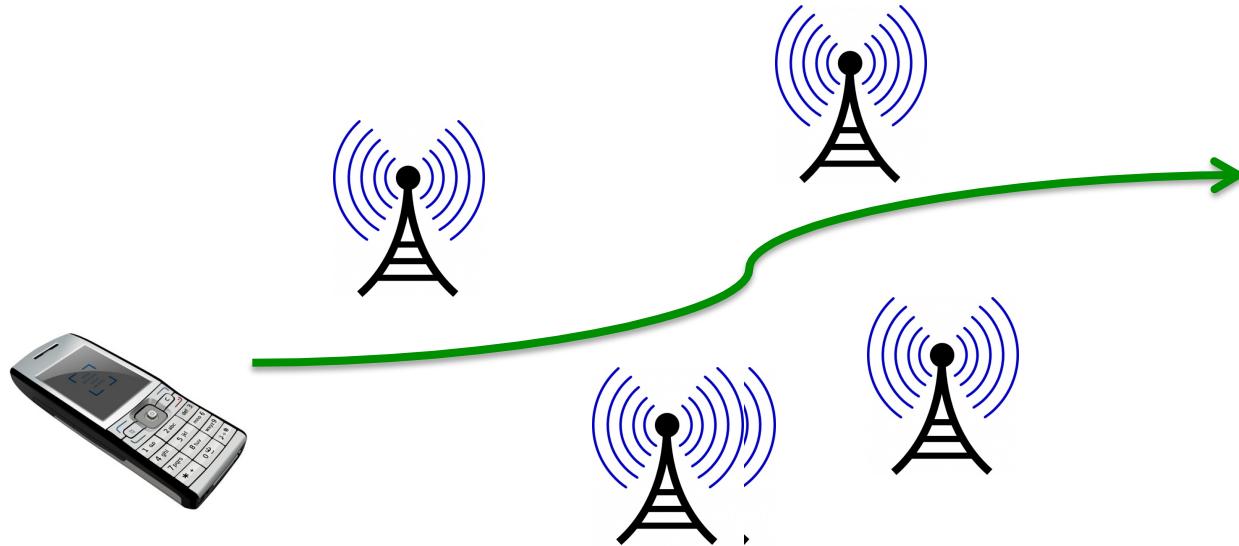


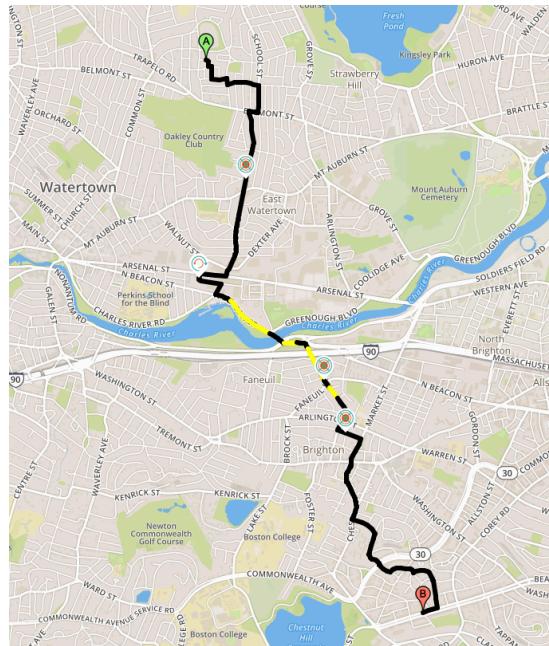
Accurate, Low Energy Trajectory Mapping For Mobile Phones



Arvind Thiagarajan, Lenin Ravindranath,
Hari Balakrishnan, Sam Madden, Lewis Girod
MIT CSAIL

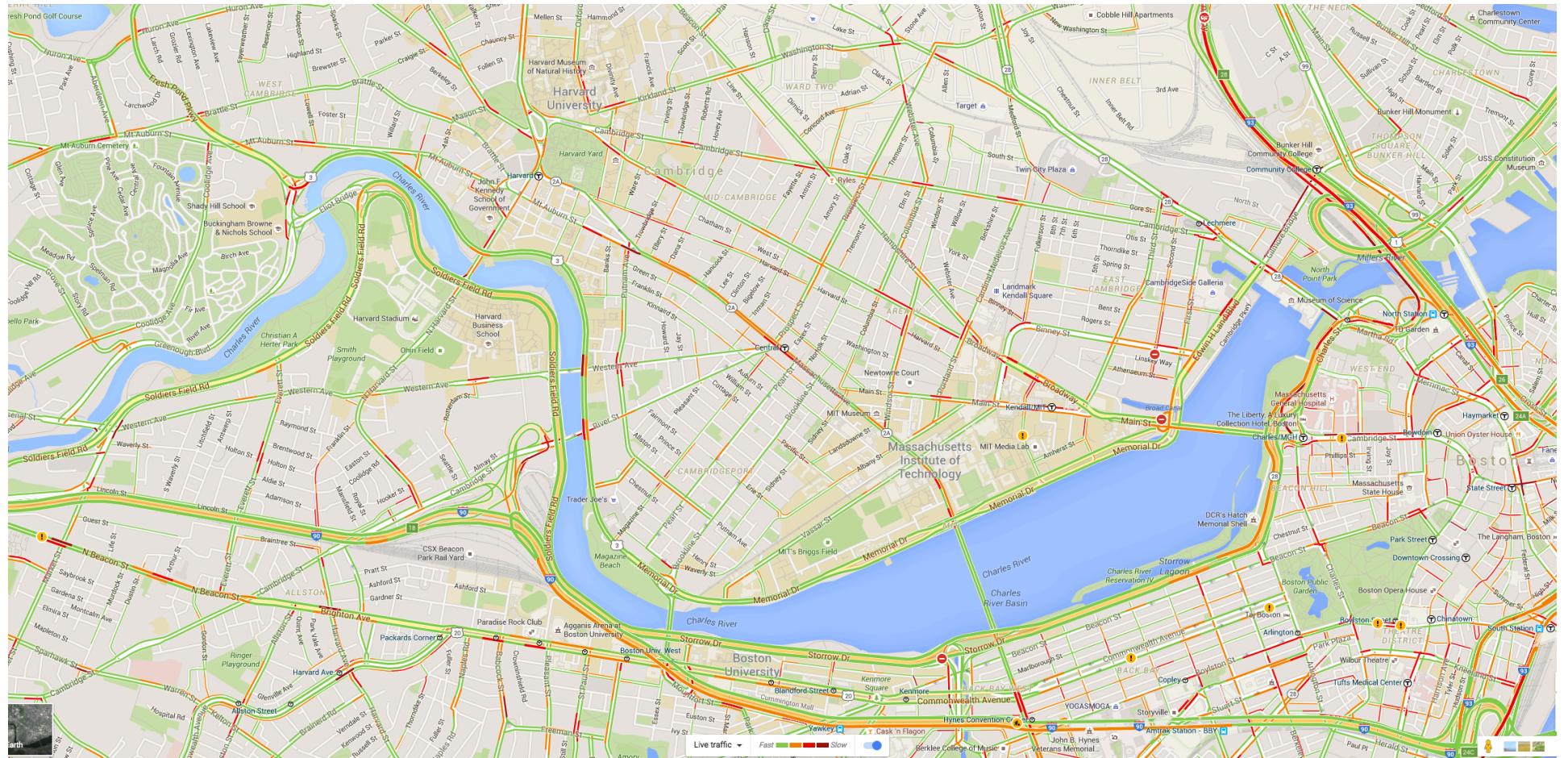
Goal

- Find the *trajectory* i.e. *sequence of locations* visited by a mobile device



- Applications need to find path both *accurately* and *energy-efficiently*

Traffic Estimation



Crowdsourced Data

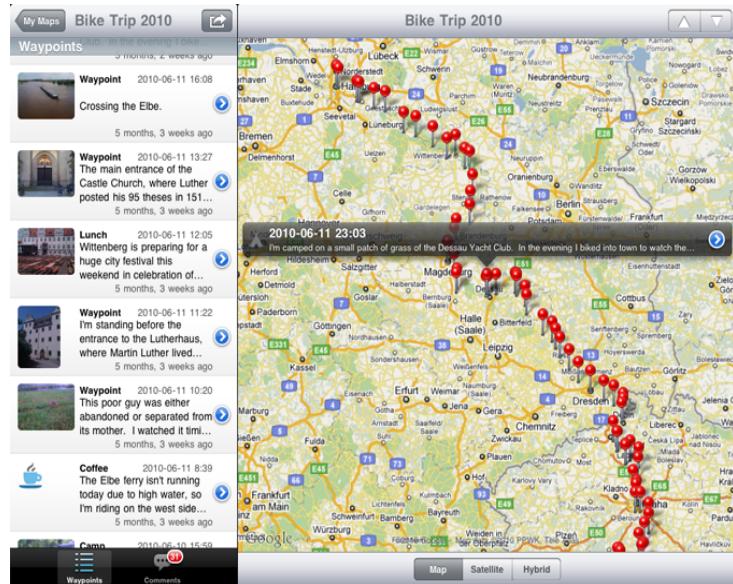


Battery dies in ~6 hours if monitoring with GPS

Bike Routes



<http://www.jonathanokeeffe.com/strava/multi-ride-mapper/>



TrackMyTour iPhone App:

Allows users to keep track of their trips and annotate them

The screenshot shows a running route map on the left and a "My Goals" summary on the right. The route map displays a path through neighborhoods like Olmsted Park, Hyde Square, and Jamaica Plain. The "My Goals" summary includes:

- Current Goals (3)**
 - Run a distance of 26mi (Target: Jul 06, 2013) - Progress: 6 / 26 mi (24%)
 - Lose a total of 10lbs (Target: Jan 01, 2013) - Progress: 0 / 10 lbs (0%)
 - Run a total of 100mi (Target: Jan 01, 2013) - Progress: 74 / 100 mi (74%)
- Past Goals (2)**
 - 75 of 75mi (Achieved: Run / Total Distance by Oct 18, 2012)
 - Lose a total of 5 lbs (Achieved: Target: Sep 1, 2012)

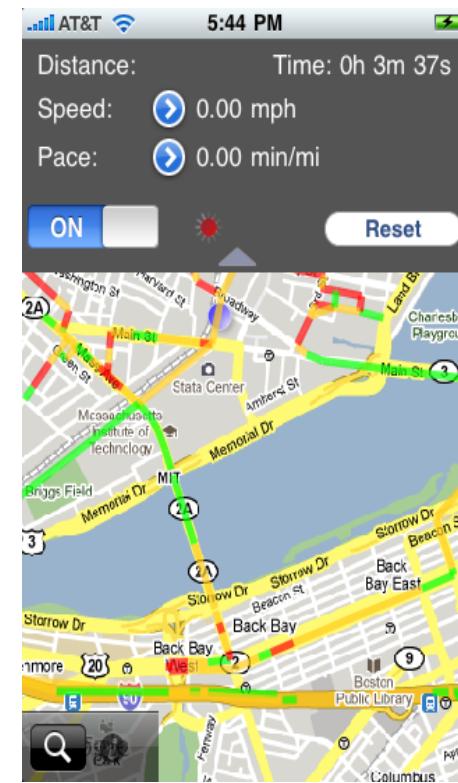
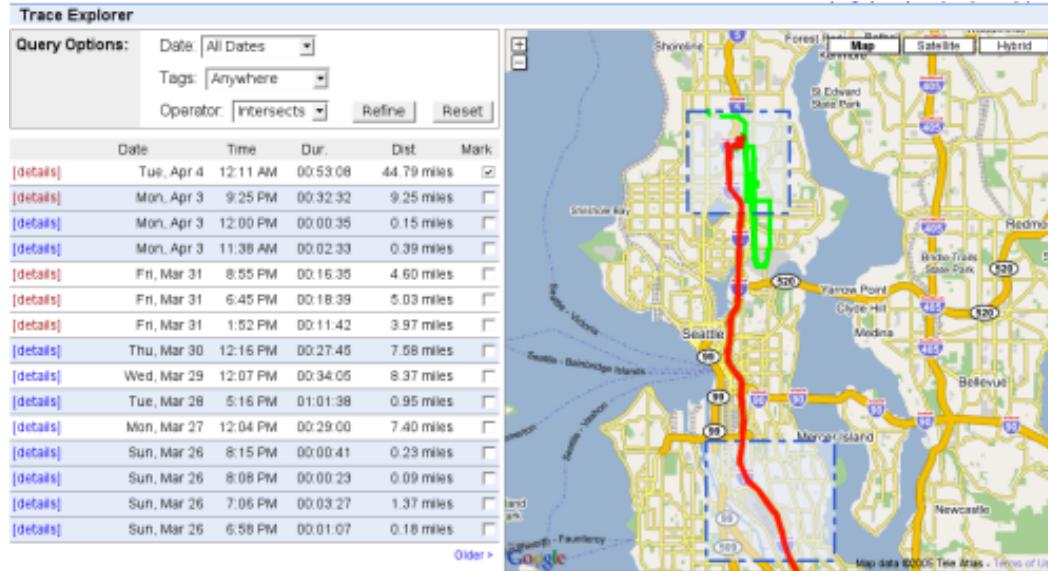
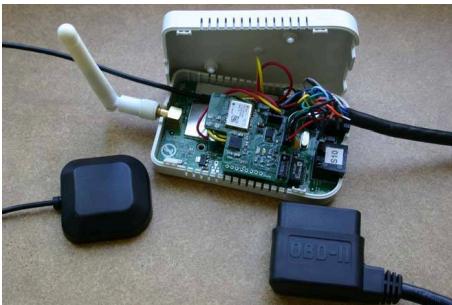
At the bottom, there are "Stop" and "Pause" buttons.

