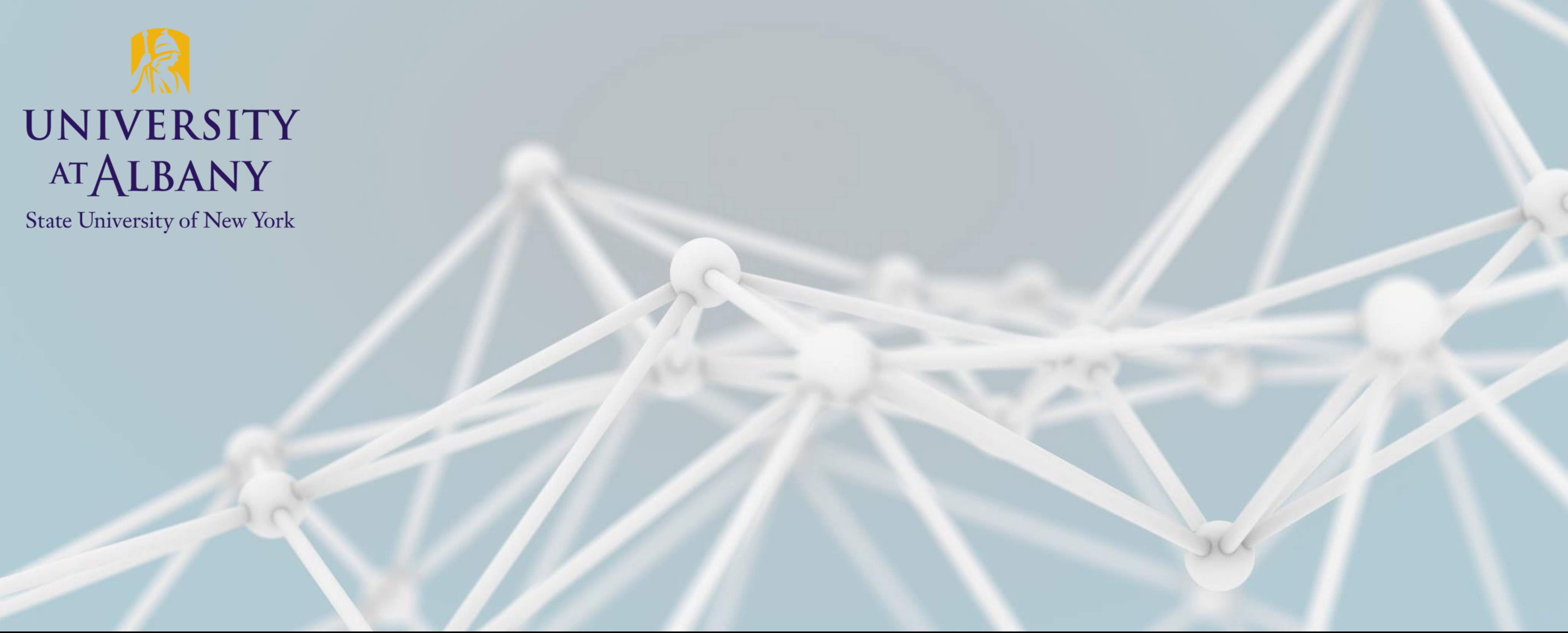
UNIVERSITY AT ALBANY
State University of New York



Autonomous Spectrum Enforcement: A Blockchain Approach

MAQSOOD CAREEM
ADVISER: AVEEK DUTTA
DEPT. OF ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY AT ALBANY, SUNY

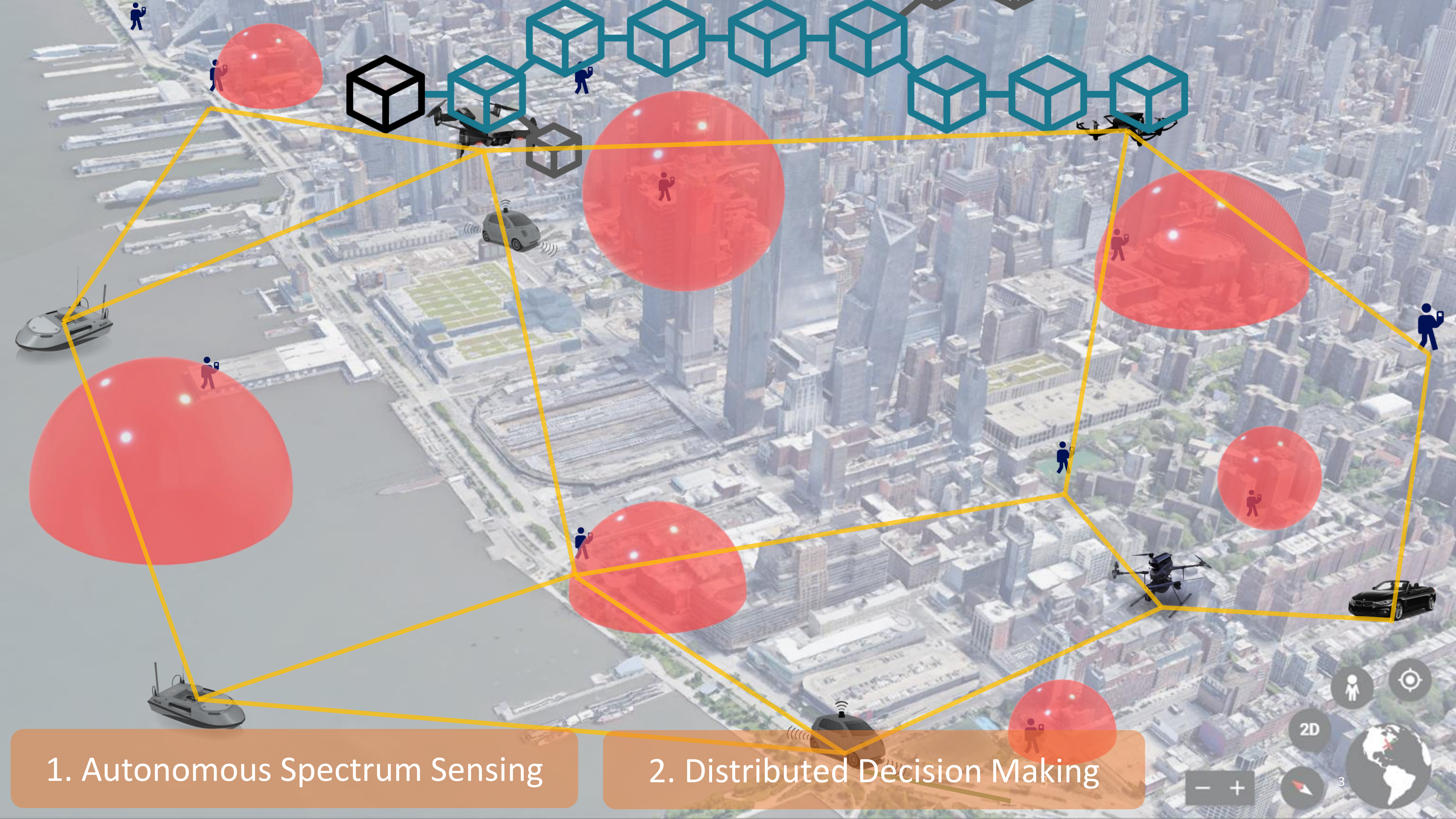
Motivation

- Advent of Spectrum Sharing demands Enforcement of Spectrum policies.
- Dynamic nature of violations necessitate use of Autonomous Agents.

Problem Statement: 1. Requires efficient schedule for multi-modal agents.

2. Requires distributed inferences among trust-less agents

Autonomous Enforcement System:
“Multi-modal agents autonomously sense, make decisions and enforce policies”

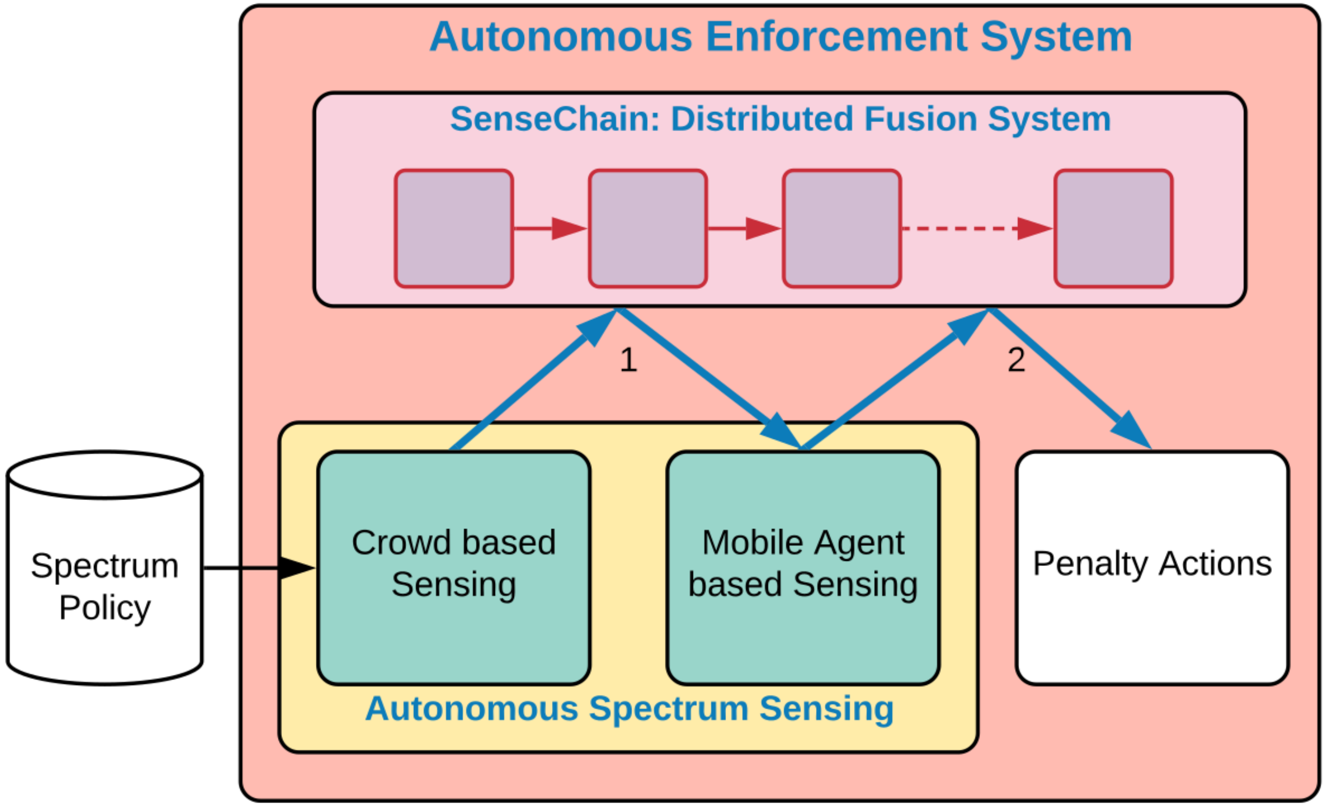

1. Autonomous Spectrum Sensing
2. Distributed Decision Making
2D

Autonomous Enforcement System

The *"Target"* (Violator) :
Entity that violates spectrum policies

The *"Sensors"* :
Agents that sense and detect infractions.

The *"Validators"* :
Agents which make decisions and collect evidence



1. Determine Schedule for Mobile Agents using Crowd measurements
2. Aggregate sensing results to detect violations and estimate locations

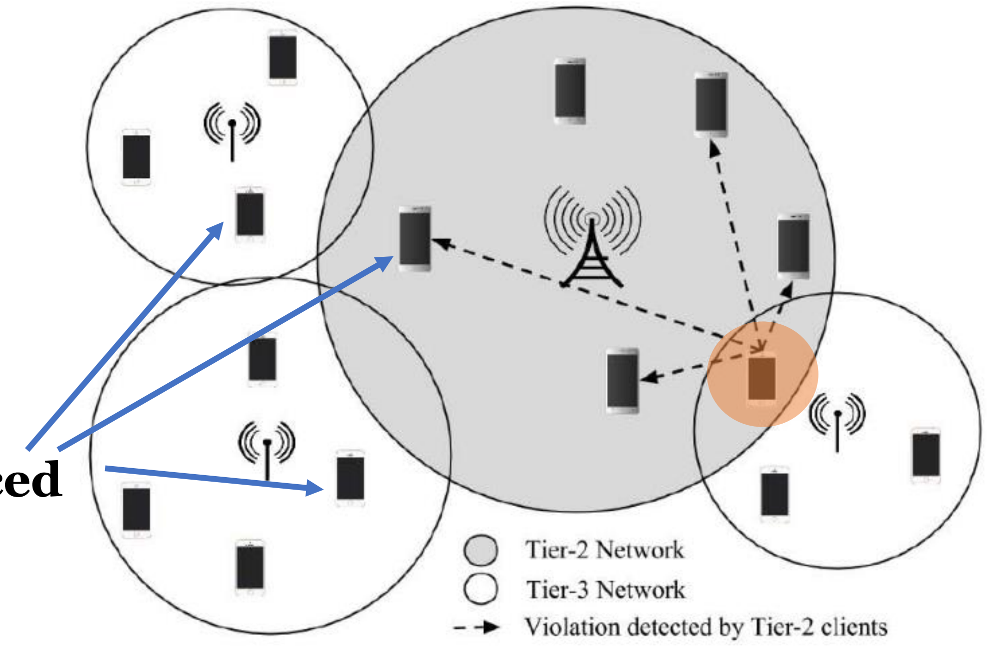
Autonomous Spectrum Sensing

"Spectrum Enforcement and Localization Using Autonomous Agents With Cardinality," Maqsood Ahamed Abdul Careem, A. Dutta and W. Wang in IEEE TCCN.
"Multi-Agent Planning with Cardinality: Towards Autonomous Enforcement of Spectrum Policies," Maqsood Ahamed Abdul Careem, Aveek Dutta and Weifu Wang in IEEE DYSPAN 2018.

