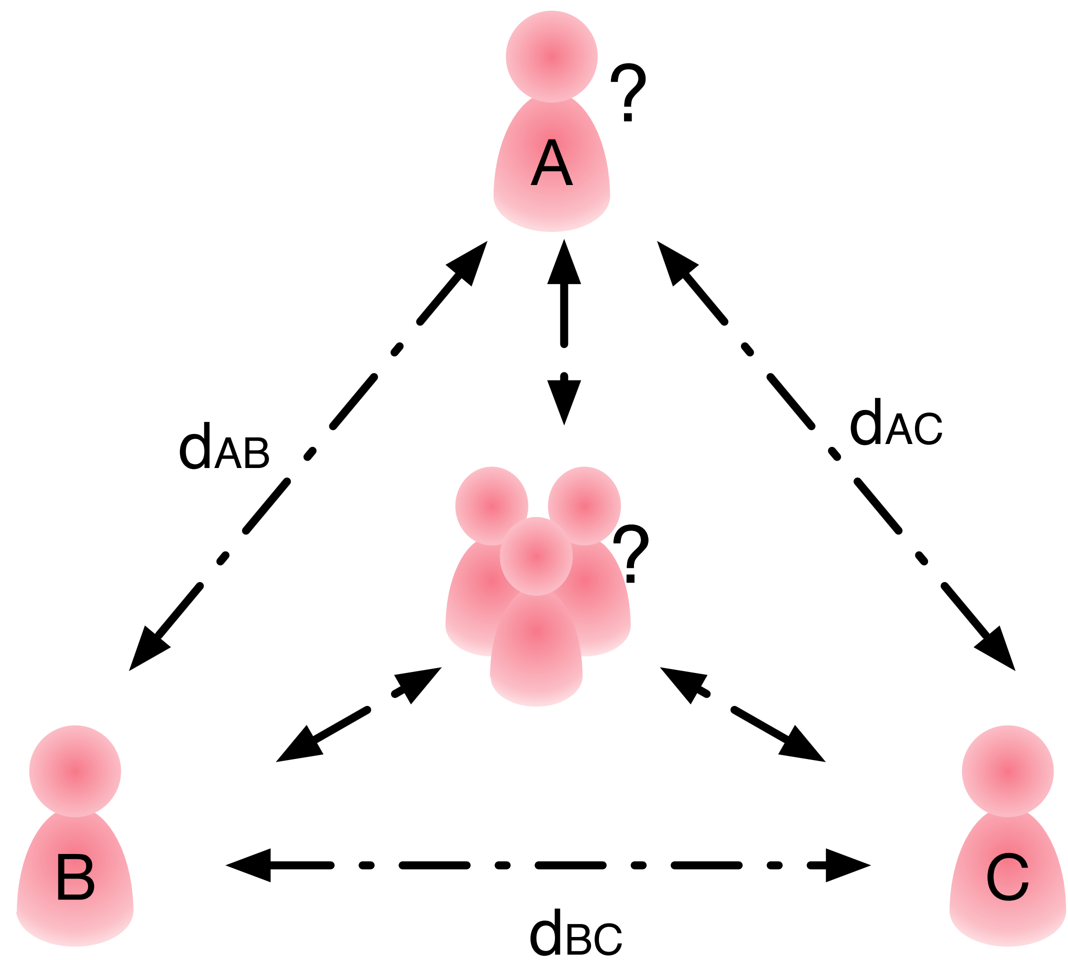
Privacy-Preserving Inference in Crowdsourcing Systems

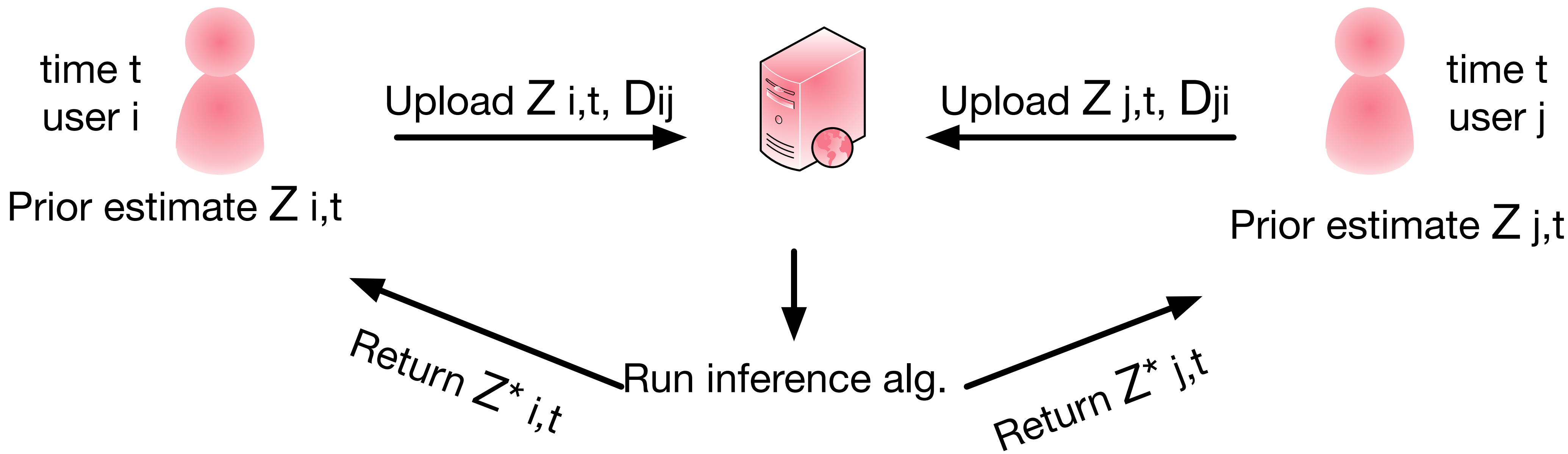
Liyao Xiang
Supervisor: Baochun Li
Oct. 9, 2017
University of Toronto

