## A Probabilistic Approach to Location Verification in Wireless Sensor

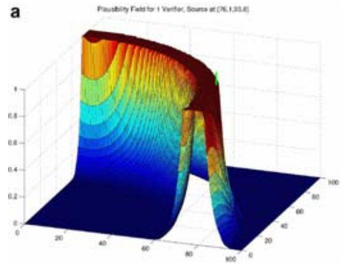
**Ekici, E.; Mcnair, J.; Al-Abri, D., "A Probabilistic Approach to Location Verification in Wireless Sensor Networks," Communications, 2006. ICC '06. IEEE International Conference on , vol.8, no., pp.3485-3490, June 2006.**

**Presented by Matthew Tan Creti and DongHoon Shin**



## Probabilistic Location Verification Overview

- **Goal is to provide the *plausibility* that a node is at the location it claims**
- **A claimant node will broadcasts its location**
- **Verifier nodes observe the hop-count and determine the probability of that hop-count occurring given the claimed Euclidean distance between the nodes**
- **Verifier nodes combine their results to calculate the plausibility that the claimed location is correct**



## Assumptions

- All sensor nodes perform localization using some non-secure method
- Small number of malicious nodes
- Small number of verifier nodes that know their exact location
- Verifiers are secure and cannot be compromised
- Malicious nodes have the same hardware as sensor nodes



## CDF of k-hop distance

- $k$ = observed hop count
- $\bar{r}$ = average 1-hop distance
- $\bar{r}_k$ = expected value of k-hop distance

$$\bar{r}_k \equiv E[d_k] = k \cdot \bar{r}. \qquad (2)$$

- $\sigma_k^2$ = variance of k-hop distance
- The probability that the k-hop distance $d_k$ is less than distance d given an observed hop count k

$$Pr\{d_k < d \mid K = k\} = \int_{-\infty}^{d} \frac{1}{\sigma_k\sqrt{2\pi}} e^{-\frac{(\delta-\bar{r}_k)^2}{2\sigma_k^2}} d\delta = \frac{1}{2}\left[1 + erf\left(\frac{d-\bar{r}_k}{\sigma_k\sqrt{2}}\right)\right], \qquad (8)$$

## PMF of the Number of Hops

- $(x_i, y_i)$= the claimed location
- $(x_v, y_v)$= the verifiers location

$d = \sqrt{(x_v - x_i)^2 + (x_v - x_i)^2}$

- Using Bayes Theorem and the CDF of k-hop distance we can find the probability that it takes k hops to reach a distance between d-$\varepsilon$ and d+$\varepsilon$

$$
\begin{aligned}
&Pr\{K = k \mid d - \epsilon < d_k \le d + \epsilon\} \\
&= \frac{Pr\{d - \epsilon < d_k \le d + \epsilon \mid K = k\} \cdot Pr\{K = k\}}{Pr\{d - \epsilon < d_k \le d + \epsilon\}} \\
&= \frac{\frac{1}{2}\left[erf\left(\frac{d+\epsilon-\bar{r}_k}{\sigma_k\sqrt{2}}\right) - erf\left(\frac{d-\epsilon-\bar{r}_k}{\sigma_k\sqrt{2}}\right)\right] Pr\{K = k\}}{Pr\{d - \epsilon < d_k \le d + \epsilon\}}. \qquad (9)
\end{aligned}
$$

- The unconditional probabilities can be calculated beforehand and stored in tables



## Using the PMF to Find Trust

- Suppose the PMF is {0.2, 0.3, 0.4, 0.1} for hop counts of {4, 5, 6, 7}, and the observed hop count is $k^* = 5$
- We need some kind of metric that indicates how much we can trust the claim
- The maximum probability of the PMF is

$$P_v^{max}(d) = \max_{n \in N} Pr\{K = n | d - \epsilon < d_k \le d + \epsilon\}, \qquad (12)$$

- The probability slack function is

$$S_v(d, k_v^*) = P_v^{max} - Pr\{K = k_v^* | d - \epsilon < d_k \le d + \epsilon\}$$

