Multi-Agent Planning with Cardinality: Towards Autonomous Enforcement of Spectrum Policies

Maqsood Careem, Aveek Dutta and Weifu Wang

Department of Electrical & Computer Engineering
University at Albany, SUNY

UNIVERSITY AT ALBANY
State University of New York

Motivation

ENTITIES

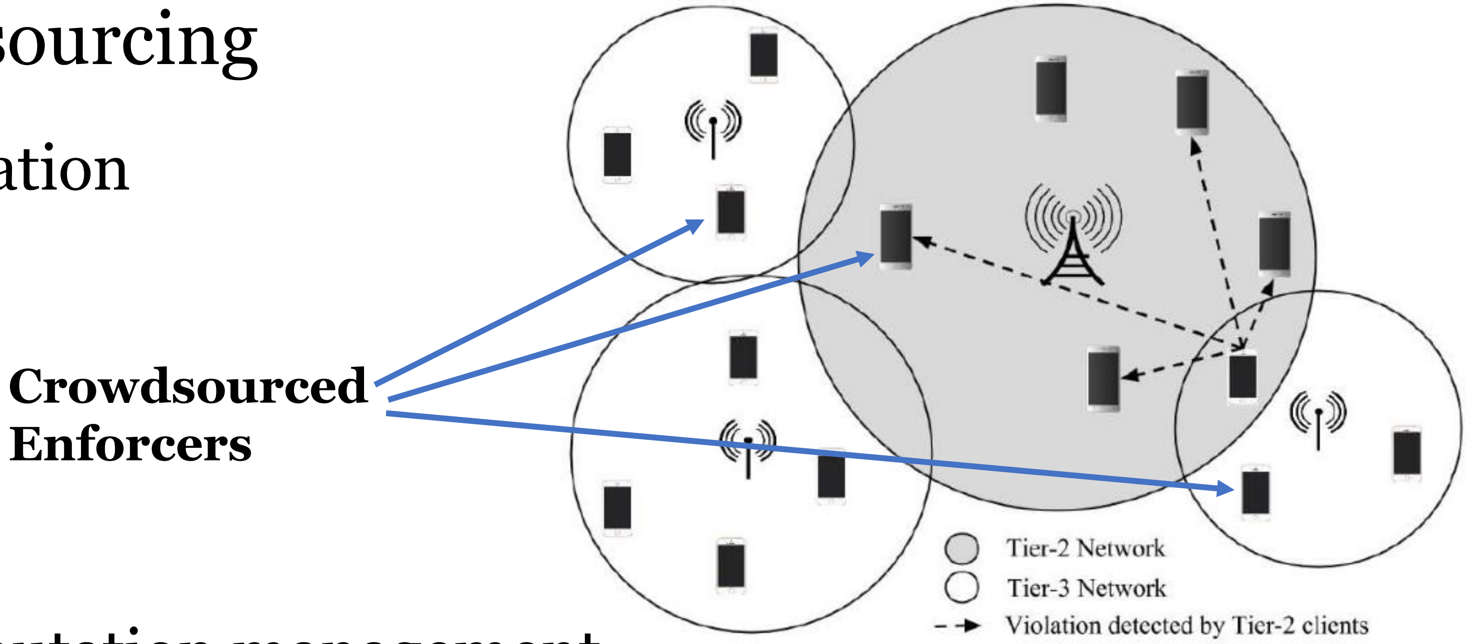
The *“Target”* (Violator) : Entity that violates spectrum policies

The *“Agent”* (Enforcer) : Entity that is deployed to detect and locate infractions (Enforcement Tasks).

The *“Dispatch”* : Enforcing agency with the authority to deploy agents as necessary and collect evidential information within its jurisdiction.

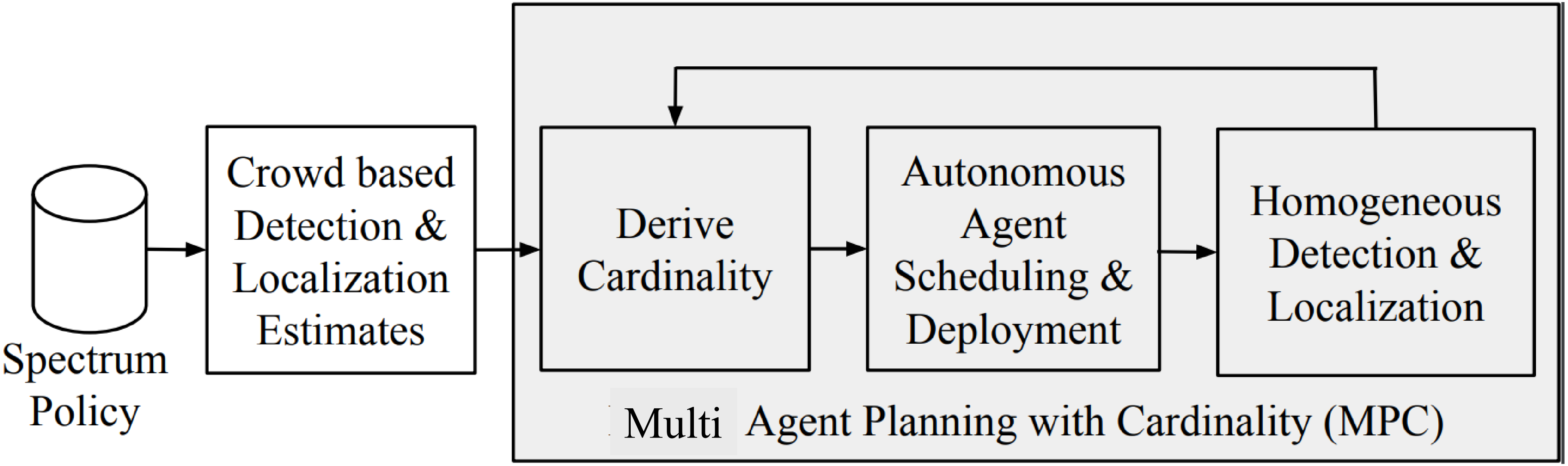
Beyond Crowdsourcing

- Shortcomings of Crowdsourcing
  - Provide approximate location
  - Limited mobility
  - Limited resources
  - Require incentives
  - Free-riders, trust and reputation management



Prior work: "See Something, Say Something": Crowdsourced Enforcement of Spectrum Policies.
Aveek Dutta, Mung Chiang, *IEEE Trans. Of Wireless Communication, Sept. 2015"*

Hybrid Enforcement



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

What is Cardinality?

