UC San Diego
JACOBS SCHOOL OF ENGINEERING
Computer Science and Engineering

AsiaCCS 2020

Hunting Sybils in Participatory Mobile Consensus-Based Networks

Nickolai Verchok, Alex Orailoğlu

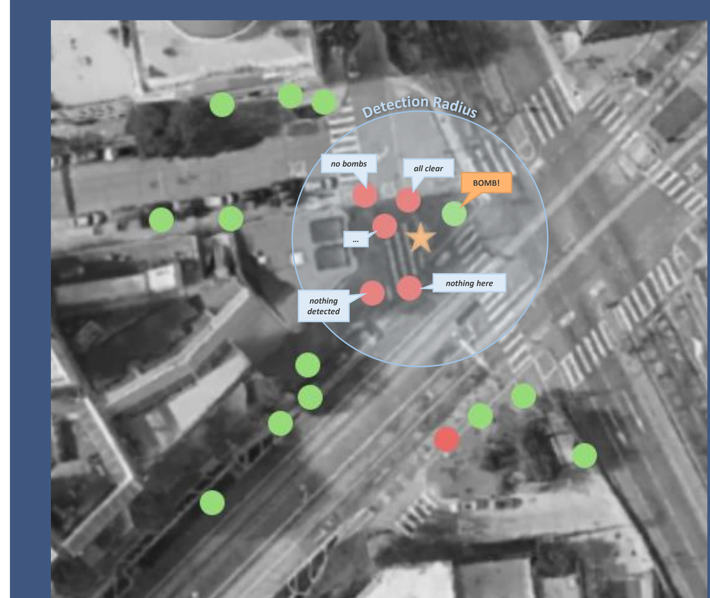
NSF 1739684

What's the Problem?

Participatory Sensing is a promising domain; e.g. Smartphone-mounted bomb detection sensors **distributed to crowds** can passively provide wide coverage

Anyone can Join ⇒ Vulnerable to disinformation

- Adversary can inject imitant forgeries (**Sybils**) into network
- Sybil clusters can easily overpower local decisions and give adversary **unlimited leverage from afar.**

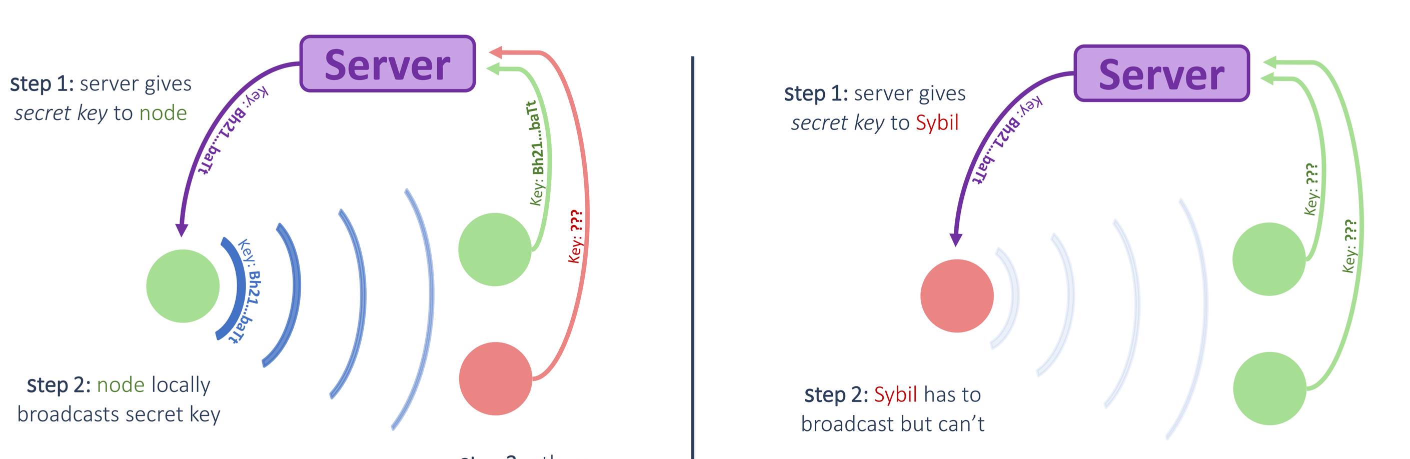


A bomb, honest nodes, and Sybils are shown. The Sybil cluster can claim to see *no bomb* and overpower local honest data, which would be deemed a false positive.