Beyond Crowdsourcing

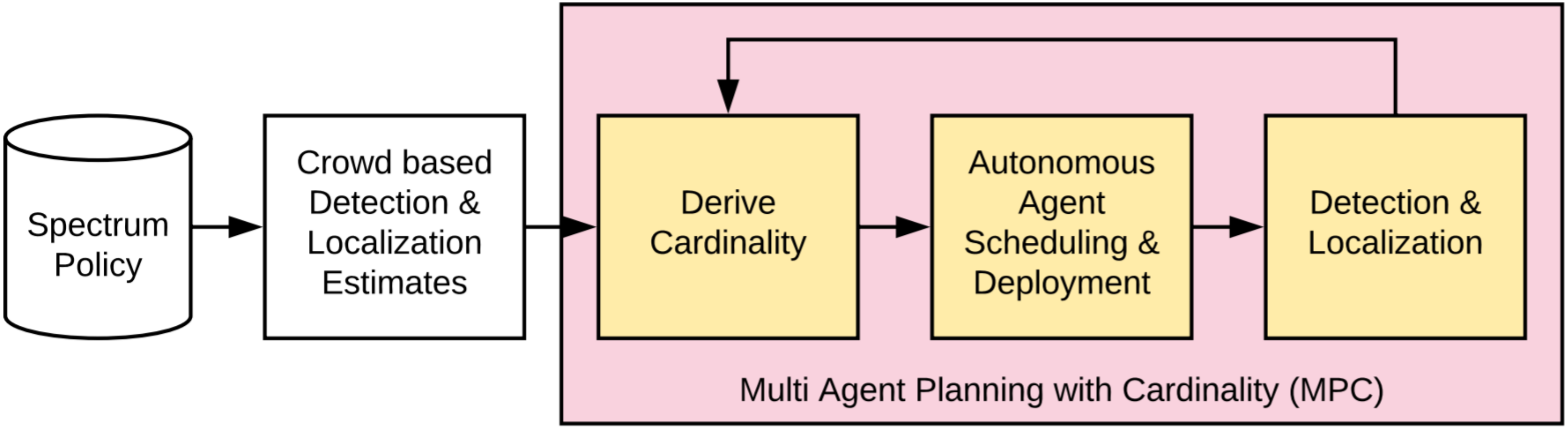
- Crowdsourced measurements [1]
  - Trust & Incentives
  - Limited Mobility & Resources
  - Provide approximate location & detection
- Accuracy
  - Detection of a *bad* source
  - Location estimate
    (low Geometric Dilution of Precision)

Crowdsourced Enforcers



[1] "See Something, Say Something": Crowdsourced Enforcement of Spectrum Policies. Aveek Dutta, Mung Chiang, *IEEE TWC, Sept. 2015"*

Hybrid Autonomous Sensing



Goal: Dispatch appropriate amount of resources (agents) to the right location in the shortest possible time.

Localization: Multilateration

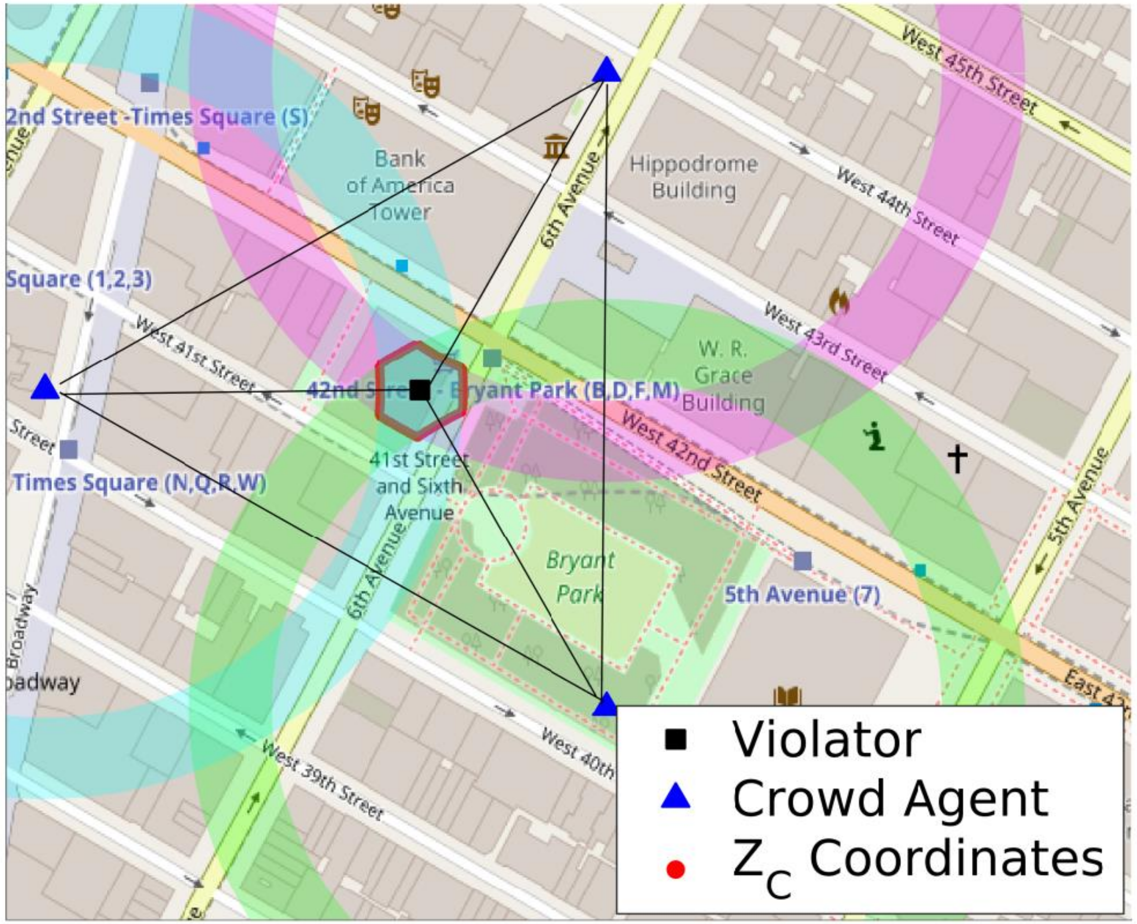
$$PL = A + B\log(d) + C \implies d = 10^{\frac{PL-A-C}{B}} \qquad (1)$$

where, $A = 69.55 + 26.16\log(f_c) - 13.82\log(h_b) - 3.2(\log(11.75h_m))^2 - 4.97$

$B = 44.9 - 6.55\log(h_b)$ and $C = 0$ (Large metropolitan areas)

$PL\ [\text{dBm}] = P_t\ [\text{dBm}] - \text{SNR}\ [\text{dB}] - P_N\ [\text{dBm}]$

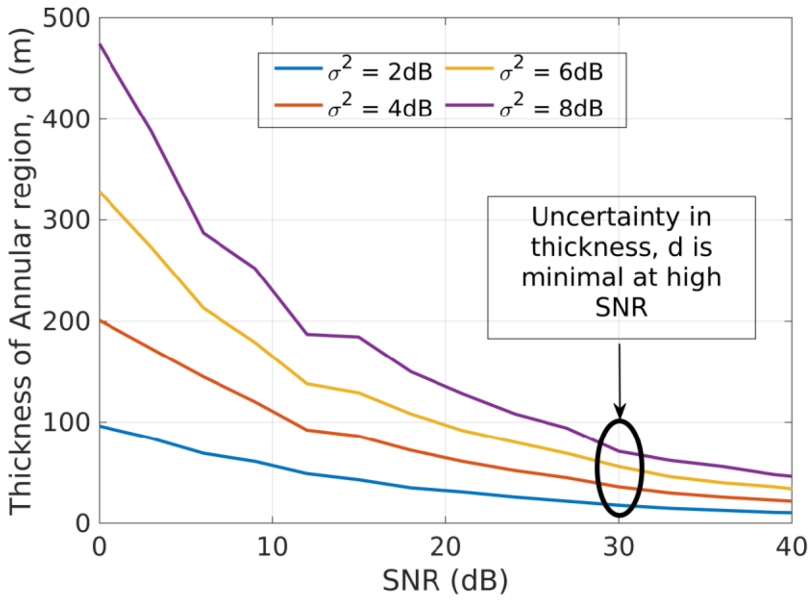
Hata-Urban channel model



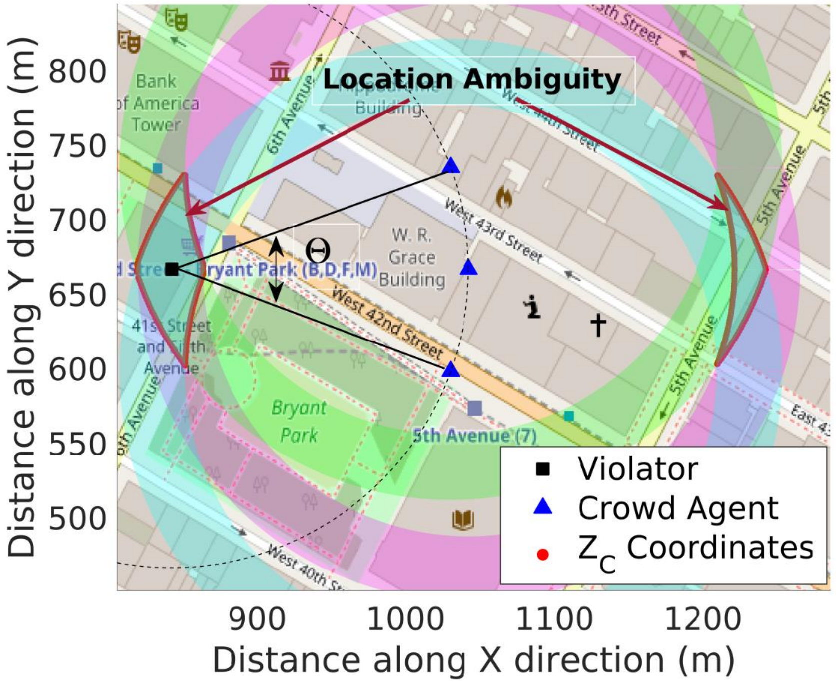
- Uncertainty from
  - Assumption about P_t
  - Measurement noise in SNR
  - Approximation of the channel model
- Use [SNR ± (X=x)]dB where X ~ N (μ, σ²)
  - $d = d_{outer} - d_{inner}$ (from (1) above)

