Object Localization Using RFID

Kirti Chawla, Gabriel Robins, and Liuyi Zhang

Department of Computer Science, University of Virginia
Charlottesville, Virginia, 22904, USA
{kirti, robins, lz3m}@virginia.edu

***Abstract* — Object localization is a key primitive in pervasive computing environments, where numerous applications depend on the rapid and accurate position estimation of objects. We present a general RFID–based localization framework that reliably determines the positions of objects with unprecedented accuracy and speed. This is achieved by varying the power levels of the RFID readers, calibrated against reference tags of known sensitivity. Our implementation and experiments are able to localize objects to an accuracy of 15 cm within a few seconds, and this compares favorably with previous techniques. We also suggest several practical optimizations for further enhancing the speed and accuracy of the method.**

Keywords –RFID, localization, positioning algorithms

I. Introduction

Radio frequency identification (RFID) technology is rapidly transforming pervasive computing applications by offering new capabilities and a richer user experience [13]. Capabilities such as object identification, real time tracking, and object localization are at the heart of numerous innovative RFID applications [9] [11]. While RFID technology enables object identification and tracking, it does not normally include object localization (i.e., positioning) capabilities. We propose to address this limitation by developing an RFID–based localization framework that accurately and quickly determines the positions of objects. In other words, our system offers a GPS-like positioning capability in an RFID environment.

Obstacles to localization accuracy, speed and reliability, include environmental interferences and occlusions (e.g., the presence of liquids and metals), orientation and spatial arrangement of tags, ambient RF noise, tag sensitivity variations, readers' locations, etc. These factors can weaken, scatter, or occlude radio waves, and thus lead to unreliable detection and inaccurate positioning of objects [4] [5].

Several RFID-based localization techniques have been proposed, either focusing on mobile objects (e.g., a robot) or stationary objects (e.g., a wallet) [6] [7] [12] [14]. However, previous techniques tend to sacrifice speed and accuracy in localizing objects in order to obtain reliable estimates (i.e., repeated measurements should consistently yield the same outcome). Unfortunately, these resulting speed and accuracy degradations tend to reduce the efficacy of client applications.

We propose a localization framework that enables accurate object position estimation, without compromising either speed or reliability. Our localization method varies the power levels of the readers, calibrated against a set of reference tags of known sensitivity, to accurately estimate target tag positions in a region of interest. Although we initially tested this methodology indoors to localize stationary objects, our framework is quite general and can be applied to many other scenarios, including outdoor environments, 3D localization, moving objects, various tag types, different combinations of tags, antennas and readers, etc. Our framework is highly scalable and can accommodate a wide range of requirements and tradeoffs among power, cost, accuracy and speed.

We implemented, tested and evaluated the proposed framework, and experimentally confirmed its accuracy, speed and reliability in localizing objects. In order to ensure high reliability and accuracy in localization, our methodology addresses various practical issues such as "binning" the calibrated tags according to their detection sensitivities, which can vary significantly even among "identical" tags (due to manufacturing variability).

This paper is organized as follows. In Section II, we describe the proposed localization framework. We present several localization algorithms and heuristics in Section III. We experimentally evaluate the proposed framework in Section IV, and conclude in Section V with extensions and future directions.

II. The Localization Framework

The proposed localization method is based on continuously varying the power levels of the RFID readers in order to infer distance and position information about target tags. We use reference tags at known locations to help calibrate the power vs. distance relationships, and we employ several readers in order to reduce the localization uncertainty when inferring the position of target tags, as illustrated in Figure 1.

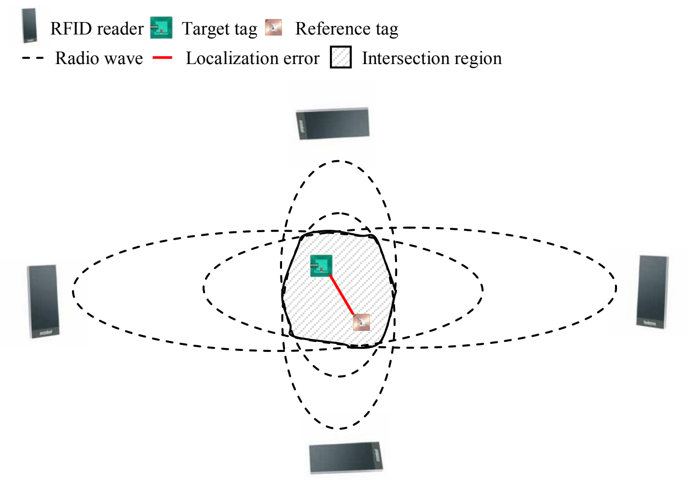

Figure 1. Working principle of the proposed localization method

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

Figure 1 depicts the intersection region covered by the lobes of radio waves emitted by different readers. Based on the relative power level that is necessary for a reader to detect a target tag, we can infer the distance between that tag and the reader. Moreover, several such power-distance correlations obtained from different readers can help localize a target tag with greater precision.