Running

Trash Track



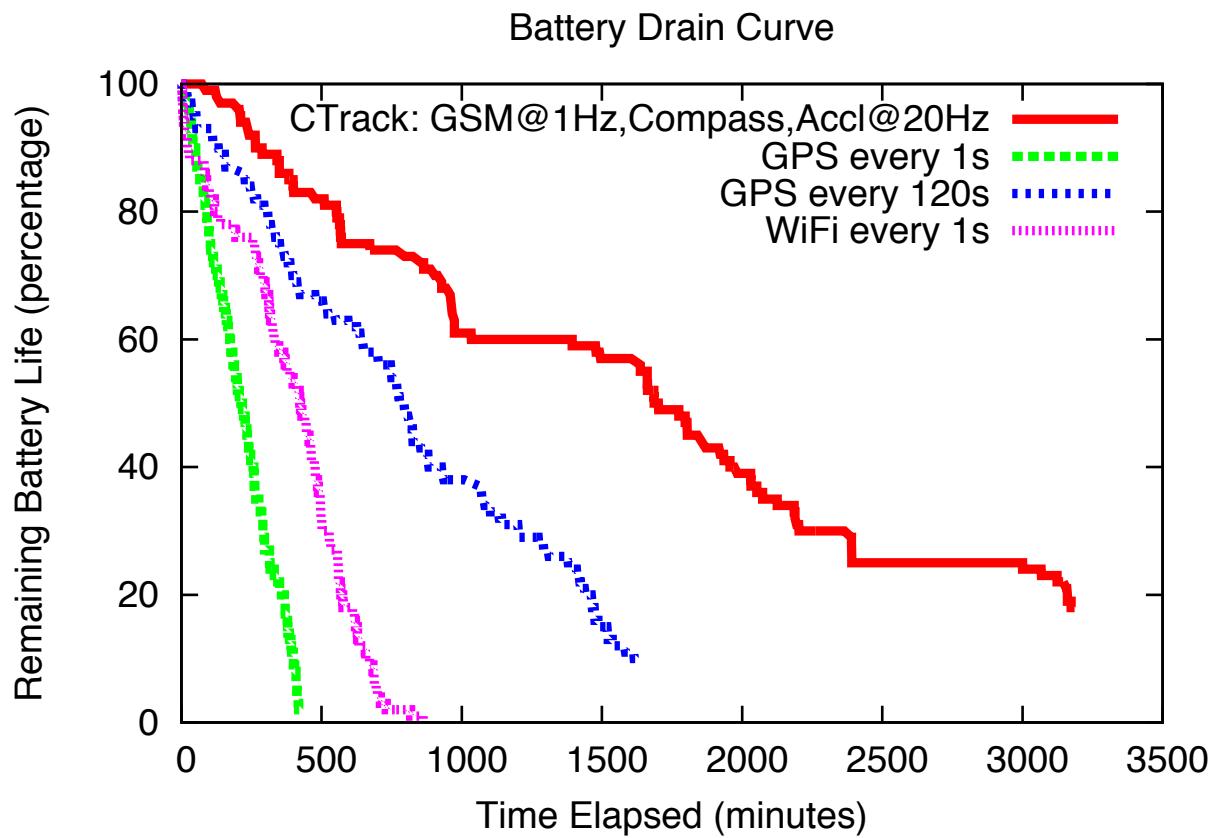
Context: CarTel Project



Crowdsouce tracks to estimate traffic on road segments

Limitation of GPS: Energy

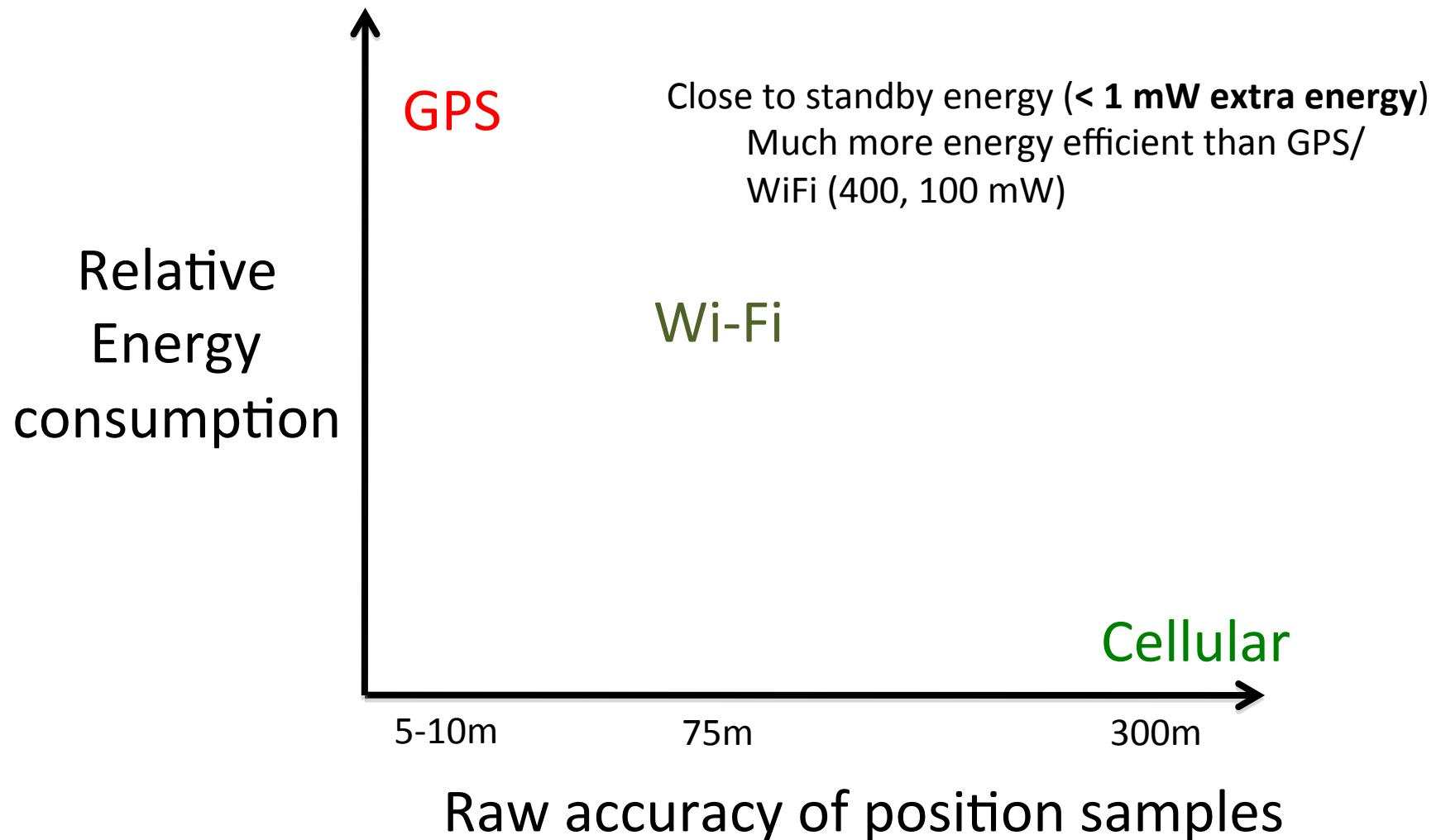
- GPS signals are energy-intensive to acquire & process
- Frequent GPS sampling drains battery fast



6 hrs on iPhone 3GS vs
18 hrs w/o GPS
(dim screen)

Android G1 phone: 6-8 hours with 1 Hz GPS

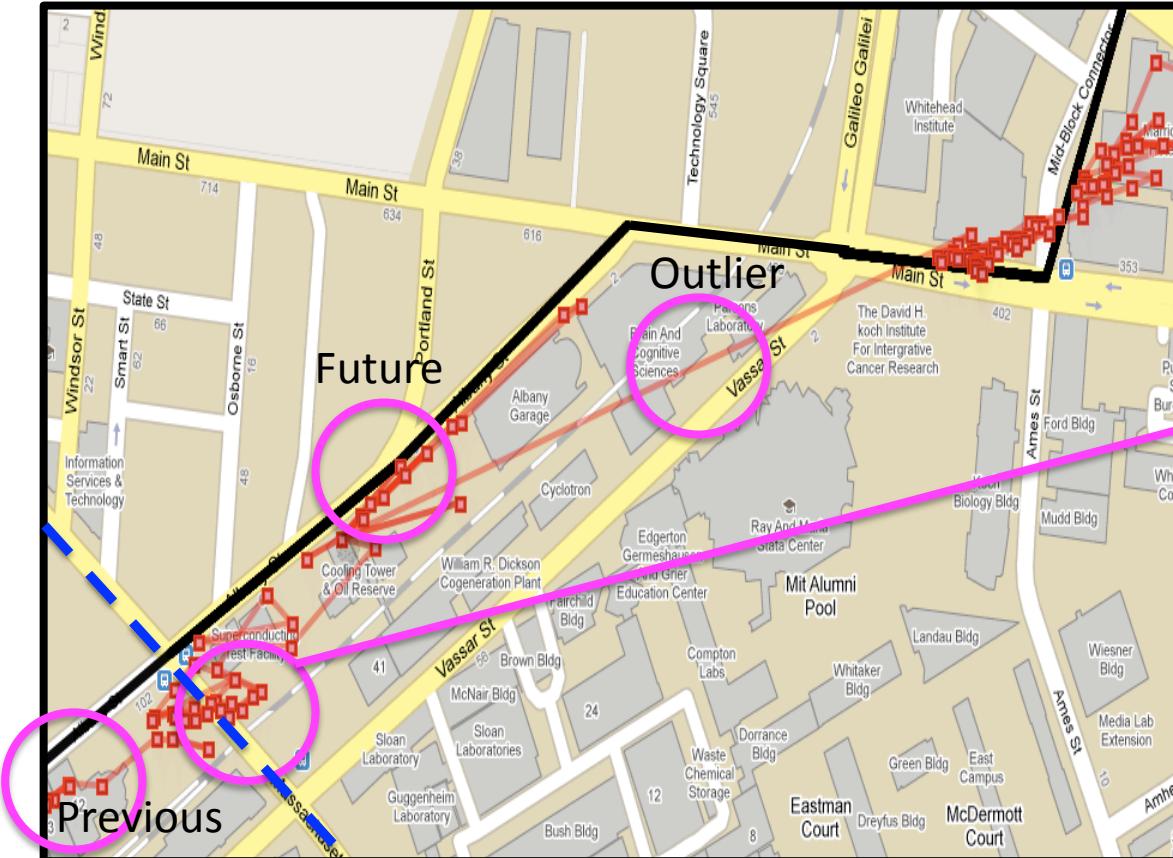
Approach: Use low-power sensors



Outline

- Prior work
 - Intermittent GPS (Microsoft – Krumm)
 - Wi-Fi (VTrack – from same group as CTrack – CarTel project)
- CTrack paper
 - Cellular fingerprints
 - Better energy
 - Accuracy?

Noisy Data (Wi-Fi example)

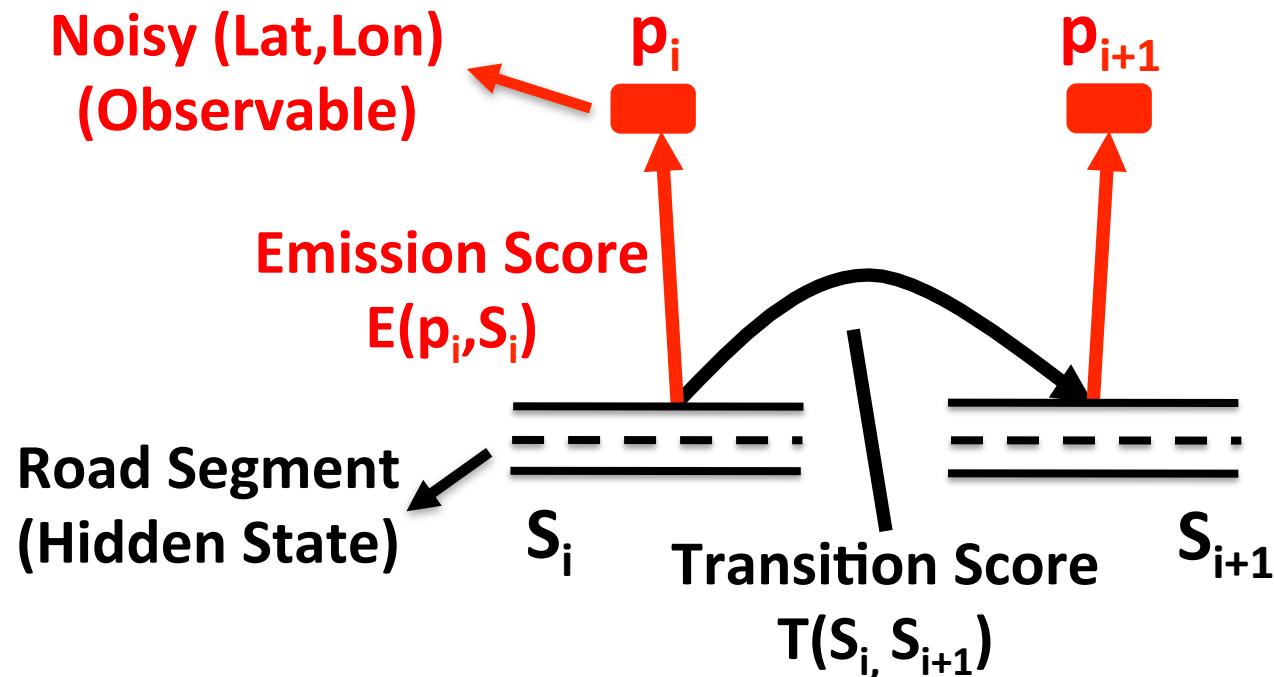


The closest road to
a position
sample is **not**
where it originally
came from

- Exploit both previous and future location information
- Don't overly weight *any one* location sample
- Find a *continuous (unbroken) sequence* of roads

Solution: Hidden Markov Model

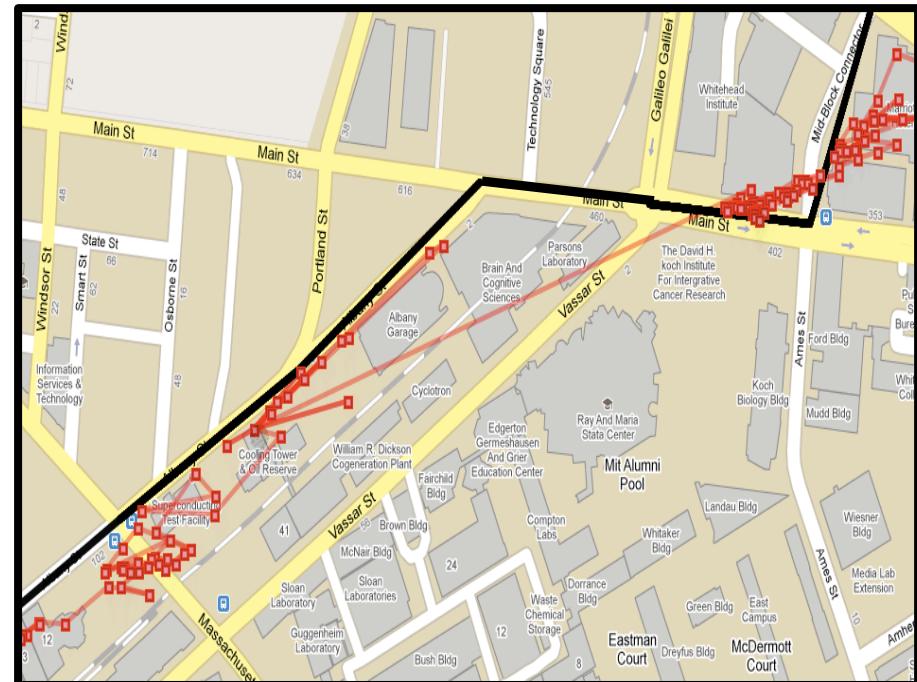
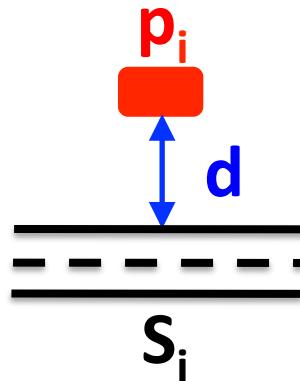
- Maps noisy observations (coordinates) to hidden states (underlying road segments)



Dynamic Program Finds Best State Sequence (cf. Viterbi)
“Best” => Max Product of Emission and Transition Scores

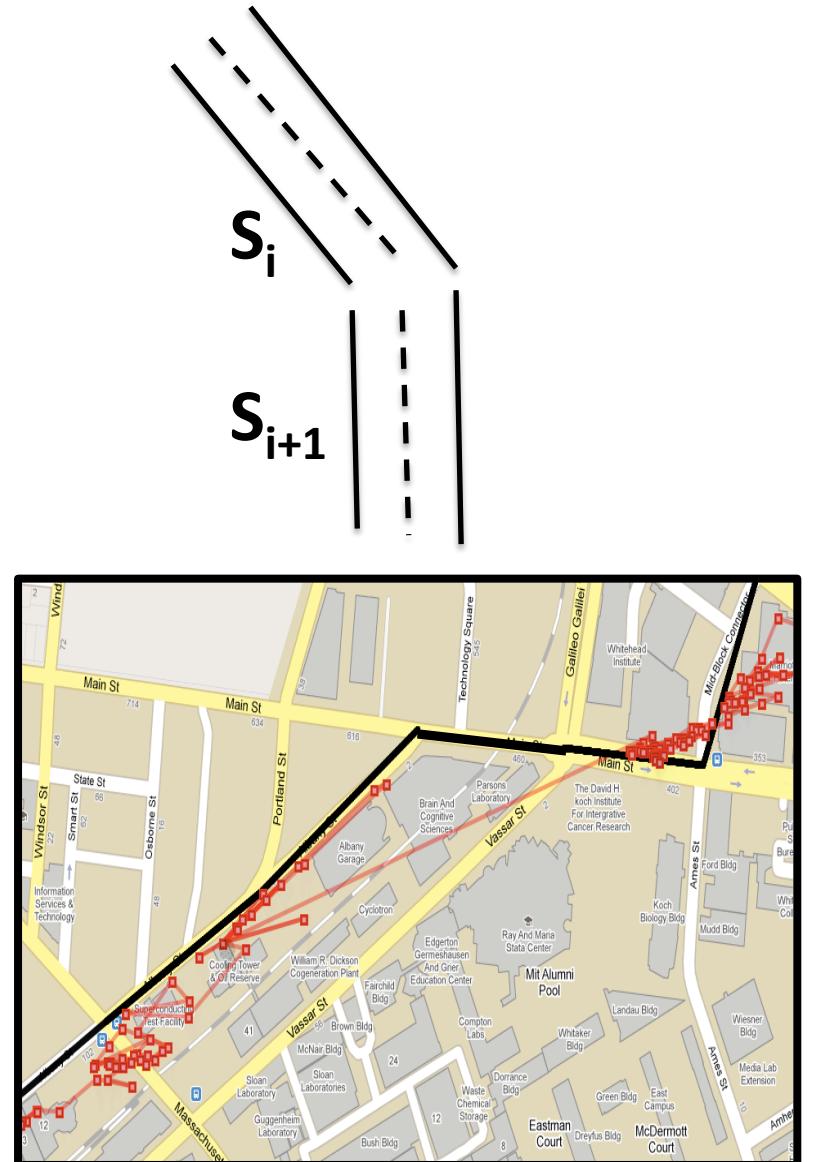
Emission Score

- Emission Score $E(p_i, S_i) = e^{-d^2/\sigma_{\text{sensor}}^2}$
 - Intuition: pts closer to a segment are more likely to come from it
 - σ_{sensor} depends on GPS/WiFi/Cellular