Multilateration under noise

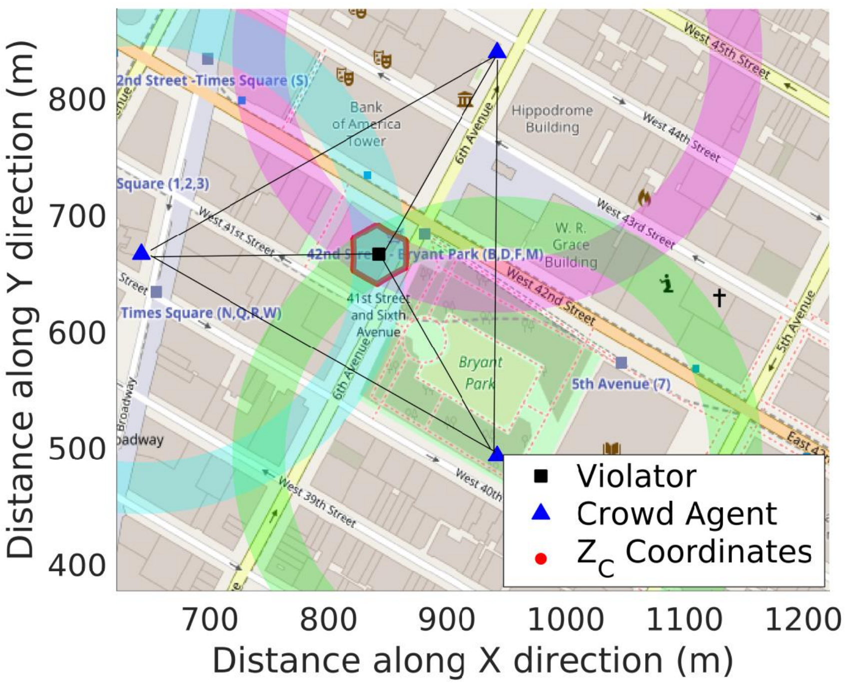
Close Agents → Less uncertainty



Crowdsourced → High GDOP



Ideal arrangement → Low GDOP

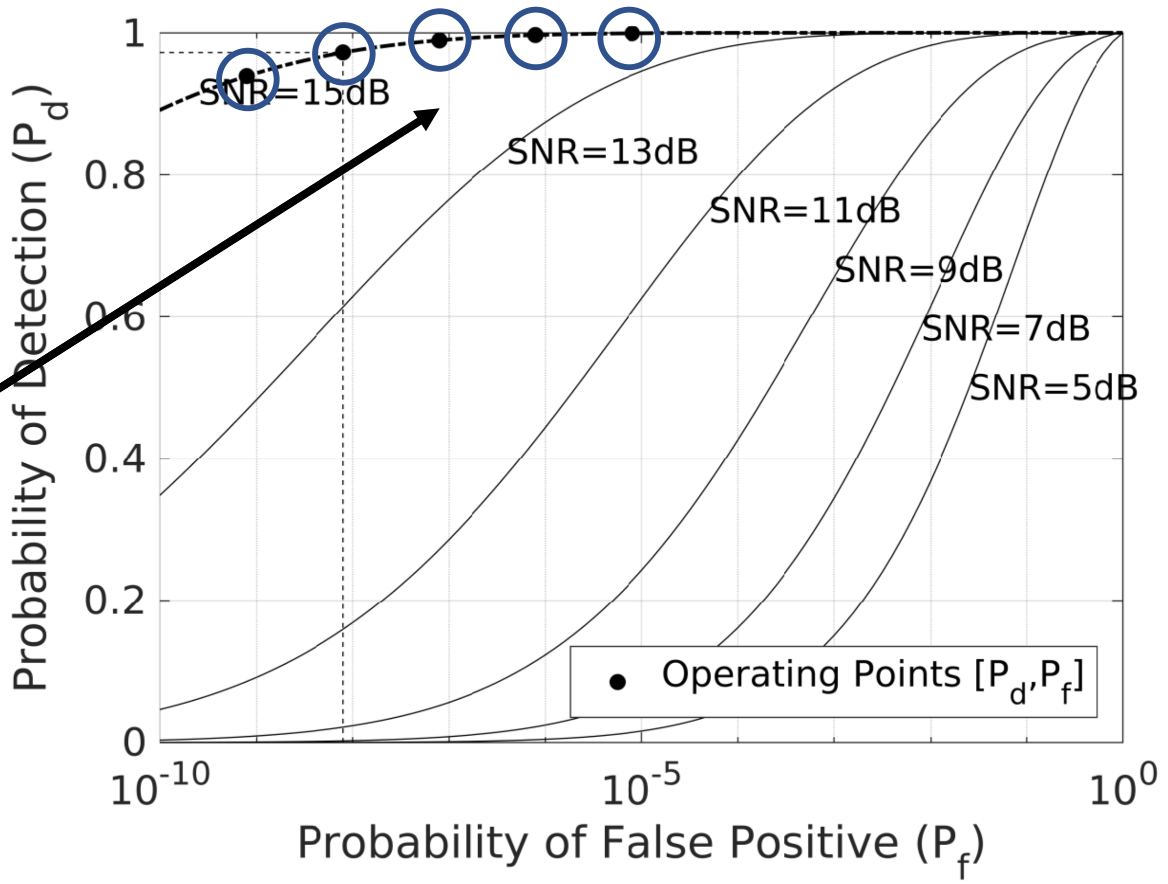


Number of Agents, their proximity and orientations affect the Localization

ROC and Impact on Detection

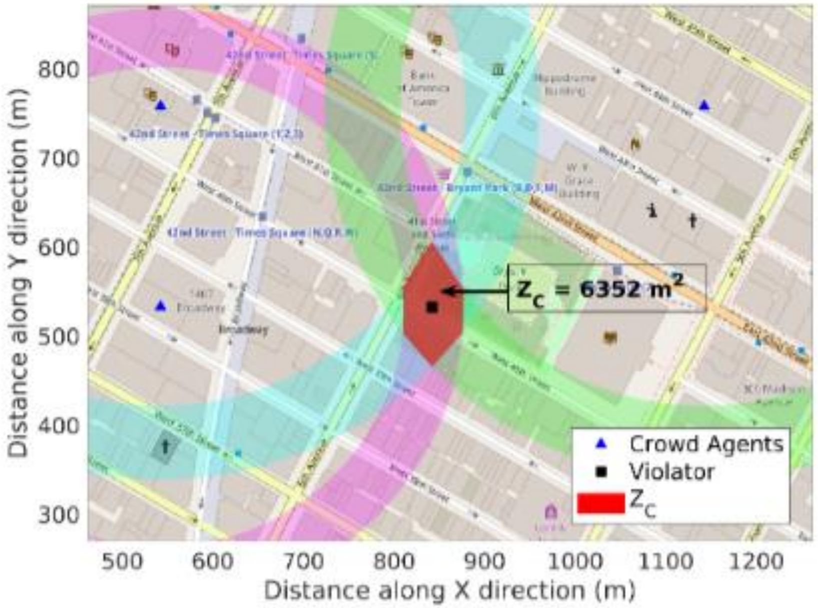
- Agents rely on ROC to choose an OP based on SNR
- Agents can use any detector [2]
  - e.g., Neyman-Pearson ROC

Close Enforcers have high SNR and can operate at desirable levels of [P_d, P_f]

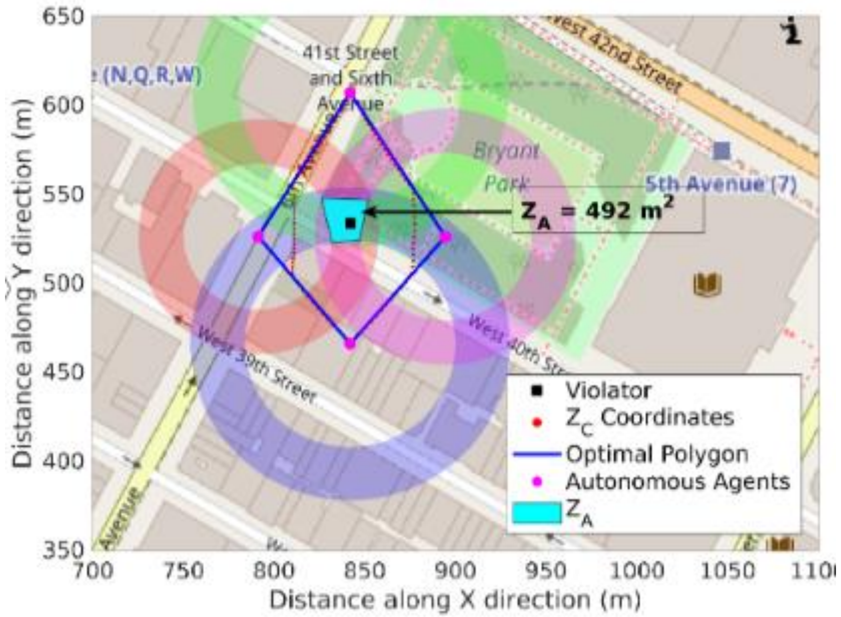


Multi-Agent Planning with Cardinality

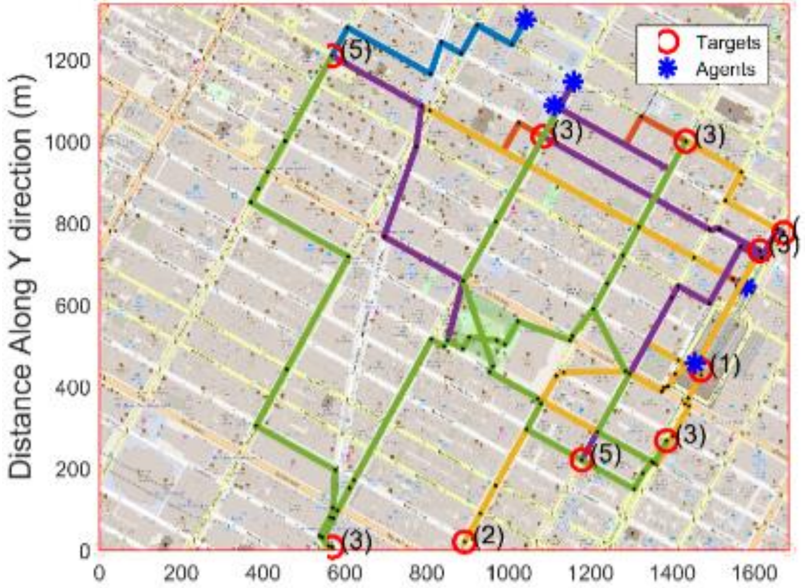
Crowd Sourced Localization



Autonomous Agent Localization



Scheduling



Route agents to Optimal Polygon circumscribing Z_c – 92% Improvement

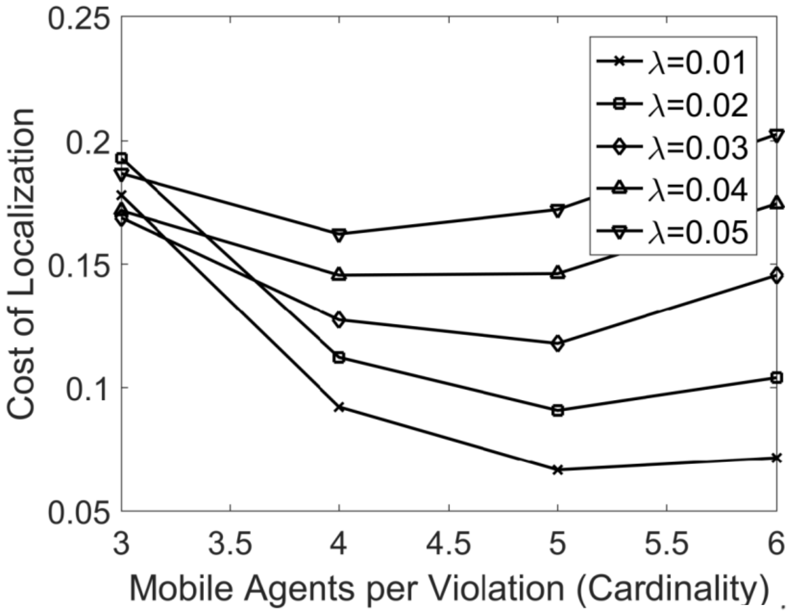
Schedule optimal number of agents to all targets

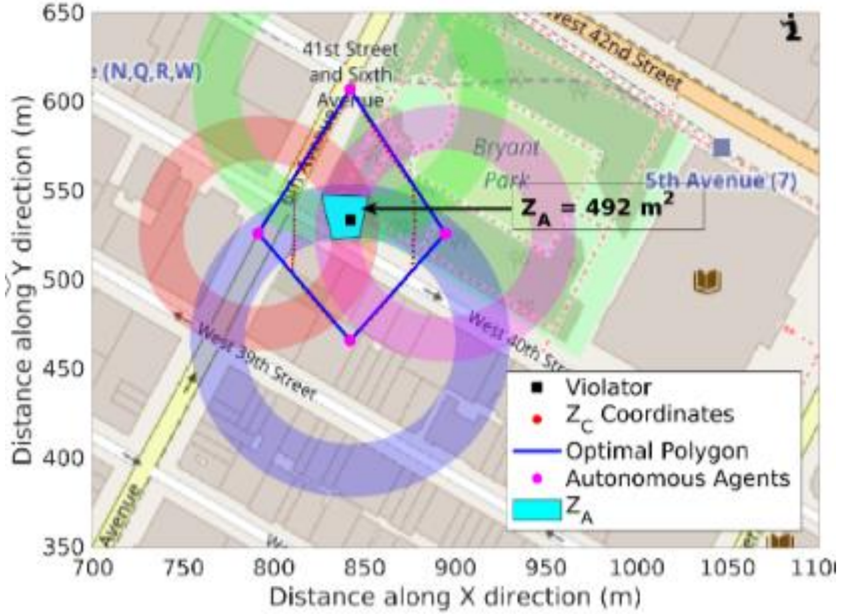
Step-A: Optimal Cardinality: Impact on Localization

$$Cost\ of\ Localization = \frac{\mathcal{Z}^i_{A,j}}{\mathcal{Z}_{C,j}} + \lambda i$$

Improvement in Localization Accuracy (over crowd)

# of agents deployed to target



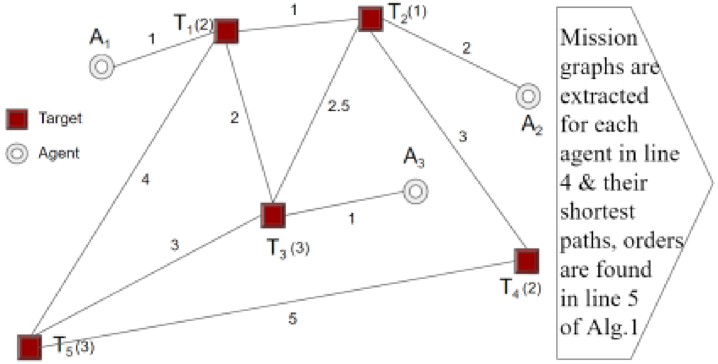


Definition: Cardinality $C_j = \arg\min_i \frac{\mathcal{Z}^i_{A,j}}{\mathcal{Z}_{C,j}} + \lambda i$

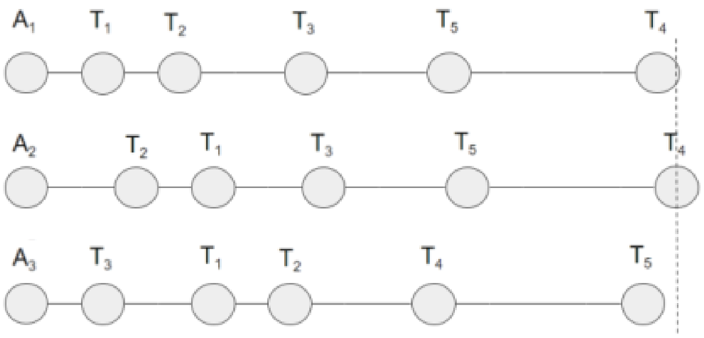
Step-B: Scheduling Algorithm

$$Cost\ of\ Scheduling = max_{\forall i} c(P_i) = max_{\forall i} l_i$$

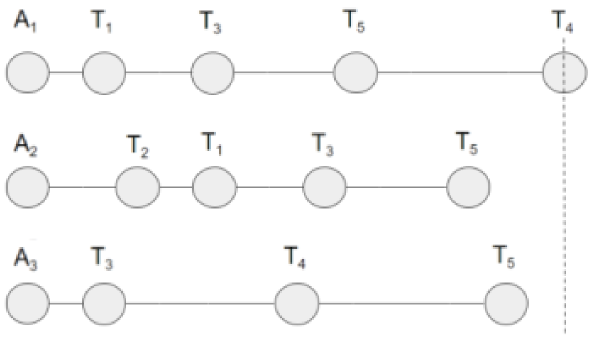
Schedule: $\mathcal{P} = \{P_1, ..., P_n\}$



(a) City map with 3 agents, 5 targets with different cardinality and edge weights



(b) Iter 1: Initial Path Estimate: A_2-costliest agent, T_4-farthest redundant target



(f) Iter 5: Remove T_1 from A_3's path, A_1-costliest agent with all cardinality fulfilled

Analysis of Scheduling Algorithm

Claim 1: The Schedule is NP-hard.

Lemma 1: Algorithm for Schedule is Polynomial $O(nm^4)$
n-# agents, m-#targets.