The reference tags serve as a practical mechanism used to initially calibrate the power vs. distance relationships, in order to avoid relying on possibly erroneous formulas, unpredictable environmental conditions, etc. This constitutes a "feedback mechanism" that enables our system to dynamically adapt to unknown variables (e.g., noise, occlusions, interferences, etc.) that may adversely affect tag readability and localization.

While the use of reference tags ascertains the actual power-distance relationships, it may also introduce errors in position estimates of target tags. When target tags are detected by varying the reader power levels, positions of the reference tags detected at the same power-level are used to infer (by interpolation) the position of target tags. This is a source of possible localization error, as depicted in above illustration. We apply different heuristics to minimize this error, based on the minimum reader power levels necessary to detect reference and target tags, as detailed in the next section.

III. Algorithms and Heuristics

We now describe three localization algorithms that incorporate the basic principles of the proposed localization framework, discuss possible sources of localization error, and present heuristics to minimize the error. The proposed localization method uses varying reader power levels to infer the position of target tags. We give three localization algorithms that control this key parameter (i.e., reader power level) in different ways in order to establish tradeoffs between accuracy and speed, as described below.

A. Localization Algorithms

In the first localization algorithm, we linearly increment the reader power level to determine the minimum power level at which reference (and therefore target) tags are detected. The variable *Power_Step* determines the size of the power level increment. The convergence time for the algorithm to find the minimum power level for tag detection is dependent on this *Power_Step* variable (i.e., the smaller this step size, the longer it may take to reach the desired detection threshold, but could yield greater localization accuracy). For example, if power level is varied between 0 and 33 dBm, and the *Power_Step* is 0.25 dbm, then this algorithm will iterate up to (33 / 0.25) + 1 = 133 times to ascertain the minimum detection power level.

The algorithm varies the reader power level from lowest to highest to determine a minimum tag detection power level (other possible power varying strategies will be discussed later). While this approach finds the minimum detection power levels, it may require too long to converge. Optionally, we can instead vary the power level from highest to lowest, since tags are not typically located very near the reader, but rather are often found closer to the far end of the reader detection range. Thus, stepping the power level down instead of up will tend to reduce the average number of iterations to determine the minimum detection power level.

```
Input: Tag_ID, Power_Step, Direction_Flag
Output: Minimum detection power level

if (Direction_Flag = LOW_TO_HIGH) then
    Power = MIN_POWER_LEVEL
    repeat
        if (Power > MAX_POWER_LEVEL) then
            return NOT_FOUND
        end
        Set reader power-level to Power
        Search for tags until successful or time-out
        if Tag_ID is found then
            return Power
        end
        Power = Power + Power_Step
    end
else
    Power = MAX_POWER_LEVEL
    Found_Power = NOT_FOUND
    repeat
        if (Power < 0) then
            return NOT_FOUND
        end
        Set reader power-level to Power
        Search for tags until successful or time-out
        if Tag_ID is found then
            Found_Power = Power
        else
            return Found_Power
        end
        Power = Power  – Power_Step
    end
end
```

Figure 2. Algorithm I: Linear search for the minimum power-level

Figure 2 describes this algorithm, called "Algorithm I". The algorithm takes as input a unique tag id (*Tag_ID*), power step (*Power_Step*), and increment direction flag (*Direction_Flag*), and returns the minimum reader power level at which that tag becomes detectable. The time this algorithm requires to process a tag is linearly proportional to the number of distinct power levels used during the search. Thus, to process N tags using P power levels, this algorithm will run within time $O(N \cdot P)$ in the worst case.

The overall running time can be further reduced by using a *binary search* on the power level instead of a linear search. This will enable a faster convergence on the minimum detection power level, requiring at most $O(N \cdot \log P)$ steps to process N tags with a resolution of P power levels. We call this binary–search based approach "Algorithm II".