Sybils must be detected and ignored.

Key Idea

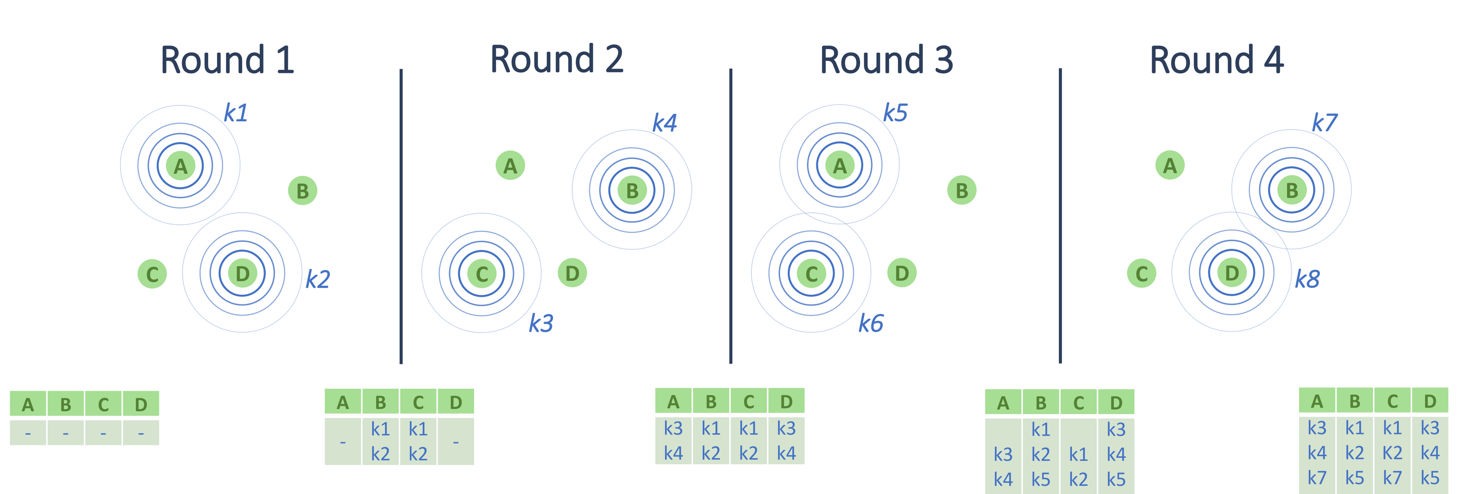
- Force nodes to **communicate locally**: Sybils will be helpless!



Sybils not present; cannot see key

Sybils not present; cannot broadcast

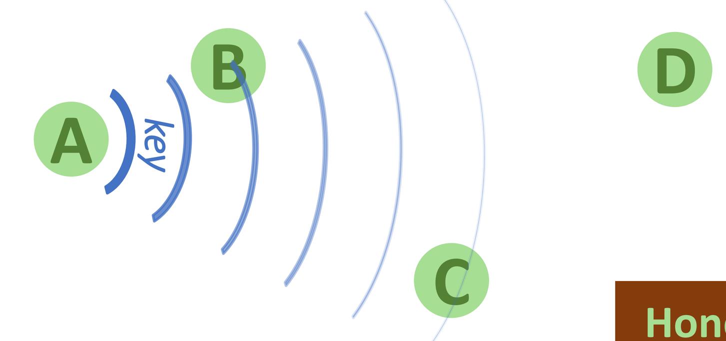
- *No nodes are trusted*, so all nodes forced to communicate with each other in both directions; **all directed pairs formed.** Done in ***logarithmic time*** in **discretized** rounds.