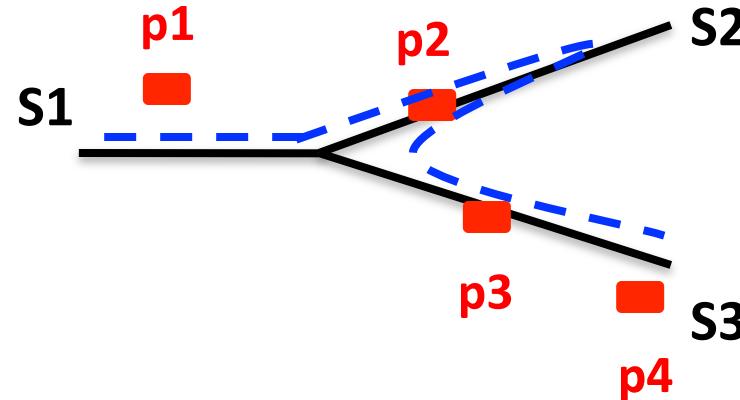


Transition Score

- Transition Score $T(S_i, S_{i+1})$
 - 0 if segments are not adjacent or not enough time has been spent on S_i
 - 1 if segments are adjacent and enough time has been spent on S_i
- **Speed constraint** is essential:
because algorithm jumps around and follows noise in the input data without it
 - Decreases error significantly

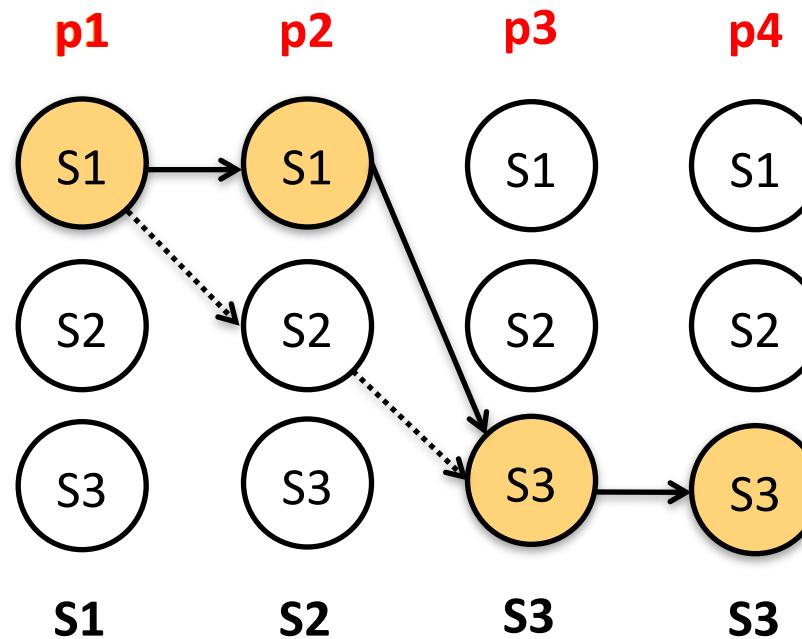


VTrack In Action



S1 S1 S3 S3 is
most likely match

S1 S2 S3 S3 has
score 0, isn't
permitted
(speed constraint)



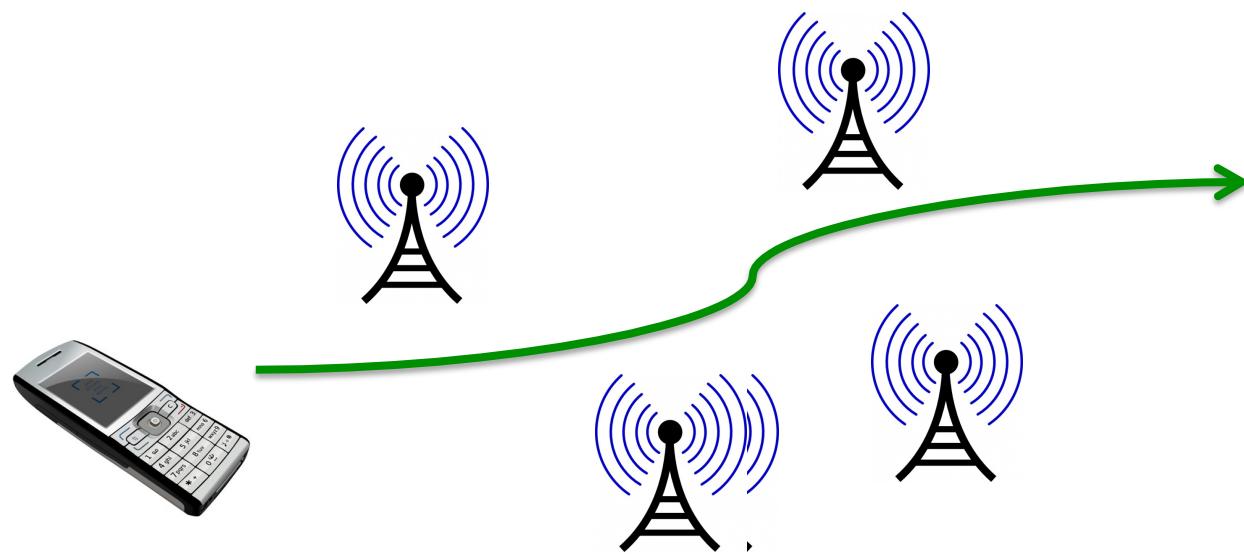
Handling Gaps: Interpolation

- VTrack's HMM maps input to output samples one-to-one
- We need frequent (1 Hz) input because we want output to be continuous (so we can enforce adjacency constraint)
- Interpolate gaps, *then* run HMM (linear interpolation)



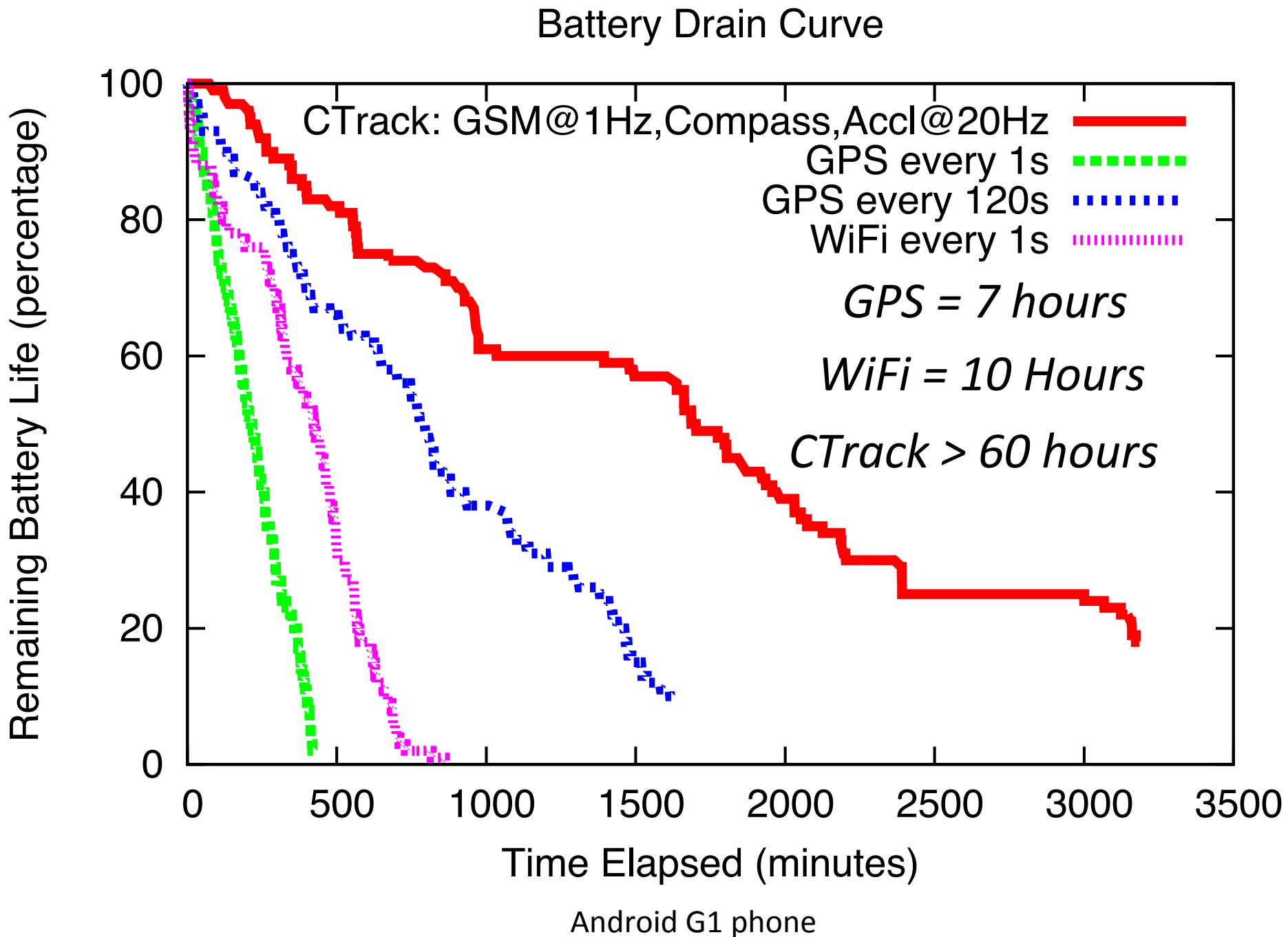
CTrack Problem Statement

- Can we develop techniques to process cellular signal information to produce accurate trajectories?
- How accurate?



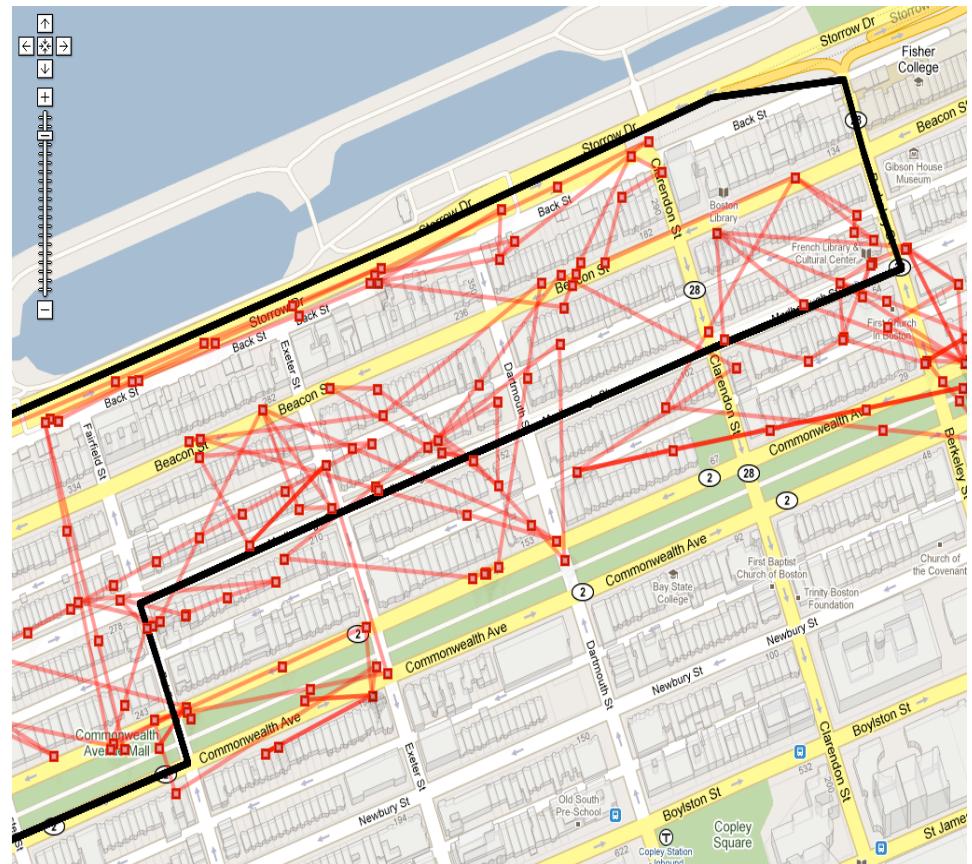
CTrack: Accurate Trajectory Mapping with Inaccurate Cellular Signals

- Consumes *no* extra energy
- New techniques achieve good enough accuracy for track-based apps
 - “75% as accurate as 1 Hz GPS”
 - “As accurate as GPS every 2 minutes”
 - As energy-efficient as GPS every 4 minutes and much more accurate”
 - Over “3x better” than previous cellular (GSM) systems
 - (I’ll explain what these mean)
- Optionally, augment with low-energy “*sensor hints*”
 - Compass to detect turns (15 μW @ 1 Hz)
 - Accelerometer to detect movement (60 μW @ 10 Hz)



Existing Cellular Location Systems Aren't Good Enough To Find Tracks

- State-of-the-art is “radio fingerprinting”
(E.g. PlaceLab)
- OK for best static localization estimate
- But poor at identifying tracks

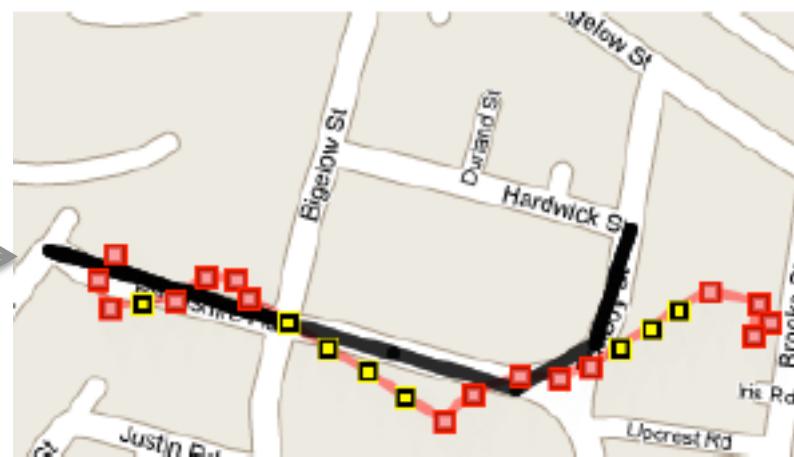


Existing Map-Matching Algorithms Perform Poorly w/ Cellular Radios

```
d8:30:62:5f:be:da, RSSI -94  
00:0f:b5:3d:43:20, RSSI -58  
00:18:0a:30:00:a3, RSSI -51  
• • • • • • • • • • • •
```

```
42.361,-71.09  
42.361,-71.09  
42.362,-71.091  
• • • • • • •
```

```
42.361,-71.09  
42.361,-71.09  
42.362,-71.091  
• • • • • • •
```