Theorem 1. *Algorithm 3 is 3-approximation for the Scheduling Problem.*

Proof Overview:
Costliest paths returned by Algorithm 3 and OPT - l_p and l_q^*

Goal: To find a relationship between l_p and l_q^*

Using: 1) Properties of Minimum Spanning Tree (MST)
2) Properties of Algorithm 3.

Cases: 1) The targets in $P_p \subseteq$ the targets in P_p^*
2) The targets in $P_p \nsubseteq$ the targets in P_p^*.

Property 1. *If $T_y^i = 0$, then l_i is <u>no worse</u> than twice the optimal cost l_i^*. i.e, $l_i \leq 2.l_i^*$.*

Furthermore, the following properties can be observed based on the design of Algorithm 3 and the definition of OPT.

Property 2. *Since, Algorithm 3 and OPT both return the costliest paths among all the agents (say l_p and l_q^*), the paths travelled by any other agent, <u>must not</u> be costlier than l_p or l_q^*. Thus, for any agent $i \in A$ we have, $l_i \leq l_p$ for Algorithm 3 and $l_i^* \leq l_q^*$ for OPT.*

Property 3. *In Algorithm 3 and OPT, all targets must be visited by the same number of agents (Definition 2 in §V).*

Property 4. *If a target t_k is removed from an agent i's path, it must have been the costliest path at some prior iteration of the algorithm (line 8–15). So, if agent p is the costliest agent at the end of the algorithm, the increase in agent i for visiting t_k <u>must be</u> such that $l_i + l_i(t_k) \geq l_p$.*

Property 5. *From Table I, we can express the costs l_i and l_i^* of agent i as,*

$$l_i = l_i(T_x^i) + l_i(T_y^i)$$
$$l_i^* = l_i^*(T_x^i) + l_i^*(T_z^i)$$

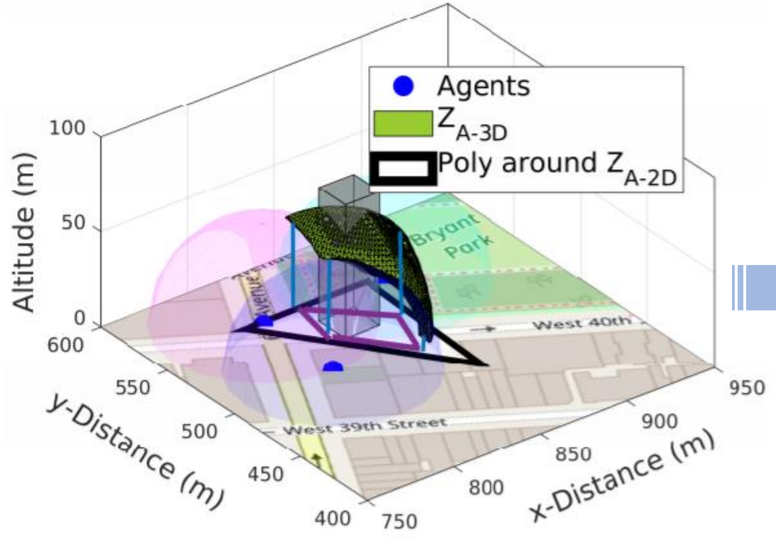
3D Localization and Detection: UAVs

$$\mathrm{PL} = \mathrm{PL}_{out} + \mathrm{PL}_{\mathrm{in}} + \mathrm{PL}_{\mathrm{tw}} + \mathcal{N}\left(0, \sigma_P^2\right)$$

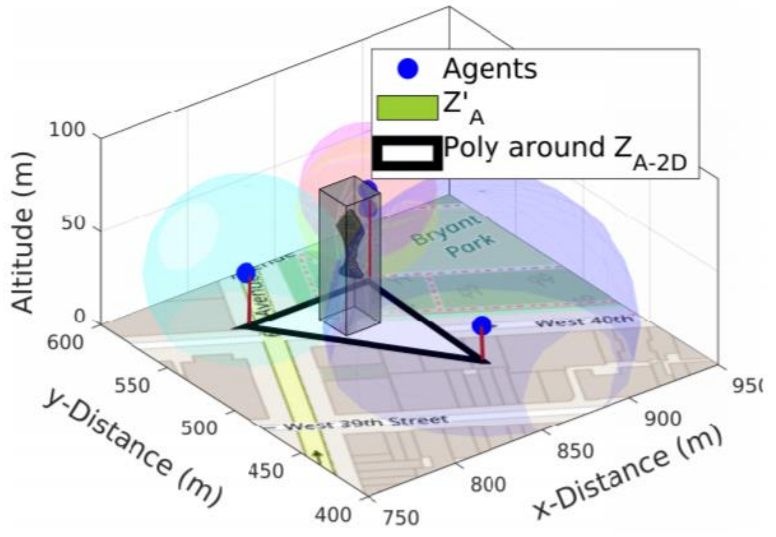
$$\mathrm{PL}_{out} = A + B\log(d) + C \quad \text{and} \quad \mathrm{PL}_{\mathrm{in}} = 0.5 d_{\mathrm{2D-in}}$$

$$\mathrm{PL}_{tw} = \mathrm{PL}_{npi} - 10\log_{10}\sum_{i=1}^{N}\left(p_i \times 10^{-0.1 L_{material\_i}}\right)$$

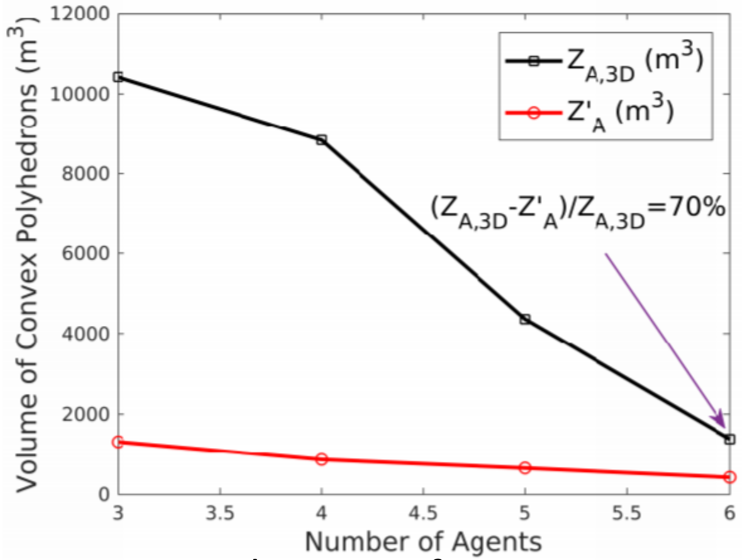
Outdoor-to-Indoor channel



UGV Localization



UAV Localization



Localization of UAVs vs UGVs

Evaluation Framework

Spectrum Sensing and Geographical Simulator

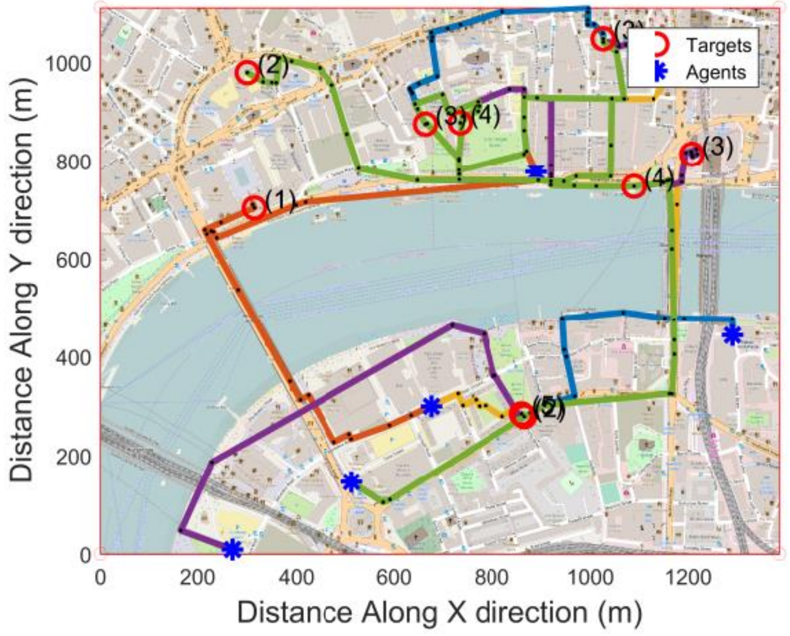
1) Open Street Map

2) Building Tags (OSM Buildings)

Autonomous Sensing Performance

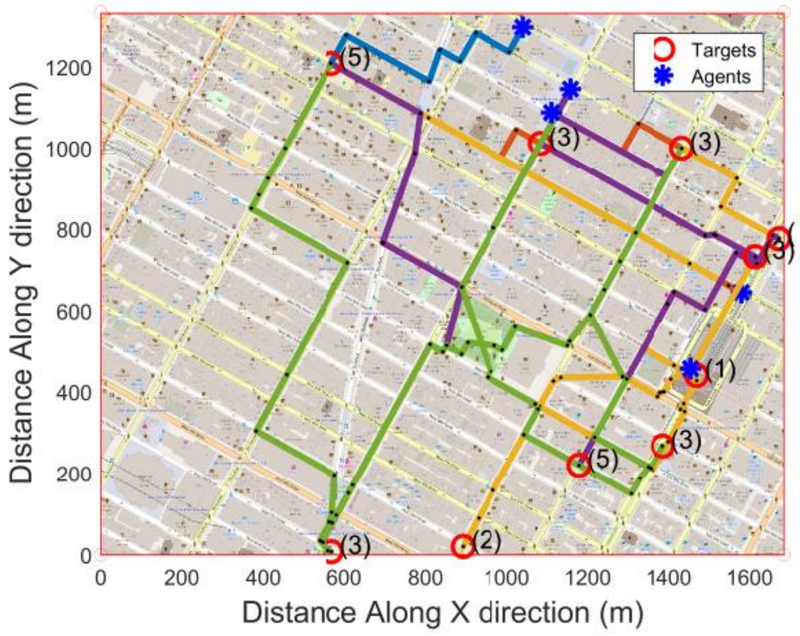
- Scheduling Costs in different cities

London



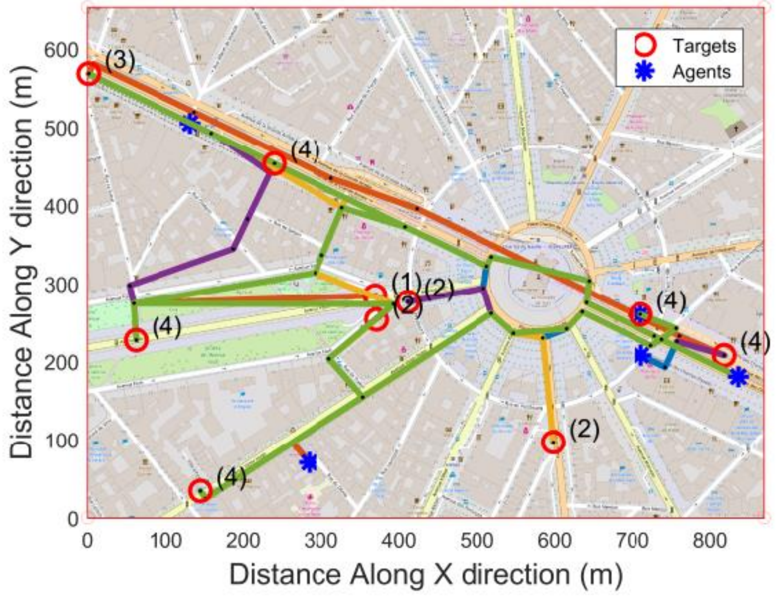
(a) Example routing in London. Cost metric = $2.618 km/km$

New York



(b) Example routing in NYC. Cost metric = $2.374 km/km$

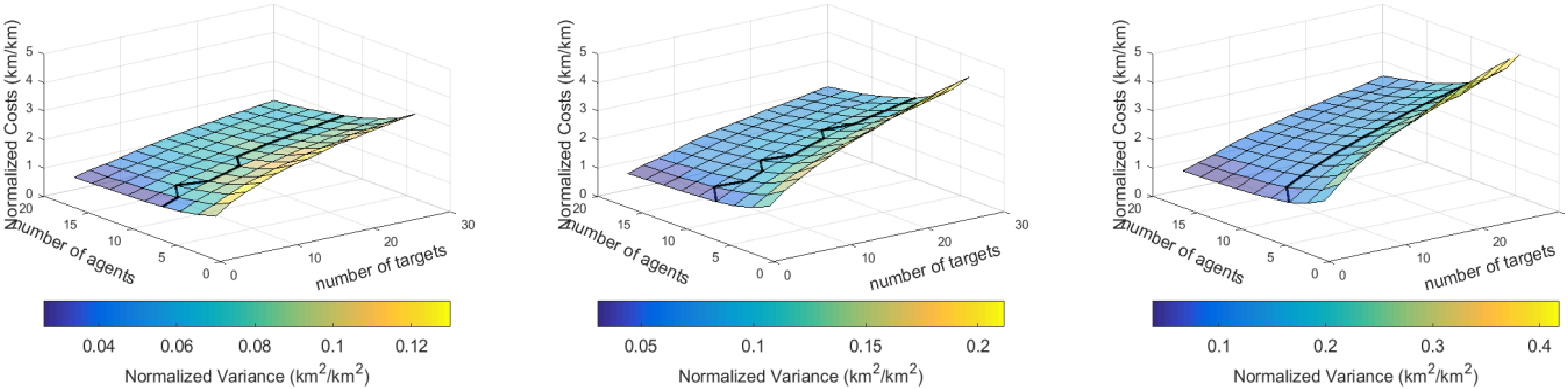
Paris



(c) Example routing in Paris. Cost metric = $2.874 km/km$

Parametric Analysis: Scheduling

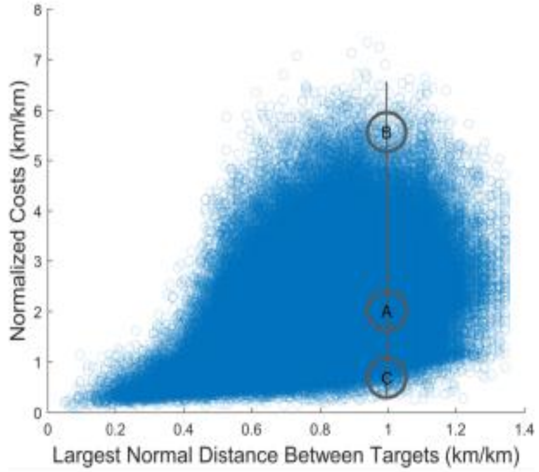
London New York Paris



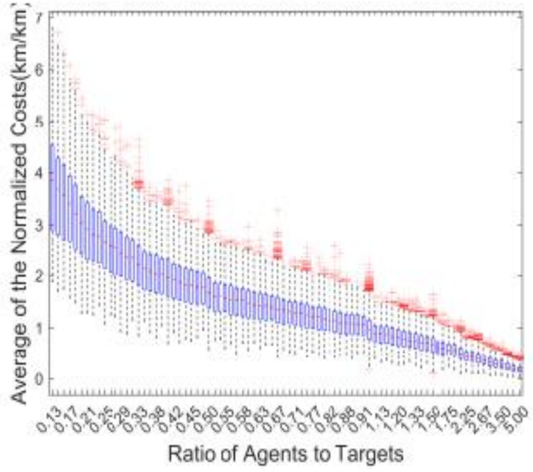
(a) Normalized cost metric in London (b) Normalized cost metric in NYC (c) Normalized cost metric in Paris

Fig. 3: Normalized cost metric for **Average Cardinality = 3** for (a) London (b) NYC and (c) Paris.The dark line highlights the points beyond which the cost variation is below 10%. The variance is indicated using the color scale.