- Cardinality
  - Number of unique, mobile agents visiting targets, to achieve a target accuracy
- Dimensions of Accuracy
  - Detection of a *bad* source
  - Location estimate (Geometric Dilution of Precision)
- Assumed Method
  - **Geometric trilateration** to locate a target
  - Find cardinality that minimizes the **GDOP**

Accuracy and GDOP

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

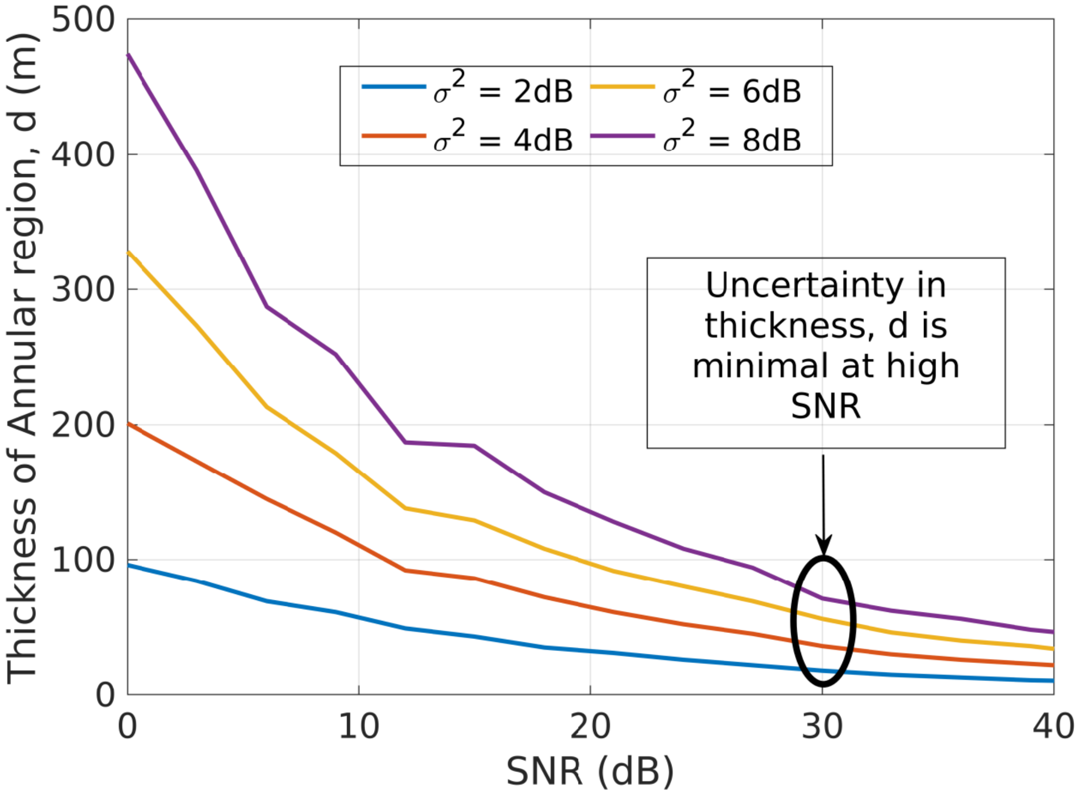
where, $A = 69.55 + 26.16\log(f_c) - 13.82\log(h_b) - 3.2(\log(11.75h_m))^2 - 4.97$

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

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

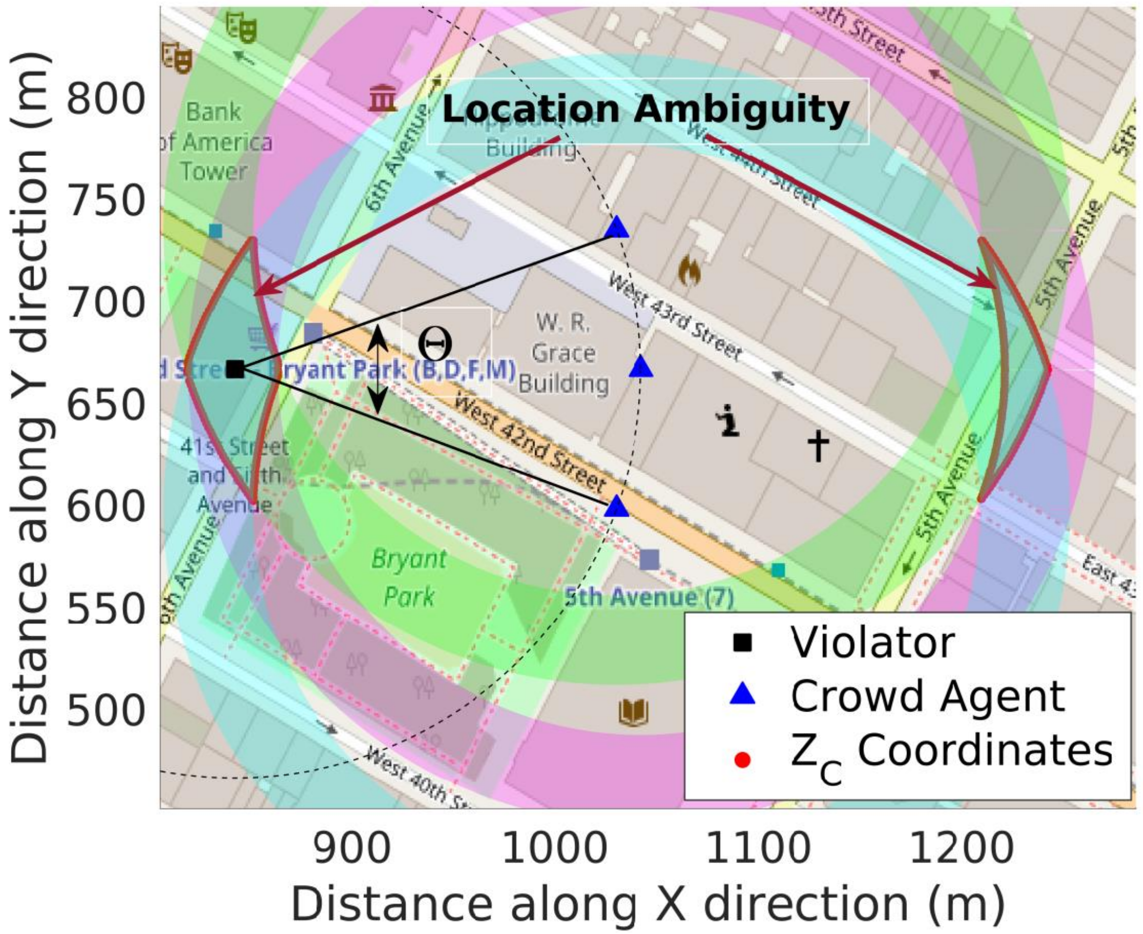
Hata-Urban channel model

- 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)
- Thickness of the annulus (d) decreases with SNR
  - Closer the Enforcers to the target, lesser is the uncertainty