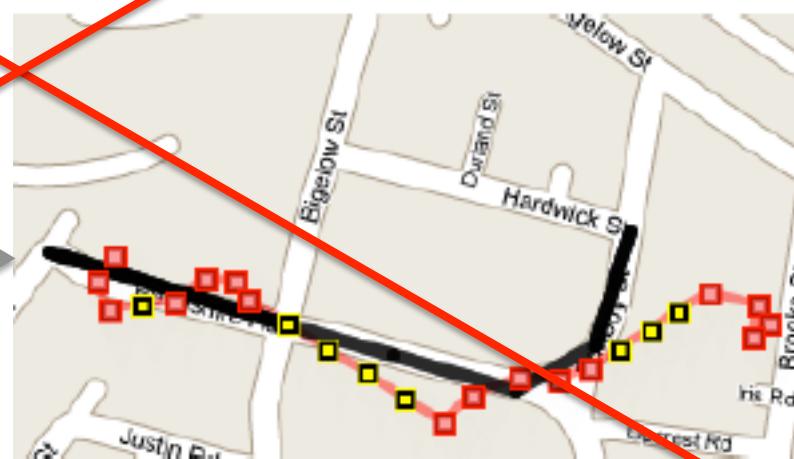
Krumm et al. (SAE World Congress '07), VTrack (SenSys '09)

Existing Map-Matching Algorithms Ok For GPS/WiFi, But Poor For GSM

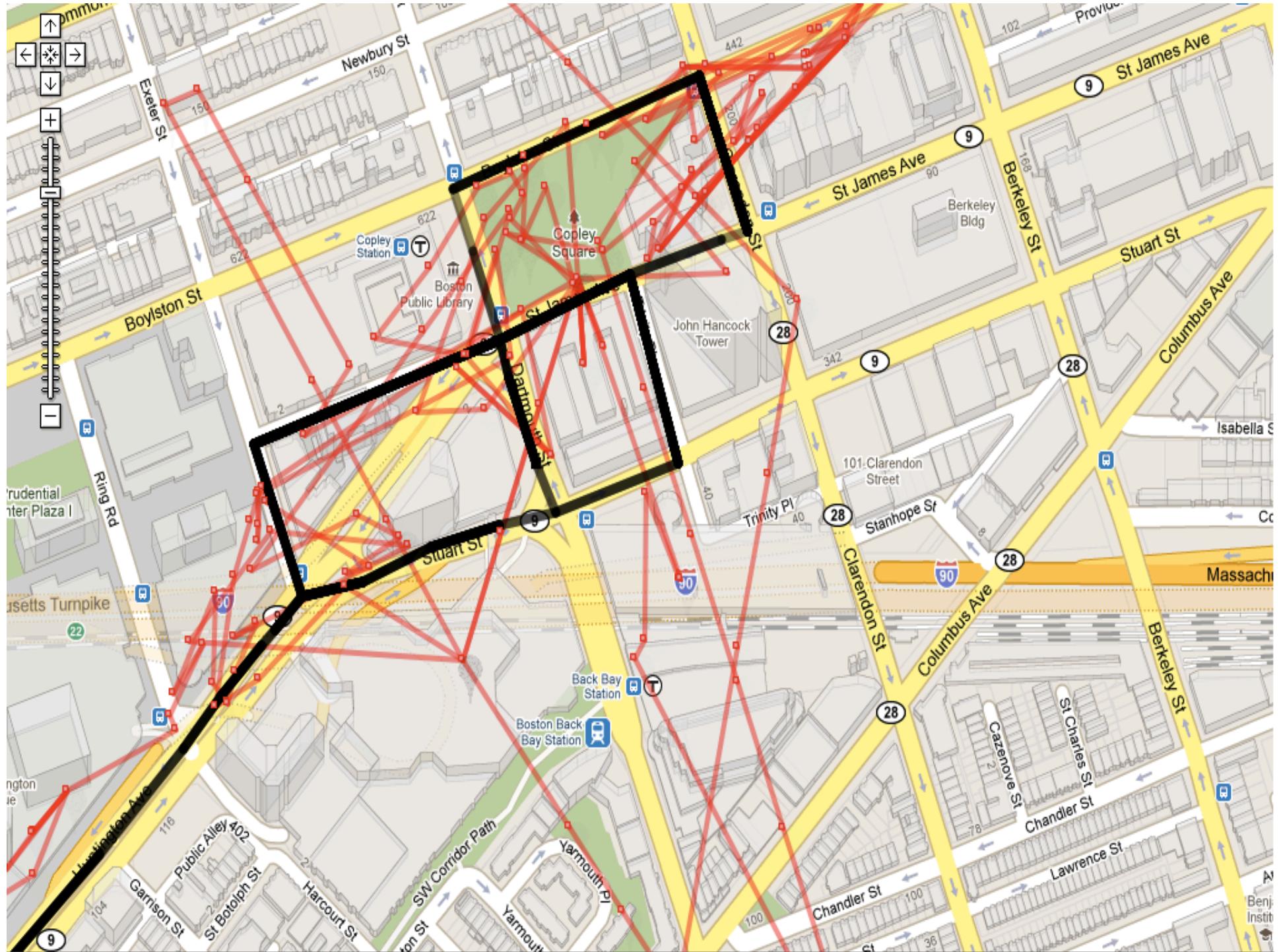
d8:30:62:5f:be:da, RSSI -94
00:0f:b5:3d:43:20, RSSI -58
00:18:0a:30:00:a3, RSSI -51
• • • • • • • • • • • •

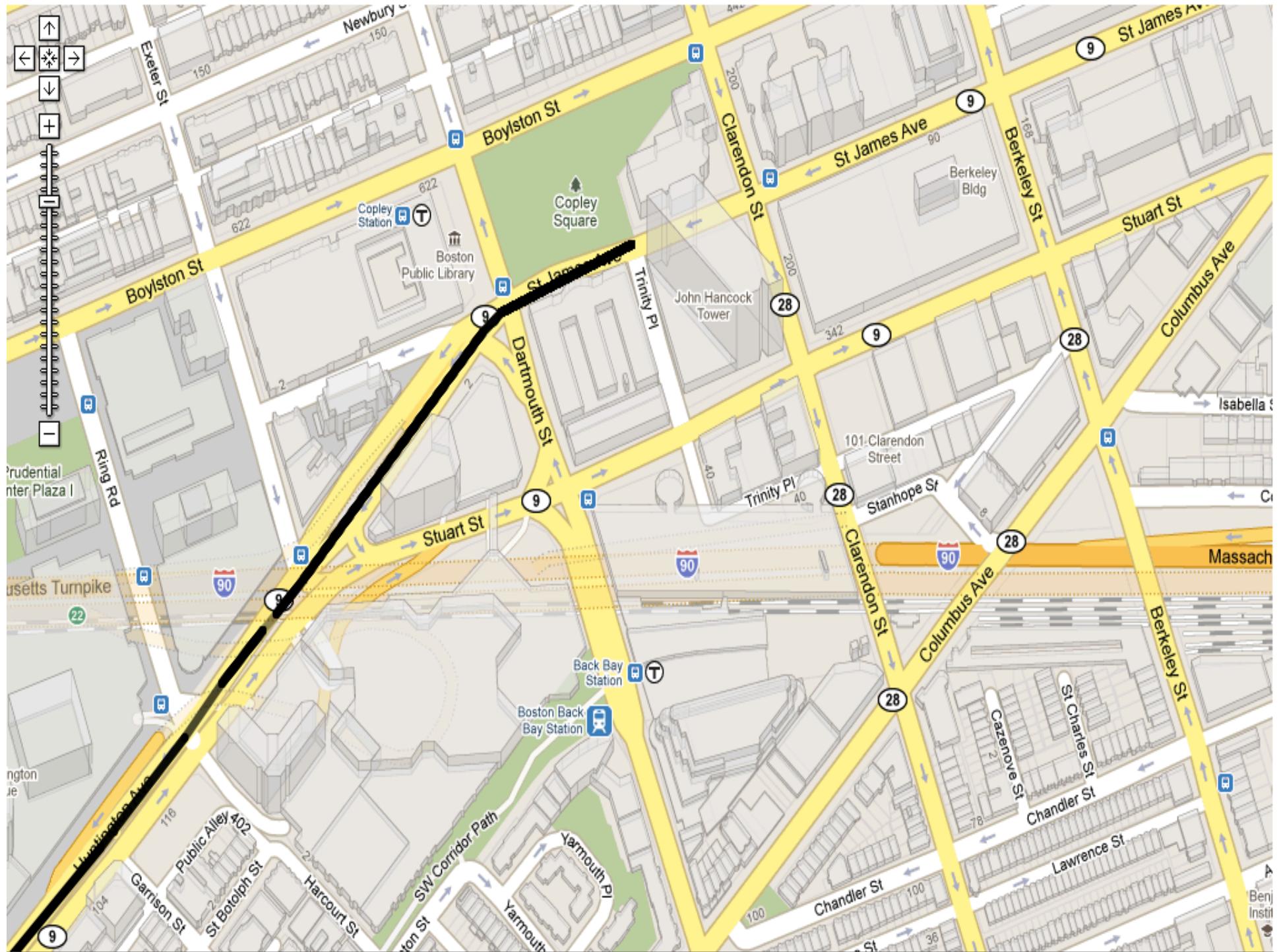
42.361,-71.09
42.361,-71.09
42.362,-71.091
• • • • • • •

42.361,-71.09
42.361,-71.09
42.362,-71.091
• • • • • •



Krumm et al. (SAE World Congress '07), VTrack (SenSys '09)

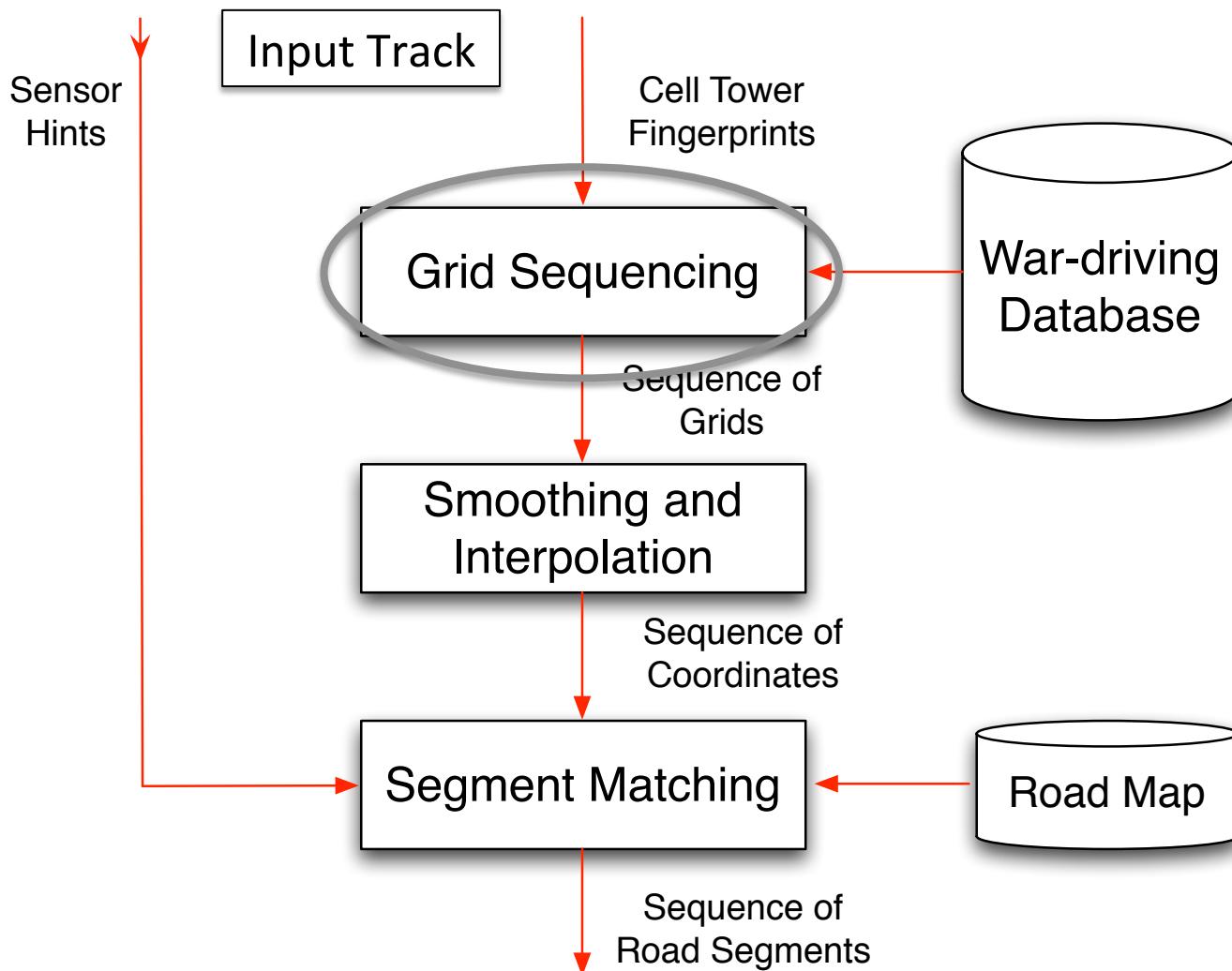




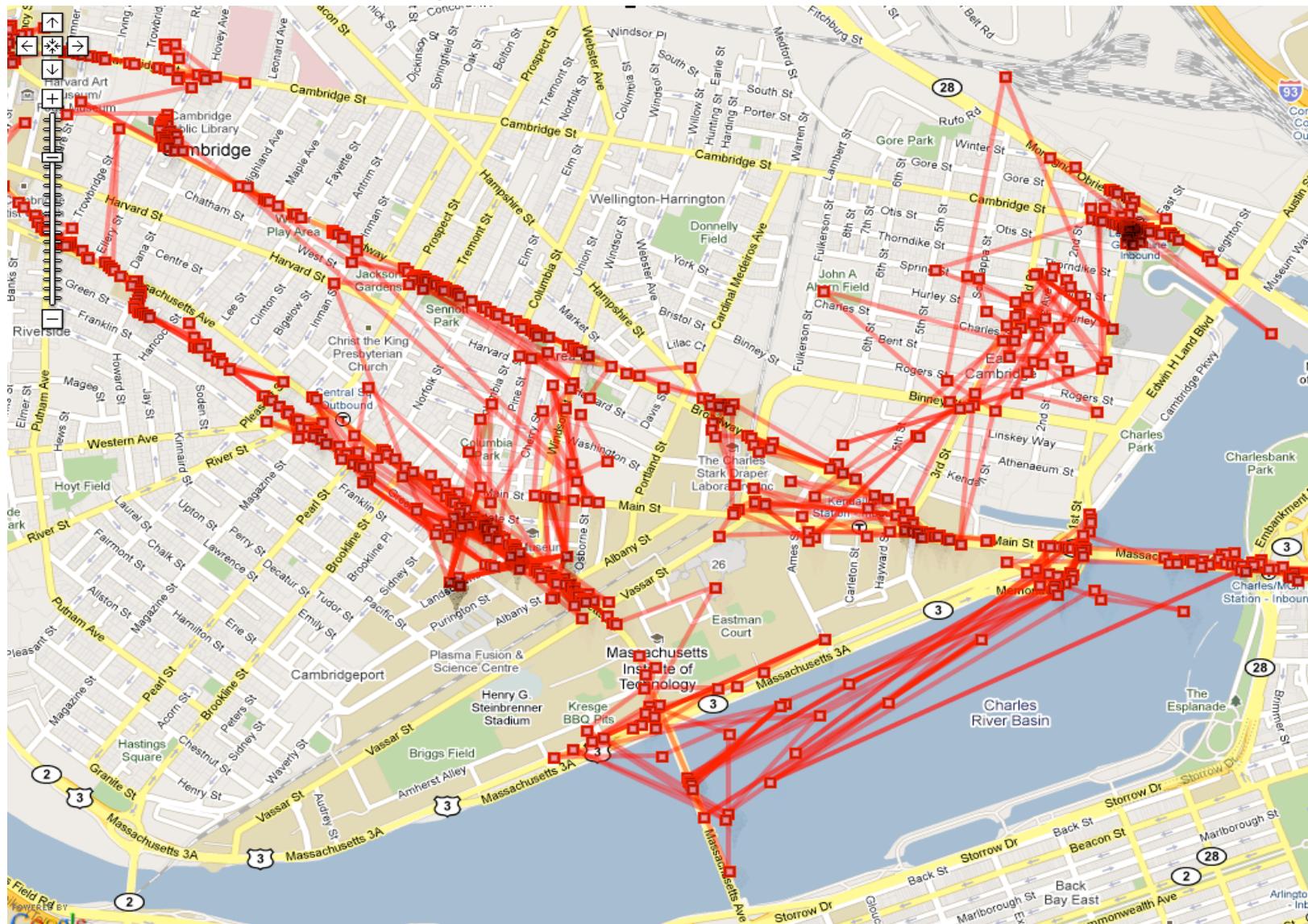
Key Insight in CTrack

- Do *not* convert radio fingerprints to (lat, lon) coordinates and then sequence them on map
- Instead, first *sequence* GSM fingerprints on a spatial grid
- This insight is crucial: it reduces error by 3x