Localization via Crowdsourcing



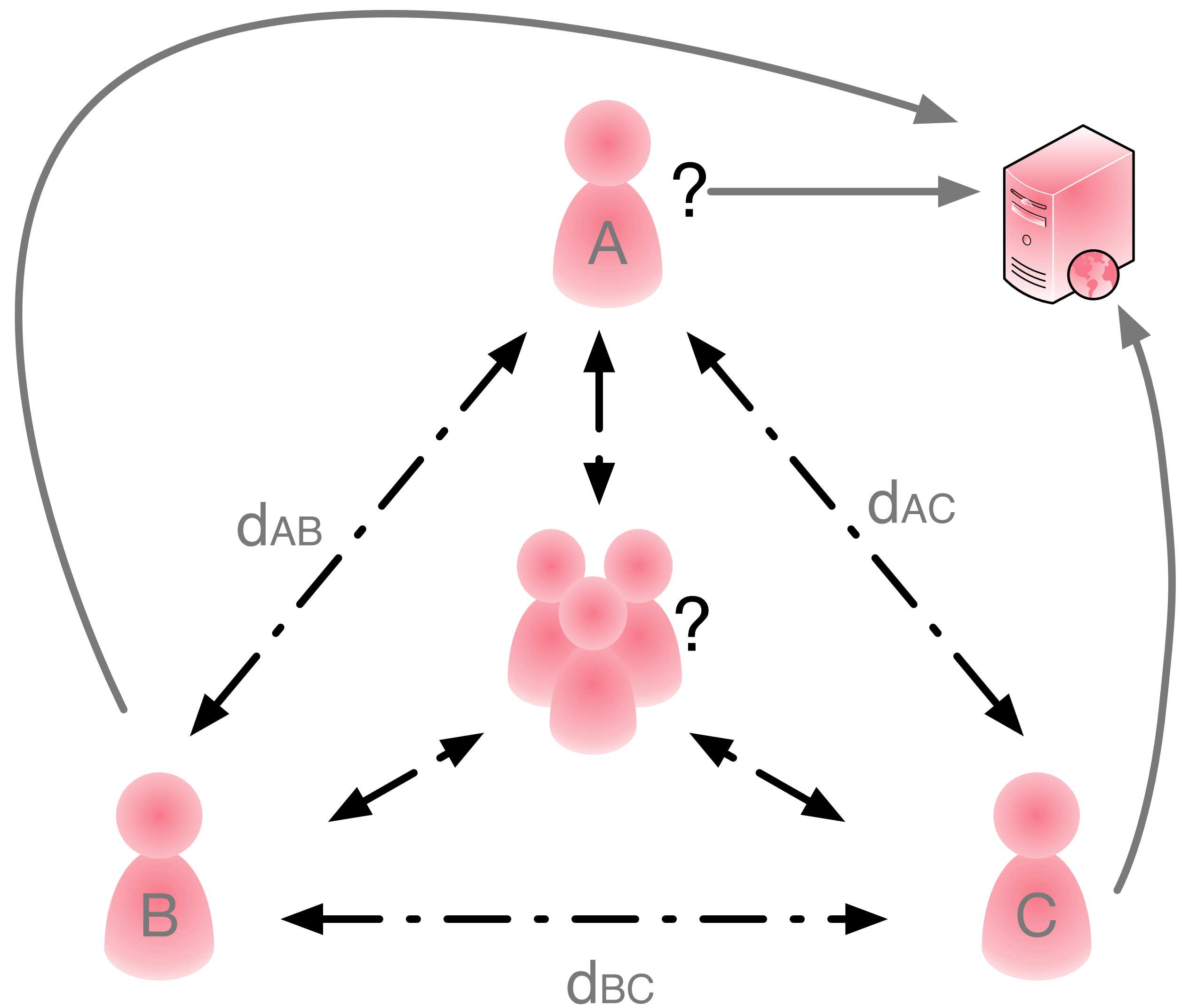
- In a crowd, some users know about their locations while some don't. With distance observations between them, how to localize each user?

