Hiding Stars with Fireworks: Location Privacy through Camouflage

Based on paper written by Joseph T. Meyerowitz and Romit Roy Choudhury

Presentation by Róża Chojnacka

Faculty of Mathematics, Informatics and Mechanics
University of Warsaw

November 2, 2011

Outline

- Location based services
- Existing work and limitations
- CacheCloak
- System evaluation
- Results and analysis
- Distributed CacheCloak
- Conclusion

What is an LBS?

- A Location-Based Service (LBS)
  - an information or entertainment service
  - accessible with mobile devices through the mobile network
  - utilizing the ability to make use of the geographical position of the mobile device

Applications

- Requesting the nearest business or service, such as an ATM or restaurant
- Receiving alerts, such as warning of a traffic jam or receiving a discount coupon
- Geolife : provides a location-based to-do system

LBS

- LBS services rely on an accurate, continuous and real-time stream of location data
- Constant identification and tracking throughout the day
- Users may by hesitant to using LBSs

Privacy protection vs usefulness

- Degraded spatial accuracy
- Increased delay in reporting user's location
- Temporarily preventing the users from reporting locations at all

The user's location data may be less useful after privacy protections have been enabled

Trusted vs untrusted LBS

- Trusted LBS
  - Cannot be used anonymously, must know your identity
    - A banking app might confirm that financial transactions are occurring in a user's hometown
- Untrusted LBS
  - Can reply meaningfully to anonymous or pseudonymous users
    - “Where are the nearest ATMs?”
- CacheCloak can eaither act as a trusted intermediary for the user or a distributed and untrusted intermediary

K-Anonymity

- A user cannot be individually identified from a group of *k* users
- Send a sufficiently large “k-anonymous region” instead of a single GPS coordinate
- Decreases spatial accuracy
- May prevent meaningful use of various LBSs, especially in low density scenarios

CliqueCloak

- Wait until at least *k different queries have been sent from a particular region*

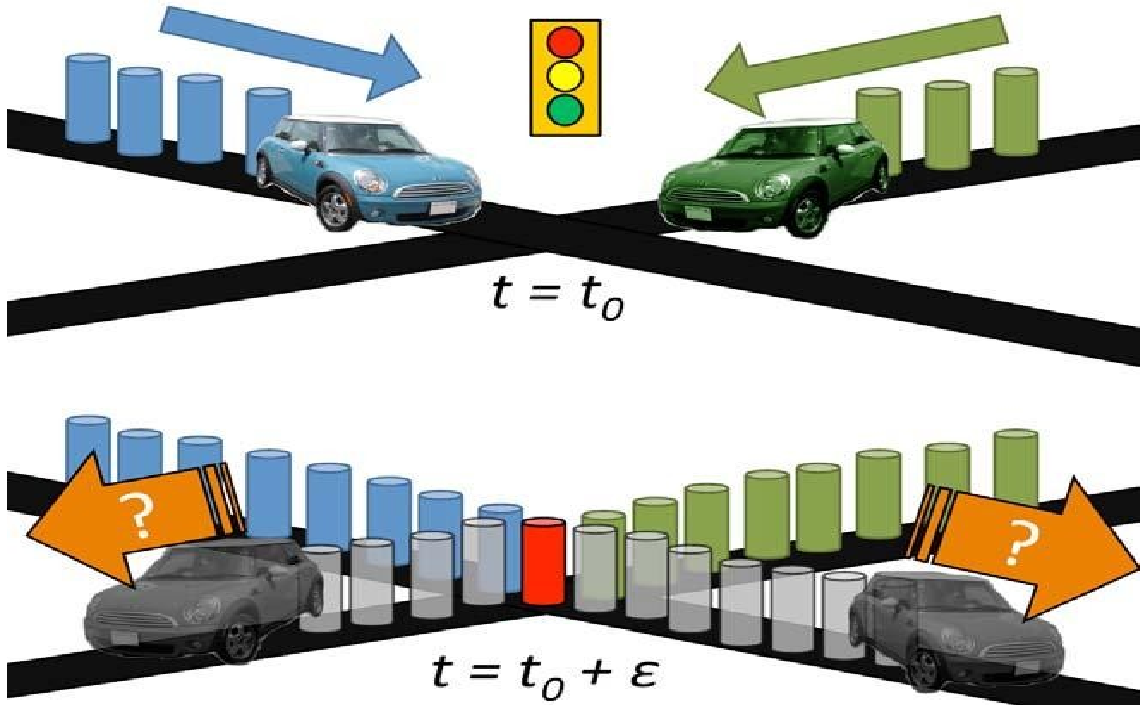
  This allows the k-anonymous area to be smaller in space but expands its size in time

- *Real-time operation suffers*

Pseudonyms

- Each new location is sent to the LBS with a new pseudonym
- Frequent updating may expose a pattern of closely spaced queries
- Very effective when requests are infrequent

Pseudonyms with “Mix Zones”



- A mix zone exists whenever two users occupy the same place at the same time e.g. when two users approach an intersection
- The attacker cannot determine whether the users have turned or have continued to go straight

Pseudonyms with “Mix Zones”

- Rarity of space-time intersections, especially in sparse systems
- It is much more common that two users' paths intersect at different times

Path Confusion

- Extends the method of mix zones by resolving the same-place same-time problem
- Incorporate a delay in the anonymization
  - t_0 - the first user passes an intersection
  - t_1 - the second user passes an intersection
  - $t_0 < t_1 < t_0 + t_{delay}$

Path Confusion

- Path Confusion creates a similar problem as *CliqueCloak*
- *Real-time operation is compromised*
- *Path confusion will decide to do not release the users' locations at all if insufficient anonymity has been accumulated after* $t_0 + t_{delay}$

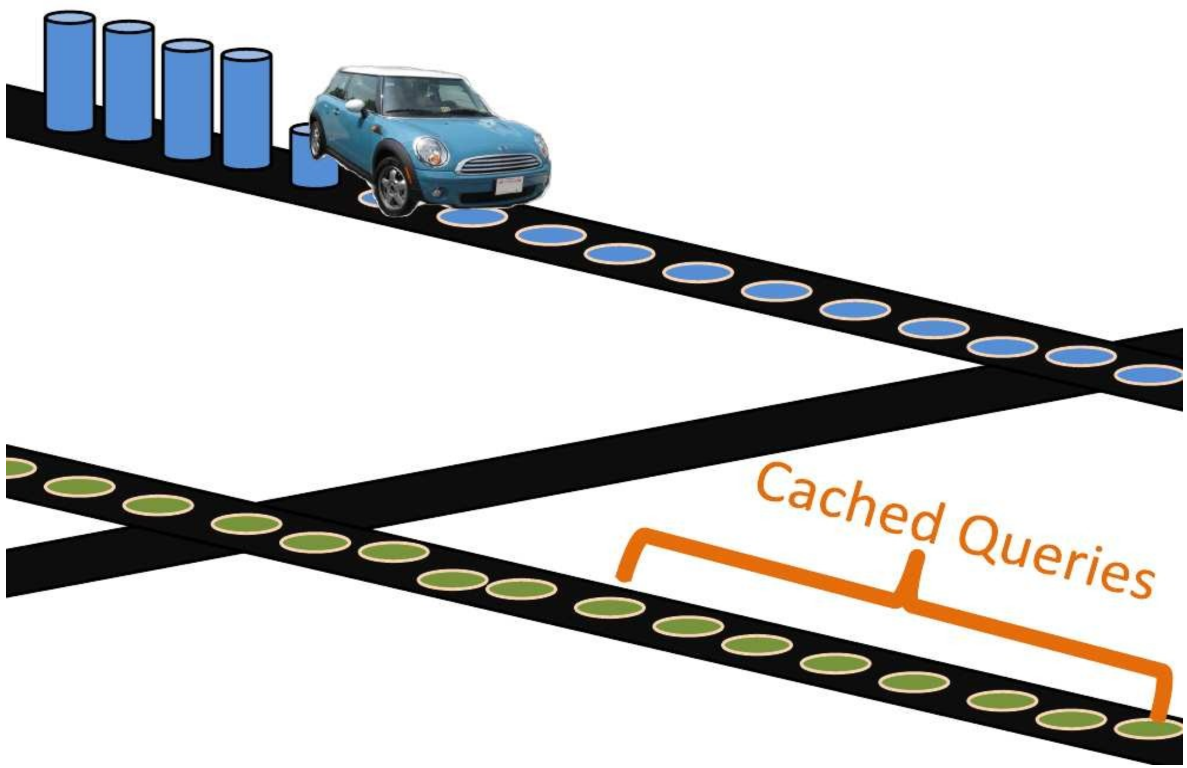
CacheCloak

- A trusted anonymizing server is needed
- On this server we have:
  - A prediction engine
  - Space for caching LBS data
  - Connections to users (wireless) and LBSs (a standard high-capacity wired link to a datacenter)

