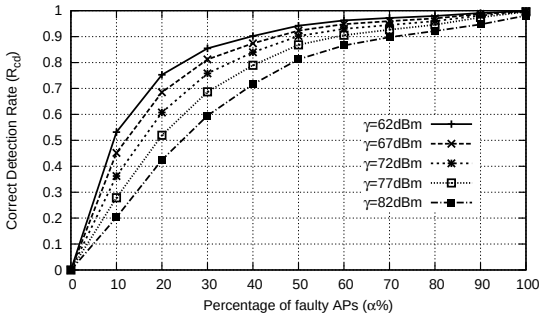



Fig. 2. The correct detection rate in the experimental setup.

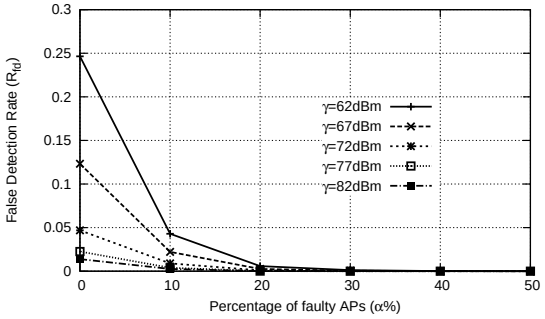



Fig. 3. The false detection rate in the experimental setup.

### A. Experimental Results

We investigate fault tolerance with respect to the accuracy degradation and a method is considered as fault tolerant, if the mean positioning error ($\mathcal{E}$) does not increase rapidly as the percentage of faulty (or attacked) APs is increased. Alternatively, we may select an acceptable upper bound on the performance, e.g. $\mathcal{E} = 5$ m, and examine the percentage of

faulty APs that each method can tolerate. We apply the fault model described in Section II-A to corrupt the original test data and the results for $\mathcal{E}$ are averaged over 100 runs using randomly selected subsets of faulty APs in each run.

We compare our hybrid method ($D^{(2)}_{sum}$, $\gamma = 72\,\text{dBm}$) against the standard KNN method and the KNN variant of [4], denoted as MED, which uses a median-based distance metric to improve the fault tolerance in case of failures or incorrect RSS readings. We also consider the probabilistic Minimum Mean Square Error (MMSE) approach of [7].

In Fig. 4, $\mathcal{E}$ is plotted for the MED, KNN, MMSE and HYBRID methods while the inset figure shows $\mathcal{E}$ more clearly when less than half of the APs have failed, i.e. $\alpha \leq 50\%$. In the fault-free case, the probabilistic MMSE method provides the best accuracy ($\mathcal{E} = 2.92\,\text{m}$). For the KNN method $\mathcal{E} = 3.03\,\text{m}$, followed by MED for which the mean error is $3.26\,\text{m}$. For the HYBRID method $\mathcal{E} = 3.04\,\text{m}$ which shows that the false detections in this case do not affect the positioning accuracy. We can also observe that the KNN and MMSE methods exhibit similar fault tolerance and if the percentage of faulty APs is low ($\alpha \leq 20\%$) the positioning error remains below $5\,\text{m}$ for both methods, which may be acceptable for some location-based applications. In the case that $\alpha = 20\%$, $\mathcal{E} = 3.34\,\text{m}$ for the HYBRID method even though the detection rate of our detection mechanism is 0.61, as shown in Section III-B.

As the number of faulty APs increases further, $\mathcal{E}$ grows sharply for both KNN and MMSE methods. The MED method can tolerate up to 30% of faulty APs ($\mathcal{E} = 3.97\,\text{m}$) and beyond that point it also fails to provide acceptable performance. On the other hand, the proposed HYBRID method proves to be extremely fault tolerant as the inset plot in Fig. 4 shows; if half of the APs are faulty, then $\mathcal{E}$ is only $4.20\,\text{m}$ compared to $6.65\,\text{m}$, $8.85\,\text{m}$ and $9.09\,\text{m}$ for MED, KNN and MMSE, respectively. If more than half of the APs are faulty then for the HYBRID method $\mathcal{E}$ is below $6.5\,\text{m}$, even when 70% of the APs are faulty. Assuming that $\mathcal{E} = 5\,\text{m}$ is the upper bound for a fingerprint-based positioning system, the HYBRID method can tolerate up to 60% of faulty APs compared to 40% for MED and only 20% for KNN and MMSE methods.