Localization via Crowdsourcing



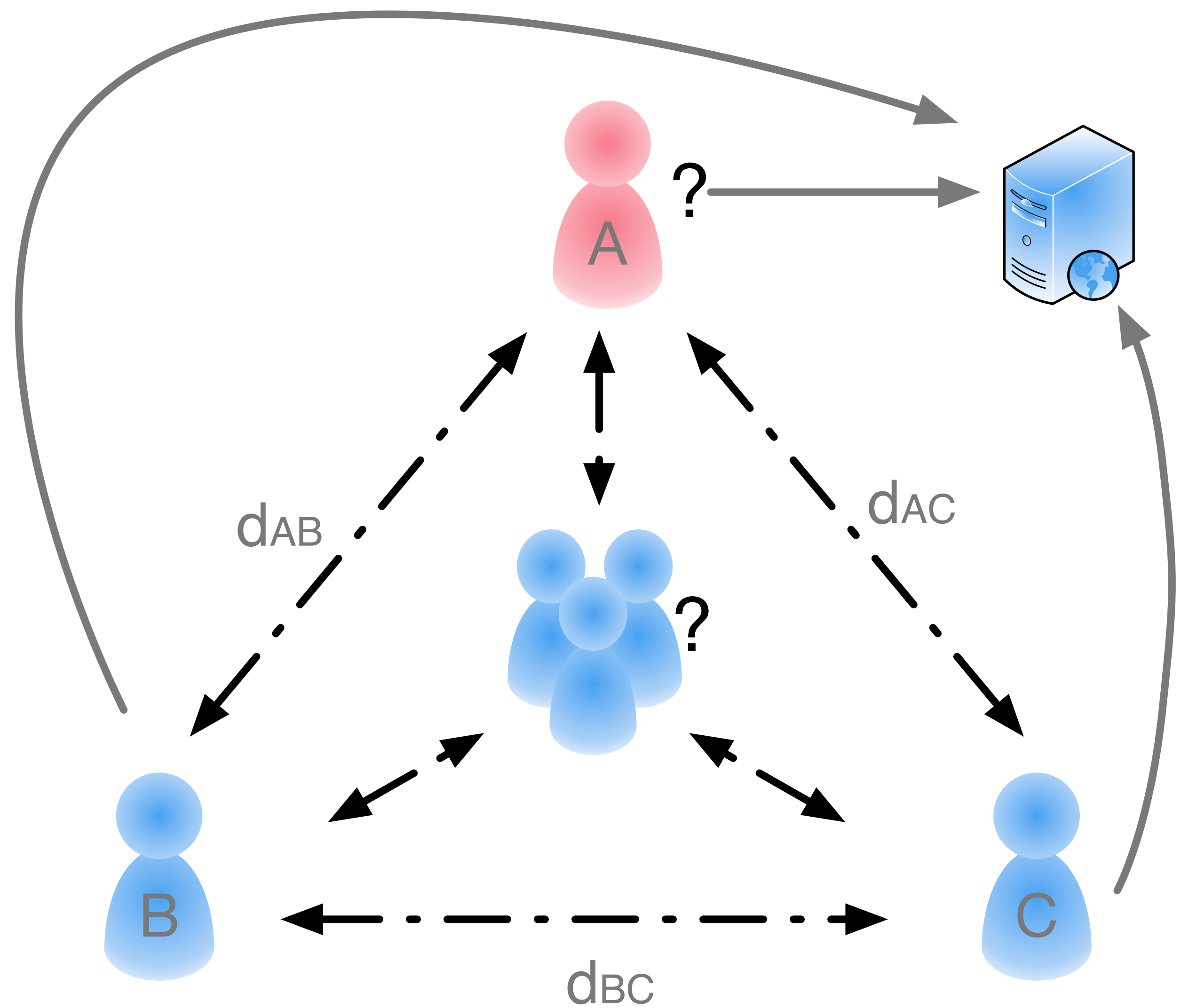
- Each user sends their prior estimates and distance observations to a central server, who returns the most likely position for each.
  - What if users would like to keep their locations private?

Privacy-Preserving Localization



- In a crowd, some users know about their locations while some don't. With distance observations between them, how to localize each user without breaching privacy?

Privacy-Preserving Localization



- In a crowd, some users know about their locations while some don't. With distance observations between them, how to localize each user without breaching privacy?

Particle Representation

- User's Location
  - A user's location is represented by a set of particles $\mathbf{Z}_{i,t} = \{z_1, \ldots, z_R\}$, $\mathbf{Z}_t = \{\mathbf{Z}_{1,t}, \ldots, \mathbf{Z}_{N,t}\}$.
  - At time t, the server finds the most likely distribution of $\mathbf{Z}_t$ given $\mathbf{Z}_{t-1}$ and $\mathbf{D}$.

$$\mathbf{Z}_t^* = \arg\max_{\mathbf{Z}_t} P(\mathbf{Z}_t | \mathbf{Z}_{t-1}, \mathbf{D}).$$

First Attempt

- To encrypt all particles and run the inference in the encrypted domain.

However, encrypted operations are constrained.

Particle Representation

- User's Location
  - A user's location is represented by a set of particles $\mathbf{Z}_{i,t} = \{ z_1, \ldots, z_R \}$. Each particle is associated with a weight $\{ w_1, \ldots, w_R \}$.
  - For example, if the location estimate is $\{z_1, z_2, z_3\}$ with probabilities {0.6, 0.2, 0.2}, then the location is more likely to be z_1 than z_3.

Particle Representation

- Users upload each particle's weight $\{E(W_1), \ldots, E(W_R)\}$ and distance observations to others E(D) in encryption.
- Server updates each particle's weight.

Privacy-Preserving Inference

- Server computes partial information Ci,r for each particle r of each user i (j is observed by i):