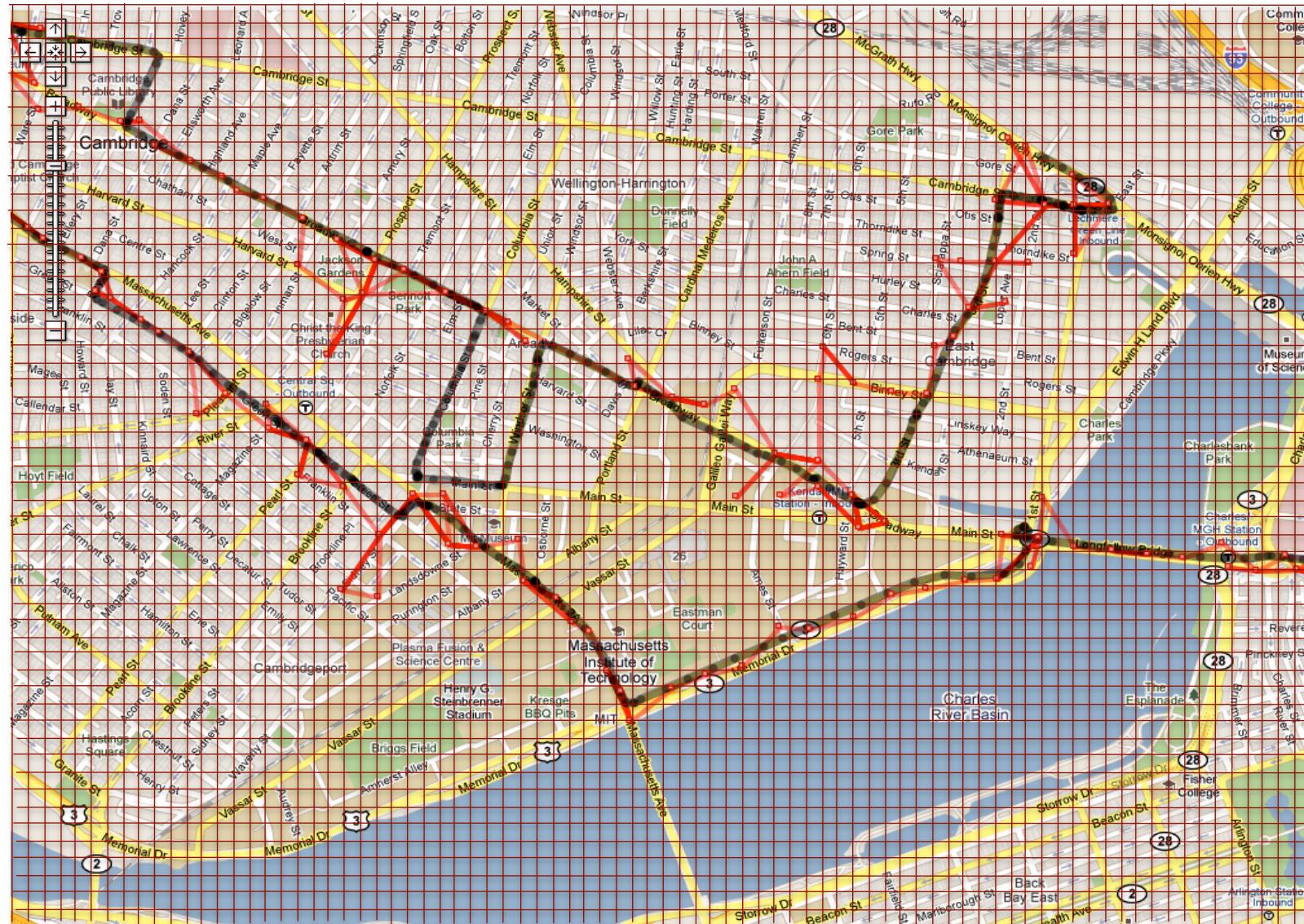
CTrack FlowChart



Raw points (Placelab)



Grid Sequence



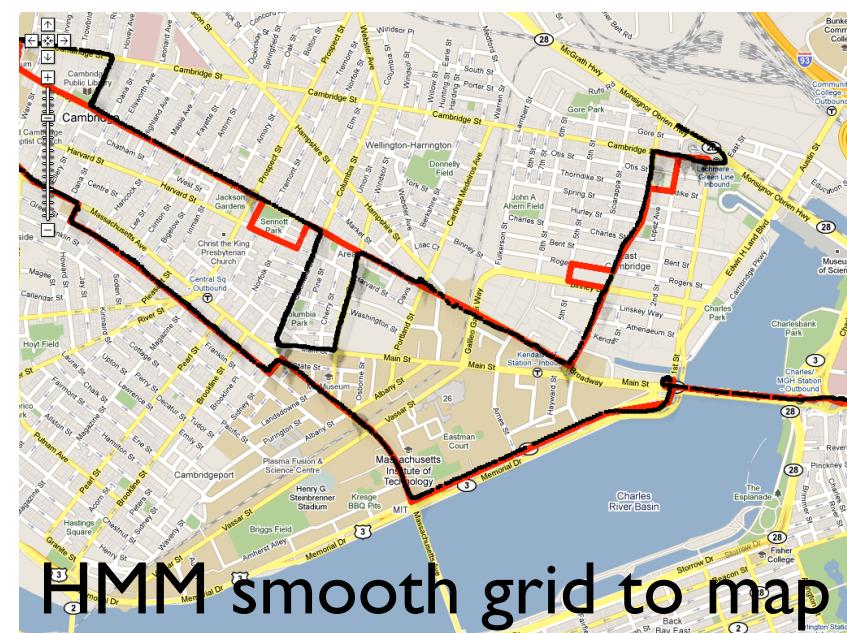
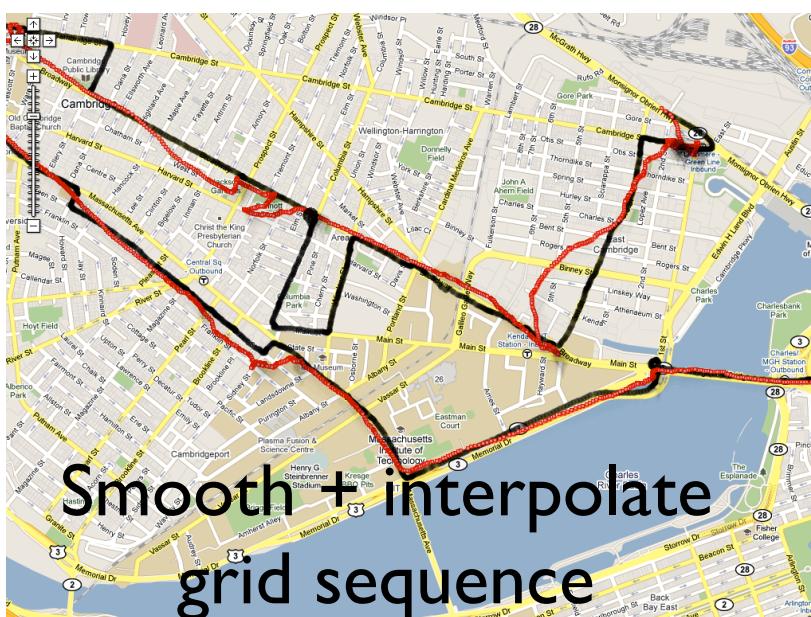
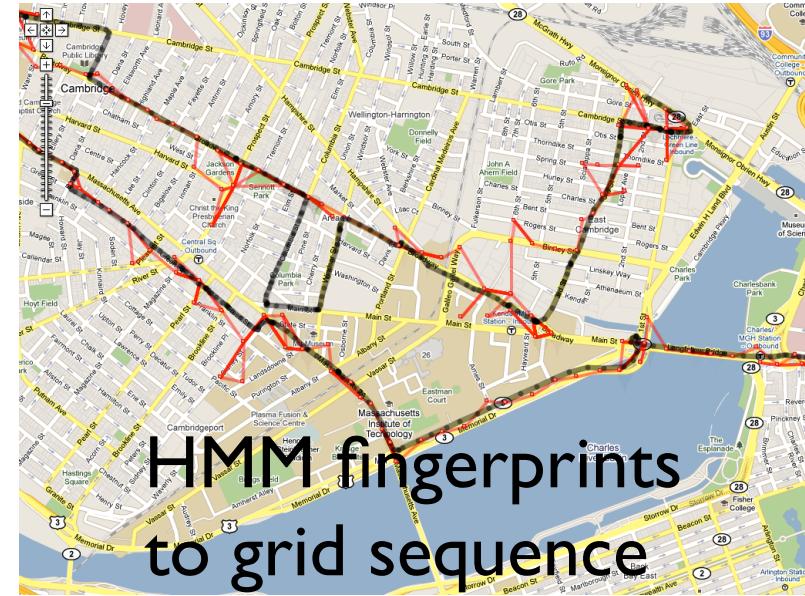
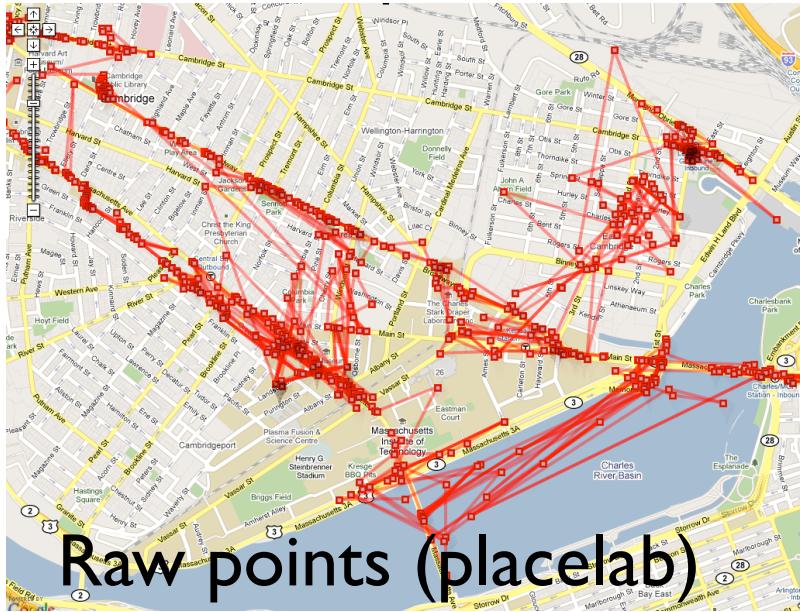
Smooth + Interpolate Grid Sequence



Smoothed Grid → Road Segments



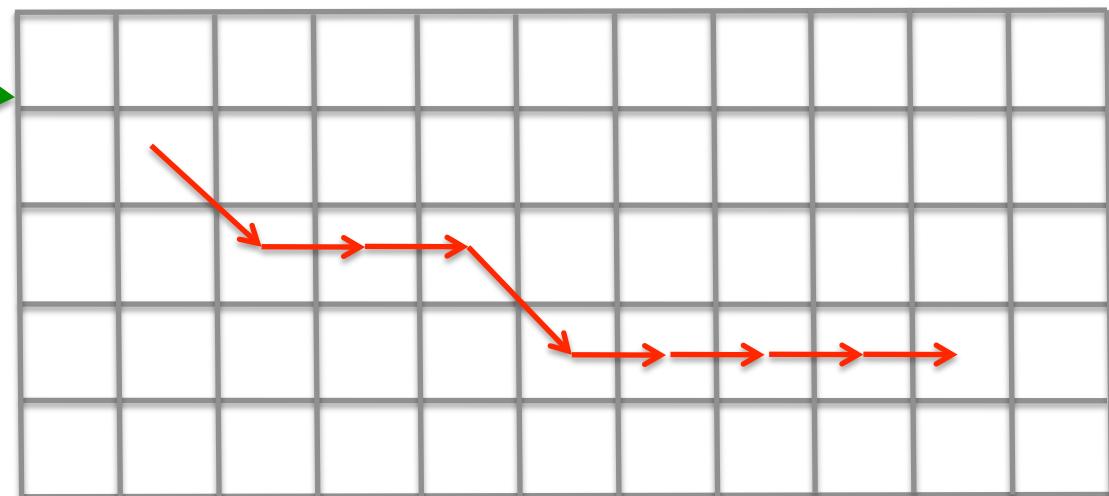
CTrack Steps



Grid Sequencing

Time	TowerId	RSSI
18.03	334490560	14
	334478599	12
	337772865	18
	334478600	14
	334470539	12
	334490699	12
19.01

Size of grid = 125 meters
Why?



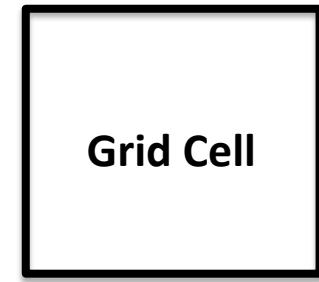
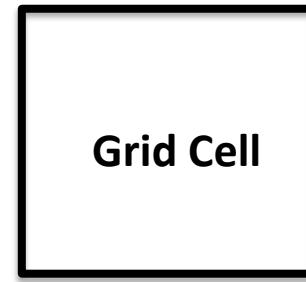
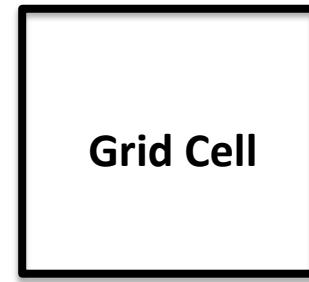
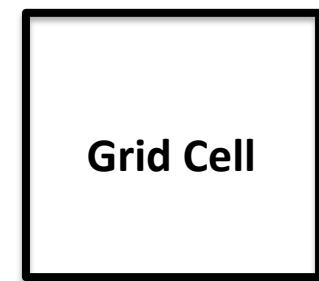
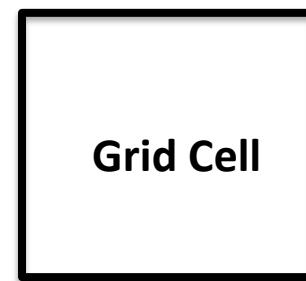
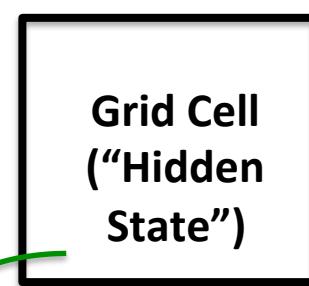
Given a sequence of GSM fingerprints (TowerID, RSSI), what is the most likely sequence of grid cells?