All directed pairs (and possibly more) made in **$2\lceil \log_2(N) \rceil$ rounds**

Challenges

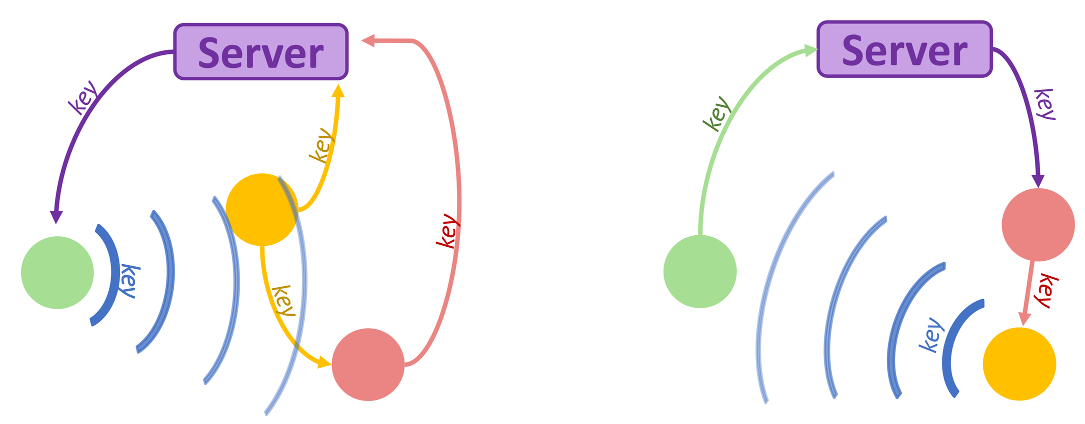
1) Connectivity is **uncertain**, and a function of distance



Pr(B sees *key*) ≈ 1.0
Pr(C sees *key*) ≈ 0.5
Pr(D sees *key*) ≈ 0.0

Honest-honest connections are **not guaranteed**

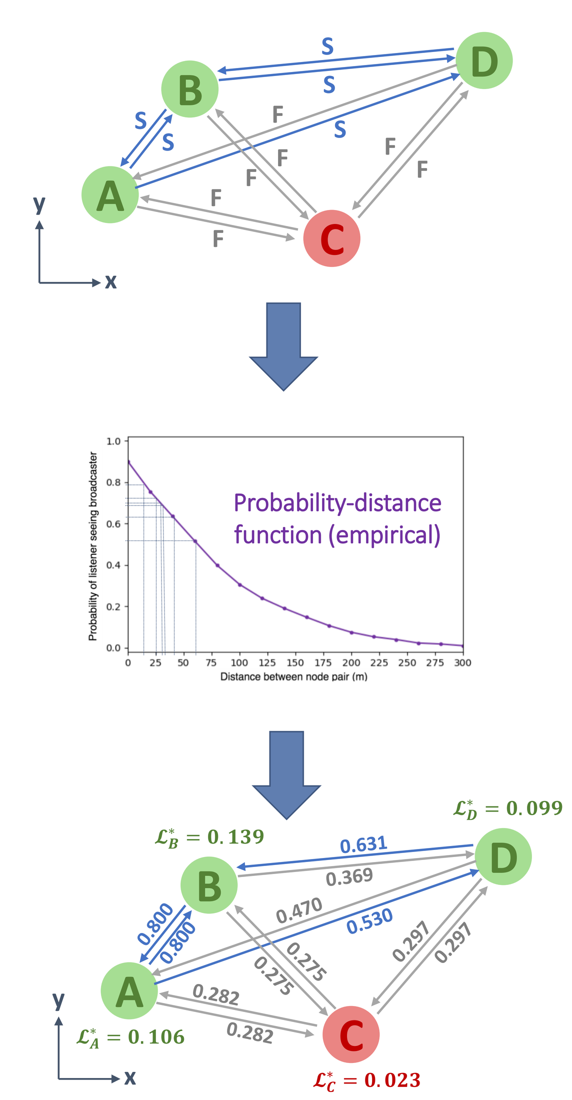
2) Adversary may have malicious nodes (physical agents on-site) who may collude and **pass keys** they hear to Sybils [left]
3) Malicious nodes may also ***broadcast on behalf of Sybils*** [right]



Malicious nodes obscure or fabricate connectivity results

Probabilistic Proximity Graphs

- After the **local communication** algorithm, all connections—*both successful and failed*—are **encoded as directed edges** with initial weights $w \in \{S, F\}$ (source = listener)
- Weights are then **transformed to probabilities** with $w \in (0,1)$, based on pairwise distances and a **probability-distance function**