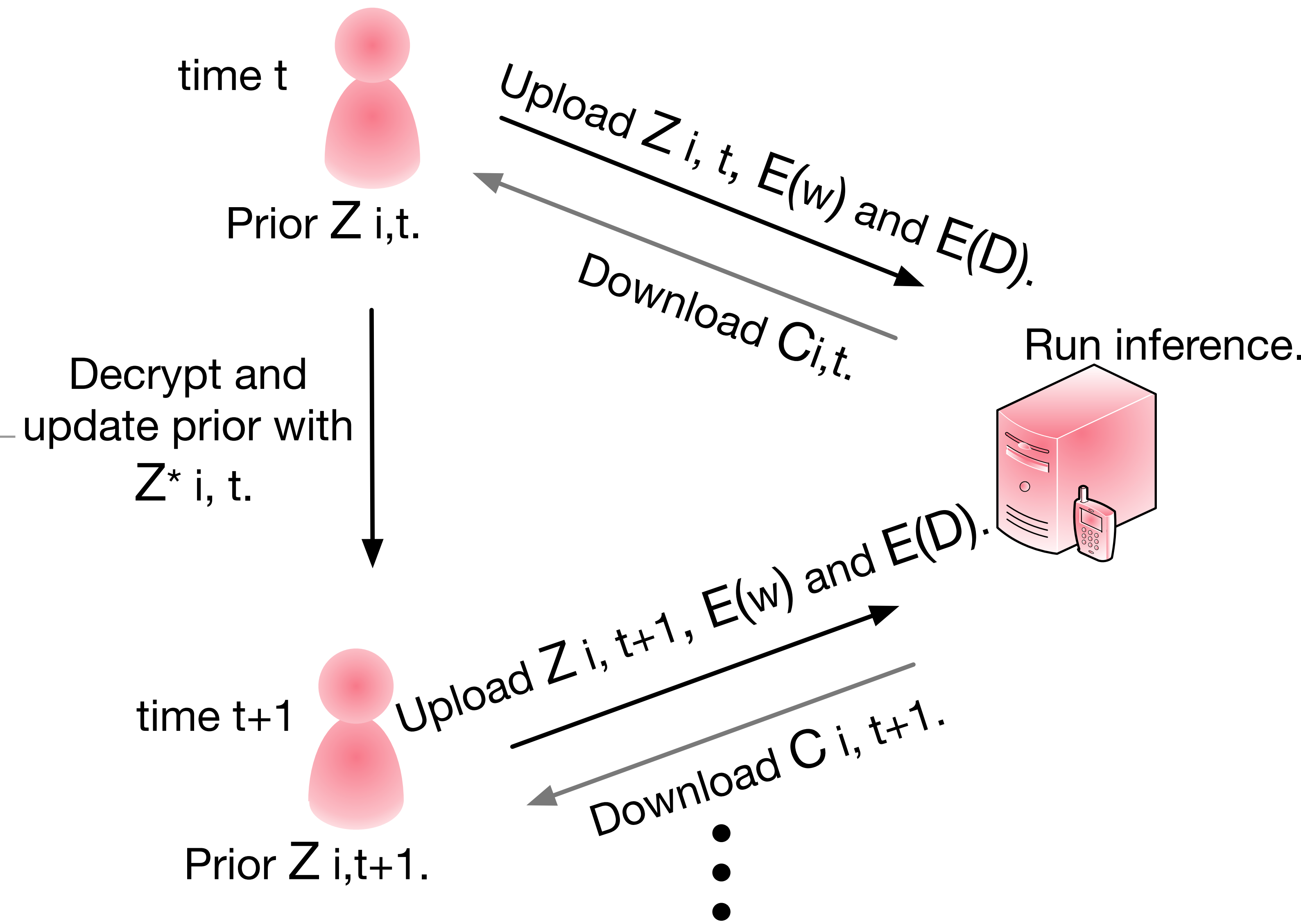
$$
\begin{aligned}
c_{i,r} = & \prod_{j \in \mathcal{N}(i)} \prod_{s \in \{1,\ldots,R\}} E_{pk}(\ln w_{j,s}) \cdot E_{pk}(d(z_{i,r}, z_{j,s})^2)^{-\frac{1}{2\sigma^2}} \\
& \cdot E_{pk}(D_{ij})^{\frac{d(z_{i,r}, z_{j,s})}{\sigma^2}} \cdot E_{pk}(D_{ij}^2)^{-\frac{1}{2\sigma^2}} \\
& = E_{pk}[\sum_{j \in \mathcal{N}(i)} \sum_{s \in \{1,\ldots,R\}} (\ln w_{j,s} - (d(z_{i,r}, z_{j,s}) - D_{ij})^2 / 2\sigma^2)].
\end{aligned}
$$

Privacy-Preserving Inference

- With secret key sk, user i updates the weight Wi,r for its particle r (djs is the calculated distance between particle s of user j and particle r of user i):

$$
\begin{aligned}
w_{i,r}^{k} &= w_{i,r}^{k-1} \exp[E_{sk}(c_{i,r})] \\
&= w_{i,r}^{k-1} \exp[\sum_{j \in \mathcal{N}(i)} \sum_{s \in \{1,\ldots,R\}} (\ln w_{j,s} - (d_{js} - D_{ij})^2/2\sigma^2)] \\
&= w_{i,r}^{k-1} \prod_{j \in \mathcal{N}(i)} \prod_{s \in \{1,\ldots,R\}} \exp(\ln w_{j,s} - (d_{js} - D_{ij})^2/2\sigma^2) \\
&= w_{i,r}^{k-1} \prod_{j \in \mathcal{N}(i)} \prod_{s \in \{1,\ldots,R\}} w_{j,s} \cdot \exp\Big(- \frac{(d_{js} - D_{ij})^2}{2\sigma^2} \Big) \\
&\simeq w_{i,r}^{k-1} \prod_{j \in \mathcal{N}(i)} \prod_{s \in \{1,\ldots,R\}} \Pr(z_{i,r}, z_{j,s} | D_{ij,t}).
\end{aligned}
$$

Privacy-Preserving Localization with Crowdsourcing



But, with R particles, adversary can still guess correct location with Prob. 1/R.

Data Perturbation

- Idea: perturb $\mathbf{Z}_{i,t} = \{ z_1, \ldots, z_R\}$ as $\mathbf{Y}_{i,t} = \{ y_1, \ldots, y_R\}$.
- Perturbation: add Gaussian noise $\mathcal{N}(0, \sigma^2)$ to $\mathbf{Z}_{i,t}$ that satisfies location differential privacy.

Privacy Definition

- Location Differential Privacy:

A mechanism M satisfies (ϵ, δ)-differential privacy iff for all z, z' that are $d(z, z')$ apart:

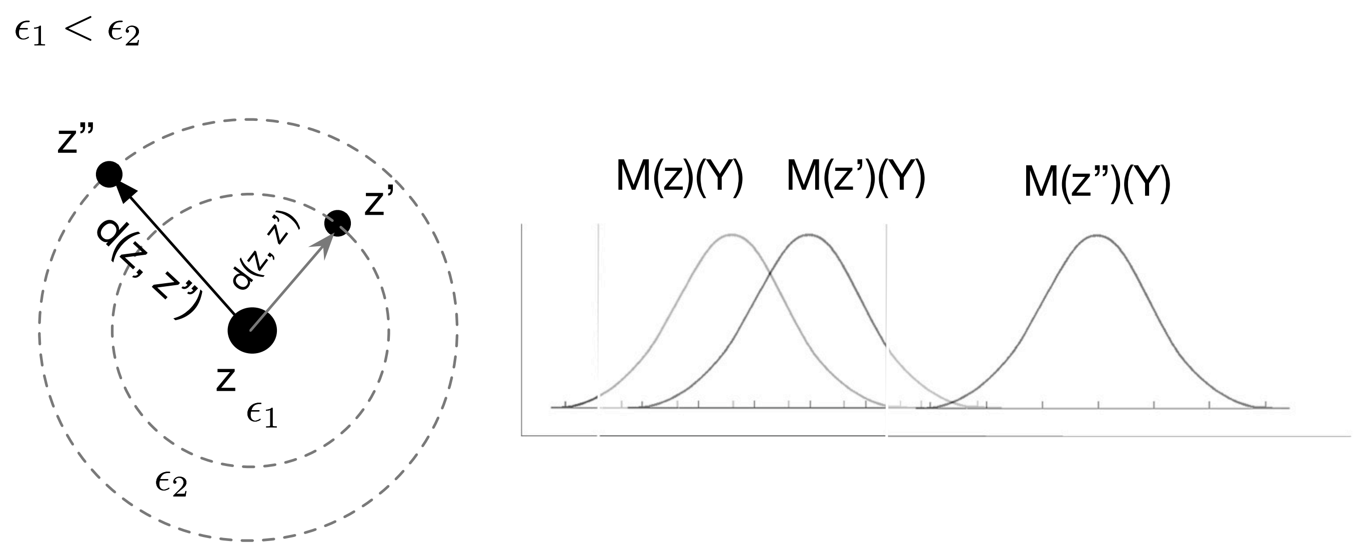
$$Pr[M(z) \in Y] \leq e^{\epsilon} Pr[M(z') \in Y] + \delta,$$

$$\text{and } \epsilon = \rho d^2(z, z') + 2\sqrt{\rho \log(1/\delta)} d(z, z'),$$

where ρ is a constant specific to the perturbation mechanism we adopt.

Interpretation of Privacy Definition

- Location Differential Privacy: the projected distributions of all the points within the same dotted circle are at most ϵ apart from each other.



- As the distance between the two locations is smaller, ϵ is smaller, indicating that it is harder to distinguish the two locations, i.e., higher privacy level.

Privacy Definition

- User Differential Privacy

If we report $Z = (z_1, ..., z_R)$ as $Y = (y_1, ..., y_R)$, then the probability of reporting Y given Z is:

$$Pr[\mathbf{M}(Z) \in \mathbf{Y}] = \prod_i Pr[M(z_i) \in Y].$$

The user enjoys (ϵ', δ)-differential privacy with

$$\epsilon' = \rho R d^2(Z, Z') + 2\sqrt{\rho \log(1/\delta) R d^2(Z, Z')}.$$

Perturbed Private Inference

- Collecting **Y**, the server computes the pairwise distances between each pair of perturbed particles as:

$$\tilde{d}(y, y') = \sqrt{||y - y'||_2^2 - 4\sigma^2}.$$

How can we guarantee the inference result the same with the unperturbed case?