HMM For Grid Sequencing

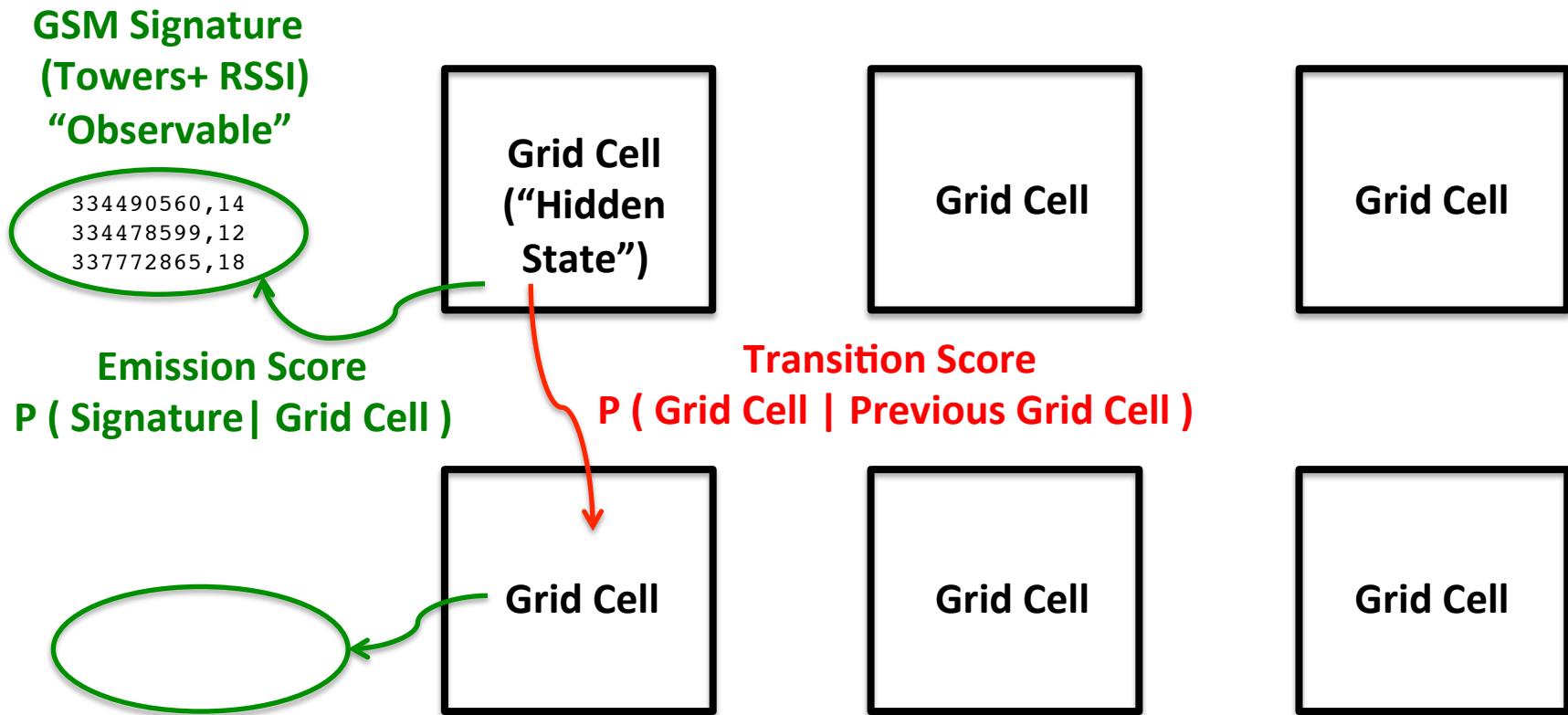
GSM Signature
(Towers+ RSSI)
“Observable”

334490560,14
334478599,12
337772865,18

Emission Score
 $P(\text{Signature} | \text{Grid Cell})$



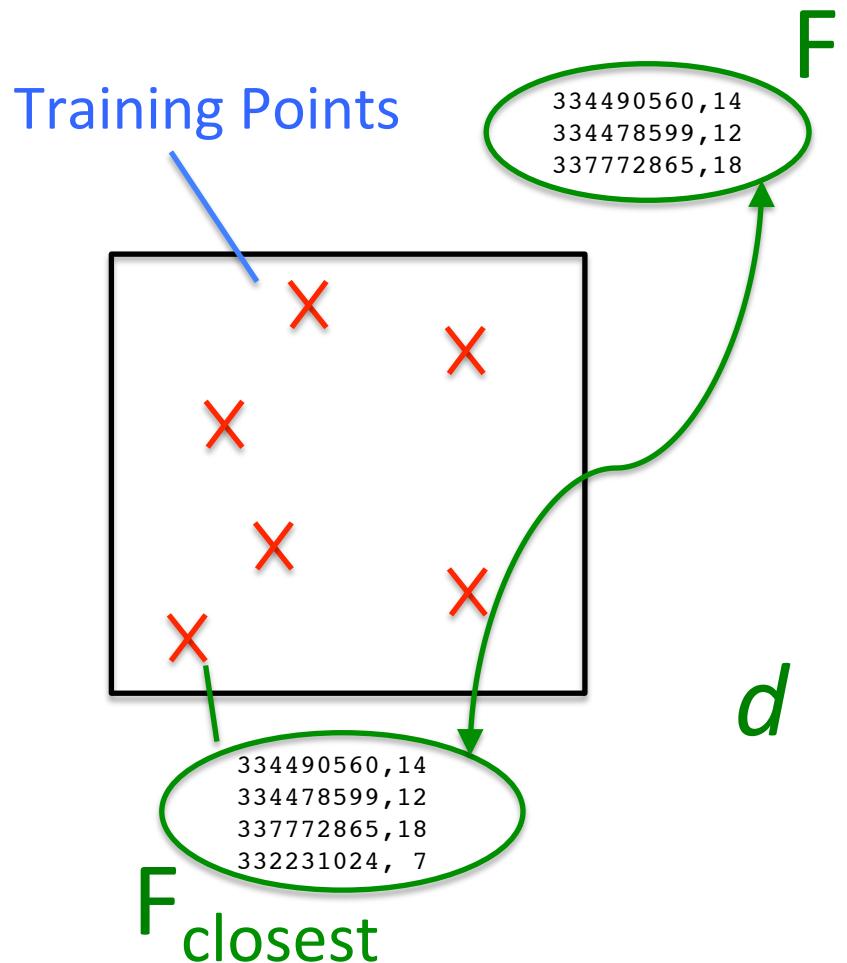
HMM For Grid Sequencing



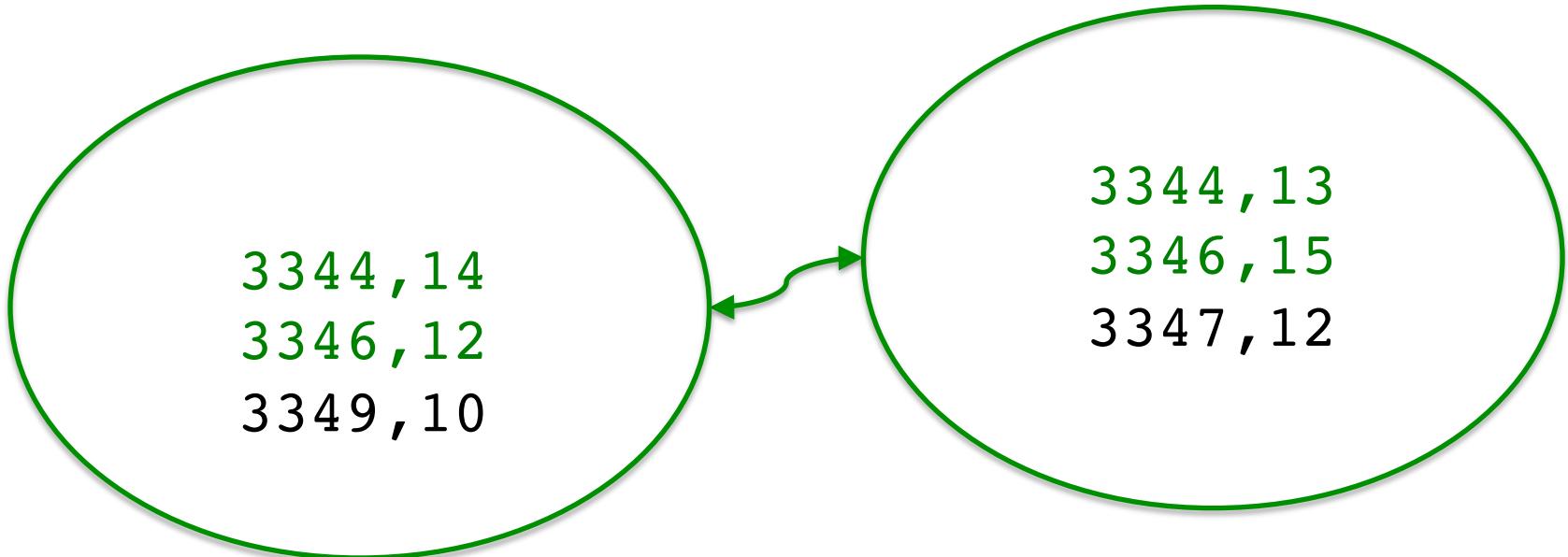
Dynamic Programming Finds Best Grid Sequence (cf. Viterbi)
“Best” => Max (Emission Score * Transition Score)

Emission Score (Grid Cell G, Fingerprint F)

- Find *closest* matching fingerprint F_{closest} to F in all training data for grid cell G
- Score is *inversely proportional* to “distance” d of F_{closest} from F in signal strength space
- Better match => smaller d => higher score



Example



$$d = \lambda * 2 + (d_{\max} - 0.5 * \sqrt{(14-13)^2 + (12-15)^2})$$

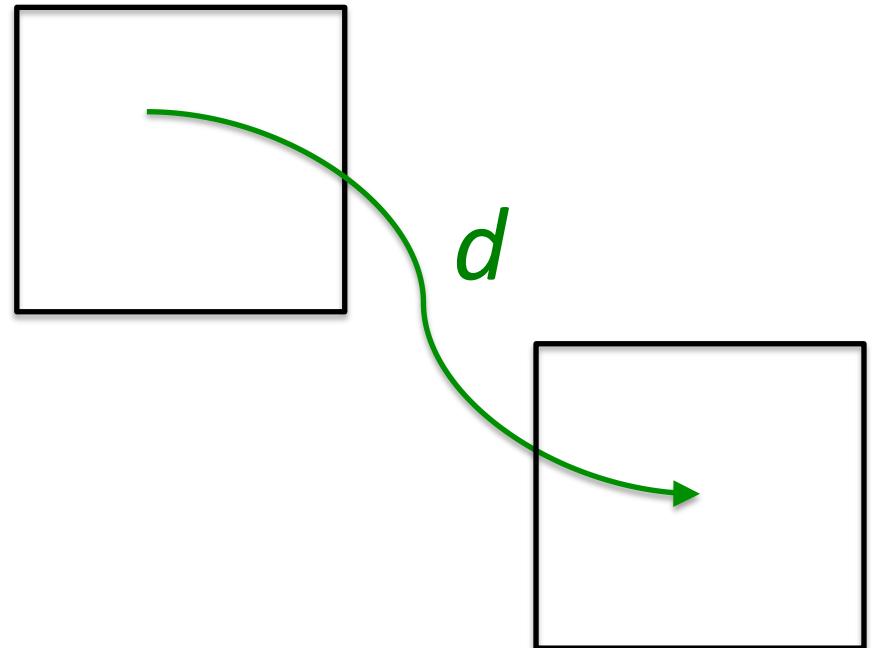
With $\lambda=3$ and $d_{\max}=32$,

Emission Score = $38 - \sqrt{10}/2$

Normalize this to $(0,1]$ range

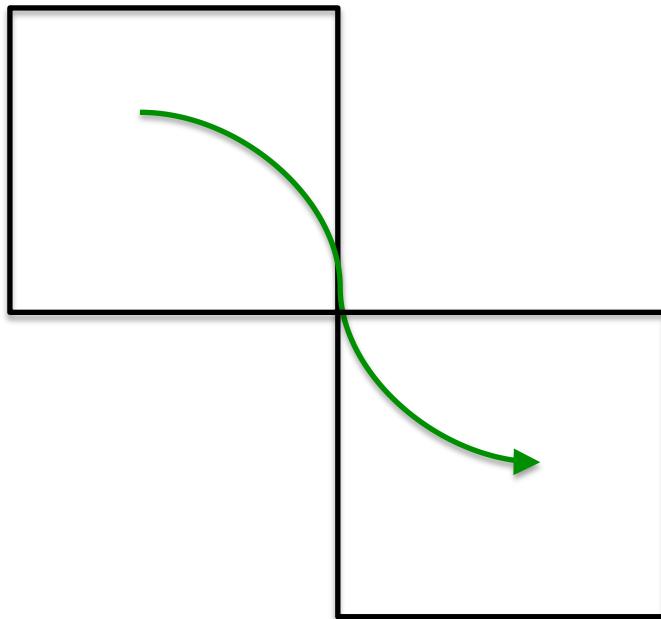
Tolerant Transition Score

- Inversely proportional to distance between grid cells
- The score is ***very tolerant*** of jumps between non-adjacent grid cells
- Necessary to tolerate large outliers/regions of poor coverage in the GSM data