Predictive privacy

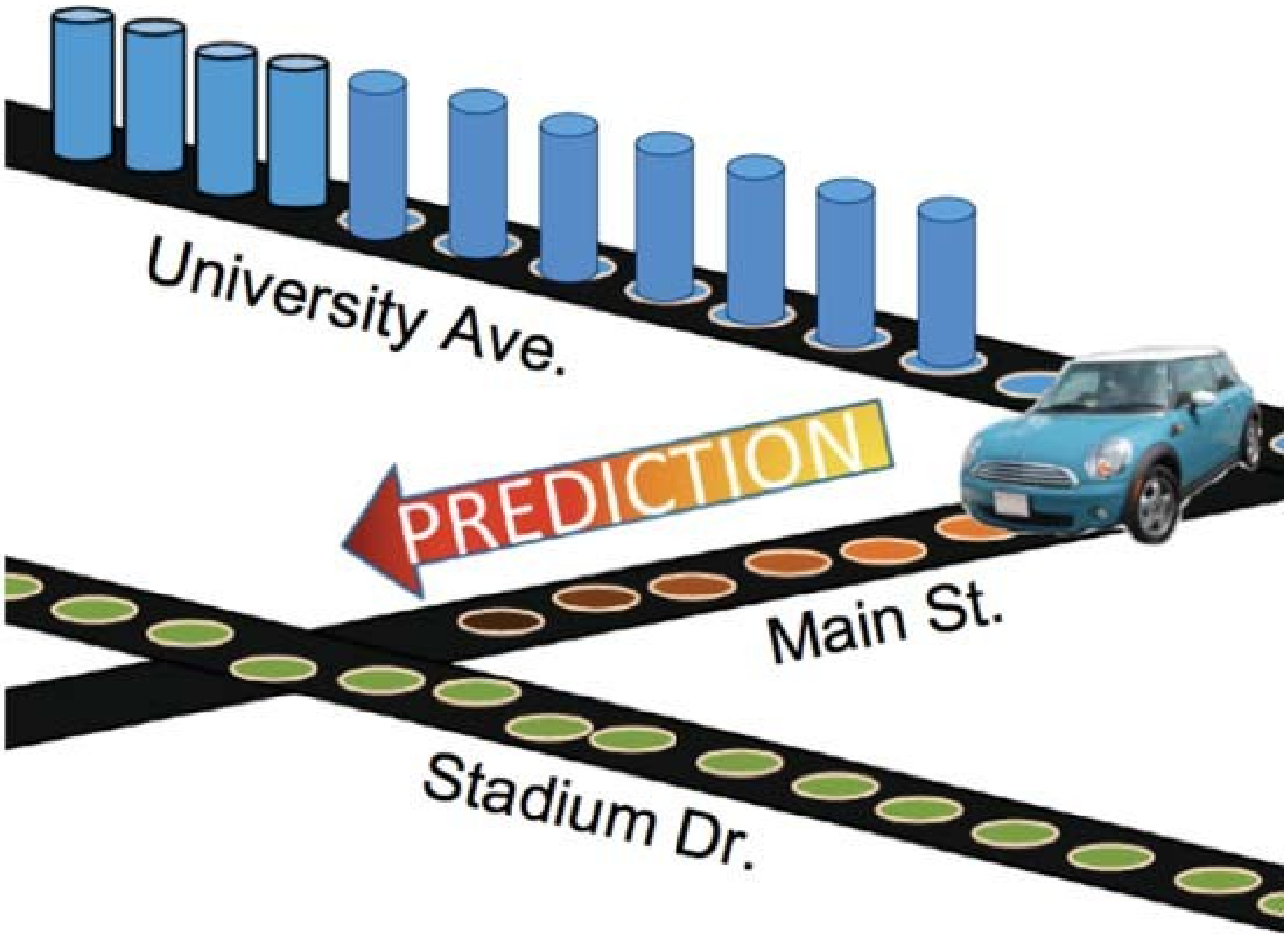
- It is a mobility prediction to do a prospective form of Path Confusion
- Predicted path intersections are indistinguishable to the LBS from a posteriori path intersections
- Keeps the accuracy benefits of Path Confusion but without incurring the delay of Path Confusion