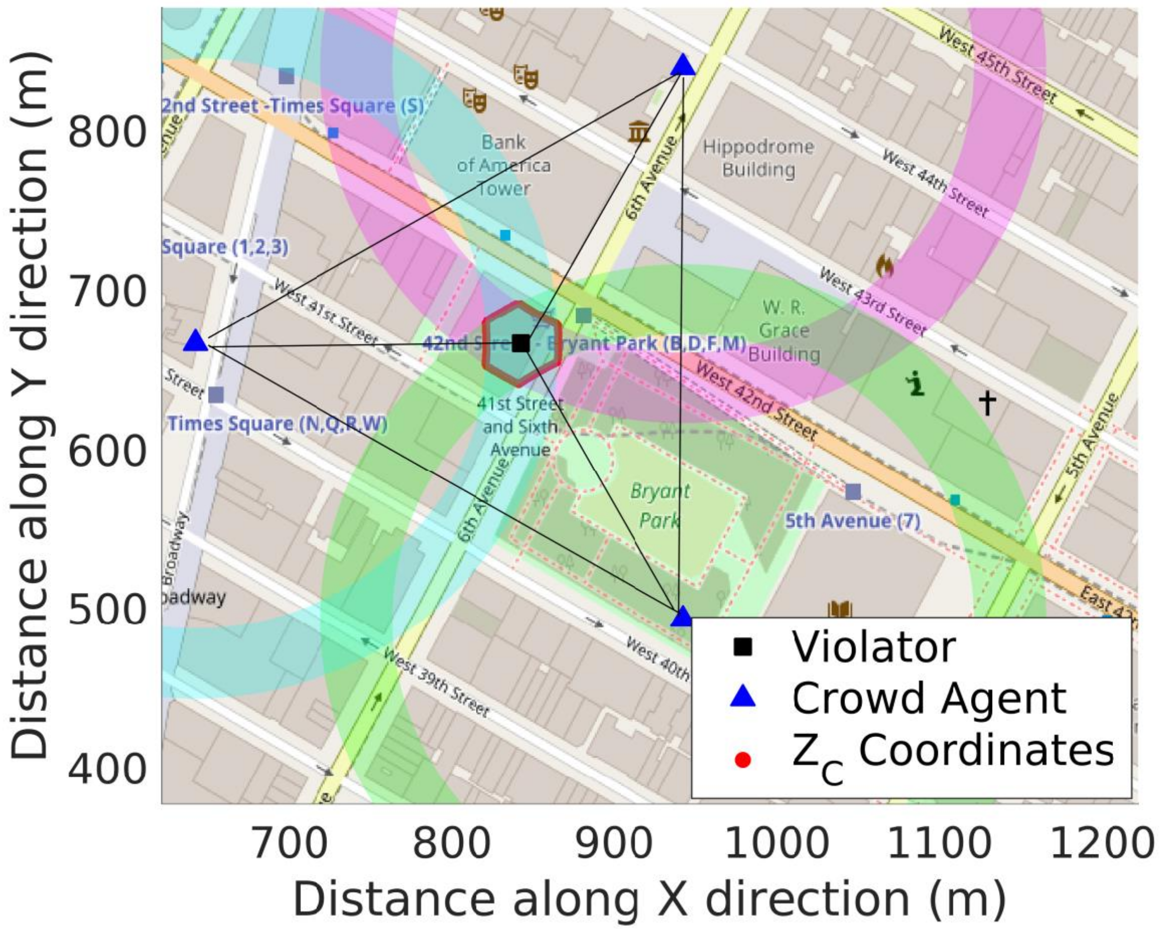


Trilateration under noise

Crowdsourced → High GDOP



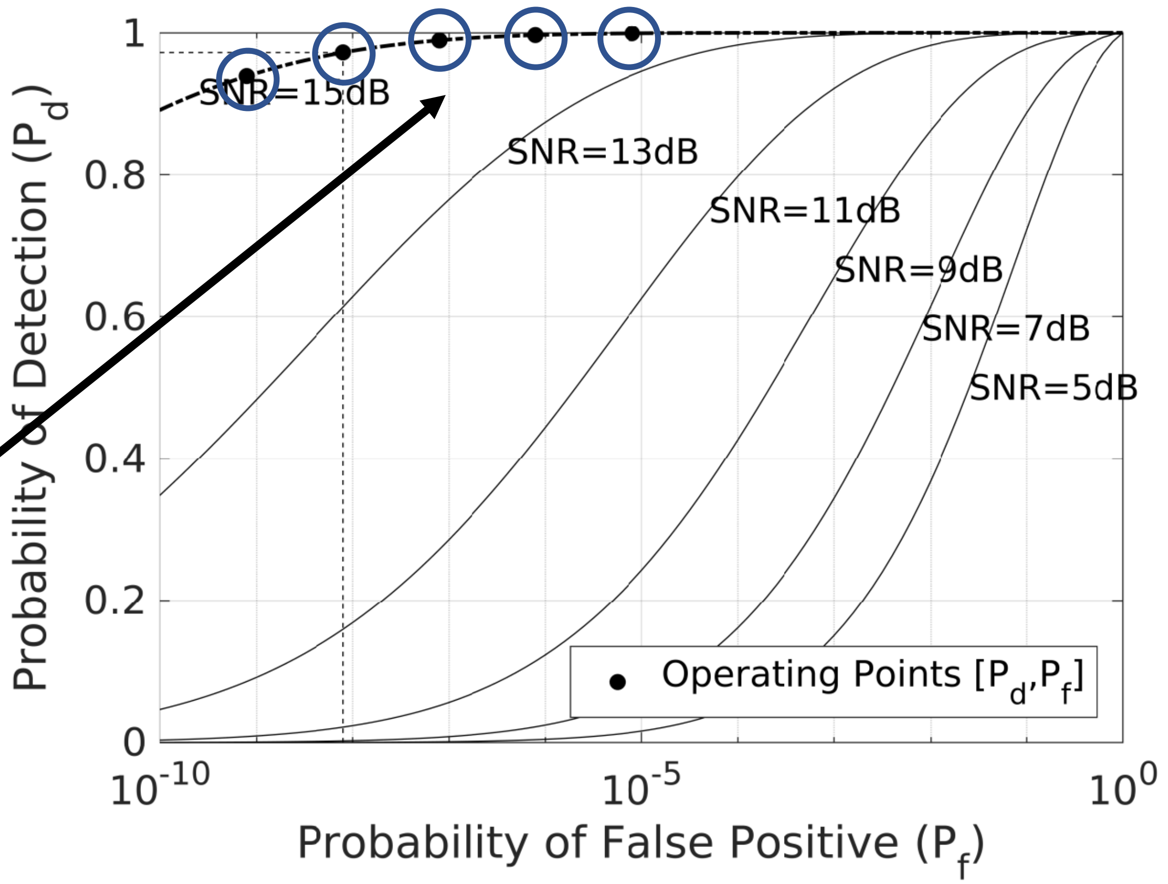
Ideal arrangement → Low GDOP



ROC and Impact on Detection

- Agents rely on a ROC curve and choose an operating point based on their SNR
- Agents can use any detector and the associated ROC
  - e.g., Neyman-Pearson ROC

Enforcers that have high SNR (closer to the target) will operate at desirable levels of [P_d, P_f], leading to maximum accuracy possible



Multi-Agent Planning with Cardinality

Algorithm 1: MPC Algorithm

Function *MPC(Map,* $\mathbb{a}$, $\mathcal{Z}_C$*)*
    $\gamma_{th} = 10m^2$; $\mathbb{t}_C$ = *getCentroids*($\mathcal{Z}_C$);
    while *True* **do**
        [$\mathcal{C}$,$\mathcal{Z}_A$] = *findCardinality*(Map, $\mathbb{t}_C$, Z_C);
        $\mathbb{t}$ = *getCentroids*($\mathcal{Z}_A$);
        $\mathcal{P}$ = *findAgentSchedule*(Map, $\mathbb{a}$, $\mathbb{t}$, $\mathcal{C}$);
        // Take measurements & evaluate actual $\mathbb{t}$
        if $\mathcal{Z}_A < \gamma_{th}$ **then** break; **else** $\mathcal{Z}_C = \mathcal{Z}_A$;$\mathbb{t}_C = \mathbb{t}$;
    end
    return $\mathcal{P}$;
end