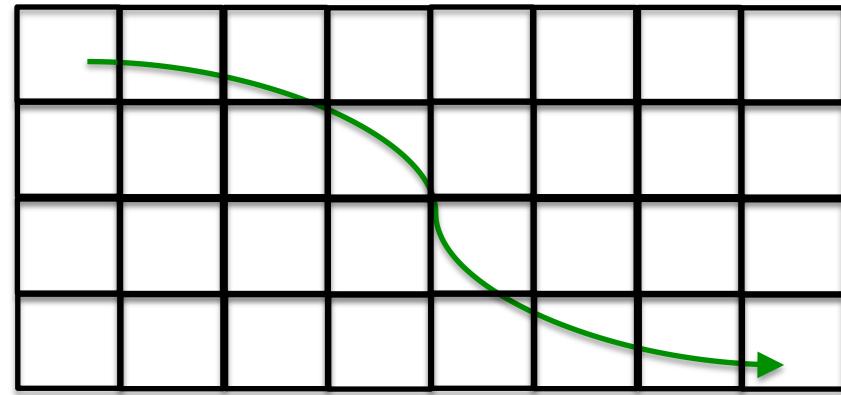


$$\text{Score} = 1/d$$

Example

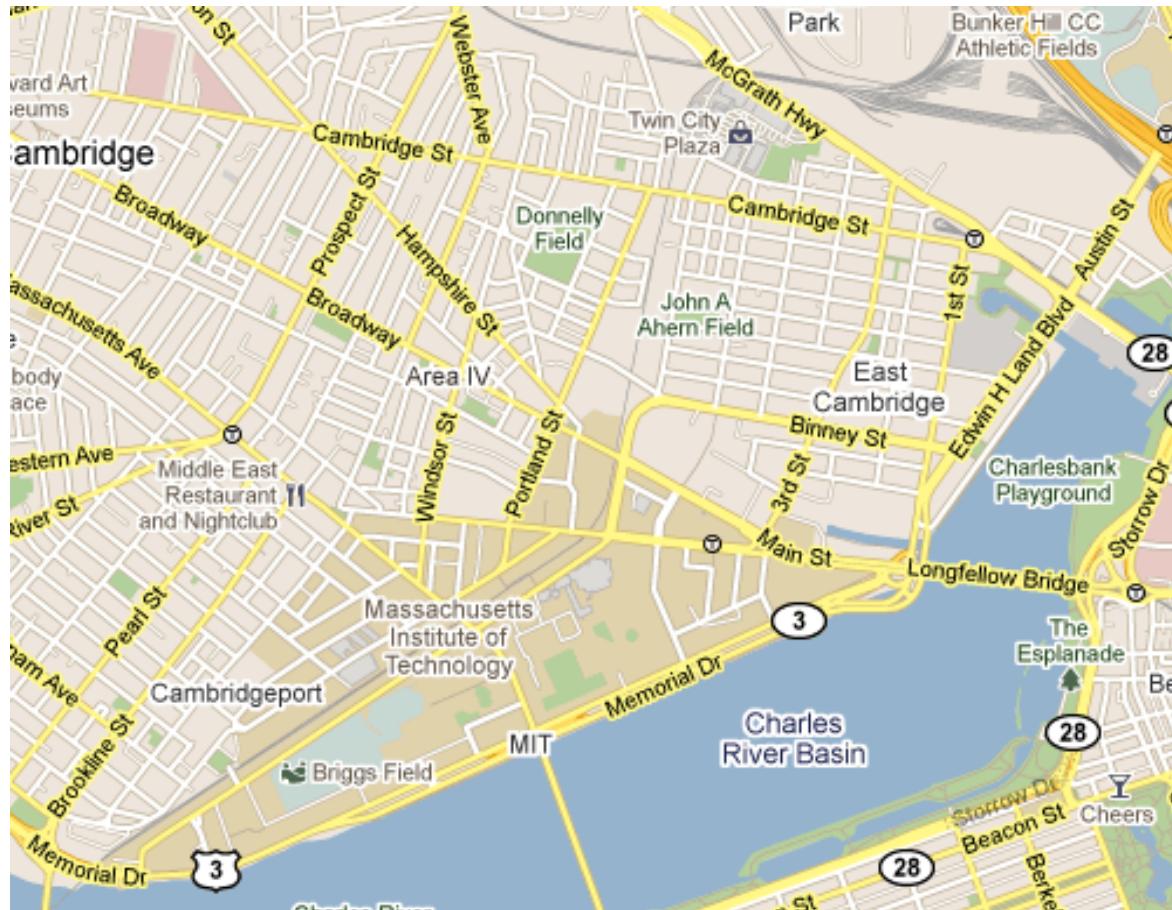


Transition Score = 1

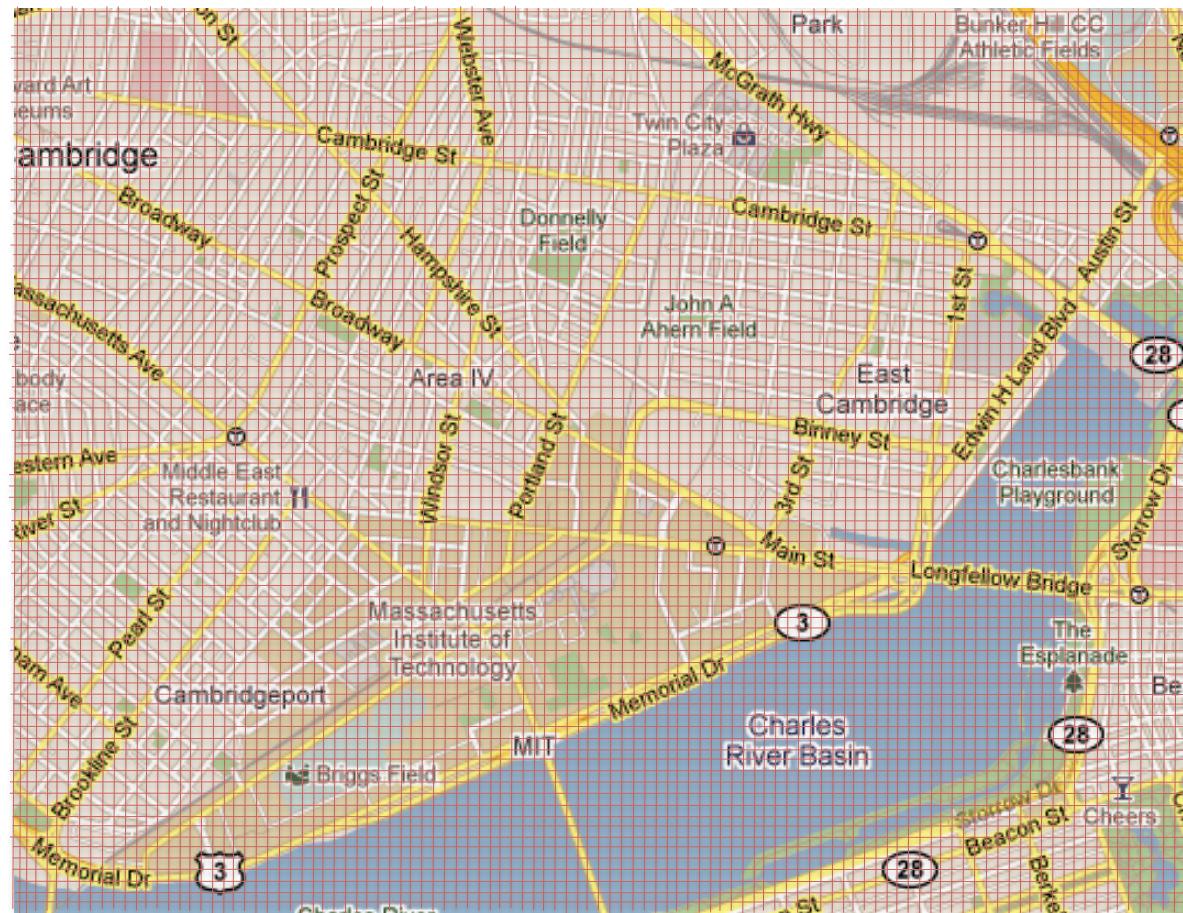


$d = 7$
Transition Score = 1/7

Grid Sequencing In Action

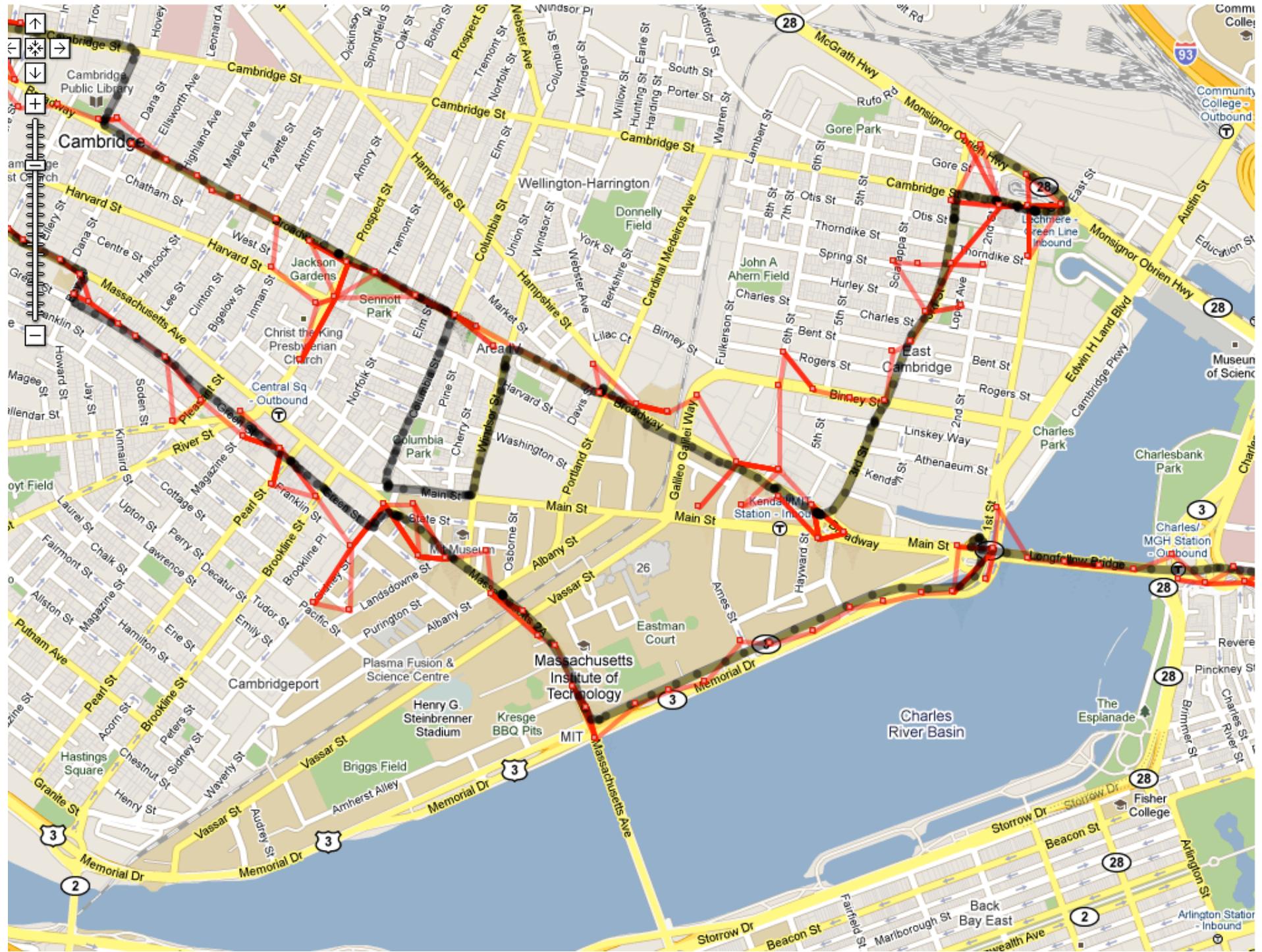


Grid Sequencing In Action



Grid Sequencing In Action

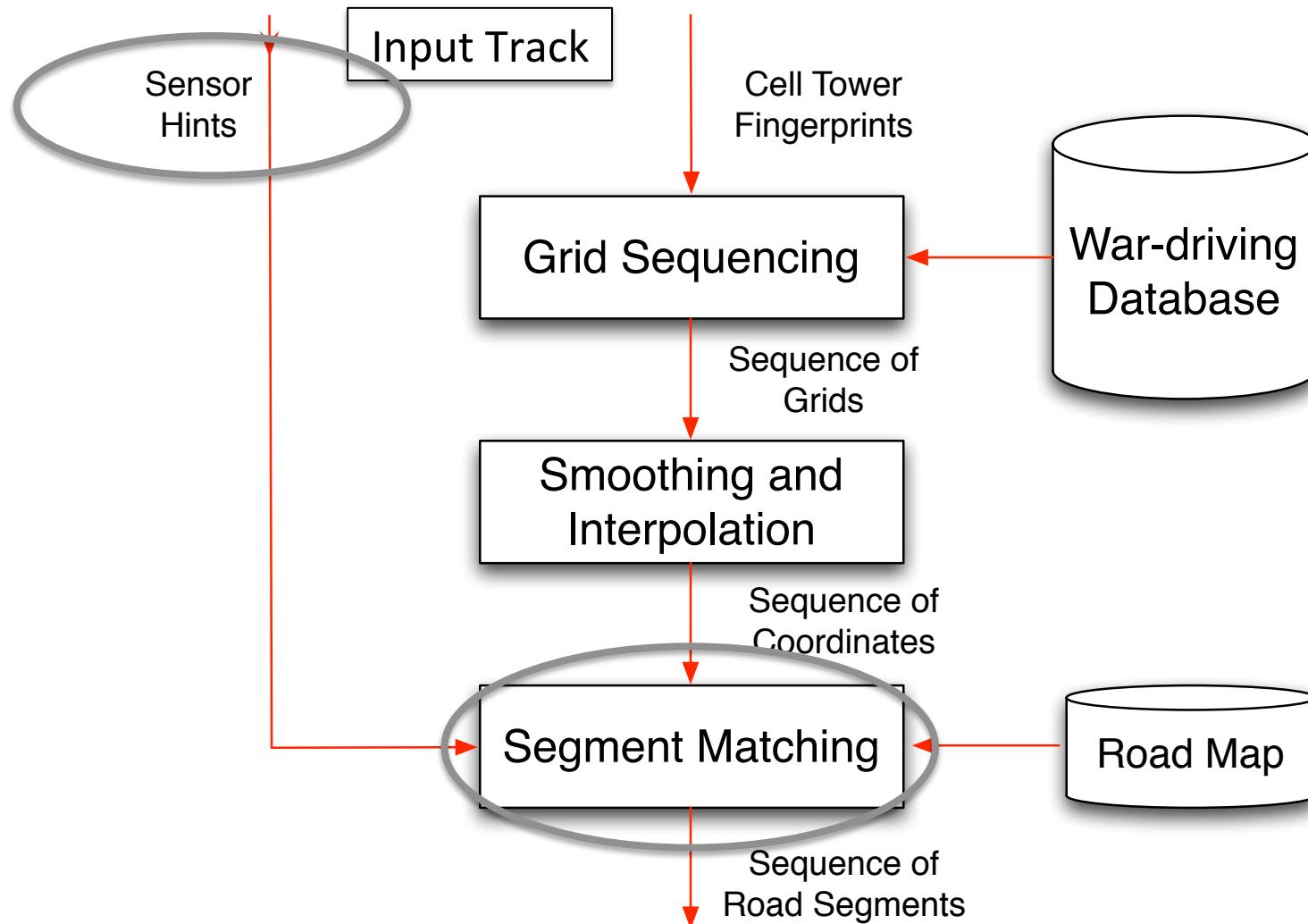




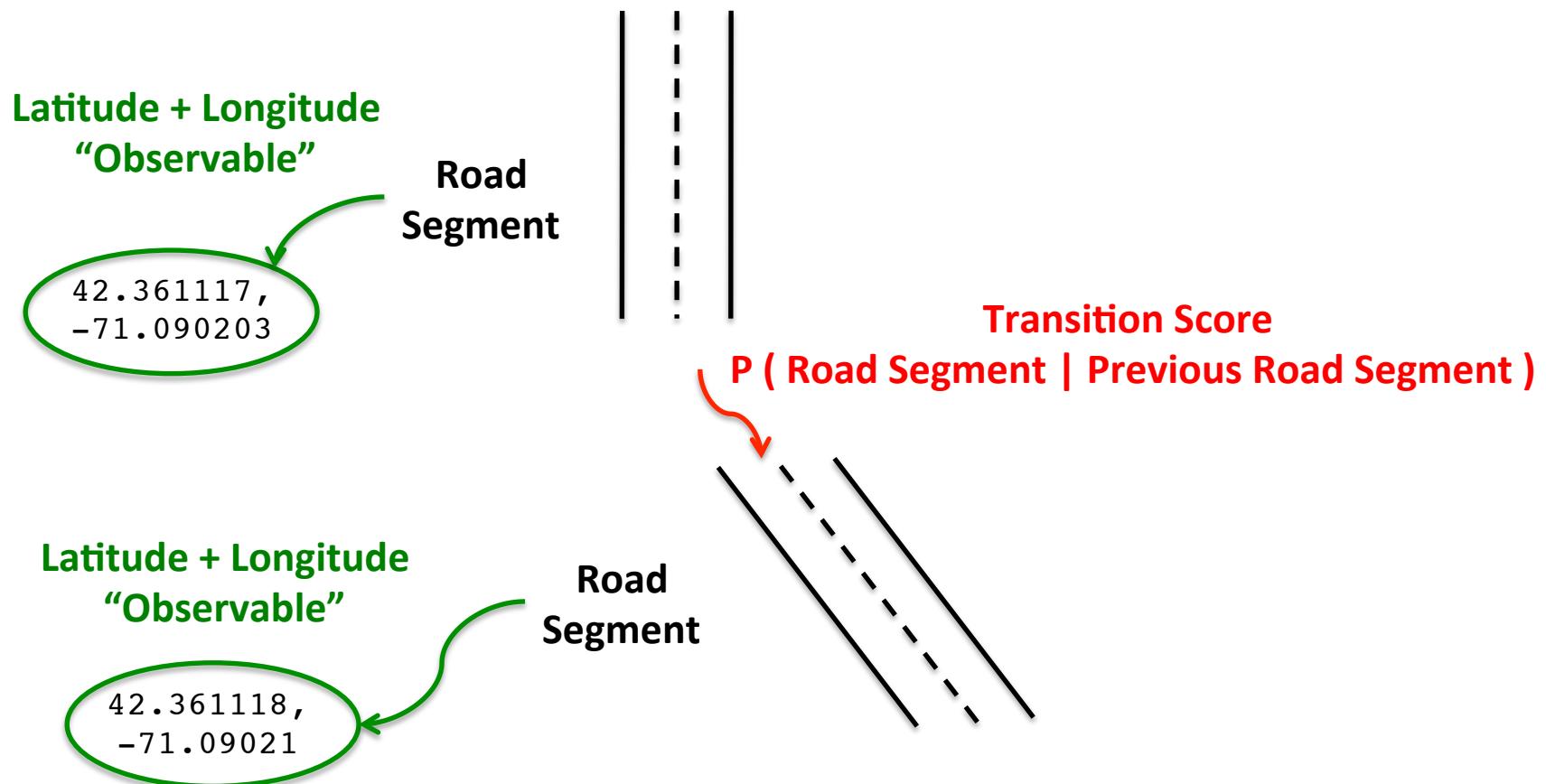
Smoothing & Interpolation



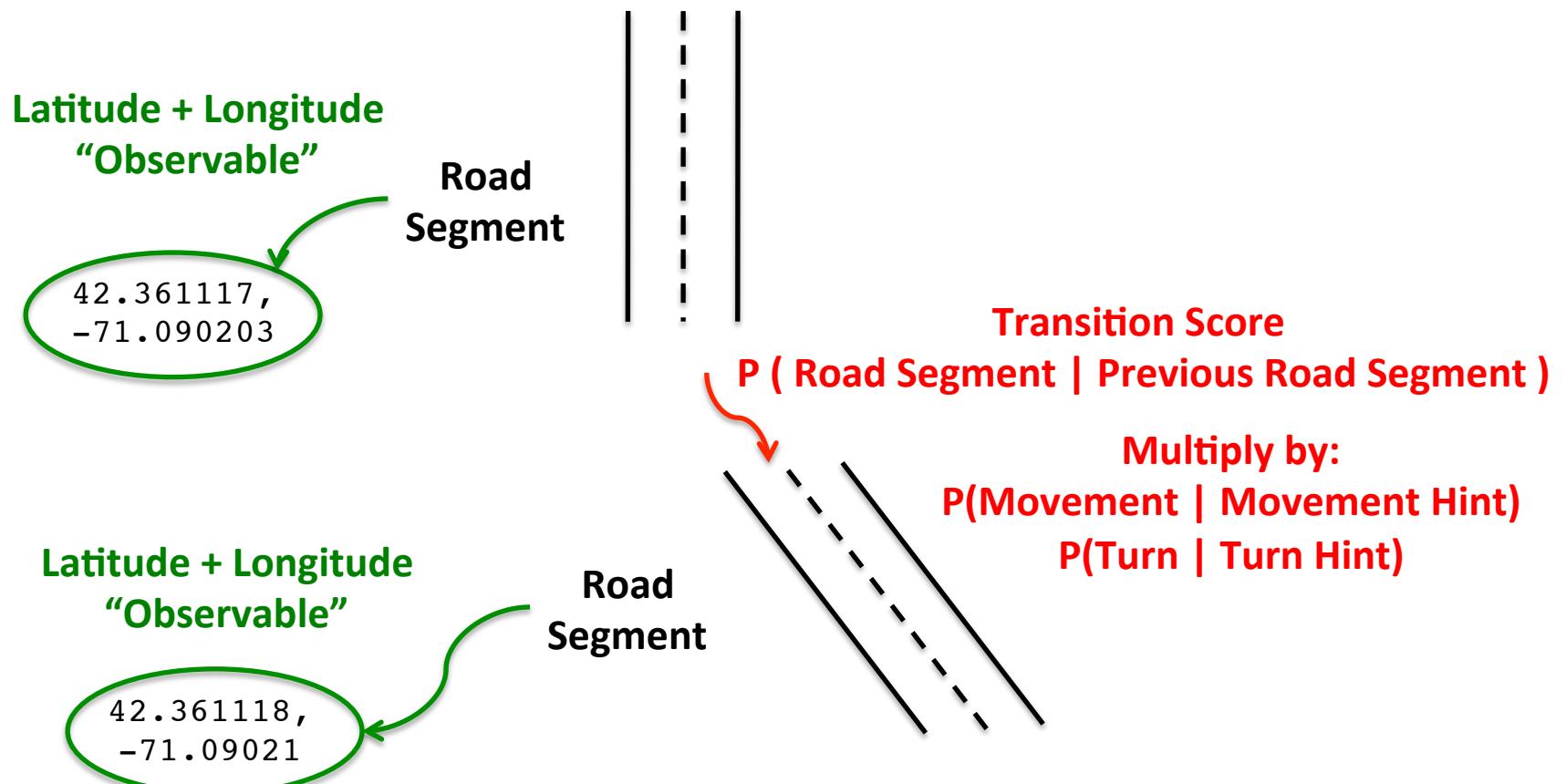
CTrack FlowChart