Predictive privacy



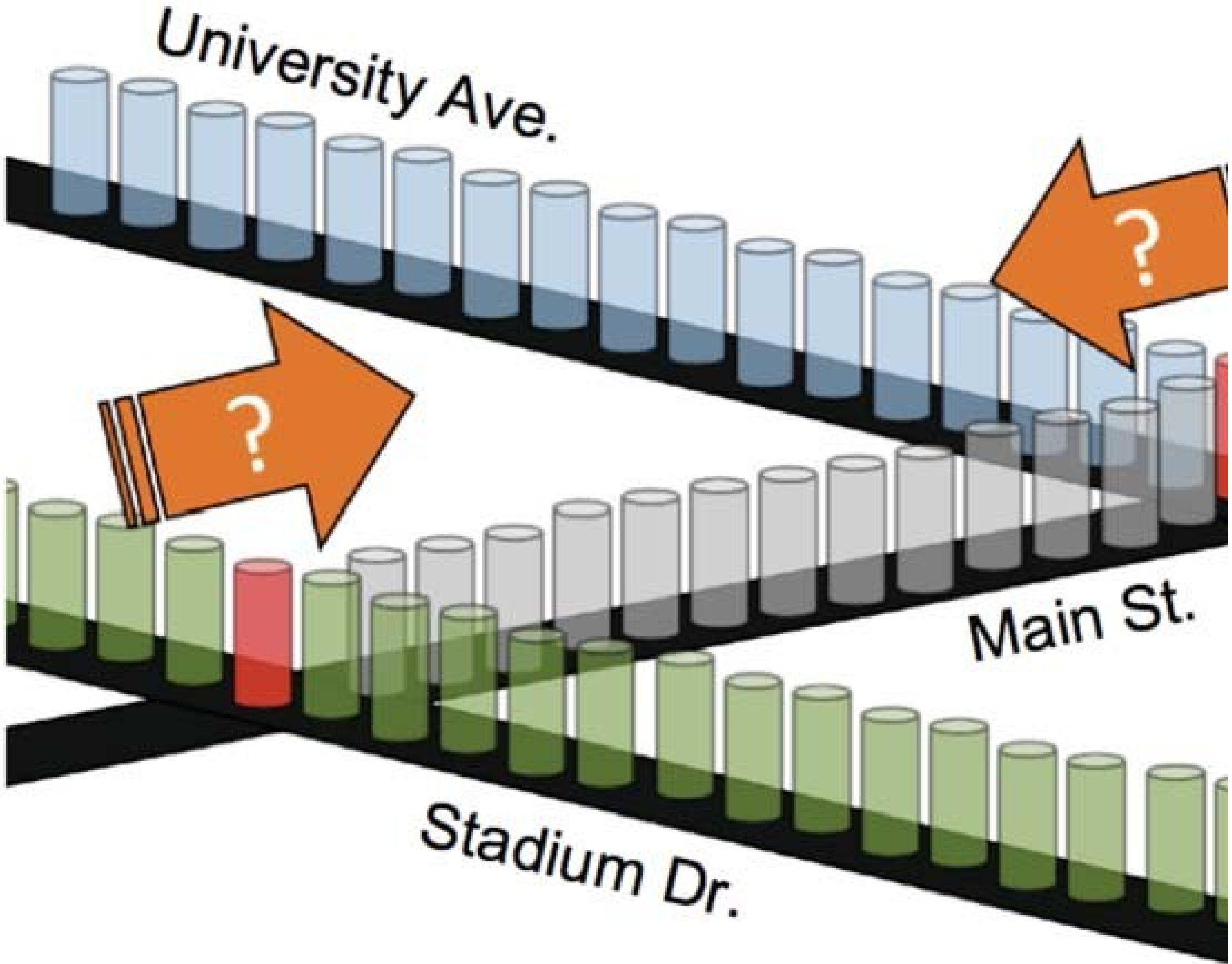
Cache hit

Predictive privacy

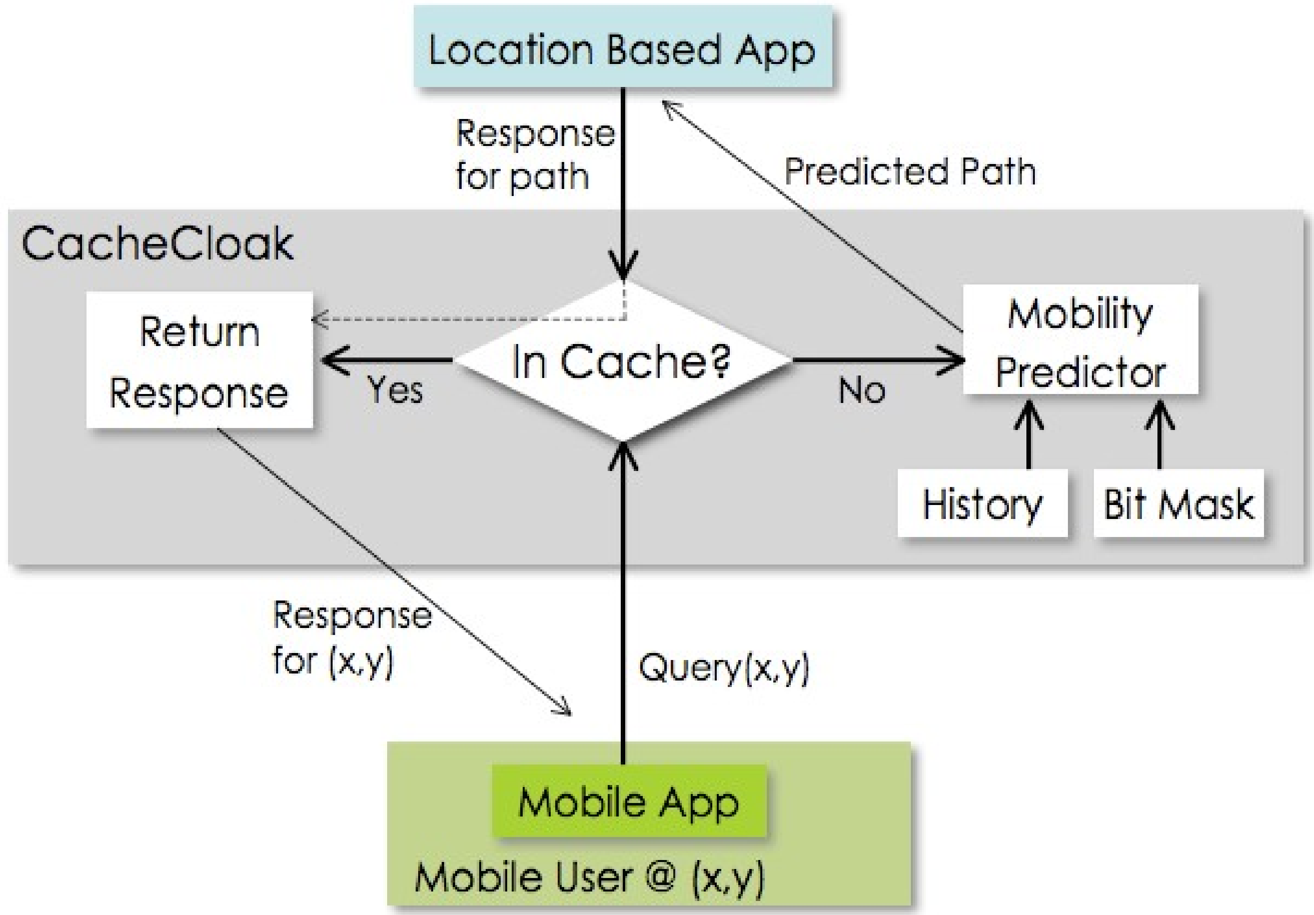


Cache miss

Predictive privacy



CacheCloak

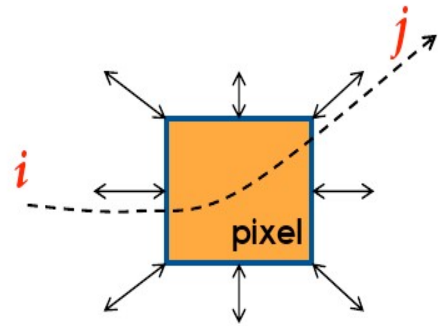

Location Based App
Response
for path
Predicted Path
CacheCloak
Return
Response
Yes
In Cache?
No
Mobility
Predictor
History
Bit Mask
Response
for (x,y)
Query(x,y)
Mobile App
Mobile User @ (x,y)

Prediction engine

- The area is pixellated into a regular grid of squares 10m x 10m
- Each “pixel” is assigned an 8 x 8 historical counter matrix C
- c_{ij} - the number of times a user has entered from neighboring pixel i and exited toward neighboring pixel j
- This data has been previously accumulated from a historical database of vehicular traces from multiple users

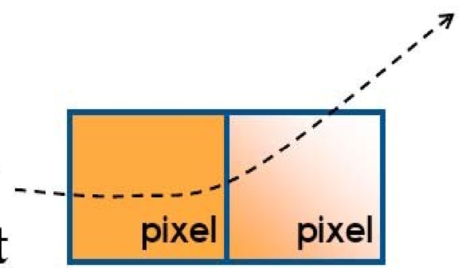
Prediction engine

Pixellate →



$$\begin{bmatrix} 0 & 0 & 0 & 0 & 34 & 57 & 0 & 0 \\ 0 & 0 & 0 & 0 & 34 & 62 & 0 & 0 \\ 0 & 0 & 0 & 0 & 283 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 7 & 31 & 316 & 0 & 0 & 0 & 0 & 0 \\ 98 & 25 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Make Prediction → **from Count Matrix**



Iterated Markov model

- $P(i|j)=\frac{c_{ij}}{\sum_i c_{ij}}$ - probability that a user will exit side j given an entry from side i
- $P(j)=\frac{\sum_j c_{ij}}{\sum_i \sum_j c_{ij}}$ - probability that a user will exit side j without any knowledge of the entering side
- Select most likely pixel max (P(j|i) for j = 1...8)
- Continue until the predicted path intersects with another previously predicted path
- Extrapolate backwards as well
- Send unordered sequence of predicted GPS coordinated to the LBS

CacheCloak

- Predictions are stored in the CacheCloak server
- Mispredicted segments of the user's path and stale data are not transmitted to the user
- Requests between the CacheCloak server and LBS are on a low-cost wired network
- Prevents absurd predictions such as passing through impassible structures or going the wrong way on one-way streets

Simulation

- Software coded in C on a Unix system
- A map of a 6km by 6km region of Durham County, NC (campus, residential areas, road networks)
- Virtual drivers obeyed traffic laws, accelerated according to physical laws and Census-defined speed limits
- The users' locations were written to the filesystem sequentially
- Trace files loaded into CacheCloak chronologically (simulation of a real-time stream of location updates from users)