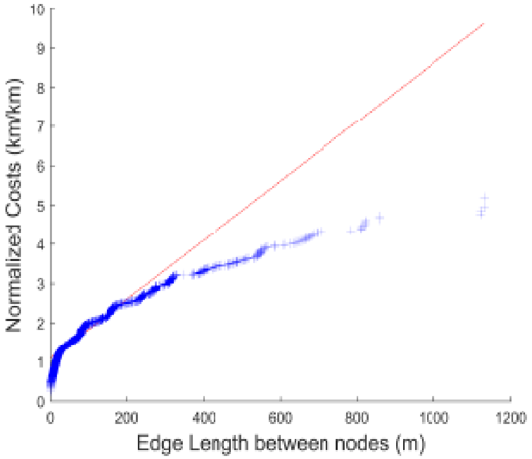
Overall System Performance



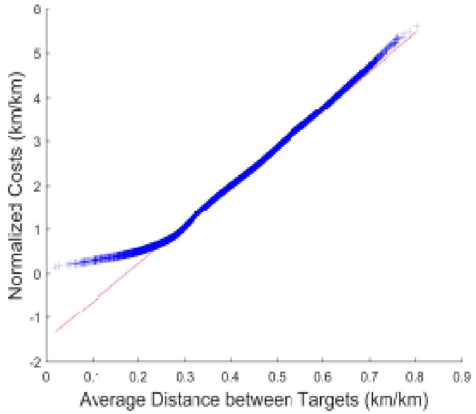
(a) Variation of Cost with maximum separation of targets.



(b) Variation of Cost with ratio of agents to targets



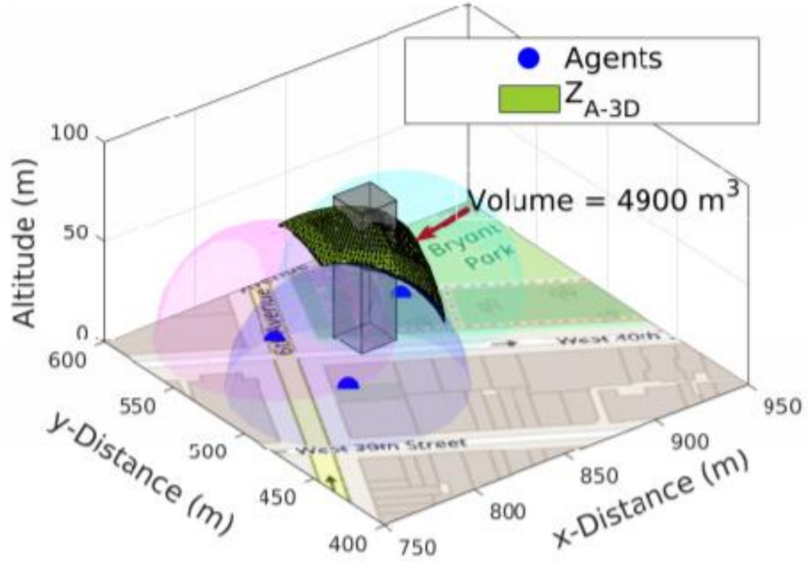
(c) Cost vs edge length



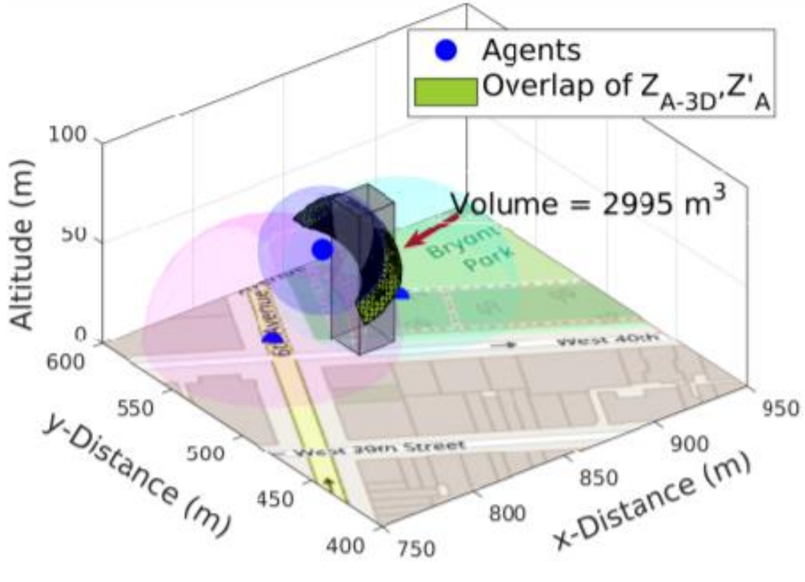
(d) Cost vs avgerage inter-target distance

Figure: Comparison of the distribution of Normalized Cost Metric for NYC with that of (a) Edge lengths and (b) Average Distance between Targets.

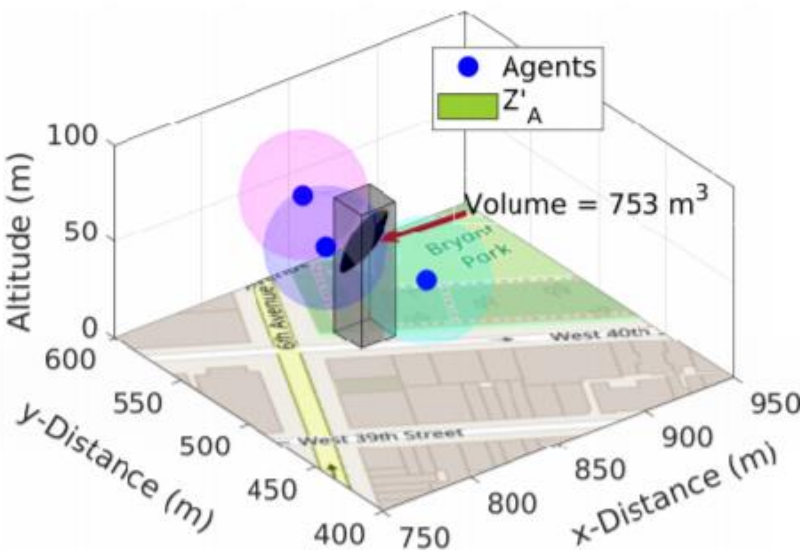
3D Localization using UAVs



(a) Localization with 3 UGVs ($\mathcal{Z}_{A,3D}$).



(b) Localization with 2 UGVs and 1 UAV.



(c) Localization with 3 UAVs.

SenseChain: Distributed Fusion System

"SenseChain: Blockchain based Reputation System for Distributed Spectrum Enforcement," Maqsood Ahamed Abdul Careem and Aveek Dutta in IEEE DYSPAN 2019.

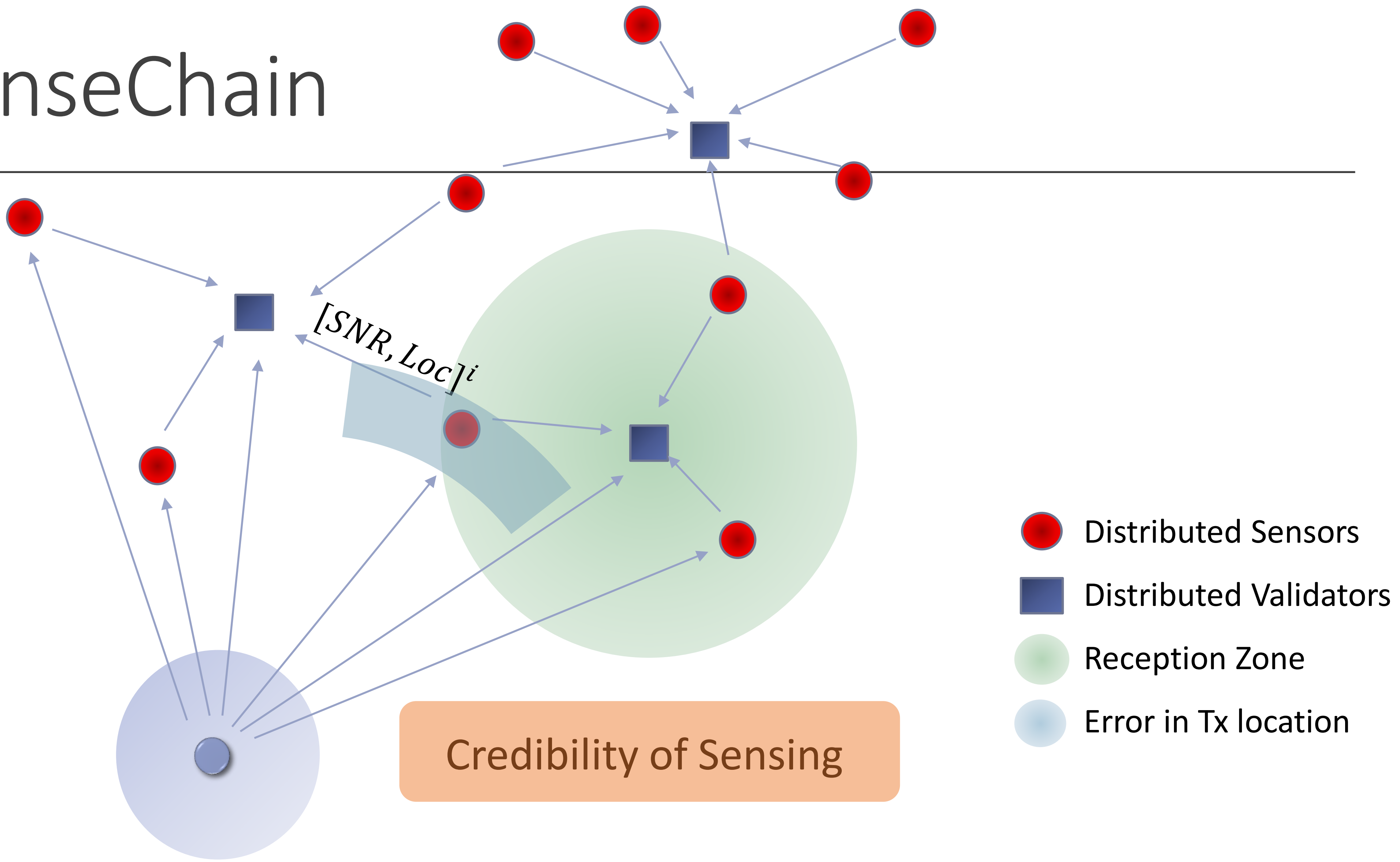
I. Contributions

Problem: Lack of Trust → Biased Inferences ⇒ Reputation of Agents

1. **Anomaly Detection**: Credibility of Sensing
2. **Heterogeneous Blockchain:** Credibility of Validation.
3. **Network protocol:** Consensus on Most credible Chain.

SenseChain: Fast & Tamper-proof distributed consensus on the reputation of sensors, among trustless entities.