- For every node i, all **incoming** edge likelihoods are multiplied to yield **total likelihood value $\mathcal{L}_i^*$ for observed combination of edge outcomes**
- **Sybils**, even with malicious node support, obtain **low $\mathcal{L}_i^*$** values and **can be detected**
- However, **scale is relative** and depends on node quantities and positions

∴ Find **distribution of $\mathcal{L}_i$** values and ***test*** the extremeness of $\mathcal{L}_i^*$

Obtaining $\mathcal{L}_i$ Distribution

- For every edge e with $P(e = S) = p_e$, define the **random variable**:

$$X_e = \begin{cases} \log(p_e) & w.p. \quad p_e \\ \log(1-p_e) & w.p. \quad 1-p_e \end{cases}$$

- Can examine **all combinations of $X_{e \in in(i)}$** values to find $\mathcal{L}_i$ **directly** for node i with incoming edges $in(i)$, but this is **exponential**
- We use **linear time** Lyapunov's CLT to get:

$$\sum X_e \sim \mathcal{N}$$

Detecting Sybils

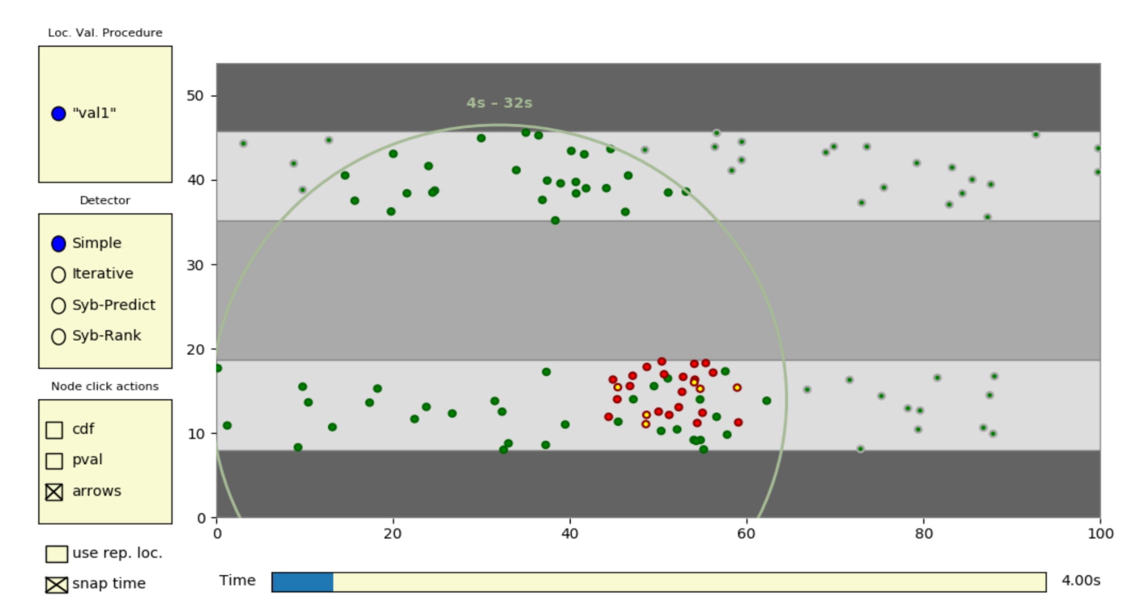
- Choose threshold τ (default $\tau \approx 10^{-6}$)
- Nodes with $P(\mathcal{L}_i < \mathcal{L}_i^*) < \tau$ are candidates
- Now, repeat:
  1) **worst** candidate is a Sybil
  2) discard it with its **outgoing** edges
  3) recalculate τ for remaining nodes

  until no candidates left
- This approach iteratively crumbles Sybil clusters one Sybil at a time