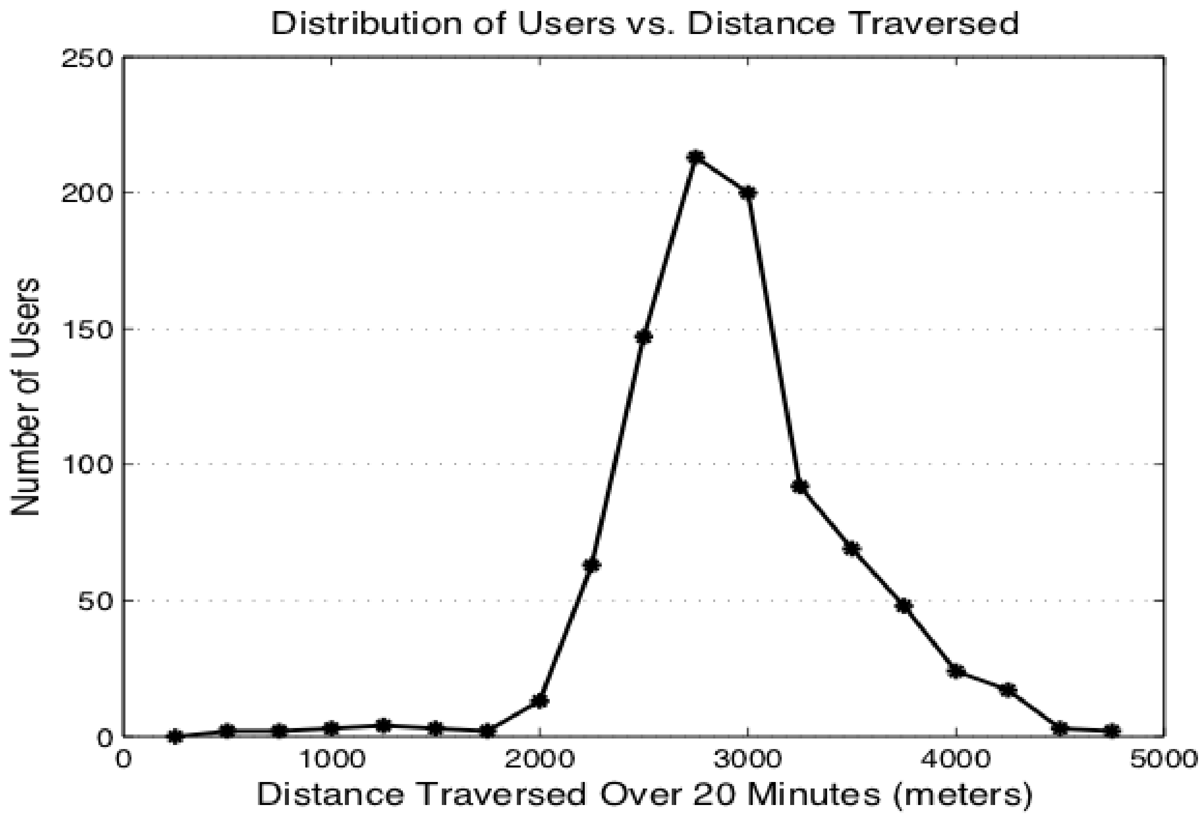
Attacker model

- An “identifying location” is a place where revealing the user's current location identifies a user
- Prevent an attacker from following a user any significant distance away from “identifying locations”

Privacy metrics

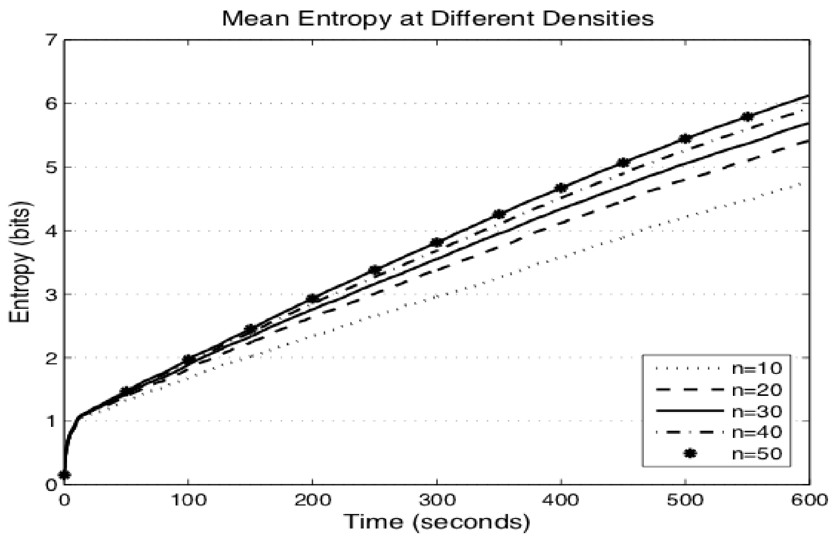
- Location entropy – a quantitative measure of privacy based on the attacker's ability or inability to track the user over time
- It gives a precise quantitative measure of the attacker's uncertainty
- $S=-\sum_{i} p_i(x,y)\log_2(p_i(x,y))$
- S – number of bits (location entropy)
- 2^S equally likely locations will result in S bits of entropy; the inverse does not strictly hold

Results and analysis

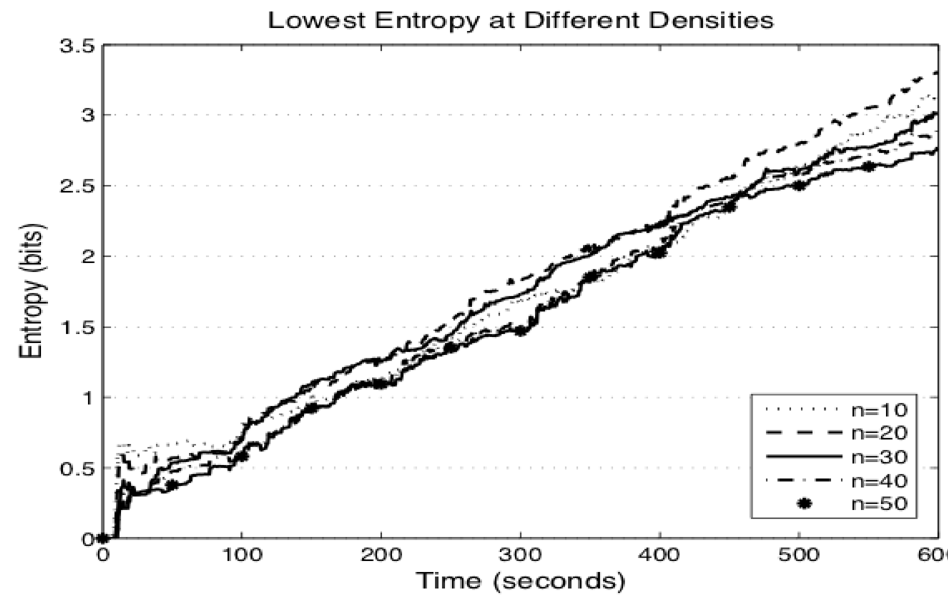


6: Distribution of users Vs. distance traversed in 20 minutes. Most users traversed around 1 to 2 miles.

Results and analysis

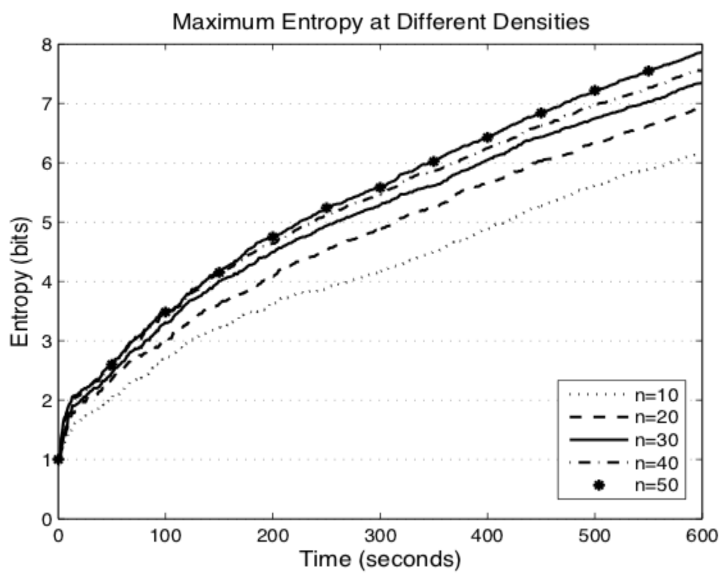


7: Mean entropy over time for different user densities. Even with 10 to 50 users, the achieved entropy is around 5 bits in 10 minutes (600s). In reality, the number of users are likely to be far greater, resulting in greater entropy.