$T = \{T_1, ..., T_m\}$ Set of m targets

$\mathbb{t} = \{t_1, ..., t_m\}$ Locations of m targets

$\mathbb{t}_C = \{t_{C,1}, ..., t_{C,m}\}$ Crowdsourced location estimates of targets

$A = \{A_1, ..., A_n\}$ Set of n agents

$\mathbb{a} = \{a_1, ..., a_n\}$ Locations of n agents

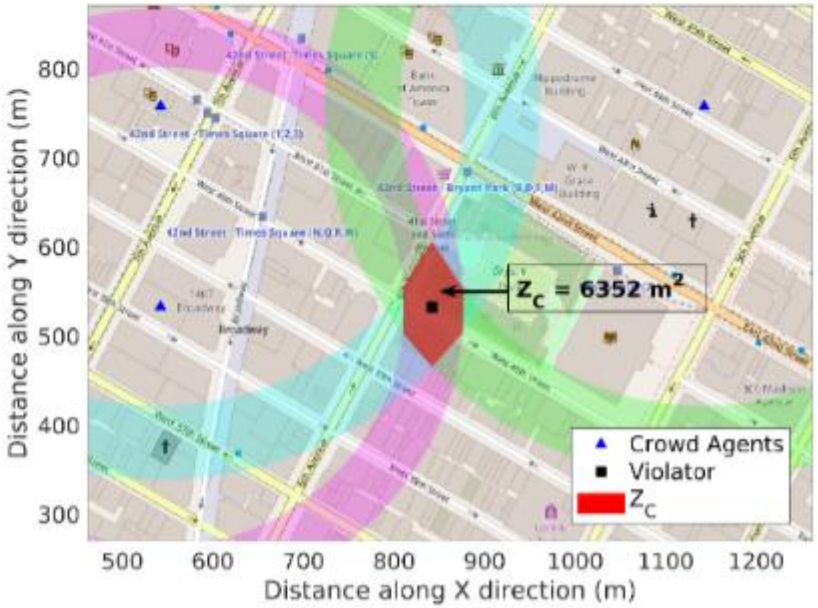
$\mathcal{Z}_C = \{\mathcal{Z}_{C,1}, ..., \mathcal{Z}_{C,m}\}$

$\mathcal{Z}_A = \{\mathcal{Z}_{A,1}, ..., \mathcal{Z}_{A,m}\}$

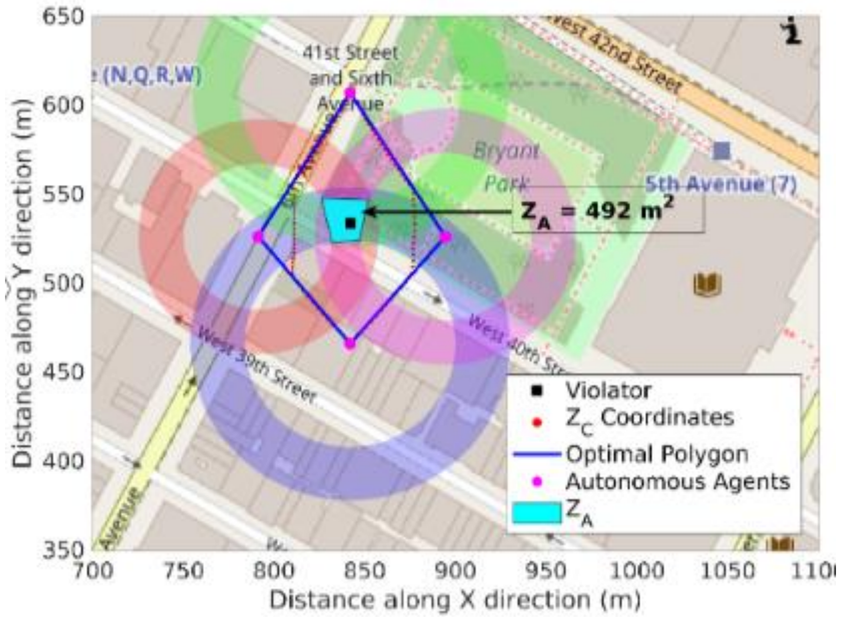
Set of m convex polygons for targets determined by crowdsourced and autonomous agent based localization

High Level View

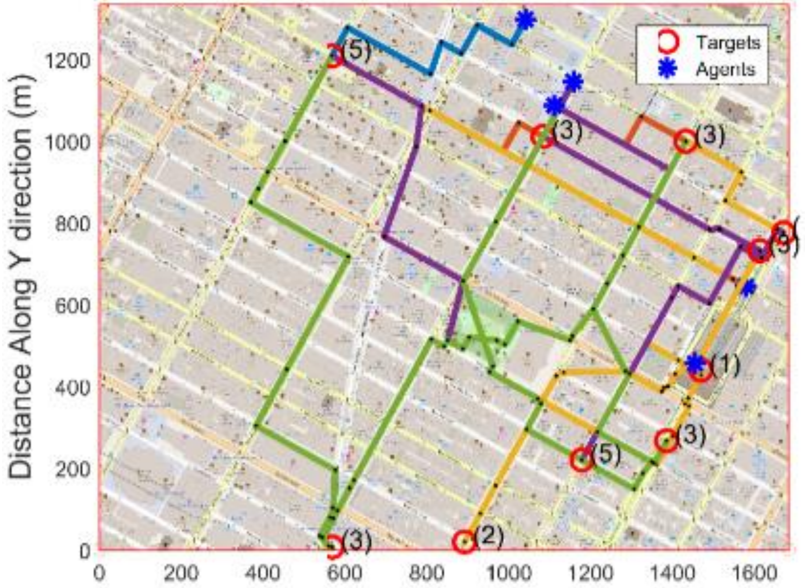
Crowd Sourced Localization



Autonomous Agent Localization



Scheduling



Optimal polygon circumscribing Z_c
92% Improvement

Schedule optimal number of agents to all targets in **minimum time**

Step-A: Determination of Cardinality

Step-A: Determination of Cardinality

Definition 1: Cost of Localization

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

- # of agents deployed to target T_j
- Polygon circumscribing Z_c
- Trade-off factor, range [0 - 0.1]
- Polygon from crowd localization

Definition 2: Cardinality

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

- *i* typically varies between 3 and 8
- Diminishing return beyond that

Algorithm to determine Cardinality

Goal: To find an optimal number and placement of agents for each target.