Results on the distribution of the positioning error are depicted in Fig. 5. For the standard KNN method, $\mathcal{E}$ is increased by $4.3\,\text{m}$ ($\alpha = 40\%$) compared to the fault-free case, while the standard deviation (*std*) is around $6\,\text{m}$; see Fig. 5a. On the other hand, the proposed HYBRID method exhibits similar performance when $\alpha \leq 40\%$; $\mathcal{E}$ is only increased by $0.7\,\text{m}$ ($\alpha = 40\%$) and *std* remains below $3\,\text{m}$ (Fig. 5b). These results indicate that the mitigation mechanism in our method can well handle the corrupt fingerprints and is actually effective in detecting outliers and reducing large errors.

## V. Conclusion

AP failures or malicious attacks introduce faults and corrupt the RSS values during positioning, thus leading to significant performance degradation in terms of the accuracy. In this work we focus on the KNN method and investigate how to improve its fault tolerance. In this direction, we develop a robust fault detection scheme to signify the presence of faults and couple this with a modified Euclidean distance metric that is preferred if faults are detected. Experimental results indicate that the proposed hybrid KNN variant is more fault tolerant compared to the standard KNN and other positioning methods.

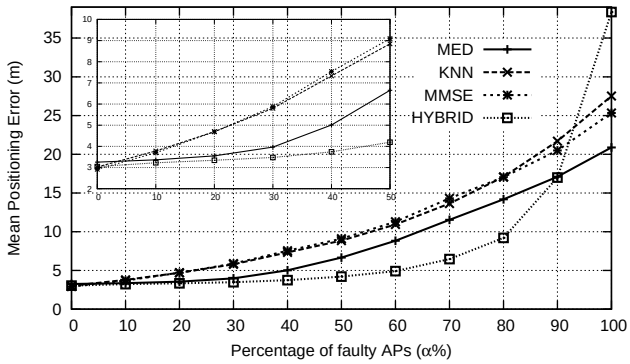
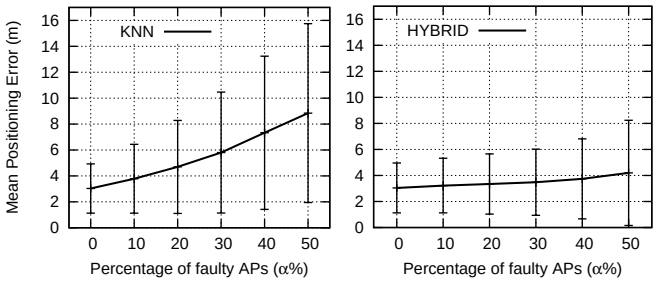



Fig. 4. Fault tolerance of positioning methods.




(a) KNN method. (b) HYBRID method.

Fig. 5. Mean and standard deviation of the positioning error.


## Acknowledgment

This work is supported by the Cyprus Research Promotion Foundation. Authors would like to thank P. Kemppi at VTT Technical Research Centre of Finland (`www.vtt.fi`) for the provision of experimental WLAN RSS data.


## References


[1] M. Kjærgaard, "A taxonomy for radio location fingerprinting," in *3rd international conference on Location-and context-awareness*. Springer-Verlag, 2007, pp. 139–156.

[2] Y. Chen, W. Trappe, and R. Martin, "Attack detection in wireless localization," in *26th IEEE International Conference on Computer Communications INFOCOM*, 2007, pp. 1964–1972.

[3] K. Lorincz and M. Welsh, "MoteTrack: a robust, decentralized approach to RF-based location tracking," *Personal and Ubiquitous Computing*, vol. 11, no. 6, pp. 489–503, Aug. 2007.

[4] Z. Li, W. Trappe, Y. Zhang, and B. Nath, "Robust statistical methods for securing wireless localization in sensor networks," in *International Symposium on Information Processing in Sensor Networks (IPSN)*, 2005, pp. 91–98.

[5] P. Bahl and V. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proceedings IEEE INFOCOM*, vol. 2, 2000, pp. 775–784.

[6] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, 2009.

[7] T. Roos, P. Myllymaki, H. Tirri, P. Misikangas, and J. Sievanen, "A probabilistic approach to WLAN user location estimation," *International Journal of Wireless Information Networks*, vol. 9, no. 3, pp. 155–164, Jul. 2002.