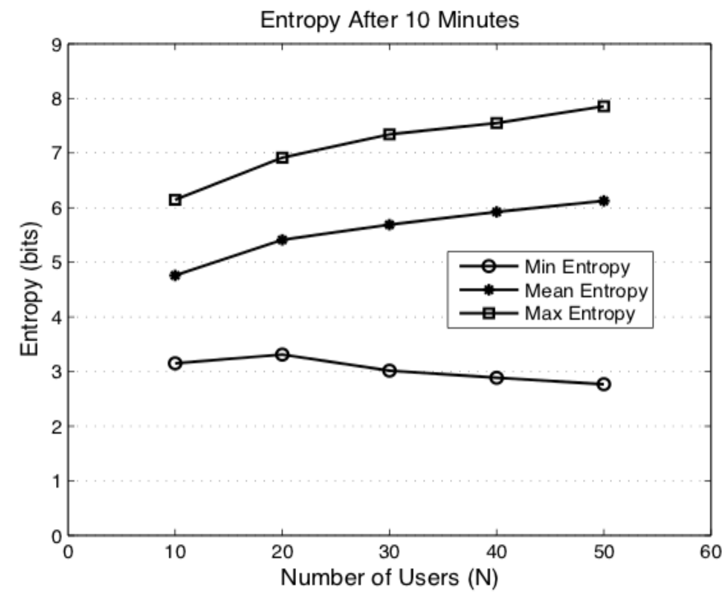


8: Worst-case entropy over time for different user densities. Around 2.7 bits of entropy achieved even with 10 users.

Results and analysis

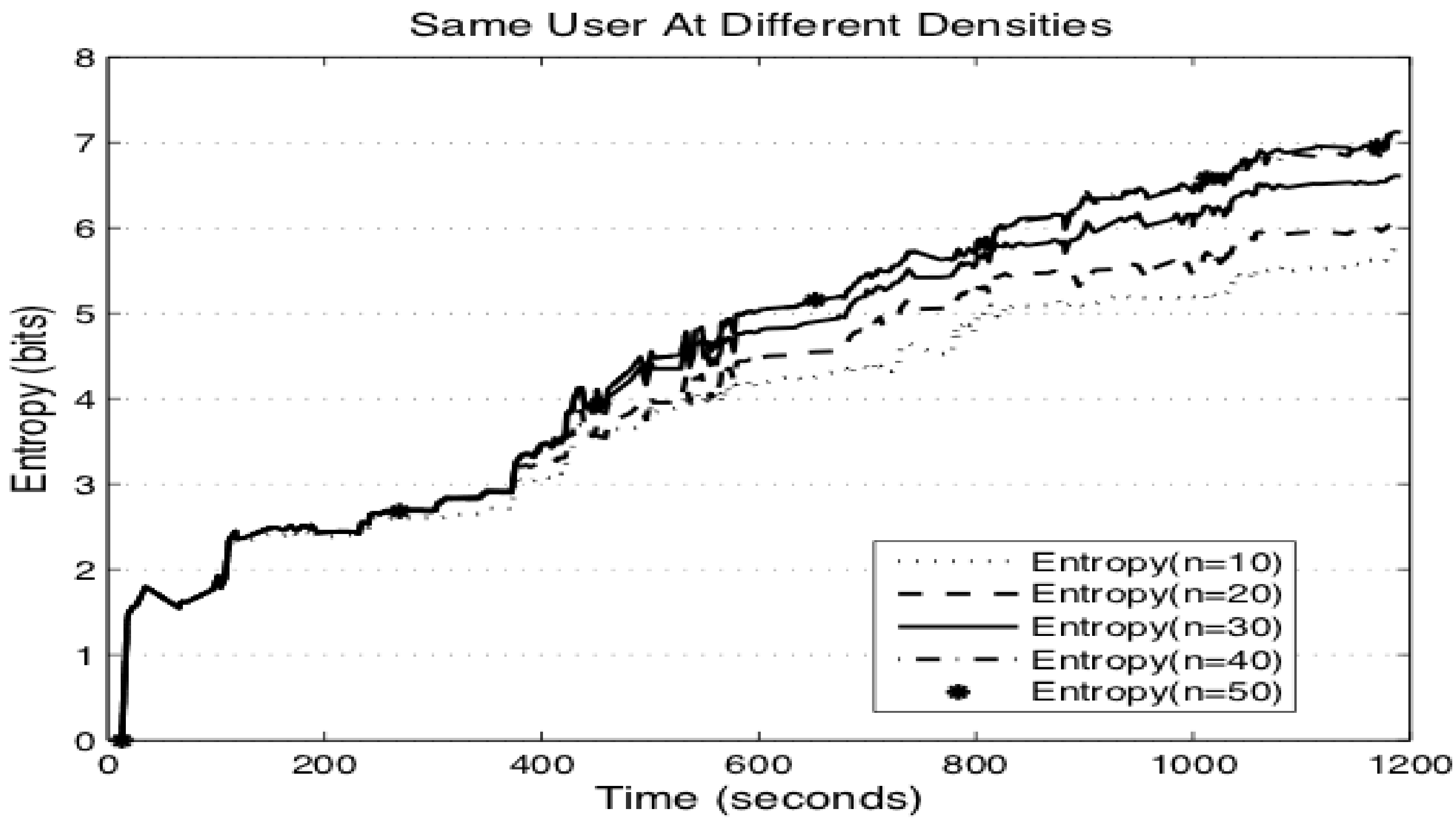


9: Best-case entropy over time for different user densities.



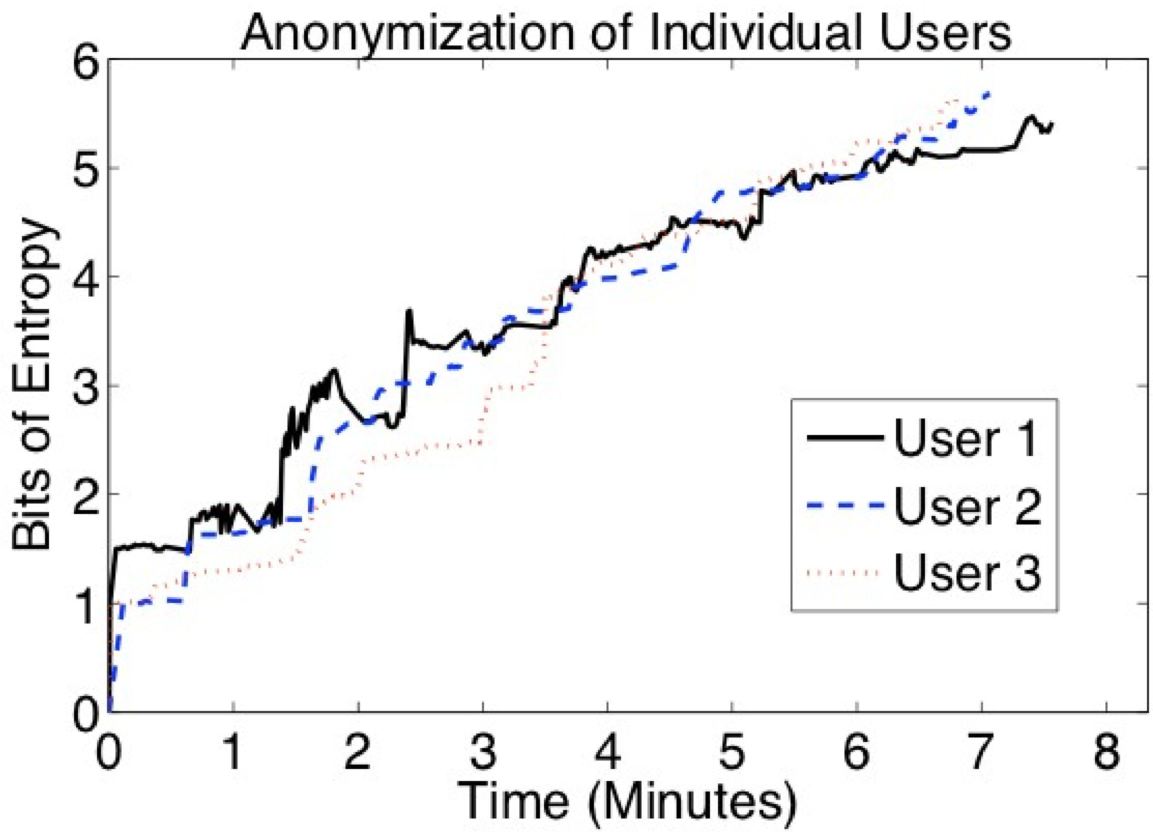
10: Entropy after 10 minutes of attempted tracking for different user densities.