II. SenseChain



III. SenseChain: Anomaly Detection

Log Distance Channel Model

$$PL_{s_i} = PL_{v_j} + 10\gamma \log_{10} \frac{d_{s_i}}{d_{v_j}} + \chi \qquad PL = P_t - P_r$$

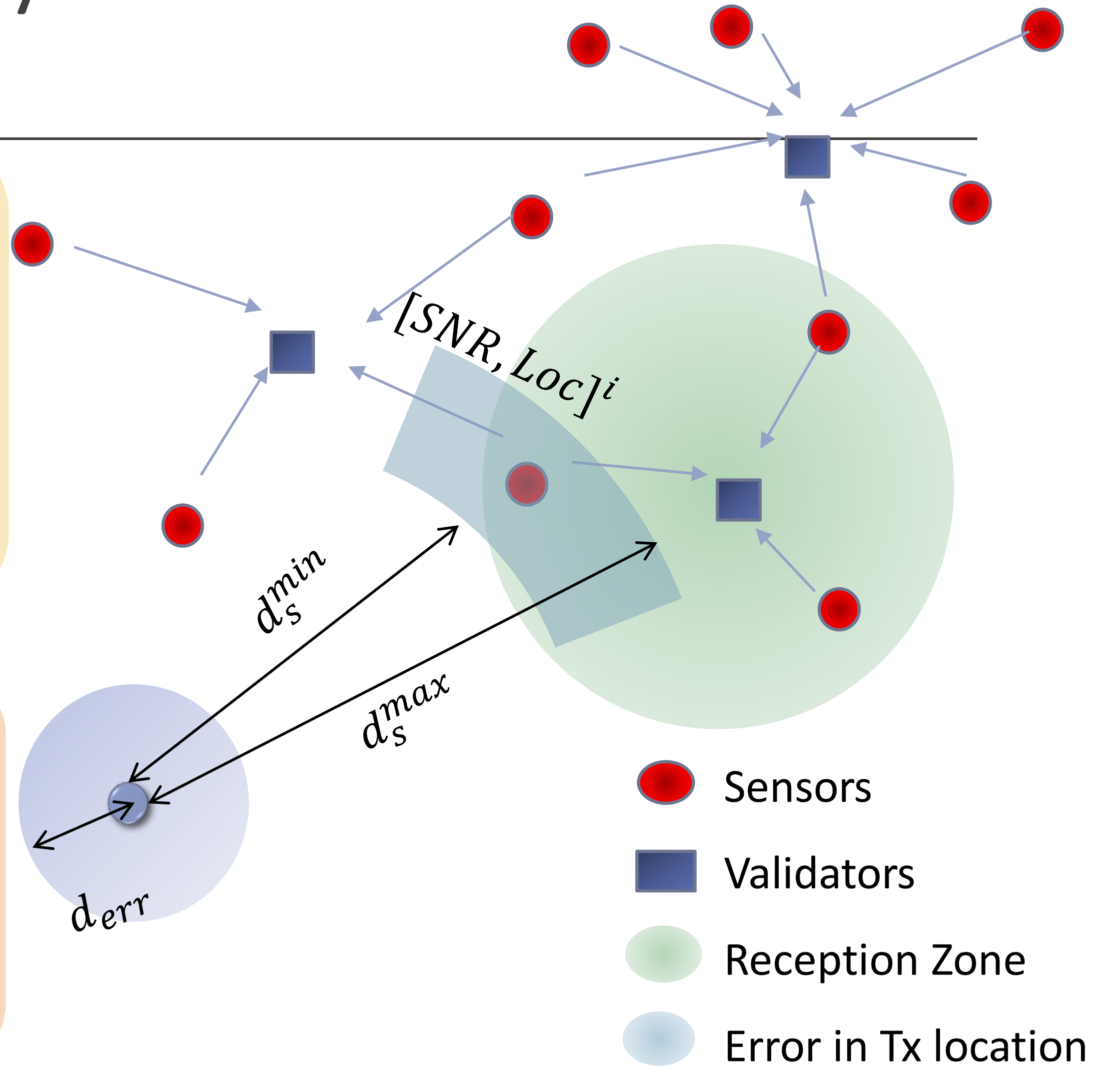
$$P_{r,s_i}(dBm) = SNR^i(dB) + NF(-96dBm)$$

$$-SNR_{s_i} = -SNR_{v_j} + 10\gamma \log_{10} \frac{d_{s_i}}{d_{v_j}} + \chi$$

Estimated Annular Zone

$$d_{s_i}^{min} = (d_{v_j} - d_{err}) \times 10^{\left(\frac{SNR^j - SNR^i - X_g}{10\gamma}\right)}$$

$$d_{s_i}^{max} = (d_{v_j} + d_{err}) \times 10^{\left(\frac{SNR^j - SNR^i - X_g}{10\gamma}\right)}$$



Anomalies and confidence score

Anomaly Detected if…

Reported location,

- Is Outside Validator Range

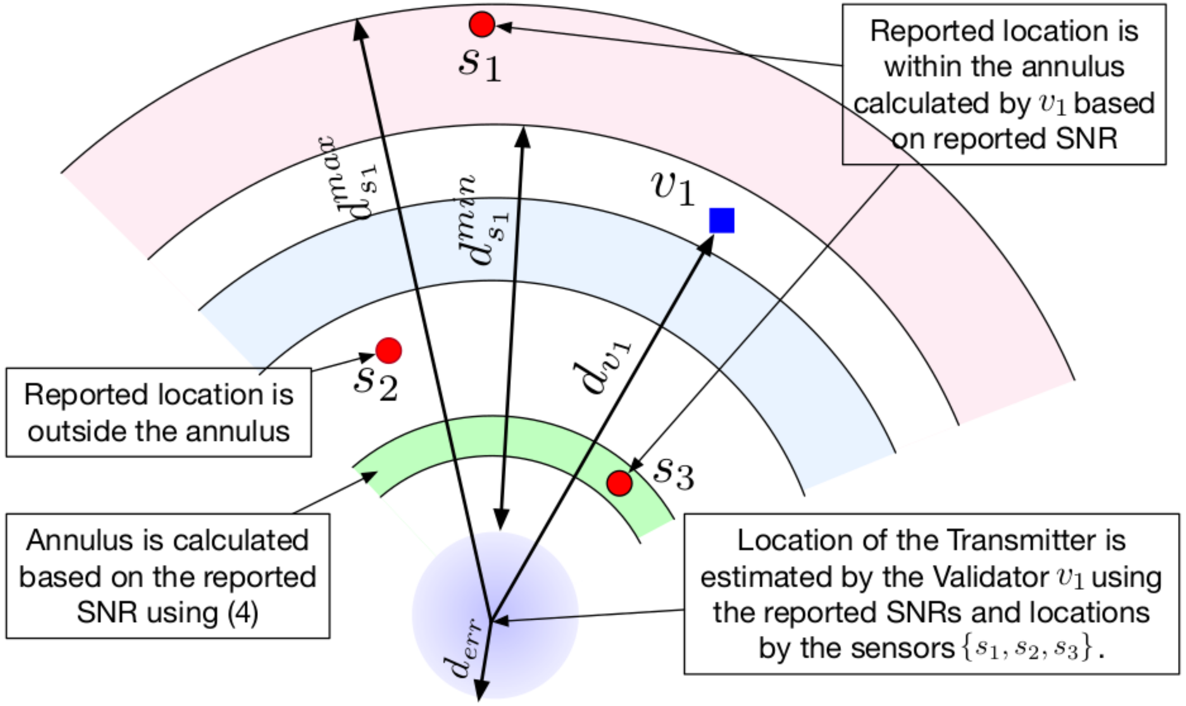
$$(d_{s_i} - d_{v_j}) > R$$

- Is Outside estimated annulus

$$\hat{d_{s_i}} < d_{s_i}^{min} \text{ or } \hat{d_{s_i}} > d_{s_i}^{max}$$

Confidence Score

$$S_{s_i} = \begin{cases} 1 - \frac{(d_{s_i}^{max} - d_{s_i}^{min})}{d_0}, \text{if } (d_{s_i}^{min} \le \hat{d_{s_i}} \le d_{s_i}^{max}) \quad \& \\ \qquad (d_{s_i}^{max} - d_{s_i}^{min} < R) \\ 0 \qquad , \text{otherwise} \end{cases}$$



Anomaly detected if reported sensor is outside annulus.

Else a confidence score represents its truthfulness.

A. Difficulty of mining

Effort in creating a Block of Information

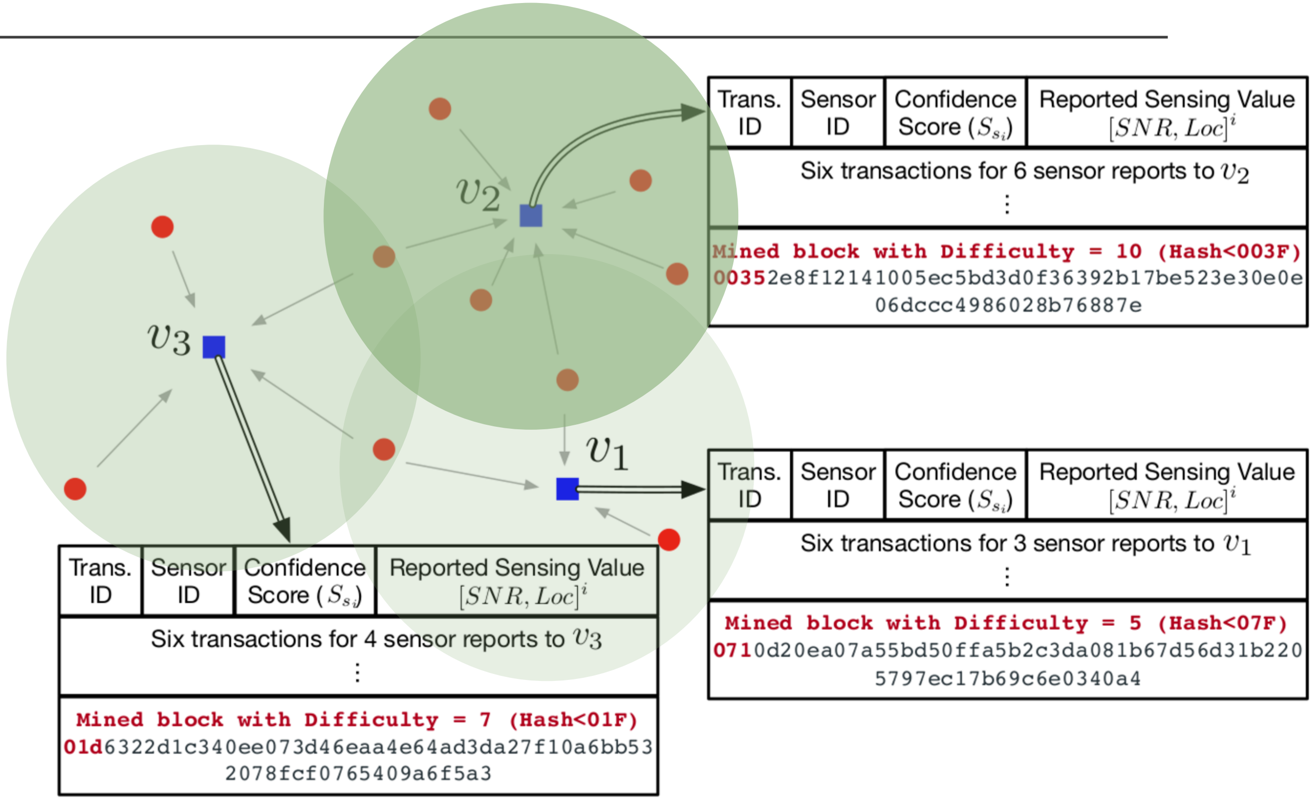
Difficulty

$$D_{v_j} = \left\lceil D_{max} \times \frac{n_{v_j}}{N} \right\rceil \qquad \forall v_j \in \mathcal{V}$$

n_{v_j} # Sensors in Reception zone of validator

N Total Number of sensors

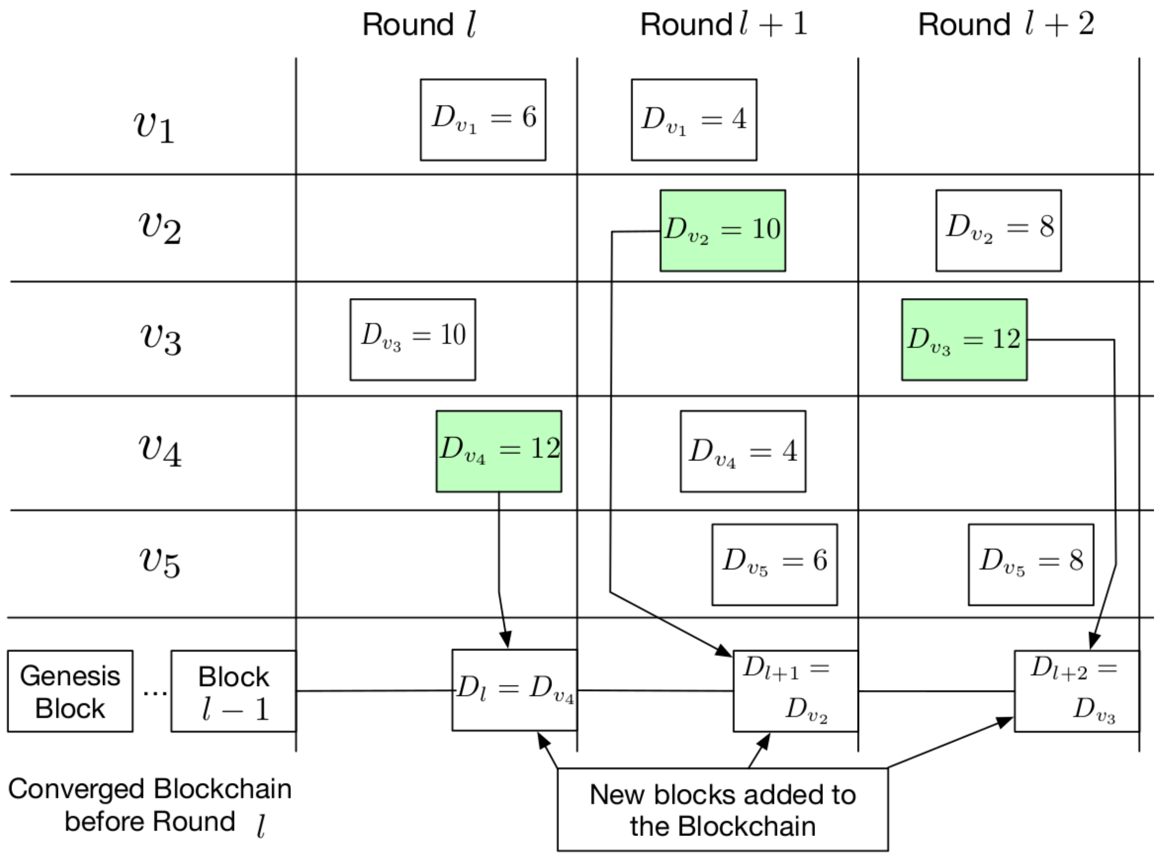
[Immutability] vs [Low Power & Fast Convergence]



Difficulty ∝ Validation Credibility (Power of the Crowd)

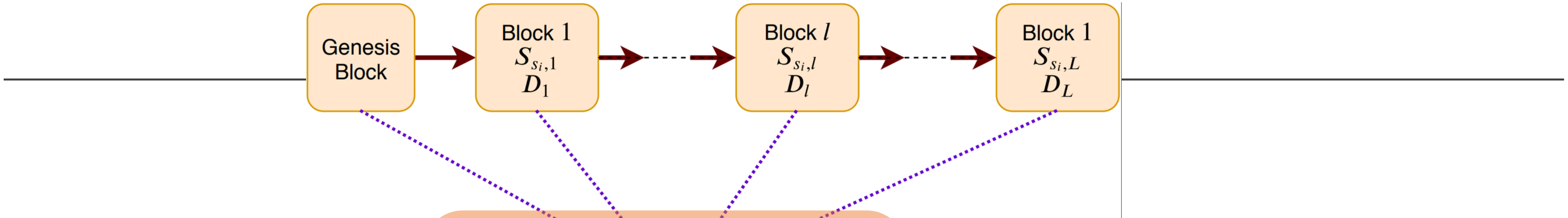
B. Most-Difficult-Chain consensus

Validators arrive at consensus on most credible chain



Most-Difficult-Chain Consensus: At each round, the most difficult mined block is added to the blockchain.

