Background -- Localization

• Data usually combined with locations
  – Fire alarm, target tracking
• Traditional GPS
  – Expensive; does not work indoors
• GPS-less localization techniques
  – AHLoS, APS-AoA, DV-Hop, Centroid, APIT, etc.

Attacks against Localization

• Challenges in defending these attacks
  – Resource constraints on sensor nodes
  – Lack of physical protection
  – Local collaboration v.s. global threat
  – Difficulty of authenticating beacon signals
Range-Based Localization

- A few beacon nodes with known locations.
- Two phases:
  - Phase 1: Estimating distance (RSSI, TDoA, or ToA)
  - Phase 2: Solving equations by using MMSE

\[
\begin{align*}
    &= d_x^2 - d_y^2 + (x_y^2 - y_x^2) \\
    &= d_x^2 - d_y^2 + (x_y^2 - y_x^2) \\
    &= d_x^2 - d_y^2 + (x_y^2 - y_x^2)
\end{align*}
\]

\[\min F = f_1^2 + f_2^2 + f_3^2\]

A, B, C: beacon nodes

Impact of Malicious Attacks

- Obtained through simulation
- MMSE with 1 malicious beacon signal + 9 benign beacon signals
- A single malicious signal => arbitrarily large location error

Attack-Resistant Location Discovery

- Goal
  - Resilient location estimation when there are malicious location references
- Our approaches
  - Attack-resistant MMSE: identify “inconsistency” among malicious and benign beacon signals
  - Voting-based scheme: have each location reference vote on the location of the non-beacon node.
Assumptions

• Use a key management protocol that provides a unique pair-wise key between any two nodes.
  – E.g., TinyKeyMan
• This implies
  – Each sensor node is uniquely identified
  – Beacon packets can be authenticated
    • The content, not the signal

Assumptions (Cont’d)

• Each sensor node uses at most one beacon signal from each beacon node
  – Represented as a location reference \(< x_i, y_i, \delta_i >\)
  – Location of the beacon node and the measured distance.
• Attacker model
  – A malicious beacon node can provide arbitrary location references

Attack-Resistant MMSE

• Observation: there is “inconsistency” between benign and malicious location references
• Intuition: identify the most inconsistent location references before final estimation
• Consistency metric (\(\zeta_i^2\)): mean square error of distance measurement

\[
\zeta_i^2 = \sum_{m} \left[ \delta_i - \sqrt{(\tilde{x}_i - x_i)^2 + (\tilde{y}_i - y_i)^2} \right]^2 \\
\leq \tau^2
\]
Attack-Resistant MMSE (Cont’d)

- Ideally, get the **largest consistent** set of location references
  - MMSE can achieve more accurate result with more benign location references
- **What we have**: check consistency, given a set of location references and a pre-defined threshold $r$
  - If $\varsigma^2 > r^2 \rightarrow$ inconsistent; otherwise, consistent
- **Two remaining questions**
  - How to determine the largest consistent set
  - How to set an appropriate threshold

---

Determining the Largest Consistent Set

- **A simple solution**
  - Try every combination of location references
  - Expensive: 10 location references, and 5 of them in the largest consistent set $\rightarrow$ at least 387 MMSE operations
- **Greedy algorithm**
  - Multiple rounds
  - Remove the most inconsistent location reference in each round
  - Not guaranteed to find the largest consistent set

---

Greedy Algorithm

A set of $m$ location references and a predefined threshold $r$

- **Consistency Test**
  - Find consistent set and output result
  - Fail to find consistent set

- $i > 3 \rightarrow$ fail to find consistent set
- $5$ location references, and $5$ of them in the largest consistent set $\rightarrow$ 50 MMSE operations on average
Threshold $\tau$

- Investigate the distribution of MSE $\varsigma^2$ when there is no malicious attack
- If the measurement errors are independent, we have

$$\lim_{n \to \infty} F[\varsigma^2 \leq \varsigma_0^2] = \Phi\left(\frac{m\varsigma^2 - \mu'}{\sigma'}\right)$$

where $\mu_i$ and $\sigma_i$ are the mean and variance of $e_i^2$, and

$$\mu' = \sum_{i} \mu_i, \sigma' = \sqrt{\sum \sigma_i^2}$$

Theoretical Results v.s. Simulation Results

The threshold should not be too small or too large.

Voting-Based Scheme

- Partition the target field into grid with $M$ small squares (cells)
- Each location reference votes on the possible locations of node
- Identify the cell (or cells) with the largest vote
Overlap Test

- No overlap between the cell and the ring \( iif \)
  - The maximum distance from A to a point in the cell \( d_{\text{max}}(A) < \max(0, \delta - \epsilon) \), or
  - The minimum distance from A to a point in the cell \( d_{\text{min}}(A) > \delta + \epsilon \).

Granularity \( M \)

- Fine granularity (large \( M \)) results in high accuracy but high computation and storage cost,
- Coarse granularity (small \( M \)) results in low accuracy but low computation and storage cost.

Iterative Refinement

- Idea
  - Repeat the basic voting algorithm on the result of the last voting round
- Stop conditions
  - Achieve the required accuracy (size of cells)
  - Size of the cell cannot be reduced anymore
- We use the second stop condition in our experiments
Simulation Evaluation

- Evaluate the ability of the proposed methods to tolerate malicious attacks
- Three attack scenarios
  - One malicious location reference \((9 + 1)\)
  - Multiple non-colluding malicious location references \((9 + 3)\)
  - Multiple colluding malicious location references \((9 + 3)\)
- Configuration:
  - 30m X 30m target field
  - Radio signal range 22m
  - Distance error evenly distributed in \((-4, 4)\)

Evaluation of Attack-Resistant MMSE

![Graph showing location error vs. location estimation error for different scenarios.]

Evaluation of Voting-Based Scheme

![Graph showing location error vs. location estimation error for different scenarios.]
Comparison

Due to the non-optimal solution given by greedy algorithm.

Implementation

• Target at MICA2 motes running TinyOS

<table>
<thead>
<tr>
<th>Code Size (byte)</th>
<th>ROM</th>
<th>RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMSE</td>
<td>2,034</td>
<td>286</td>
</tr>
<tr>
<td>AR-MMSE</td>
<td>3,226</td>
<td>396</td>
</tr>
<tr>
<td>Voting-Based</td>
<td>4,488</td>
<td>174</td>
</tr>
</tbody>
</table>

Execution Time

1 malicious location reference $e = 4, v = 10$
Field Experiment

Use RSSI to measure distance
\[ f = 4 \text{ feet} \]

\[ \epsilon = 4 \text{ feet} \]

1 Malicious Beacon

Location error created by malicious beacon

Location estimation error

MMSE
AR-MMSE
Voting

3 Non-Colluding Malicious Beacons

Location error created by malicious beacon

Location estimation error
3 Colluding Malicious Beacons

![Graph showing location error created by malicious beacon]

Conclusion

- We have been investigating various techniques to secure localization in sensor networks
  - Prevention
  - Tolerance
  - Detection and response
- Future work
  - Light-weighted secure and resilient solutions
  - Secure and resilient localization for dynamic sensor networks