Results and analysis



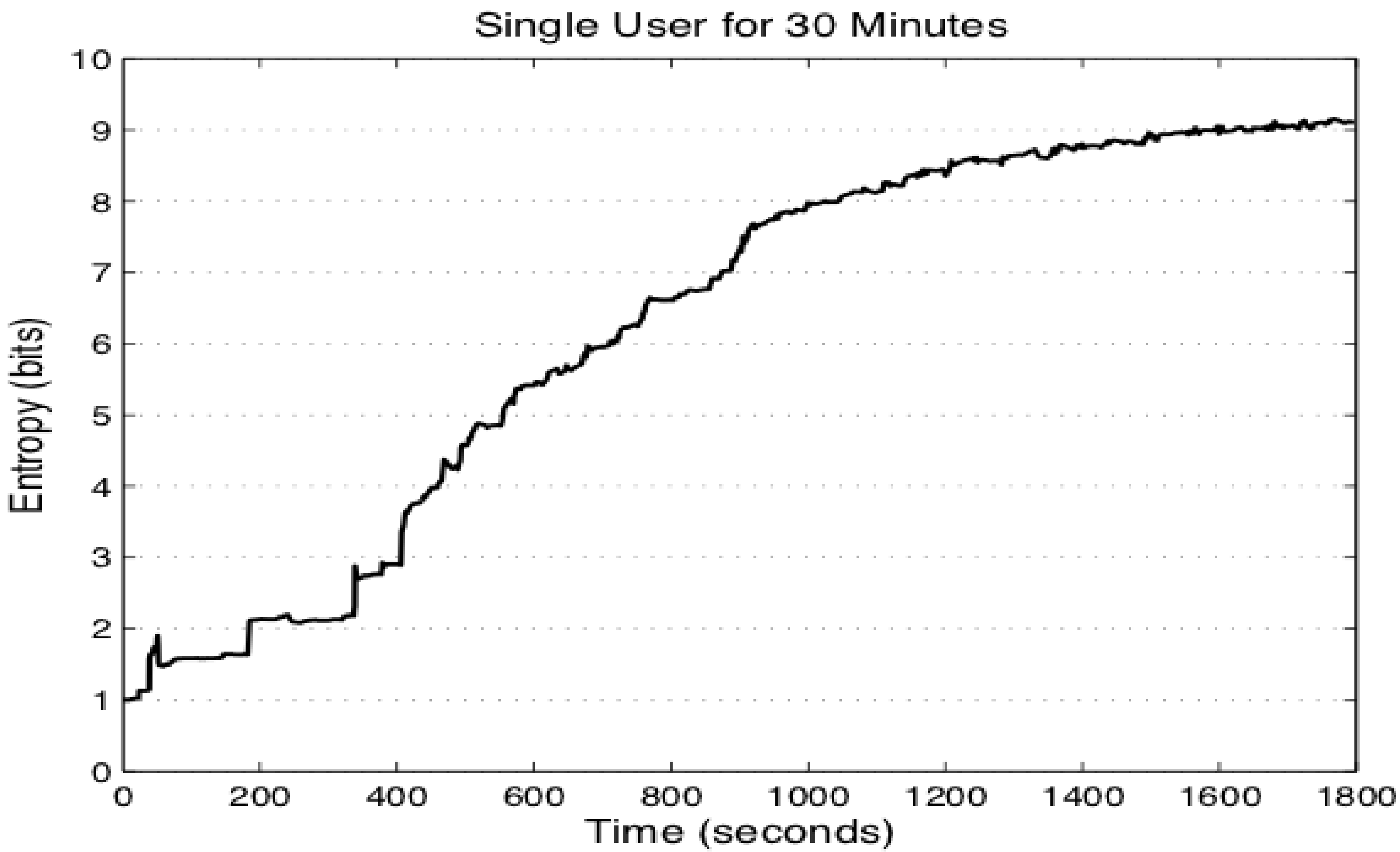
11: One arbitrary user's entropy over time in different density conditions

Results and analysis



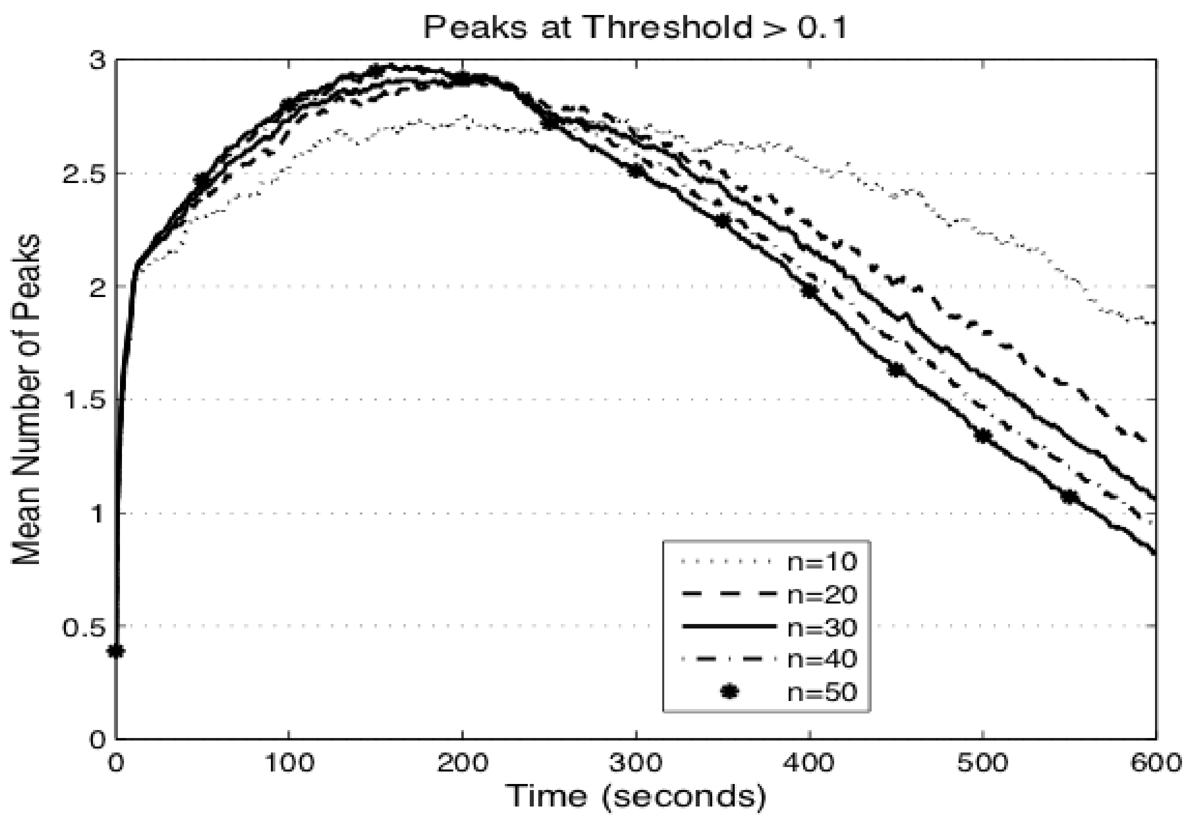
12: Three arbitrary users' entropies over time $(n = 50)$