Another efficiency optimization leverages the capability of an RFID reader to simultaneously detect a large number of tags during the same read cycle. Therefore, instead of invoking Algorithm I separately for each tag ID, we can have it determine at each iteration *all the tags* that are detectable at that power level, and separately update the status of each one.

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

Note that this is logically equivalent to running Algorithm I *in parallel* independently for each tag. Assuming that the number of tags does not exceed the maximum simultaneous tag reading capacity of the reader, this strategy will require O(P) steps using a resolution of P power levels, independently of the number of tags. We call this parallel–based approach "Algorithm III".

There are several sources of possible "localization errors", including the "round off" error inherent in identifying a target tag with the "nearest" reference tag, as well as the errors inherent in the algorithms for estimating the minimum detection power level. We next discuss these errors and outline techniques to mitigate them.

B. Localization Error Mitigation Heuristics

Apart from the errors discussed above, other factors that contribute to localization errors include variability in tag sensitivity and environmental interferences [5]. In Section IV, we discuss the impact of variability in tag sensitivity on localization errors, and suggest practical methods to reduce it. We now present eleven heuristics for mitigating localization errors, grouped into four broad categories as follows.

1) Absolute Difference: This heuristic takes into account the absolute difference between the minimum detection power levels for the neighbouring reference tags and the target tags. We suggest four heuristic variations of this type:

$$H_1: \underset{\forall J}{\mathrm{Min}}\left(\sum_{I=1}^{M} \Delta_I(R_J)\right) \tag{1}$$

$$H_2: \underset{\substack{\forall J,K\\ J\neq K}}{\mathrm{Min}}\left(\sum_{I=1}^{M} \Delta_I(R_J) + \sum_{I=1}^{M} \Delta_I(R_K)\right) \tag{2}$$

$$H_3: \underset{\substack{\forall J,K\\ J\neq K}}{\mathrm{Min}}\left(\sum_{I=1}^{M} \Delta_I(R_J) + \sum_{I=1}^{M} \Delta_I(R_K)\right) \tag{3}$$

J,K are neighbors

$$H_4: \underset{\substack{\forall J,K\\ J\neq K}}{\mathrm{Min}}\left(\sum_{I=1}^{M} \Delta_I(R_J) + \sum_{I=1}^{M} \Delta_I(R_K)\right);\ \sum_{I=1}^{M} \Delta_I(R_J) < \sum_{I=1}^{M} \Delta_I(R_K) \tag{4}$$

J,K are neighbors

2) Minimum Power Reader Selection: This heuristic employs the minimum detection power levels from two (orthogonal) readers to compute the absolute difference between the power levels of the neighbouring reference and target tags. Two such heuristic variations are given as follows:

$$H_9: \underset{\substack{\forall J,K,S,Q\\ J\neq K\\ S\neq Q}}{\mathrm{Min}}\left(\Delta_J(T) + \Delta_K(T)\right);\ \underset{\forall S,J}{\mathrm{Min}}\left(\Delta_J(R_S)\right),\ \underset{\forall Q,K}{\mathrm{Min}}\left(\Delta_K(R_Q)\right) \tag{5}$$

$$H_{10}: \underset{\substack{\forall J,K,S,Q\\ J\neq K\\ S\neq Q}}{\mathrm{Min}}\left(\Delta_J(T) + \Delta_K(T)\right);\ \underset{\forall S,J}{\mathrm{Min}}\left(\Delta_J(R_S)\right),\ \underset{\forall Q,K}{\mathrm{Min}}\left(\Delta_K(R_Q)\right) \tag{6}$$

S,Q are neighbors

3) Root Sum Square Absolute Difference: In these heuristics, we compute the square root of the sum of squares of the absolute difference between the minimum detection power levels of the neighbouring reference and target tags. The following heuristic variations are based on this approach:

$$H_5: \underset{\forall J}{\mathrm{Min}}\left(\sqrt{\sum_{I=1}^{M} \Delta_I(R_J)^2}\right) \tag{7}$$

$$H_6: \underset{\substack{\forall J,K\\ J\neq K}}{\mathrm{Min}}\left(\sqrt{\sum_{I=1}^{M} \Delta_I(R_J)^2} + \sqrt{\sum_{I=1}^{M} \Delta_I(R_K)^2}\right) \tag{8}$$

$$H_7: \underset{\substack{\forall J,K\\ J\neq K}}{\mathrm{Min}}\left(\sqrt{\sum_{I=1}^{M} \Delta_I(R_J)^2} + \sqrt{\sum_{I=1}^{M} \Delta_I(R_K)^2}\right) \tag{9}$$