Idea: Find the optimum Polygon that circumscribes Z_c, then deploy agents to the vertices of this polygon for low GDOP and high accuracy

---

Start with Location estimate from crowd - Convex Polygon $\mathbf{Z_c}$

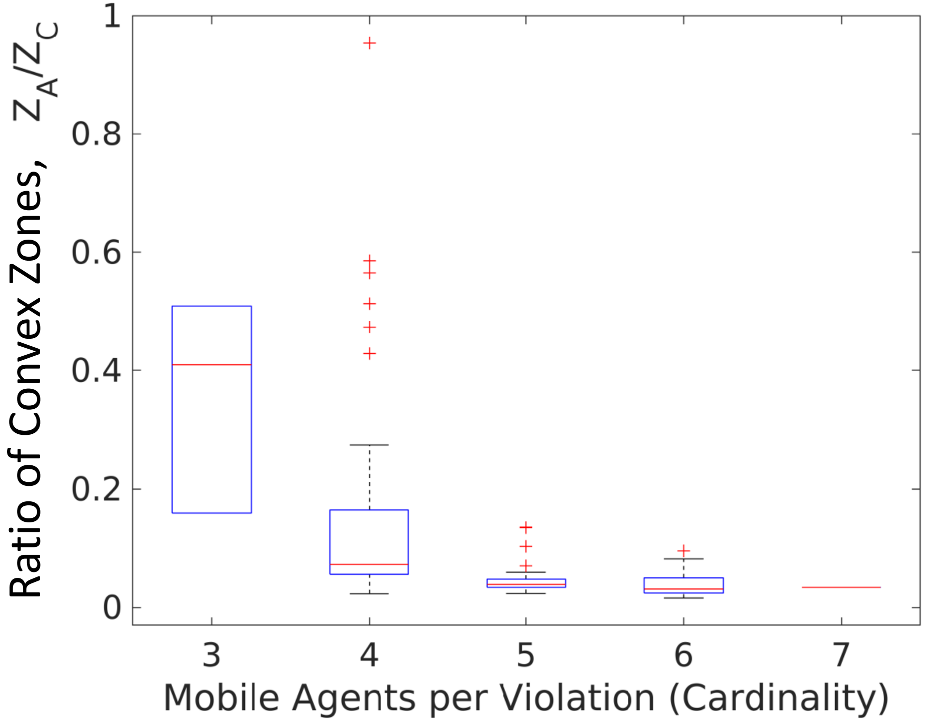
Step 1: For **3 : maxAgents (# Edges of $\mathbf{Z_c}$)**,

- **Step 1a:** Find smallest polygon (minPoly) that circumscribe Z_c
- **Step 1b:** Perform trilateration with agents positioned at **vertices of minPoly ($\mathbf{Z_A}$)**
- **Step 1c:** Calculate Cost of Localization

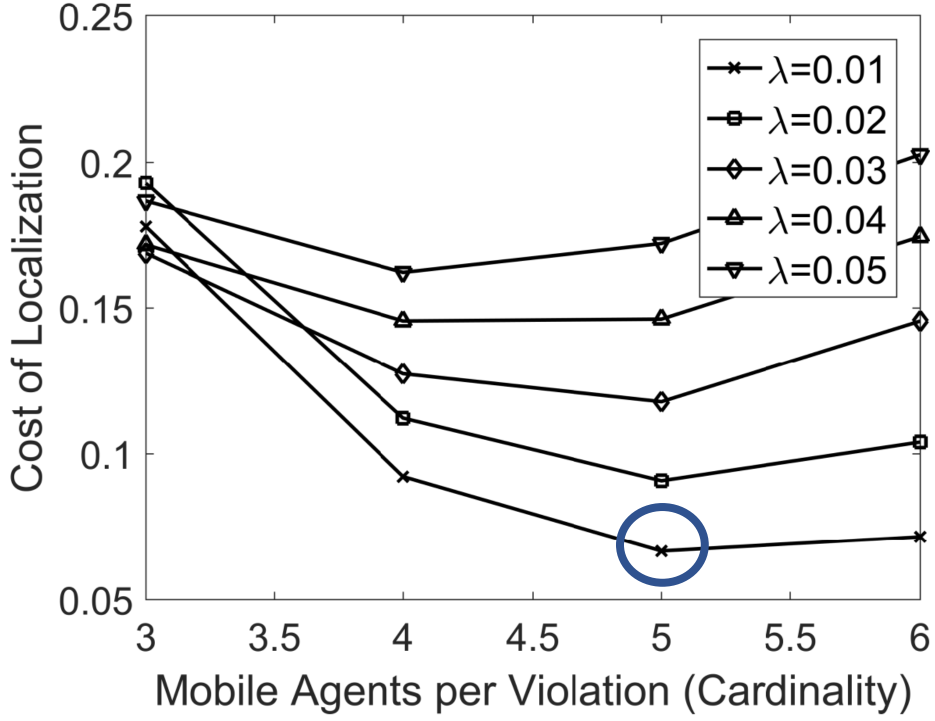
Step 2: Optimal Polygon = **minPoly** with **least Cost of Localization**

Step 3: Cardinality = **# Sides** of **Optimal Polygon**

Impact on Localization



(a) Ratio of $\mathcal{Z}^i_{A,j}/\mathcal{Z}_{C,j}$ with Cardinality



(b) Cost of Localization vs Cardinality.

Accuracy and cost of localization. For λ = 0.01, the optimal **cardinality is 5** and the median reduction in the **area of the convex polygon is 96%**.

Step-B: Schedule Autonomous Agents

Step-B: Schedule Autonomous Agents

Definition 3: Cost of Scheduling

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

P_i - Path of agent a_i of length l_i

$c(P)$ - Cost of a path with *k* vertices = Sum of Edge Weights

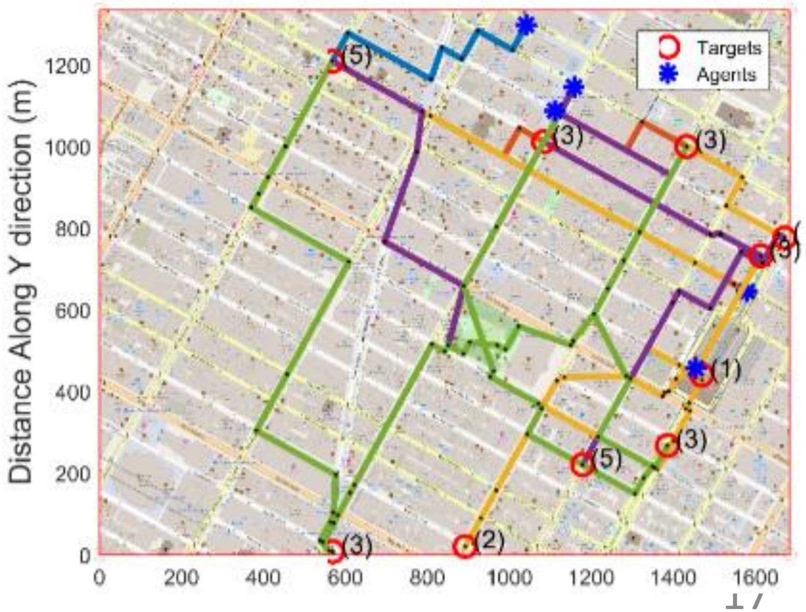
$$c(P) = \sum_{i=1}^{k} w(x_i, x_{i+1})$$

Definition 4: Uniqueness

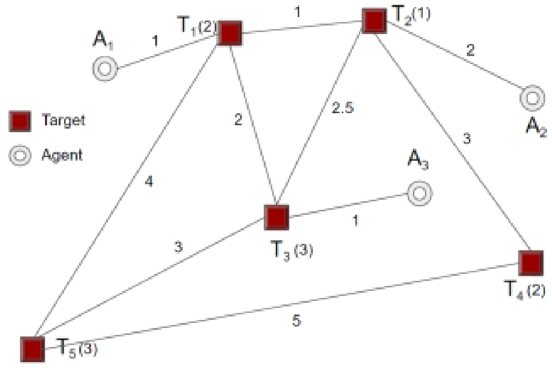
Distinct agents visit each target to fulfill its cardinality

Definition 5: Schedule

The set of paths $\mathcal{P} = \{P_1, ..., P_n\}$ of all *n* agents to visit *m* targets in the shortest possible time while fullfilling the cardinality of each Target.