Results and analysis



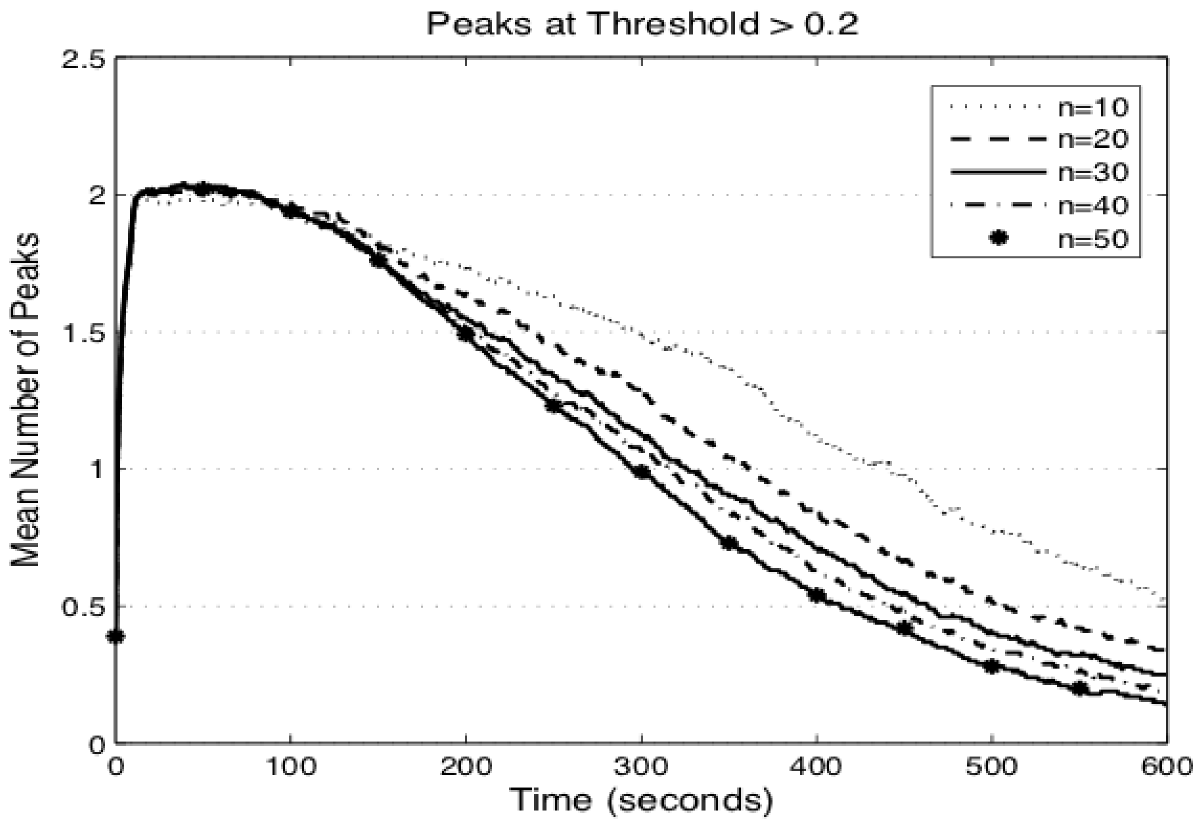
13: The time evolution of a random user's entropy over 30 minutes.

Results and analysis



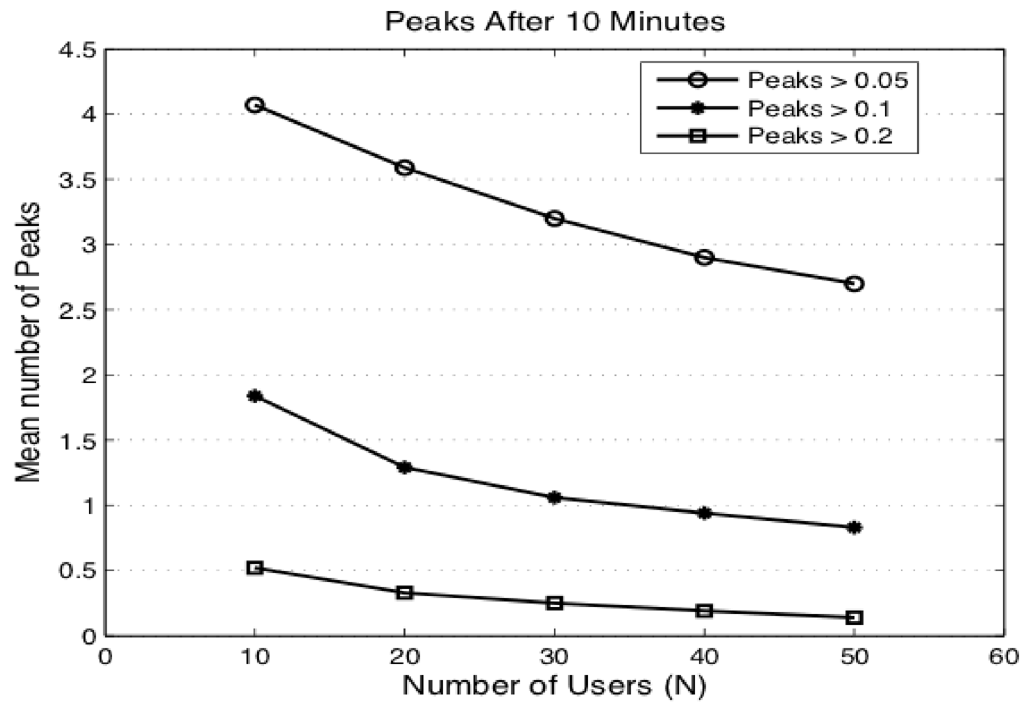
15: Average number of locations a user might be according to a 0.1 threshold.

Results and analysis



16: Average number of locations a user might be according to a 0.2 threshold.