J,K are neighbors

$$H_8: \underset{\substack{\forall J,K\\ J\neq K}}{\mathrm{Min}}\left(\sqrt{\sum_{I=1}^{M} \Delta_I(R_J)^2} + \sqrt{\sum_{I=1}^{M} \Delta_I(R_K)^2}\right);\ \sqrt{\sum_{I=1}^{M} \Delta_I(R_J)^2} < \sqrt{\sum_{I=1}^{M} \Delta_I(R_K)^2} \tag{10}$$

J,K are neighbors

4) All Heuristics Minimum: This "meta-heuristic" computes for a given target tag the minimum of all the other heuristics, as follows:

$$H_{11}: \underset{\forall L}{\mathrm{Min}}(H_L) \tag{11}$$

Where the following notation glossary applies to all of the above heuristics:

T = Target tag
R_I = Reference tag I
H = Heuristic
Power = Minimum detection power level for a tag
M = Number of readers
$\Delta_I(R) = |Power(T) - Power(R)|$
S, Q, J, K= Iteration variables for neighbourhood tags
I = Iteration variable for unmarked tag
L = Heuristic iteration variable

The above positioning heuristics are used as a post-processing step in our localization algorithm, once the minimum detection power levels of the reference and target tags have been determined. By employing different combinations of localization algorithms and positioning heuristics, a desired level of accuracy can be achieved.

A key feature of the proposed framework is the flexibility to incorporate new localization algorithms and heuristics that may be developed in the future, which can enable the framework to localize objects with higher accuracy and speed.

IV. EXPERIMENTAL EVALUATION

In this section, we present our experimental evaluation methodology, report results regarding tag sensitivity, localization accuracy and speed, and compare the overall approach to existing techniques.

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

A. Experimental Setup

We evaluated the proposed localization framework to localize stationary objects in an indoor environment using one reader connected to four antennas. Our goals for this evaluation were to first classify the tags based on their detection sensitivity (i.e., "binning" them by quality), then ascertain the localization accuracy and speed of the proposed method, and finally compare the overall performance with existing localization techniques. Table I details the experimental setup used in our experiments.

TABLE I
EXPERIMENTAL SETUP DETAILS

Type	Technology Parameters			
Workstation	CPU	*AMD Athlon 64 @ 2 GHz*	OS	*WinXP*
	RAM	*1 GBytes*	Prog. Support	*C++/C#*
	Hard Disk	*100 GBytes*	API	*M4 LIB*
RFID Backend	Reader	*ThingMagic M4*	Protocol	*EPC Gen2*
	Antenna	*Linear*	Readers	*1*
Environment	Sector Map Area	*6 square meters*	Antennas	*4*
	Room Volume	*41 cubic meters*	References Tags	*32*
Tags	Type	*EPC Gen2 UHF passive tags (96 bit)*	Model	*Impinj "Dogbone Monza 3"* 93×23mm

Our experiment was deployed in a rectangular region having an area of 6 square meters (2m × 3m). This region was divided into eight equal sub-regions called "sectors", each having an area of 0.75 square meters (1m × 0.75m). Furthermore, we divide each sector into four equal-sized sub-sectors called "quadrants", each having an area of 0.19 square meters (0.5m × 0.375m), as shown in Figure 3.

One reference tag was placed in each quadrant, with a total of 32 reference tags evenly distributed throughout the entire region. The tag type we used was an EPC Gen2 96-bit UHF passive tag, model "Dogbone Monza 3", manufactured by Impinj, Inc.

B. Binning Tags According to their Sensitivity

Manufacturing variability can dramatically affect the detection sensitivity of tags (i.e., the minimum reader power-level needed to successfully read a tag at a given location). In fact, a small fraction of any commercially obtained batch of tags are typically even "dead" altogether. The accuracy of our localization methodology depends on the uniformity of the detection sensitivities across all tags, while the localization speed will increase with higher tag sensitivities. As a pre-processing quality-control check, we therefore tested and characterized the sensitivities of all the tags, to ensure that only tags with similar (and high) sensitivities are used in our localization experiments.