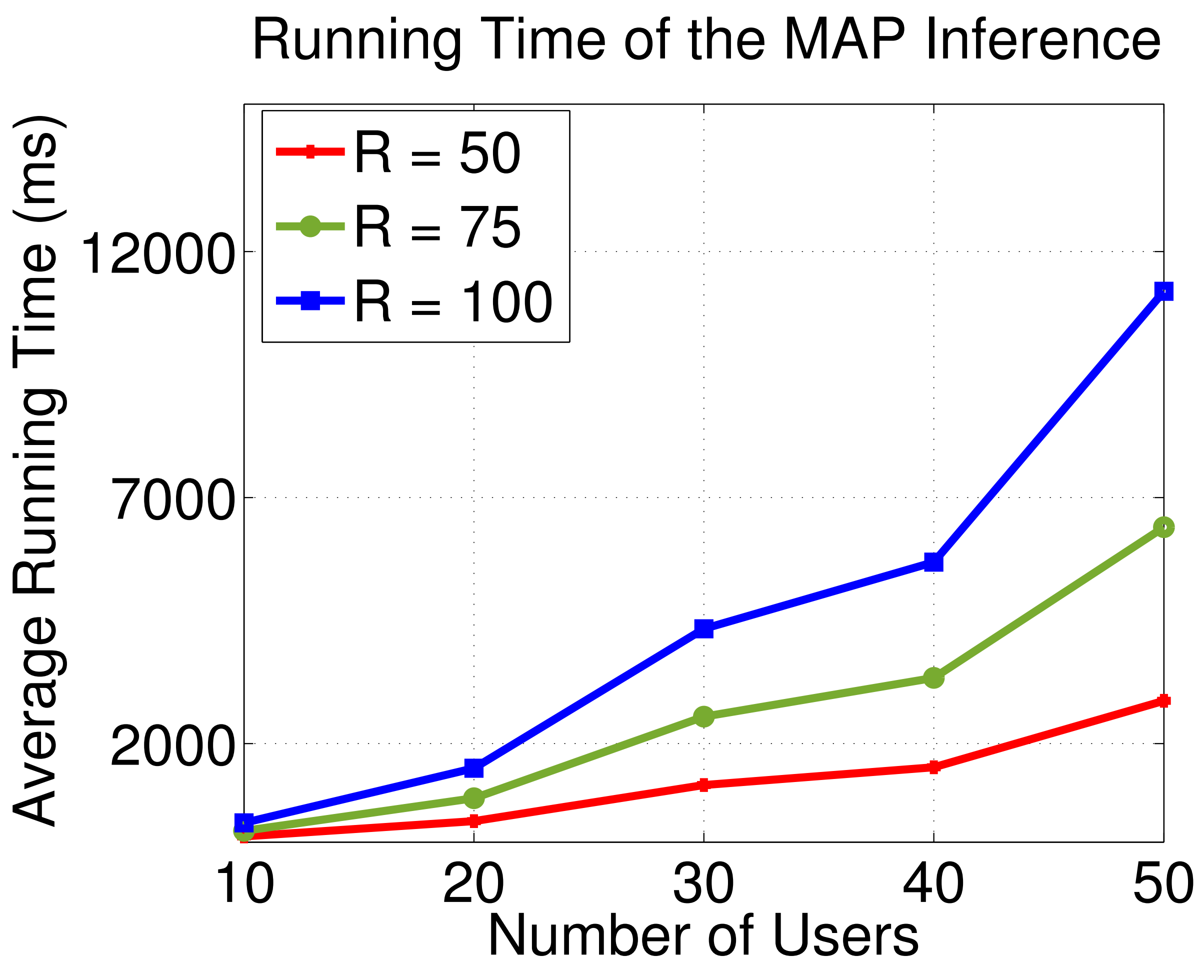
Privacy and Utility Analysis

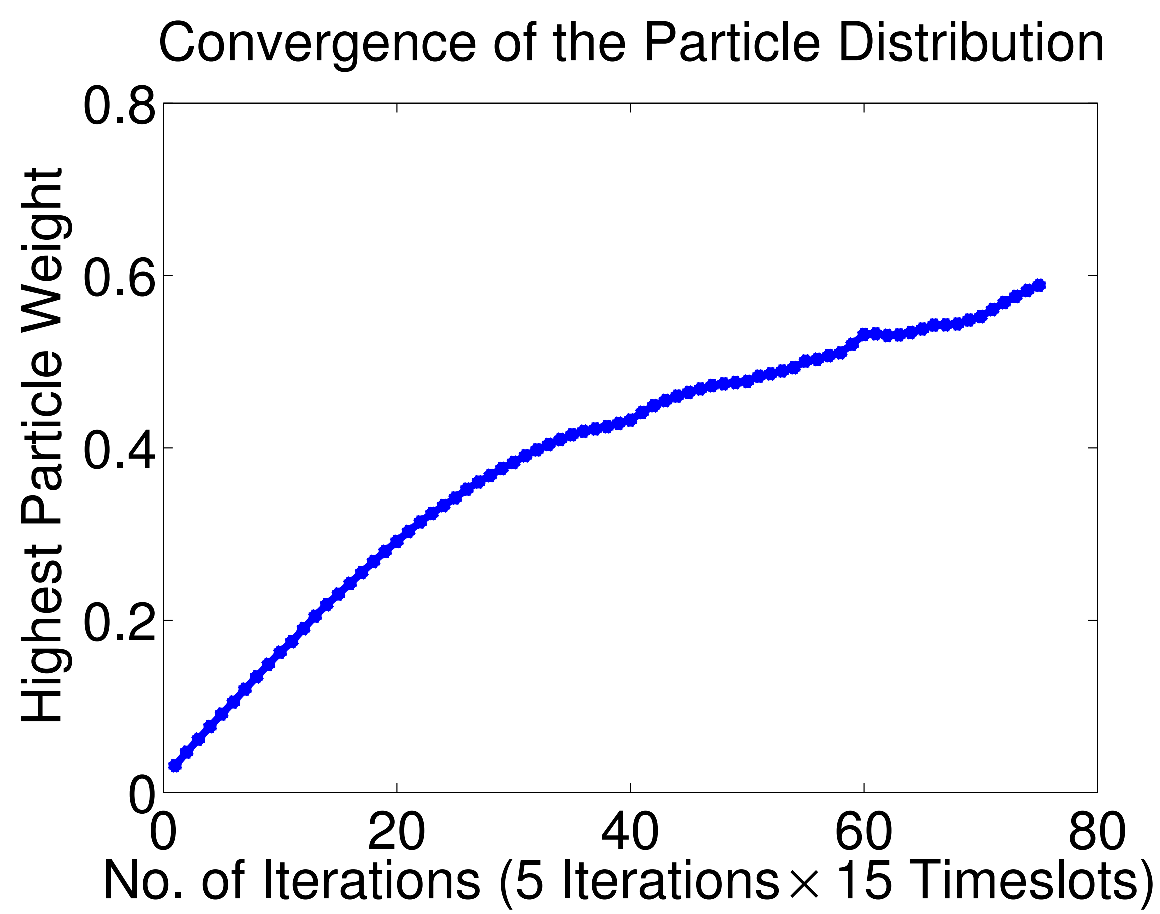
- Utility results: We proved $\tilde{d}(y, y')$ is an unbiased estimator of $d(z, z')$

- Privacy guarantee: We proved our perturbation scheme satisfies location differential privacy and user differential privacy. Compared to previous work, we improve the privacy level by $\sqrt{R}$ with the same utility level.