Results and analysis

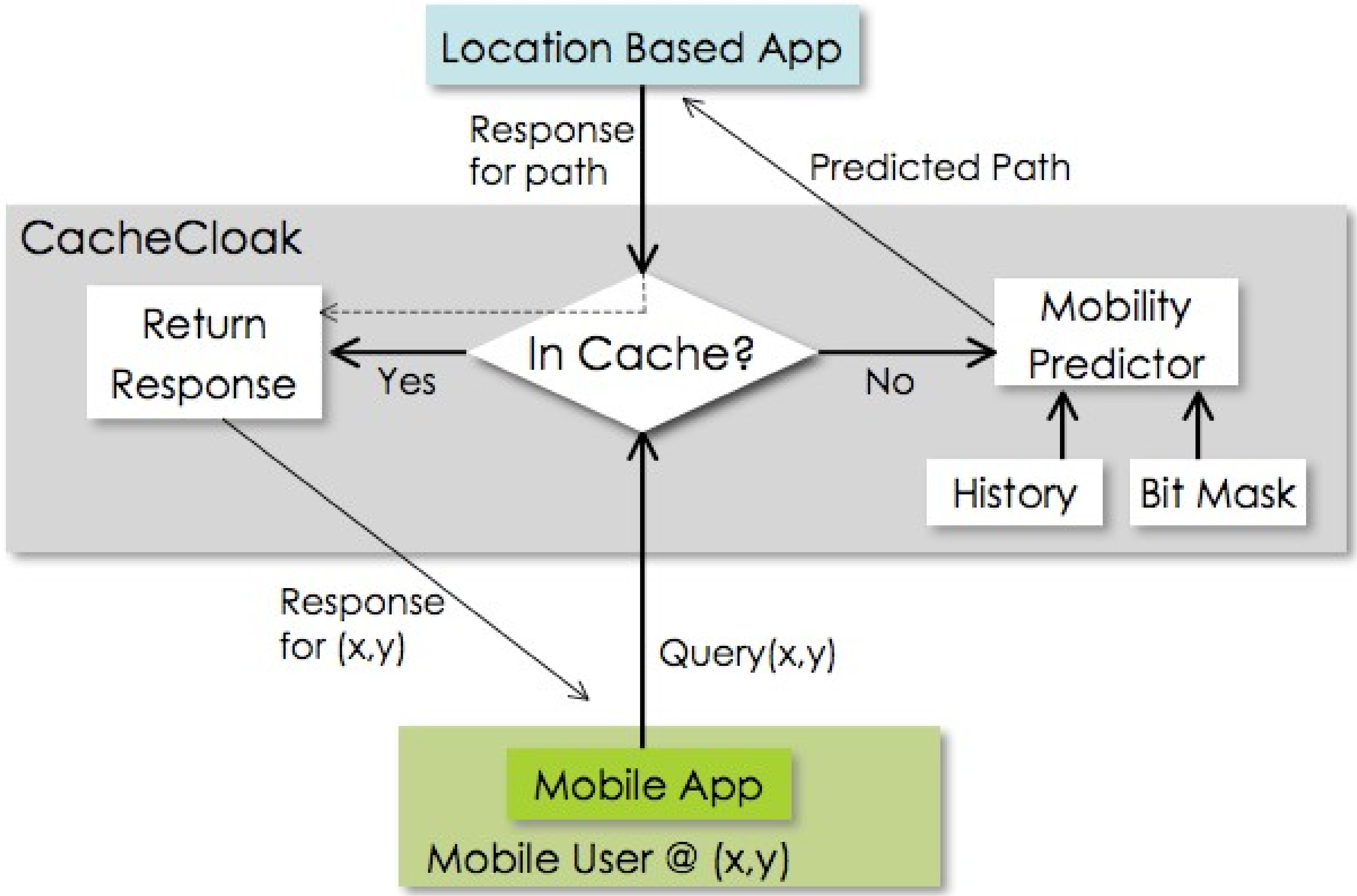


17: Variation of number of peaks left after 10 minutes at different densities and thresholds. The number of peaks for a given threshold decreases with increasing users, showing that more users offer greater opportunity to hide.

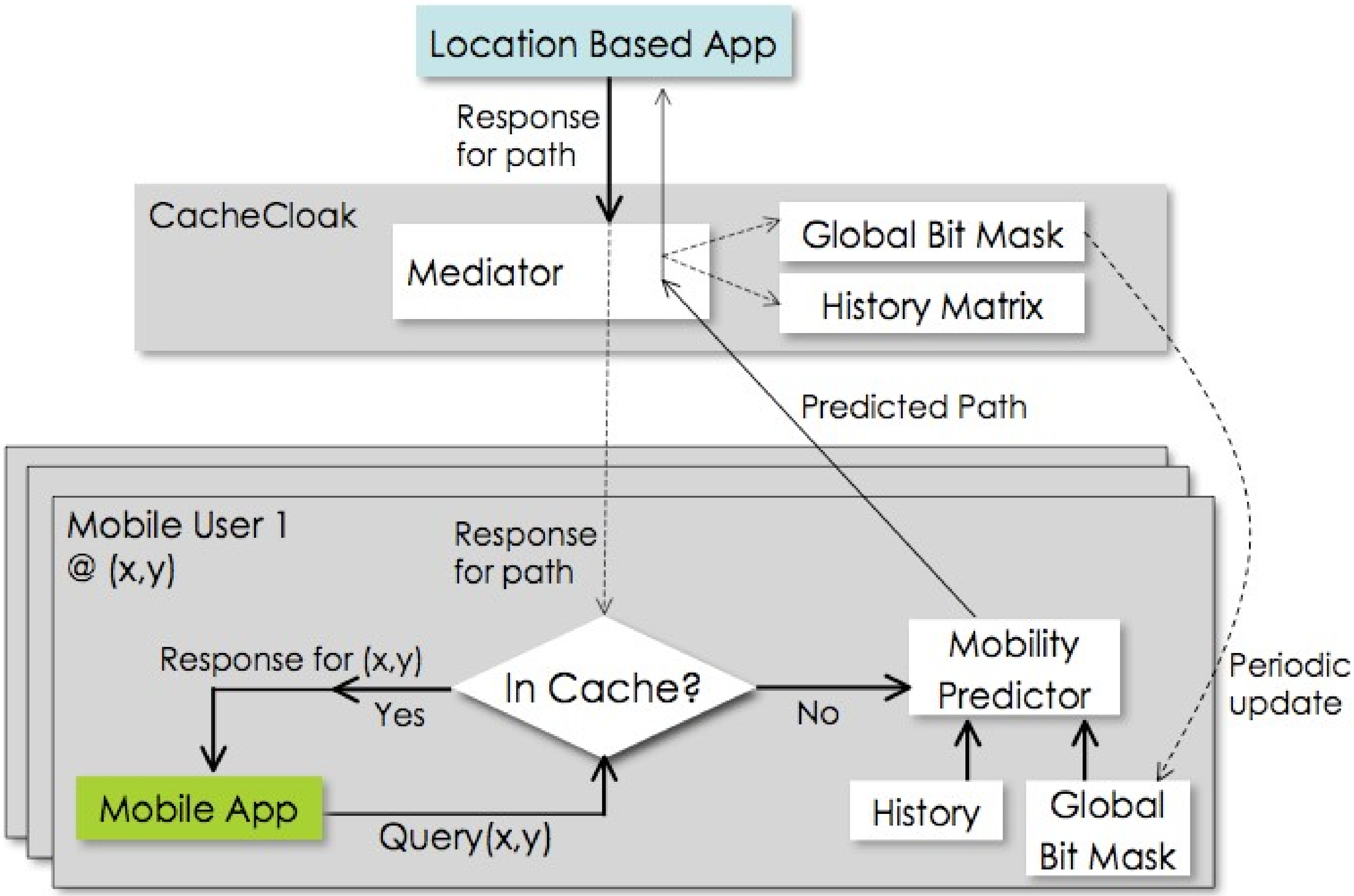
Distributed CacheCloak

- CacheCloak requires the users to trust the server
- What if the users do not wish to trust CacheCloak?
- The need to rearrange the structure of the previous system

Centralised CacheCloak (reminder)



Distributed CacheCloak



Distributed CacheCloak

- The CacheCloak server is only necessary to maintain the global bit-mask from all users in the system
- The user never reveals to CacheCloak nor the LBS its actual location

Distributed CacheCloak drawbacks

- The historical prediction matrix needs to be obtained from the server which creates bandwidth overhead
- But we con compress this data
- Users receive the same quality of service in the distributed form but their mobile devices must perform more computation

Pedestrian users

- So far only vehicular movements were taken
  - Realistic vehicular movements can be simulated easily in very large numbers
- Pedestrians follow paths just between a source and a destination just as vehicles do
- More diffucult to get enough historical mobility data to bootstrap the prediction system
  - Obtain walking directions from realistic source-destination pairs on Google Maps

Bootstrapping CacheCloak

- A new LBS starts with zero users
- If privacy cannot be provided to the first new users, it may be difficult to gain a critical mass of users for the system
- CacheCloak works well with very sparse populations
- CacheCloak can be used initially with simulation-based historical data

Conclusion

- Existing location privacy methods require a compromise between accuracy real-time operation and continuous operation
- CacheCloak eliminates the need for these compromises
- Mobility predictions are made for each mobile user
- Camouflaging users in a “crowd”
- Centralized and distributed forms of CacheCloak
- Tracebased simulation of CacheCloak with GIS data of a real city with realistic mobility modeling

Conclusion

- An attacker cannot track a user over a significant amount of time
- Can work in in extremely sparse systems where other techniques fail
- The cost of the privacy preservation is purely computational
- No new limitations on the quality of user location data
- This is a new location privacy method that can meet the demands of emerging LBSs

Questions