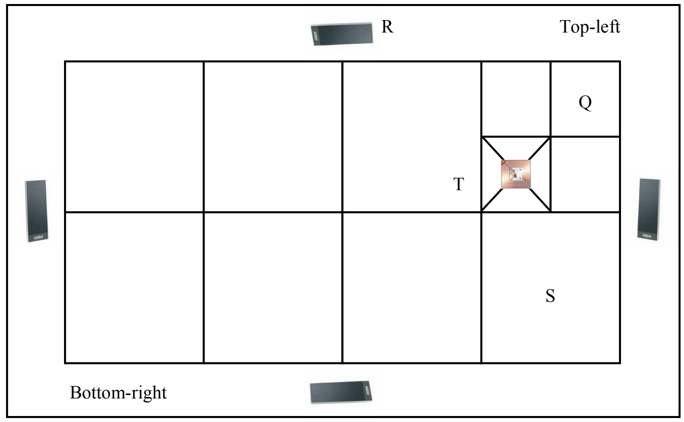

Figure 3. The experimental region with sectors (S), quadrants (Q), reference tags (T), and reader antennas (R)

Our experimental evaluation showed that tag sensitivity varied considerably across a group of 243 tags of the same type. We have characterized the tag sensitivities based on the read counts using different reader power levels. Thus, given a reader power level, if a tag has low read counts among its peers, we call it "non-sensitive". Similarly, tags with high read counts relative to their peers are labelled as "highly sensitive", while tags having equal read count are called "equally sensitive".

We have performed two experiments to quantify tag sensitivities by varying the power levels and distances between the readers and the tags. While these experiments used EPC Gen2 passive tags, our "tag binning" approach is equally applicable to other types of tags. We now describe these sensitivity analysis experiments in detail below.

1) Constant Distance / Variable Power: In this experiment, a batch of four tags was positioned at a distance of 2.5 meters from the reader's antenna, while the reader power level was varied from 25.6 dBm to 31.6 dBm, in steps of 3 dBm. We recorded the cumulative read counts of each tag for 60 seconds (3 read iterations lasting 20 seconds per iteration).

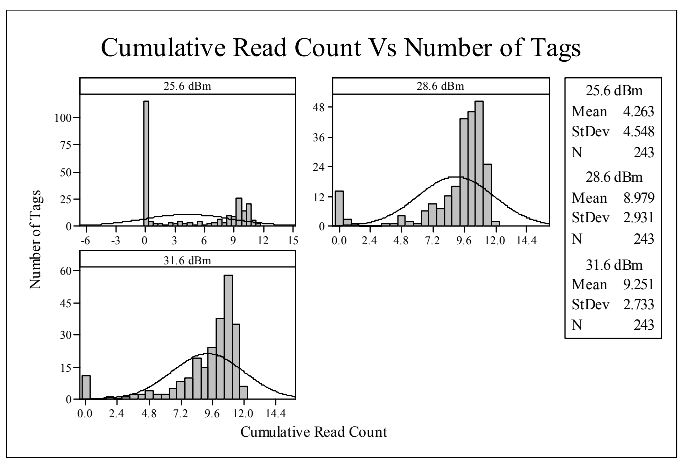

Figure 4. Tag senstivity measurements for constant distance / variable power

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

Figure 4 shows that 114 out of 243 tags had cumulative read counts of zero at 25.6 dBm, with most of the tags having read counts in the range of 3 to 9 (with some tags having read counts as high as 12). Moreover, at a reader power level of 28.6 dBm, most of the tags had cumulative read counts in the range 6 to 12. Finally, at 31.6 dBm, the cumulative read counts all ranged between 5 and 12. Tags were labelled as non-sensitive if they had zero cumulative read counts at a power level of 25.6 dBm. Tags were labelled as non-sensitive at 28.6 dBm only if they were also labelled as non-sensitive at 25.6 dBm. Similarly, we labelled tags as highly-sensitive at 25.6 dBm only if they were labelled as highly-sensitive at 31.6 dBm.

Using this process, 89 out of 243 tags were marked as highly-sensitive, 133 tags as equally-sensitive, and the remaining tags were considered to be non-sensitive. Thus, this experiment classified all 243 tags into three sensitivity categories, based on reader power levels needed for detection.

2) Variable Distance / Constant Power: In the second tag sensitivity experiment, we fixed the reader power level to 31.6 dBm and varied the distance between the tags and the reader from 1.27 meters to 3.81 meters, in steps of 1.27 meters. We labelled tags as non-sensitive if they had low read counts at 1.27 meters. Tags were labelled as non-sensitive at 2.54 meters only if they were also labelled as non-sensitive at 1.27 meters. Similarly, we labelled tags as highly-sensitive at 1.27 meters only if they were also labelled as highly-sensitive at 3.81 meters.