Performance Evaluation

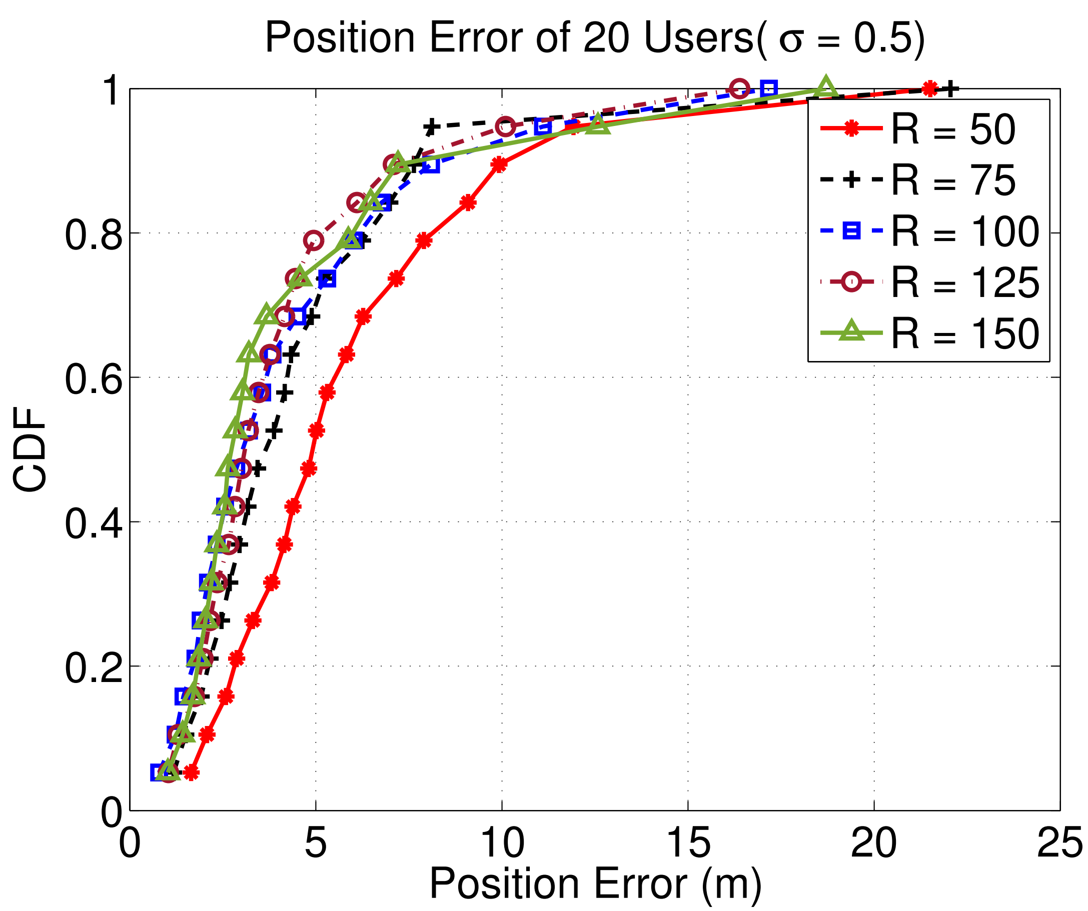
- Overhead

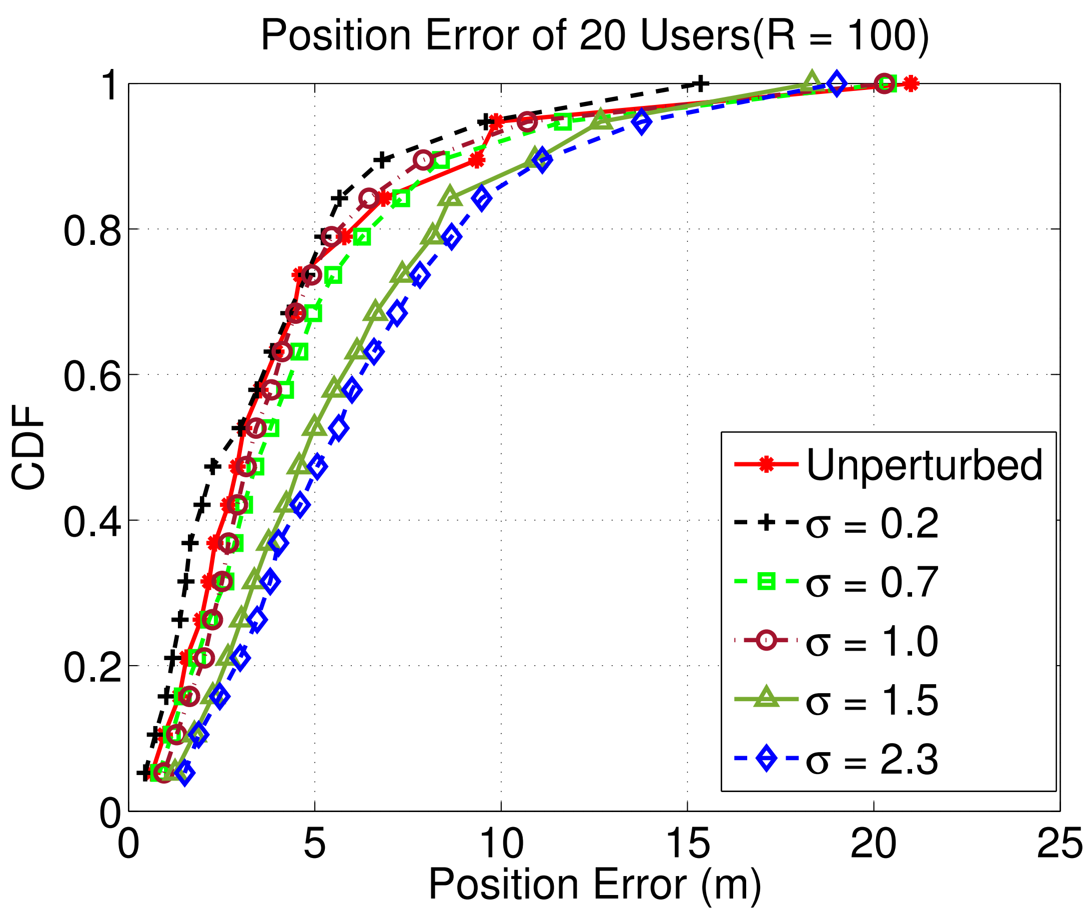




Performance Evaluation

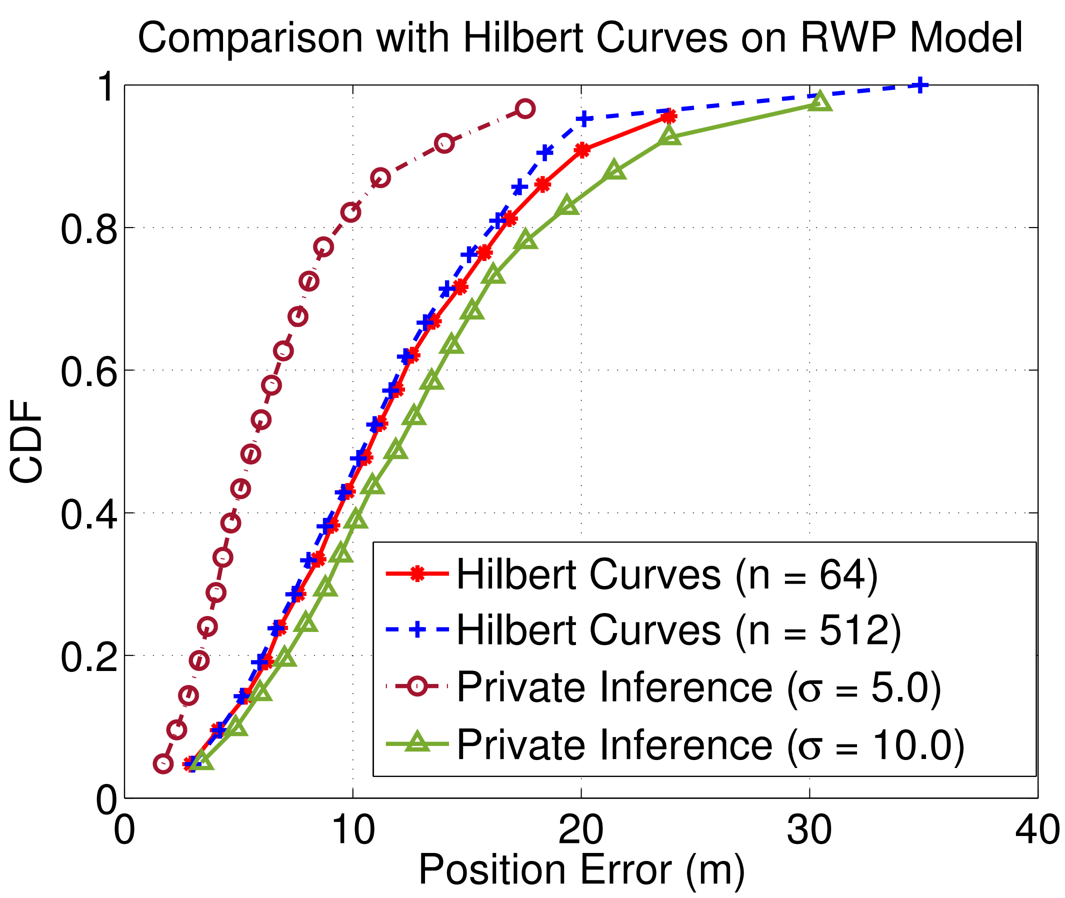