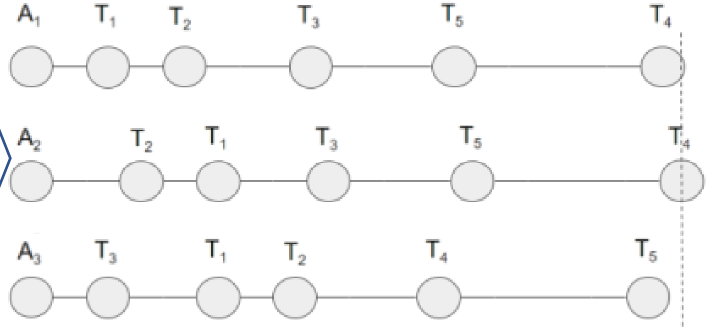


Algorithm for the Schedule

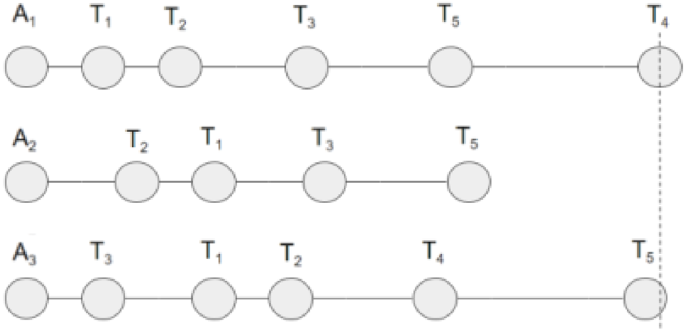


(a) City map with 3 agents, 5 targets with different cardinality and edge weights

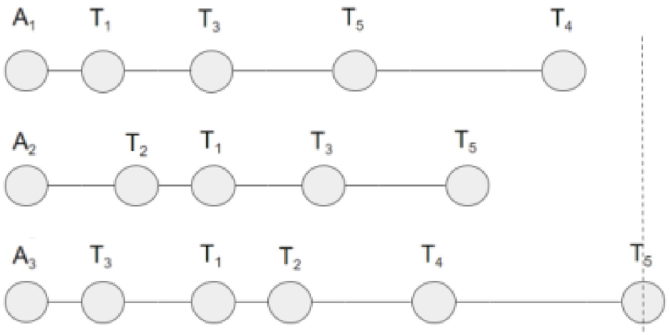
1. Compute shortest distances from each agent to all targets.
2. Apply TSP to each agent to find shortest paths to all targets.



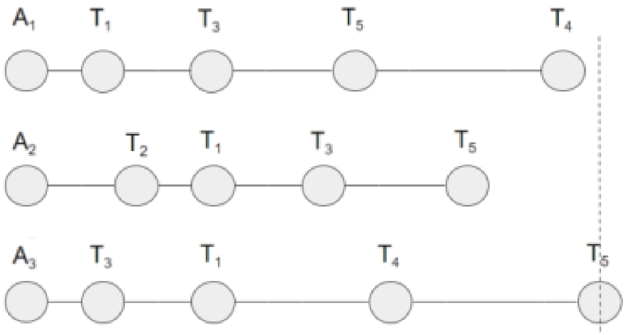
(b) Iter 1: Initial Path Estimate: A_2-costliest agent, T_4-farthest redundant target



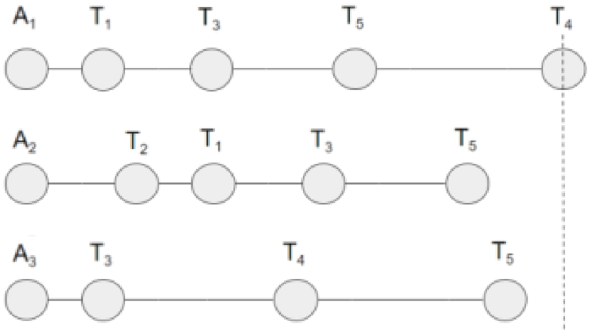
(c) Iter 2: Remove T_4 from A_2's path. A_1-costliest agent, T_2-farthest redundant target



(d) Iter 3: Remove T_2 from A_1's path. A_3-costliest agent, T_2-farthest redundant target



(e) Iter 4: Remove T_2 from A_3's path. A_3-costliest agent, T_1-farthest redundant target



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

Analysis of the Scheduling Algorithm

Claim 1: The Schedule is NP-hard.

Lemma 1: Algorithm for Schedule has complexity of $O(nm^4)$ where n is the number of agents and m is the number of targets.

Approximation Ratio for Scheduling Algorithm

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 $P_p \not\subseteq$ the targets in P_p^*.

Property 1. *If $T_y^i = 0$, then l_i is no worse 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, must not 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 must be such that $l_i + l_i(t_k) \geq l_p$.*

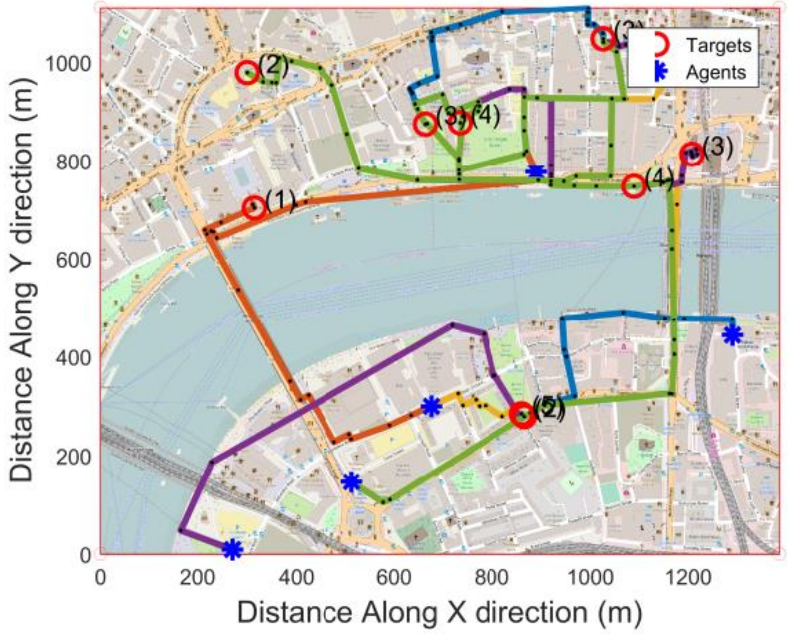
Property 5. *From Table I, we can express the costs l_i and l_i^* of agent i as,*