V. Historical Reputation & Provenance



$$\eta_{s_i} = \frac{\beta \sum_{l=1}^{L} a_{i,l} D_l (2S_{s_i,l} - 1)}{L.D_{max} \sum_{l=1}^{L} a_{i,l}}$$

$$\mathcal{R}_{s_i} = \frac{1}{1 + e^{-\eta_{s_i}}}$$

a: Availability

S: Confidence (Credibility of Sensing)

D: Difficulty (Credibility of Validation)

$$P_d^{fused} = \sum_i w_i P_{d_i} \qquad w_i = \mathcal{R}_{s_i} / \sum_i \mathcal{R}_{s_i}$$

Most Credible Reputation Assignment → Most Credible Inference

Evaluation Framework

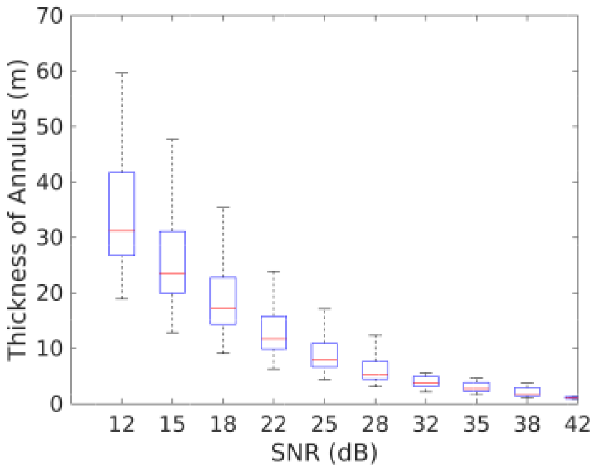
1) Sensing Environment

2) Blockchain Simulator

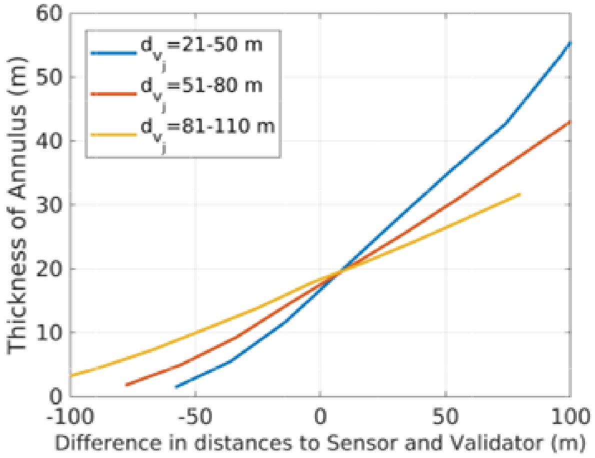
TABLE I: Simulation Parameters

Parameters	Value/Model
Area	300m $\times$ 300m
Node Distribution	Uniform Distribution
Mobility Model	Random Waypoint
Propagation Model	Log-distance propagation model [14]
Path-loss exponent (γ)	3 (urban area)
Carrier Frequency (f)	600 MHz
Number of Validators	5
Number of Sensors	20
Antenna Type	Omnidirectional
Broadcast Range	100
Maximum Difficulty (D_{max})	16
Block-wait Time ($\tau_{\mathcal{B}}$)	7 s
Target location error (d_{err})	Uniformly distributed in [20,30] m

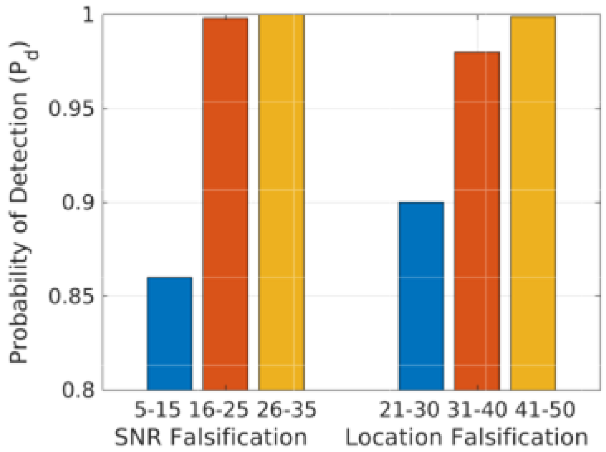
A. Performance of anomaly detection



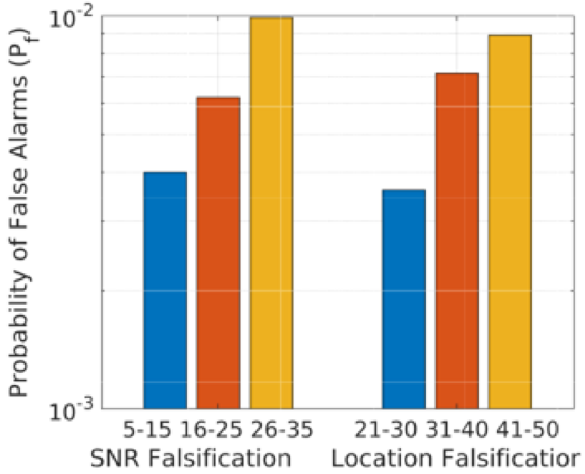
(a) Variation of annulus width with reported SNR



(b) Annulus width with Sensor and Validator distances



(c) Pd with falsification in SNR (dB) and Location (m)

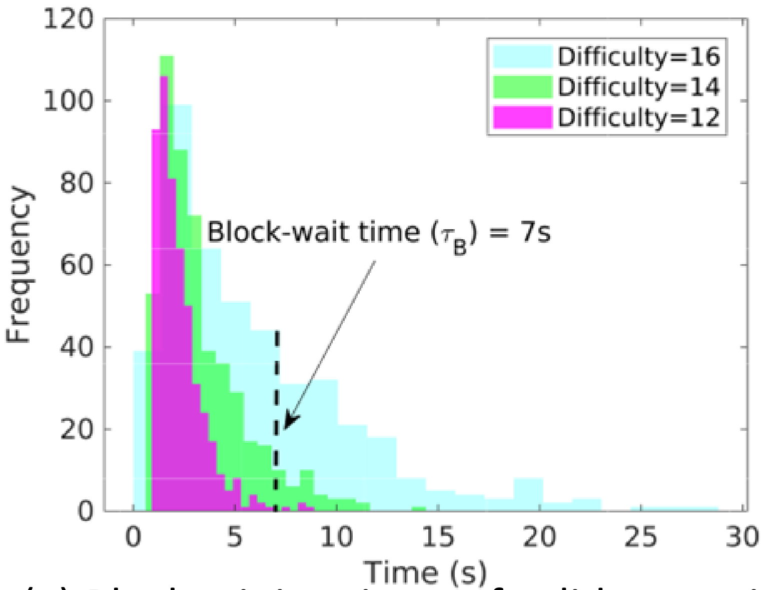


(d) Pf with falsification in SNR (dB) and Location (m)

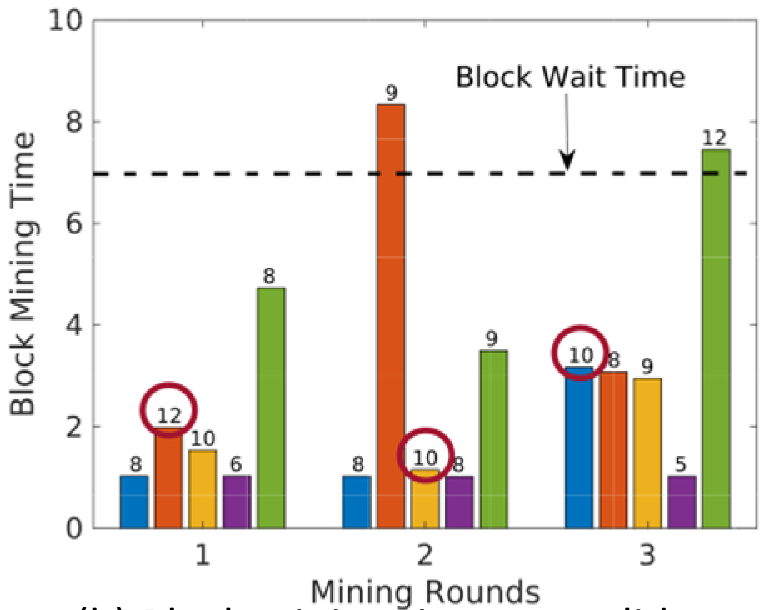
Truthfulness of Sensors can be Accurately inferred in Distributed Manner

B. Performance of Blockchain based Reputation

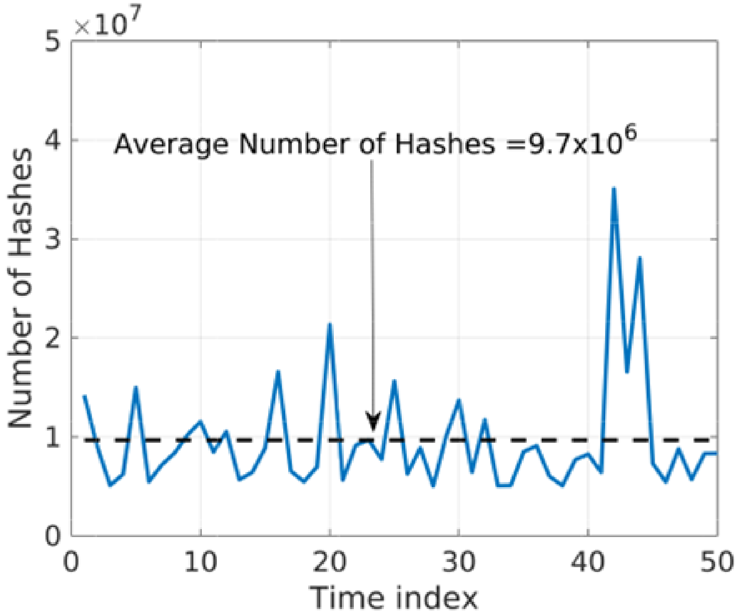
Blockchain performance:



(a) Block mining times of validators with varying difficulty targets

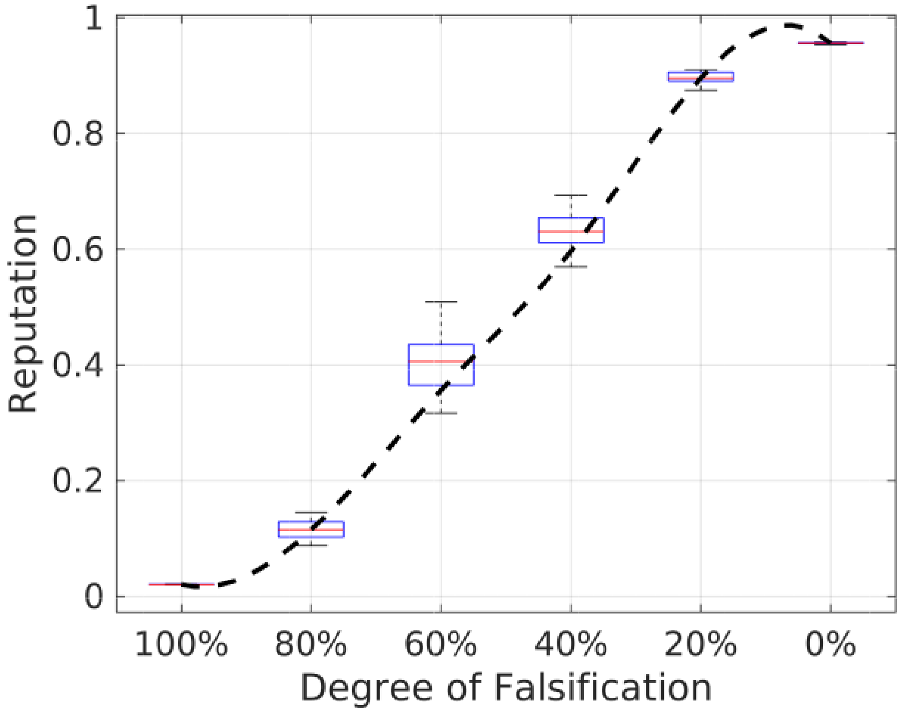


(b) Block mining time per validator and winning block in each round

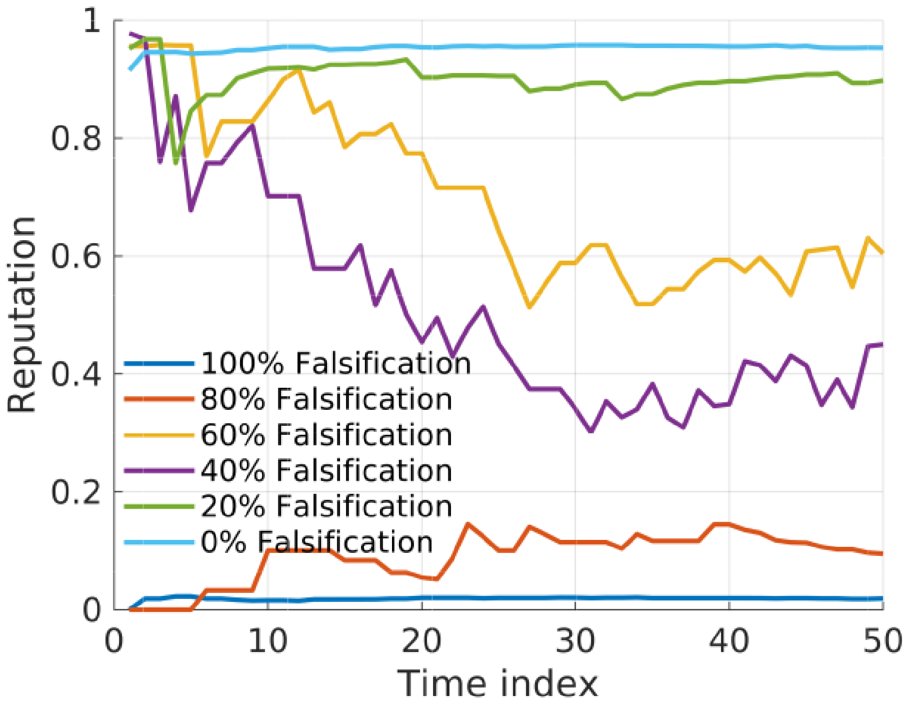


(c) The number of hashes generated by the winning validator

Reputation Assignment:



(a) Reputation with degree of falsification



(b) Reputation of falsifying Sensors over time

Reputation of Sensors represents the Degree of Maliciousness of Sensors

Conclusion

1. **Can Enforce, Distributed and Dynamic Violations in Shortest possible time with high accuracy compared to crowd or static paradigms**

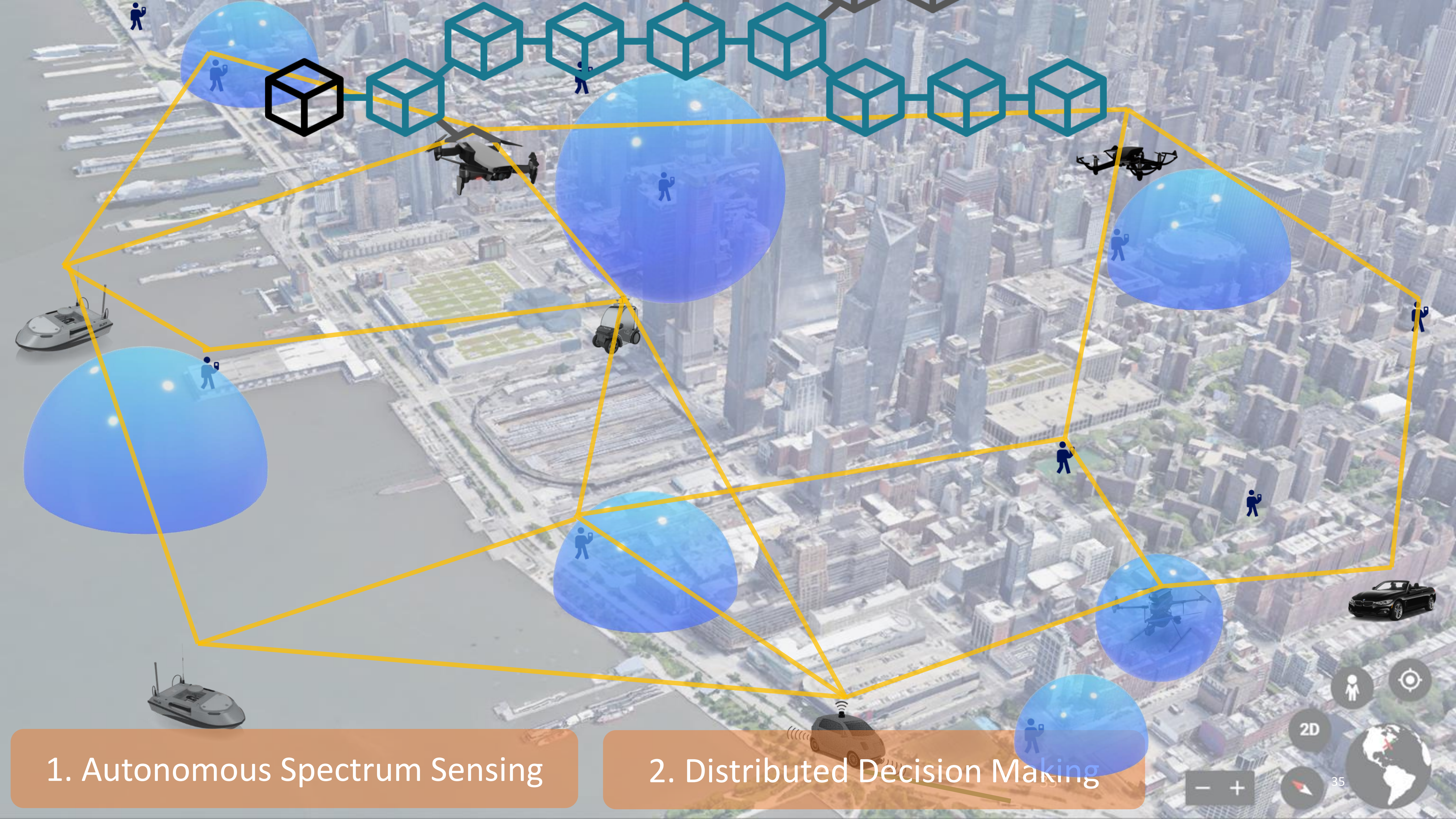
2. **Distributed Decisions can be made among trust-less agents without centralized architecture**

3. Can also be applied to Spectrum Sharing and Autonomous Spectrum Sensing.

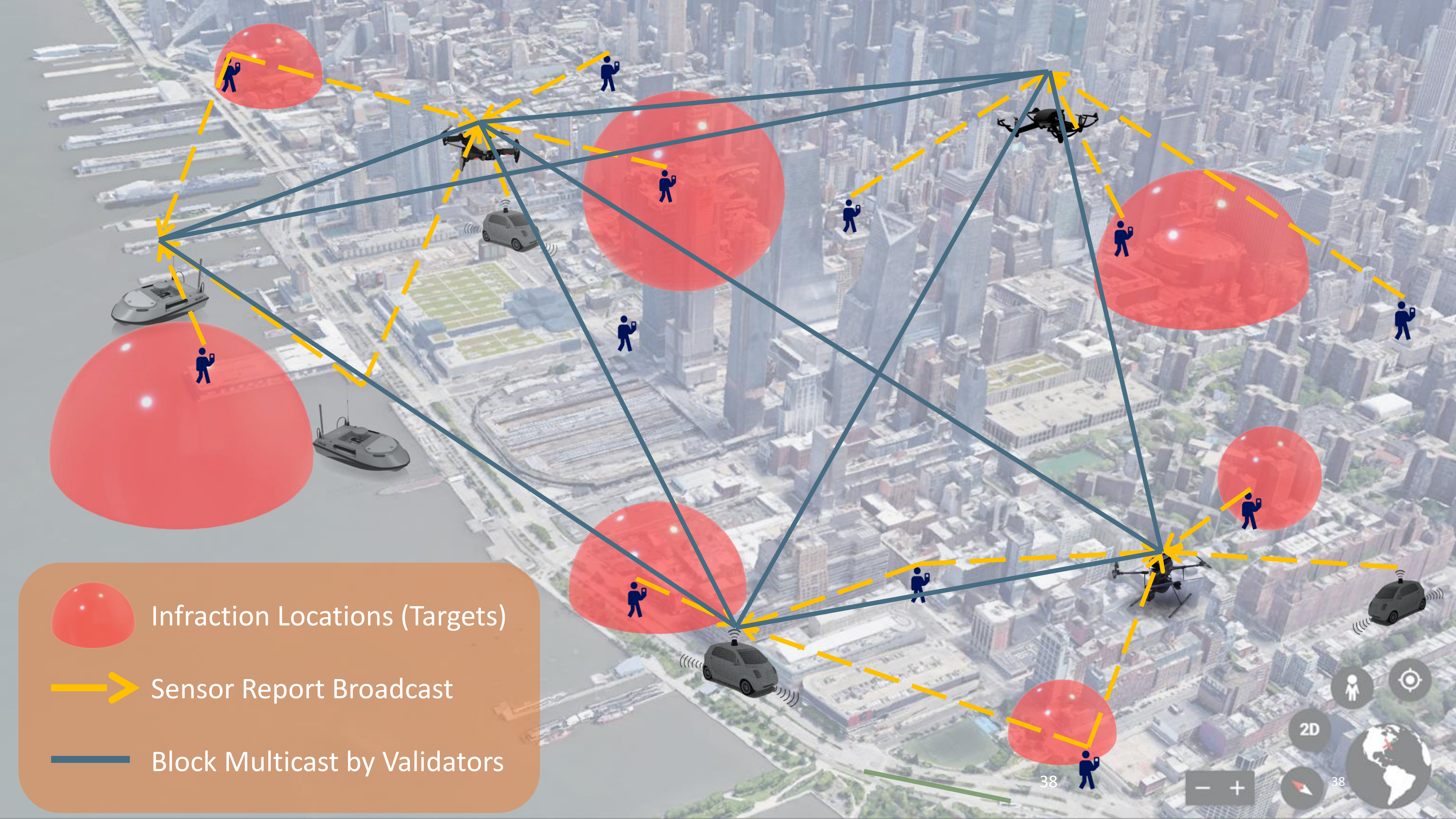
Autonomous Spectrum Enforcement system performs fully autonomously and achieves higher Enforcement accuracy and reliability compared to crowdsourced or static paradigms

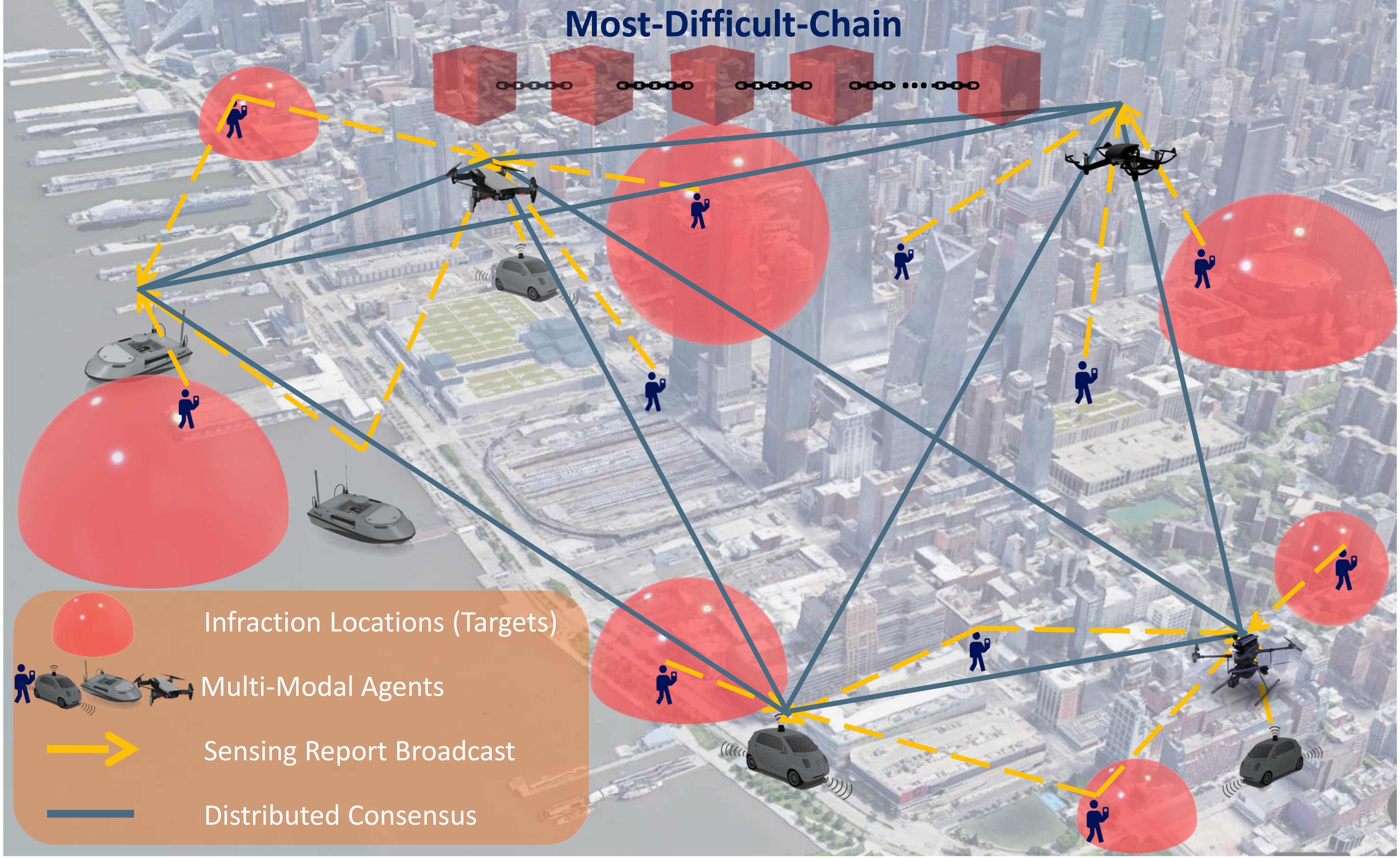
Thank you

Feedback & Questions


1. Autonomous Spectrum Sensing
2. Distributed Decision Making
2D


2D


Infraction Locations (Targets)
Sensor Report Broadcast
Block Multicast by Validators
2D


Most-Difficult-Chain
Infraction Locations (Targets)
Multi-Modal Agents
Sensing Report Broadcast
Distributed Consensus