- The amount of distrust in the claim is

$$\frac{S_v(d,k_v^*)}{P_v^{max}(d)}$$

- So in the example above our level of distrust is (0.4-0.3)/0.4 = 0.25

## Using Trust to Find Plausibility

- Trust alone is not enough
- What if the PMF at $v_1$ is {0.3, 0.6, 0.1} for hop counts {3, 4, 5} and $v_2$ is {0.1, 0.2, 0.2, 0.2, 0.2, 0.1} for hop counts {3, 4, 5, 6, 7, 8}
- When $k^*$ is 3, the distrust at $v_1$ is (0.6-0.3)/0.6 = 0.5 and the distrust at $v_2$ is (0.2-0.1)/0.2 = 0.5
- Although the trust value is the same for both values, $v_1$ can be much more confident in it answer
- Plausibility incorporates both the trust and confidence of all verifiers

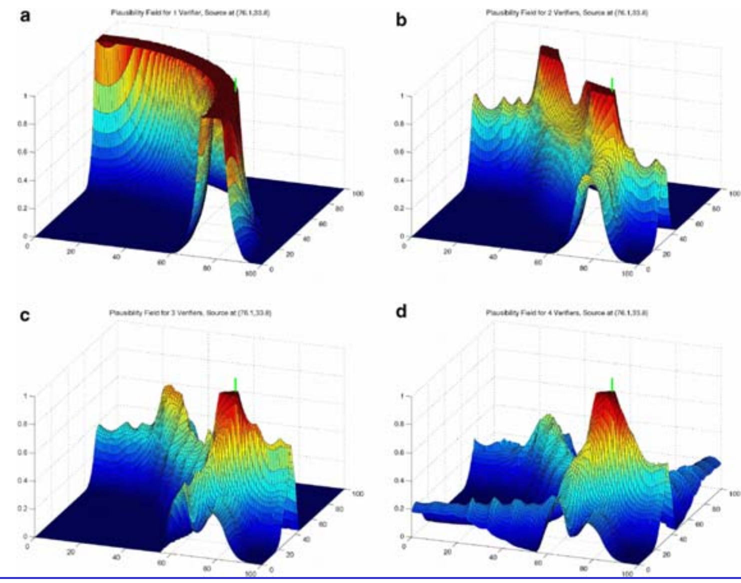
$$\mathcal{P}_i = 1 - \frac{\sum_{j=1}^{\mathcal{V}} \frac{P_j^{max} - Pr\{K=k_j^* | d-\epsilon < d_k \leq d+\epsilon\}}{P_j^{max}} \cdot P_j^{max}}{\sum_{j=1}^{\mathcal{V}} P_j^{max}} = 1 - \frac{\sum_{j=1}^{\mathcal{V}} S_j(d, k_j^*)}{\sum_{j=1}^{\mathcal{V}} P_j^{max}}. \quad (14)$$



## Probabilistic Location Verification (PLV) Algorithm

1. A node i broadcasts its location ($x_i$, $y_i$)
2. Each of the V verifies receive the message over $k_v^*$ hops and computes $d_v$
3. Each V uses $k_v^*$ and $d_v$ compute probability slack $S_v(d, k_v^*)$ and maximum probability $P_v^{max}(d)$
4. The probability slack and maximum probability of all verifiers are collected at a central node and $P_i$ is computed
5. $P_i$ is compared to thresholds that classify its trustworthiness

## Plausibility of Claimed Location for Different Numbers of Verifiers






## Attacks

- Disreputation though Impersonation
  - A malicious node could impersonate a node and send false location information to get the node blacklisted
  - Prevented by encrypting nodes identity and location claim in message
  - A unique symmetric key at every sensor node can be used by verifier nodes to decrypt messages
  - It is more practical to limit the number of keys to $N_k$ where $N_k << N$
  - This means a malicious node m would succeed to disrepute a node i with a probability of $1/N_k$

Thoughts

• The authors do not explain why the node identity needs to be encrypted rather than just signed

• Just one malicious node can disrepute $N/N_k$ of the network!

# Attacks

- Denial of Service through Payload Alterations
  - A malicious node could modify a message so that it cannot be authenticated
  - A verifier will use the altered packet only if the malicious node lies on the shortest path to a verifier
  - One approach is to collect packets for a predetermined time period, if the collected packets do not match then all of them are dropped
  - This will create a hole around the malicious node
- Denial of Service through Hop Count Alterations
  - A malicious node can change the hop count to a small number that will load to a low plausibility and blacklisting of a claimant
  - We assume a low complexity asymmetric key $k_1$
  - Let am intermediate node j receive a packet P
  - Node j forwards the packet after appending X to P and encrypting X+P using $k_1$
  - The hop count of a packet can now be inferred from its packet length
  - This solution is expensive due to using an asymmetric key!