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

Performance Evaluation

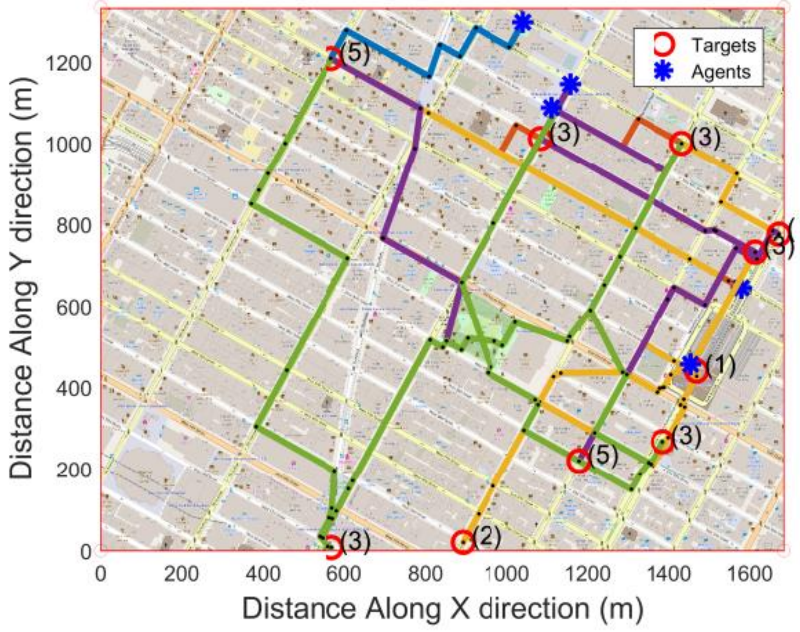
- Scheduling Costs in different cities

London



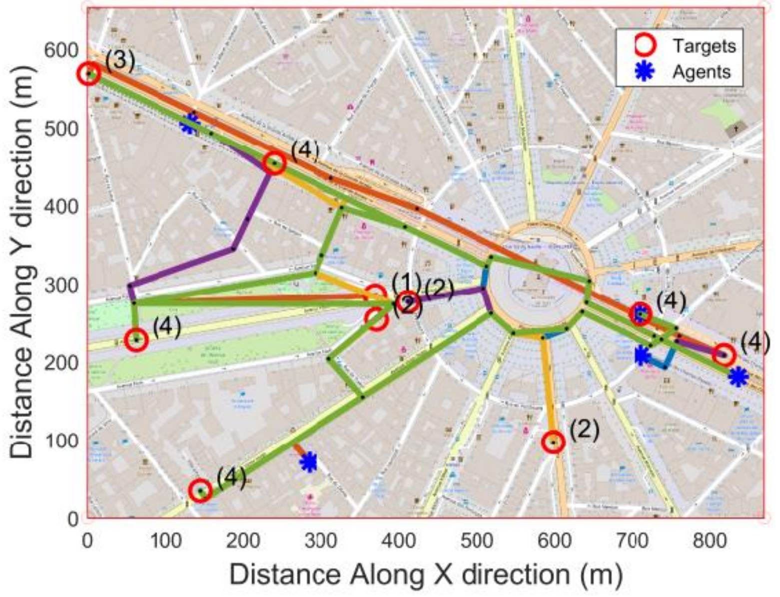
(a) Example routing in London. Cost metric = $2.618 km/km$

New York



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

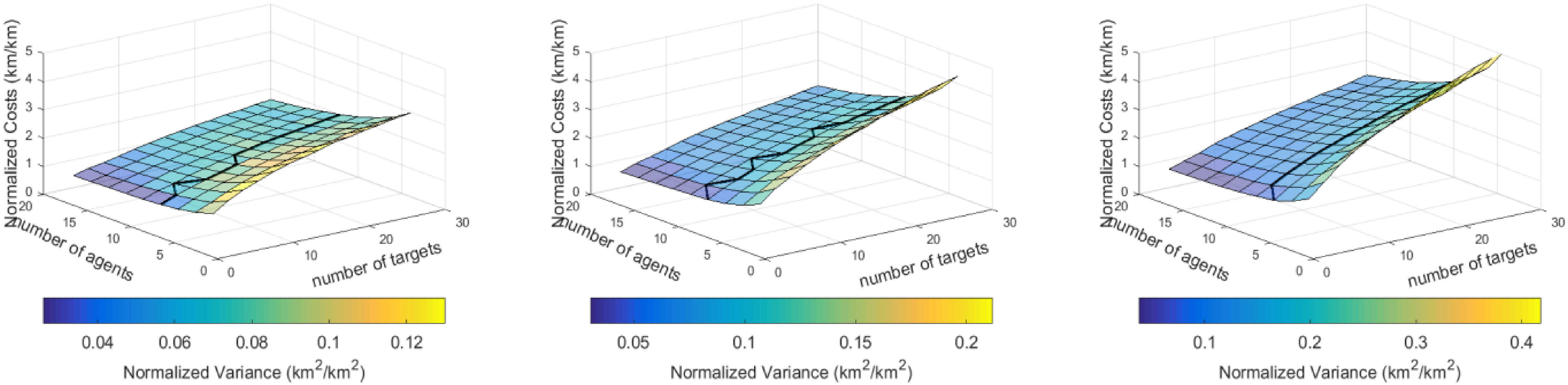
Paris



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

1. Constant Average Cardinality

London New York Paris

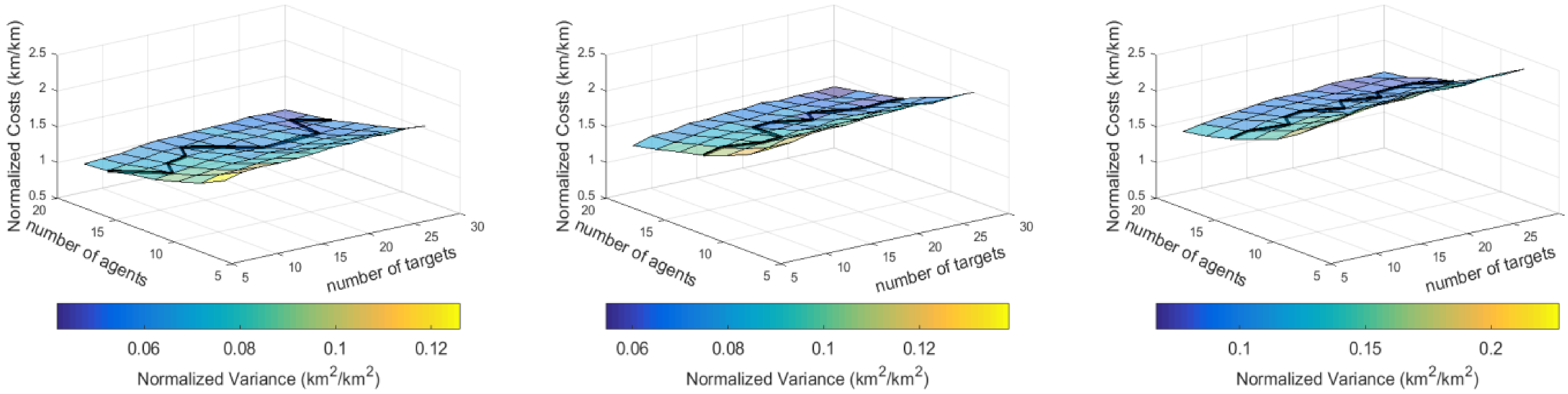


(a) Normalized cost metric in London (b) Normalized cost metric in NYC (c) Normalized cost metric in Paris

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.