- Simulation results using random way point (RWP) model.

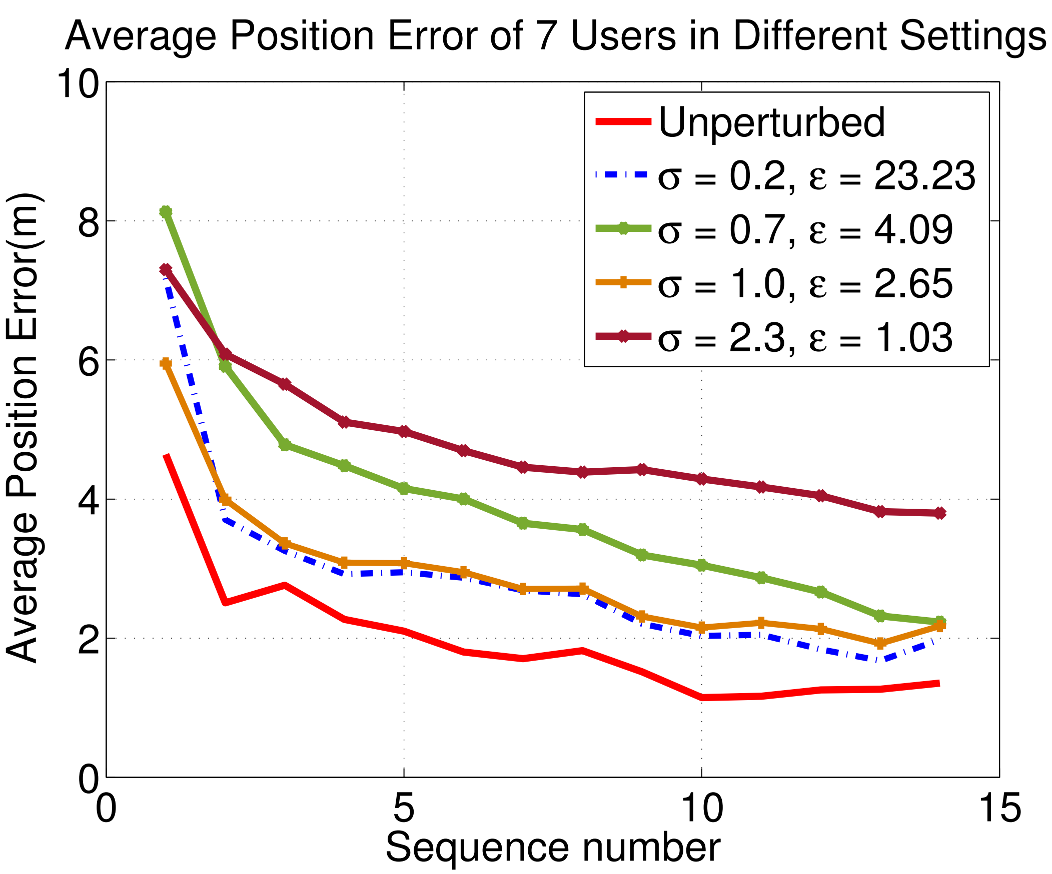




Performance Evaluation

- Comparison experiment and real-world experimental results.





Thank you!