# ROC - Simulation Results

- There is a trade off of probability of false alarm with probability of detection
- As expected 1 verifier performs very poorly
- To get a unique and small plateau in 2D at least 3 verifiers are needed
- An area closer to 1 under the ROC curve is better

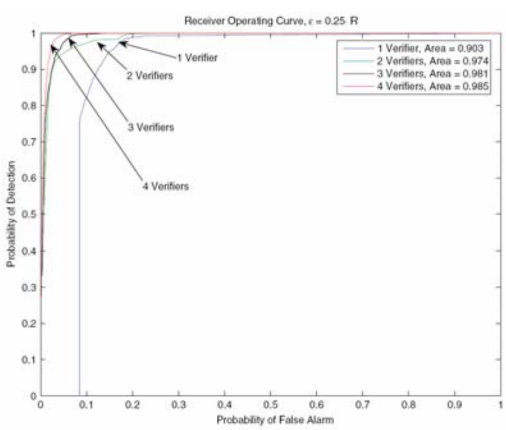



Fig. 3. Receiver operating curve.

## Node Density - Simulation Results

- A higher number of verifiers consistently result in higher classification accuracy
- Changes in node density have less effect on performance as the number of verifiers increases
- Performance decease after a point due to the accuracy of the Gaussian approximation of the distance covered in k hops decreasing

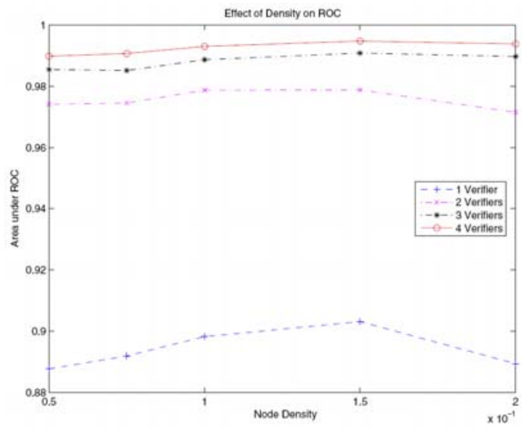



Fig. 4. Effect of node density.



## Verifier Separation - Simulation Results

- When verifiers are separated they make independent estimations of plausibility and performance is better

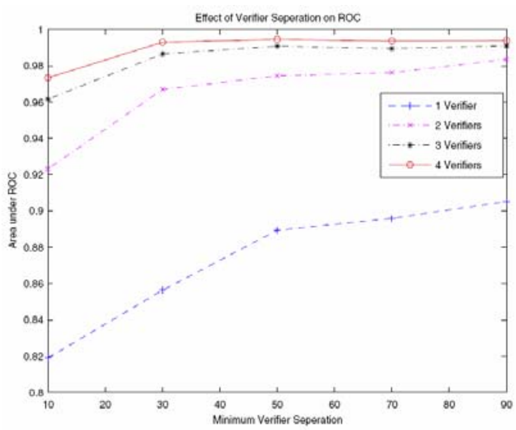



Fig. 5. Effect of minimum verifier separation.

## DOS - Simulation Results

- Malicious nodes create “holes” in the network that decrease performance

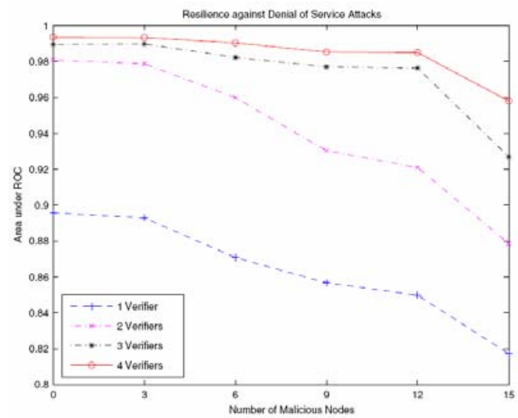



Fig. 6. DoS attacks with payload alterations.



## Extensions

- The security measures are heavy weight (i.e. asymmetric keys) or will perform poorly (i.e. $N_k$ symmetric keys)
- Wormhole attack is ignored
- Verifiers are assumed to be resistant to attacks; could a trust system be constructed that would not require this assumption?