2. Constant Total Cardinality

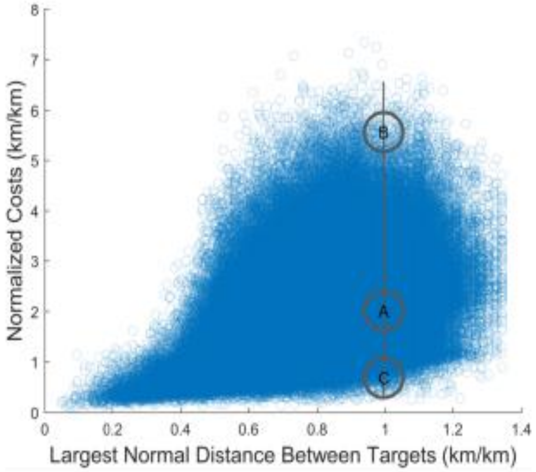
London New York Paris



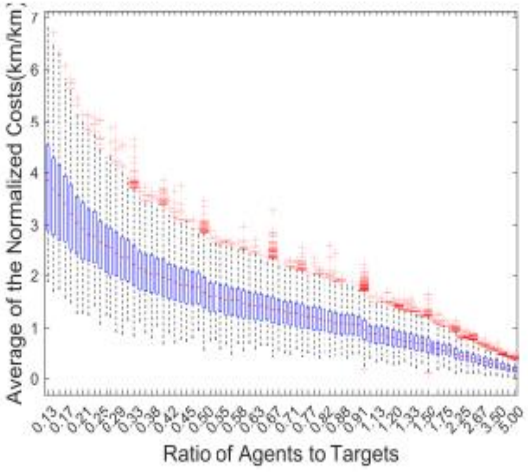
(a) Normalized cost spread in London (b) Normalized cost spread in NYC (c) Normalized cost spread in Paris

Fig. 4: Normalized cost metric for **Total Visits = 40**. The dark line highlights the points beyond which the cost variation is below 10%. The variance is indicated using the color scale.

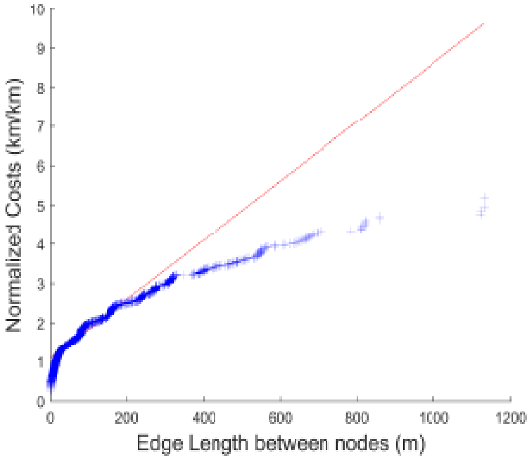
Overall System Performance



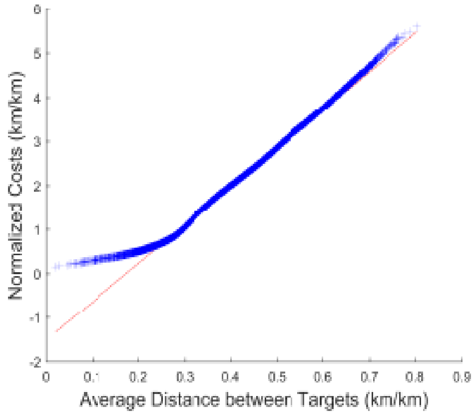
(a) Variation of Cost with maximum separation of targets.



(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.

Related Work

- Multiple Traveling Salesmen Problem (MTSP) [NP hard] [1] → Multiagent Planning with uniqueness (MPU) → But, no Cardinality
- Vehicular Routing Problems (VRP/MDVRP) → Uncontrolled number of multiple visits → No Uniqueness [2]

[1] "The multiagent planning problem," Tamas et. Al
[2] "A review of dynamic vehicle routing problems," Pillac et. Al

Conclusions

Through simulations and analysis, we draw three firm conclusions:

- The algorithm is polynomial and provides the shortest paths for the agents while conforming to the cardinality.

- The algorithm has a provable bound of 3-approximation ratio.

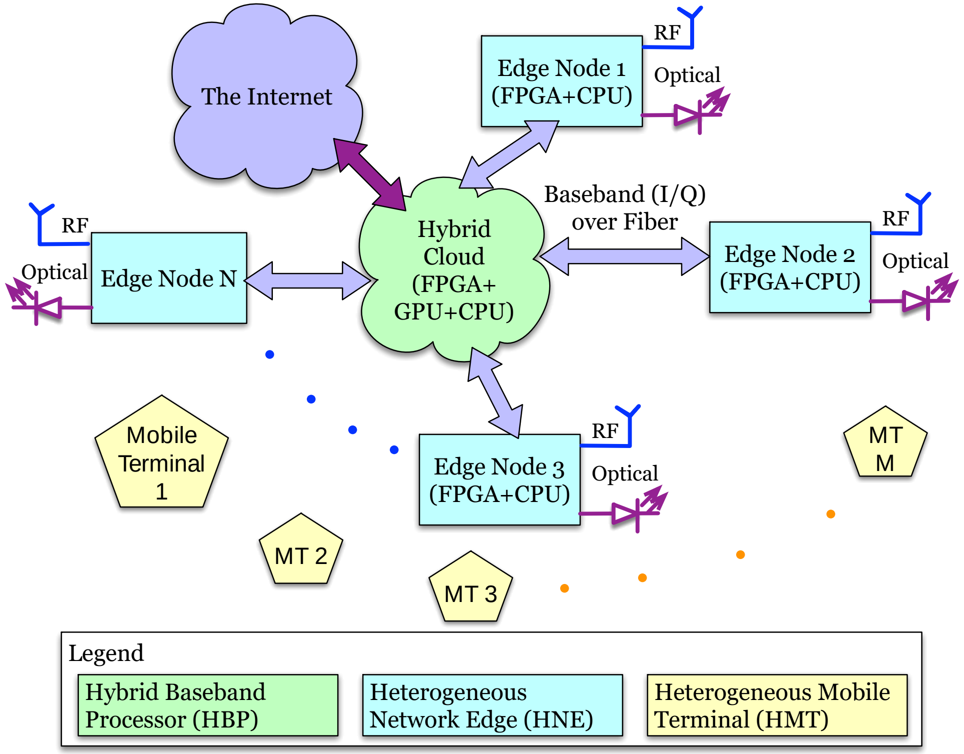
- It exhibits strong generality across different geographical regions, by producing statistically similar results for varying degree of violations.

CHRONOS

https://www.albany.edu/mesalabs/

NSF Award # 1823225 CRI: II-NEW: CHRONOS : Cloud based Hybrid RF-Optical Network Over Synchronous Links

Co-PIs: Drs. Dola Saha, Aveek Dutta and Hany Elgala



Baseband Processing in Quad-FPGA Blade Server

Thank you