This approach classified 61 out of the 243 tags as non-sensitive, 161 tags as equally-sensitive, and 21 tags as highly-sensitive, based on the minimum detection distances between the tags and the reader. Figure 5 gives the distribution of the cumulative read counts of the tags, taken over the three testing distances, for a duration of 60 seconds each.

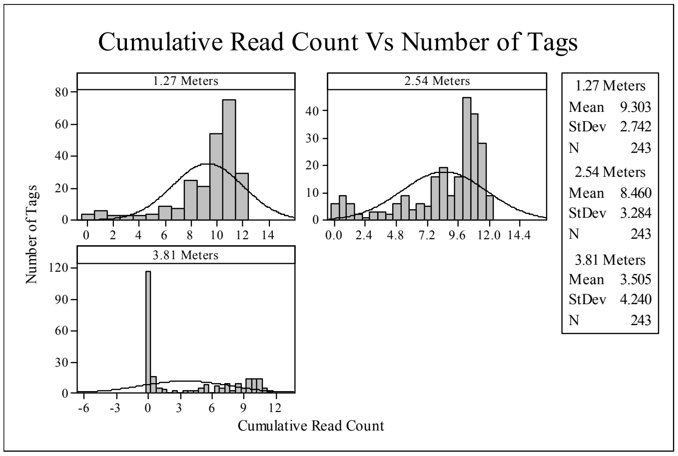

Figure 5. Tag senstivity measurements for variable distance / constant power

Based on the combined outcomes of these two sensitivity experiments, we classified 133 tags as equally-sensitive (i.e., by taking the intersection of the equally-sensitive tag sets from each experiment). In our ensuing localization experiments, we selected all reference and target tags from this equally-sensitive tag set.

C. Localization Accuracy and Speed

We measured localization accuracy by determining the effect of the parameter *Power_Step* on the minimum detection power levels. This is accomplished by determining for a given target tag, the minimum detection power levels over different power steps. These measurements are given below.

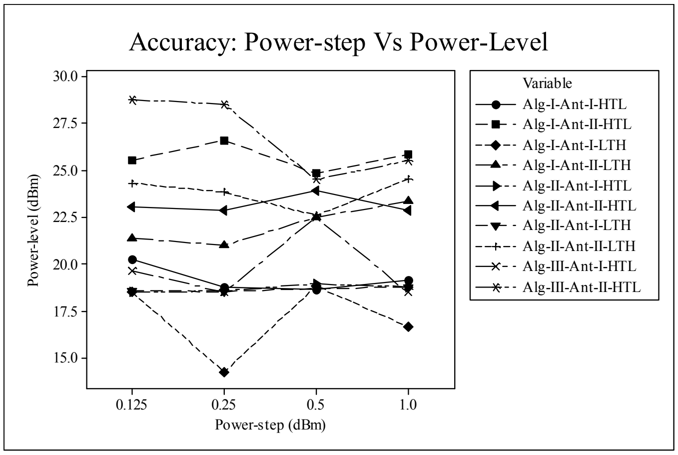

Figure 6. Power level comparison for algorithms I, II, and III

Figure 6 gives the minimum detection power levels of a tag for four different power steps, measured using the three localization algorithms using two orthogonally placed antennas. Algorithm I (in low-to-high *LTH* mode) reports the lowest minimum detection power level, while Algorithm III (in high-to-low *HTL* mode) yields the highest minimum detection power level for the same tag for all the algorithms and power steps. Since localization accuracy is based on determining minimum detection power levels, the Algorithms I, II, and III are able to trade off accuracy and speed.

The time required for localization is heavily dependent on the time required to detect tags. Figure 7 gives the time required to detect tags placed at eight random locations in the region for all three algorithms (using two orthogonal reader antennas). The data confirms our hypothesis that varying the power levels from high to low is typically more efficient for localizing tags farther away from the reader.

While Algorithm II consistently requires less time to find tags, it yields sub-optimal minimum detection power level estimates, due to the coarser granularity of the binary search as compared to the linear search of Algorithm I. Also, Algorithm III requires the smallest search time to find tags, unless the tags are placed very near to the antennas, which then enables Algorithm I to find them more quickly.

Thus, by combining different algorithms, we can choose appropriate application-driven tradeoffs between localization accuracy and localization speed.

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.