Simulation Analysis

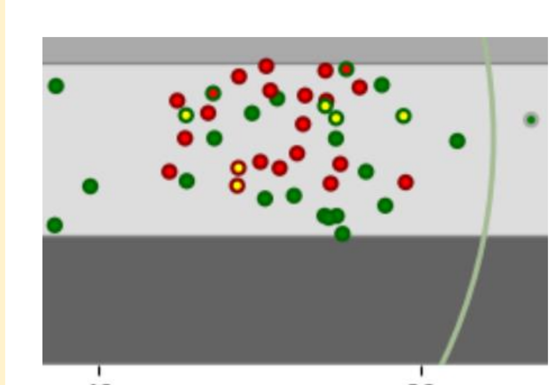
Simulator

- Generates life-scale distribution of nodes
- Simulates connectivity algorithm and successes/failures based on pairwise distance
- Creates Sybil and malicious nodes (physical adversarial agents) and employs evasive tactics
- Runs primary detection algorithm along with Graph-Based methods for comparison

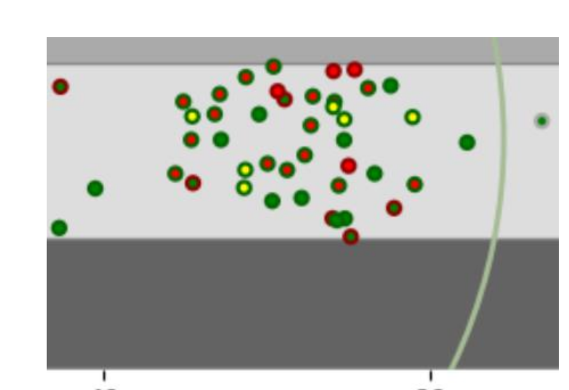


Simulation scenario: 100 honest nodes, 20 Sybil nodes, and 6 Malicious nodes (fill) on Market Street in SF. Large circle defines participants. Selected detection algorithm (*"Simple"*) predicts **real** and **Sybil** nodes (rim).

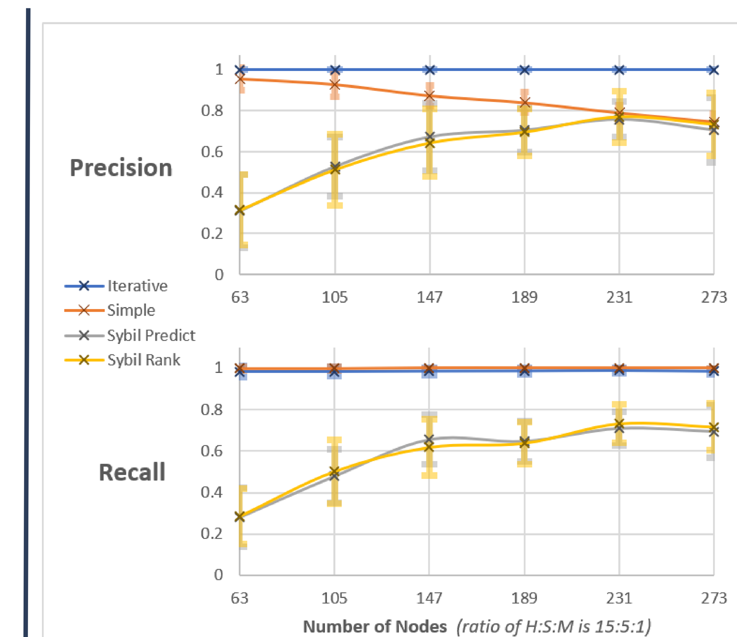
Malicious nodes use evasive tactics (challenges 2 & 3) to integrate Sybils into the proximity graph. This may have **devastating effects** on traditional Graph-Based Detection algorithms, while **our approach remains robust**.



Sybil-Rank Detector. *Performance is* ***worse*** *compared to our approach:* ***2 false negatives*** *compared to 0.*



Sybil-Predict Detector. *Catastrophic output on the same input, highlighting the fragility of Graph-Based Sybil Detection methods.*



Results. *Detection performance across 4 detection algorithms, with* ***Iterative*** *(our approach) clearly dominating in both precision and recall.*

Here the adversary uses evasive strategies to intermix Sybils with honest nodes in the proximity graph.

The most important aspect is the difference in ***variance*** *(shown as error bars), indicating that the Graph-Based Detection algorithms occasionally have* ***catastrophic failures****, as shown in the example on the left.*

The robustness of our approach comes from the smoothness of the likelihood when used as a test statistic.