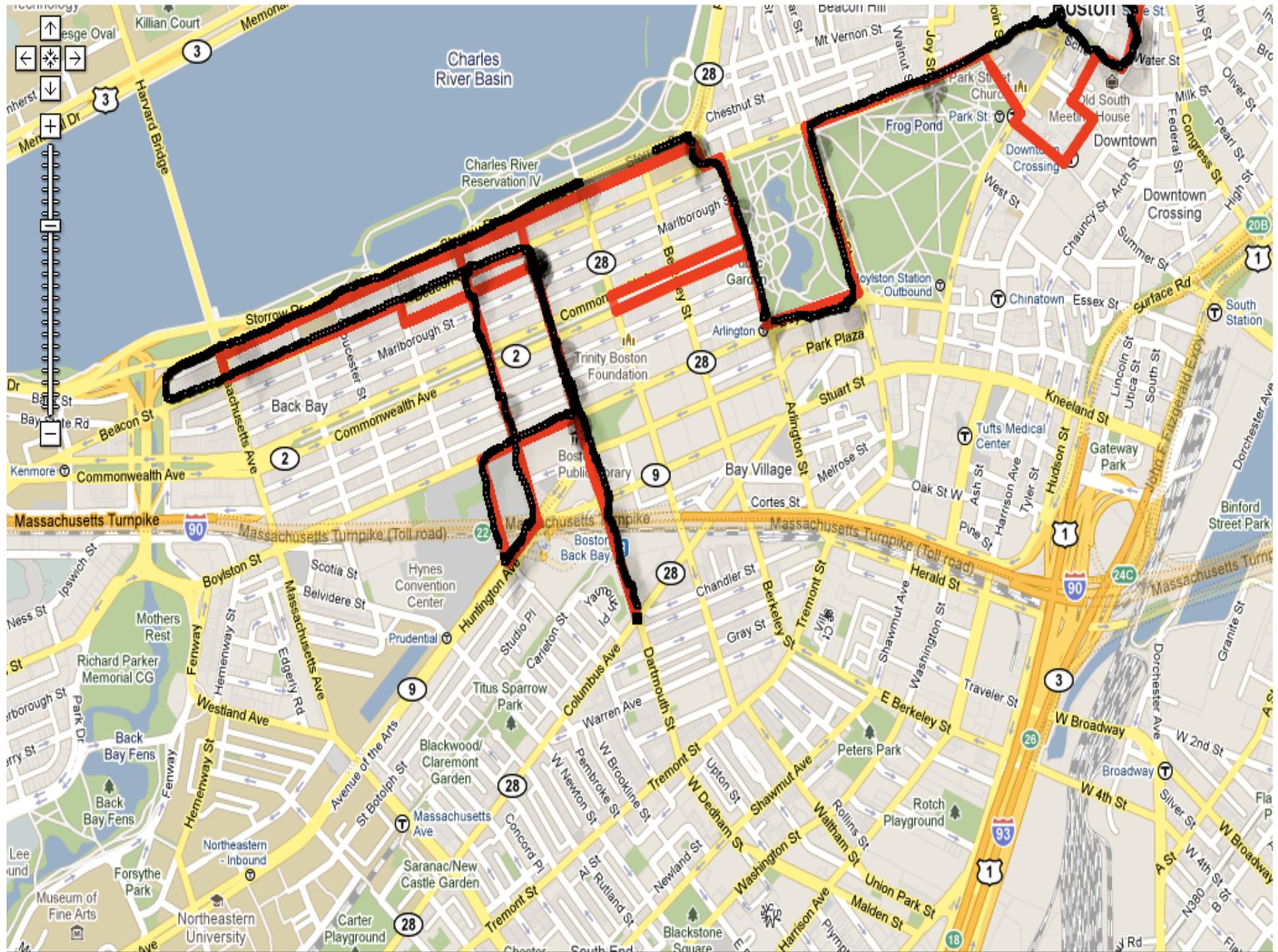
Matching (Lat, Lon) To Segments

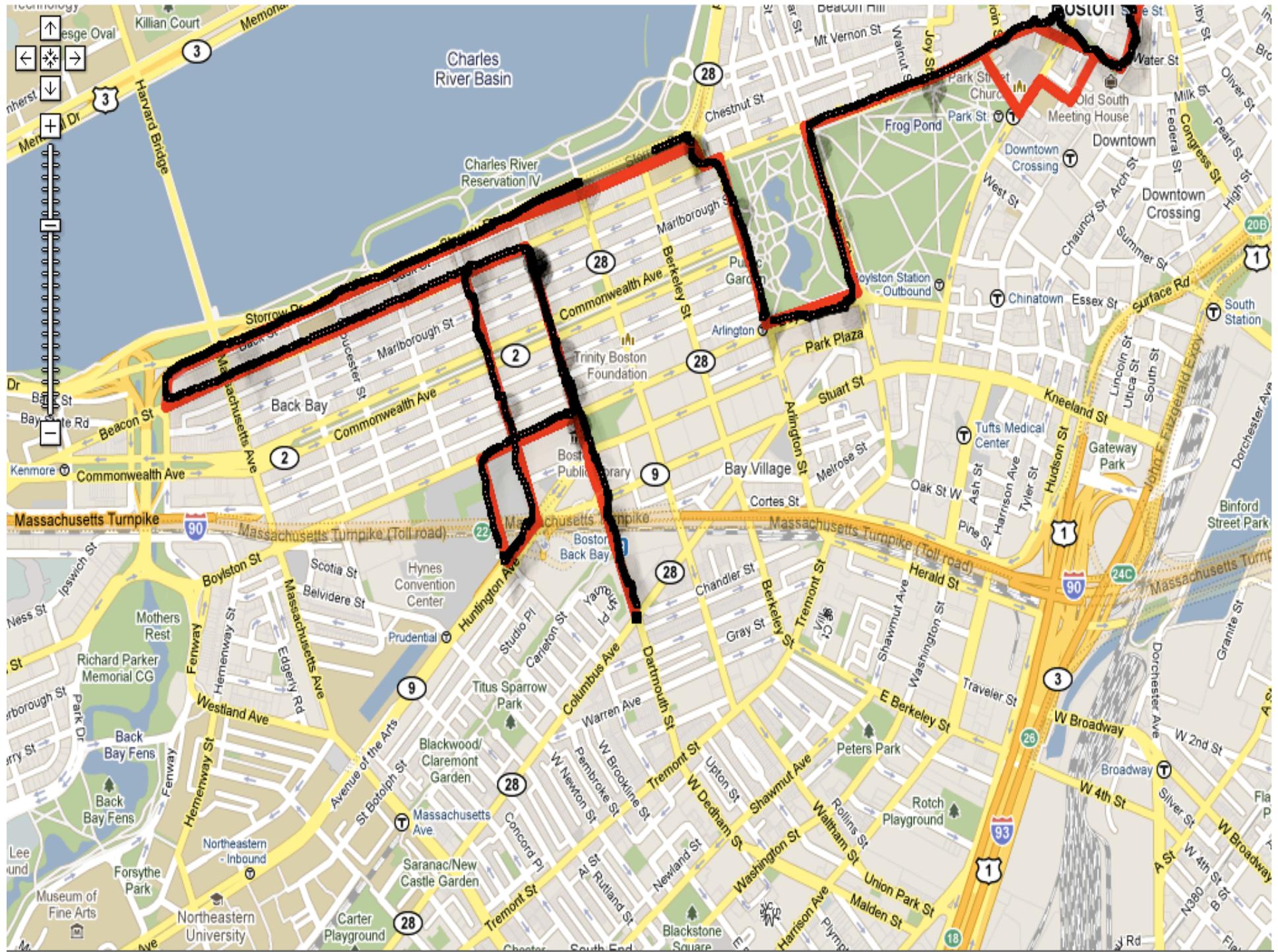


Matching (Lat, Lon) To Segments



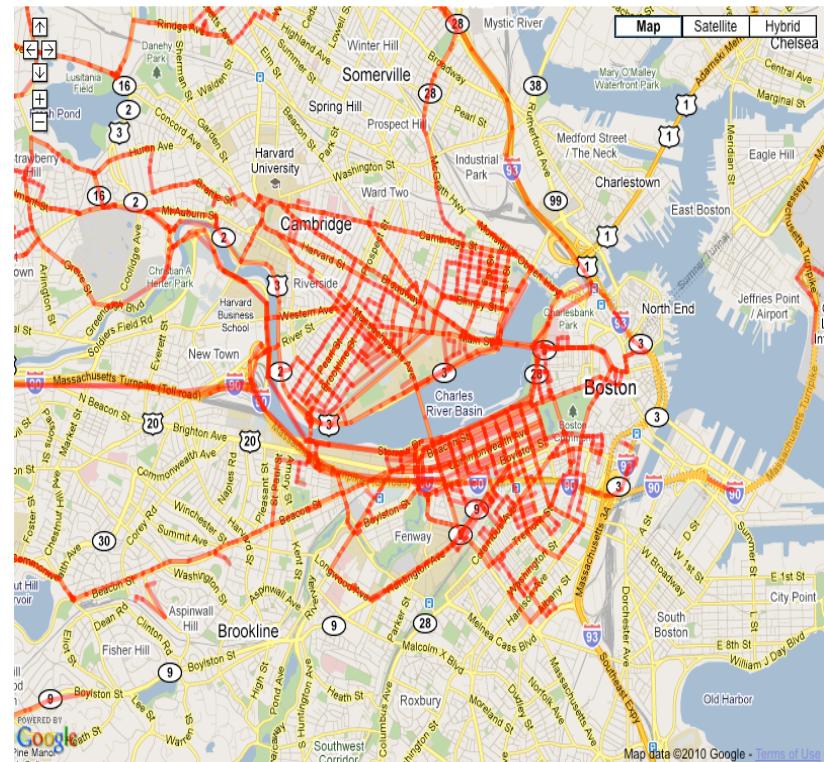
Extract 0/1 (Binary) Movement and Turn Hints For Each Time Slot





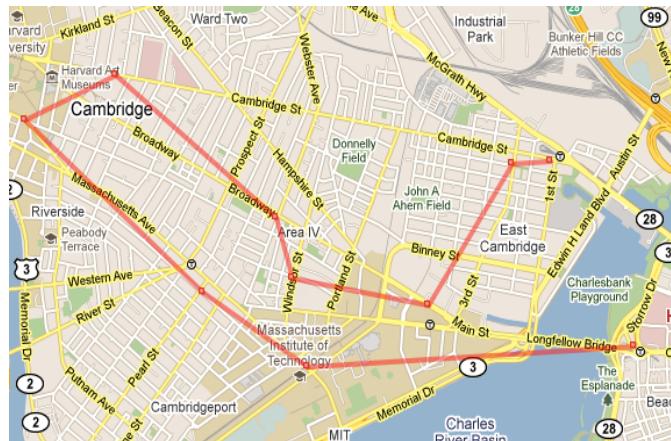
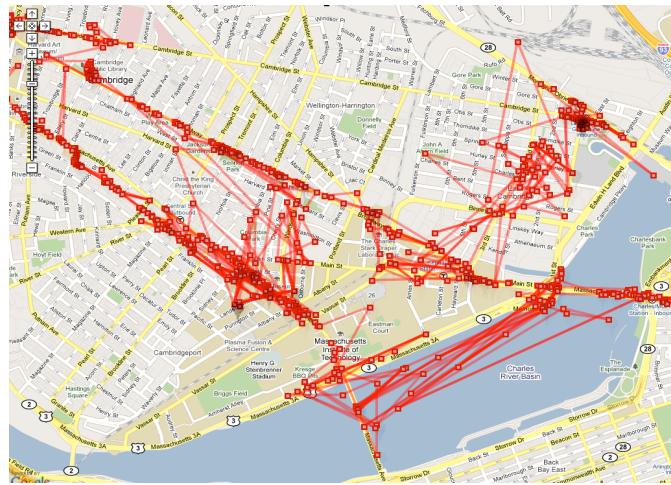
Evaluation

- 125 Hours (312 “Drives”)
 - From 16 Android phones
 - Logged GPS ground truth, GSM, accel, compass
- Selected subset of 53 drives (28 hours) as “test drives”
 - Tests lie in dense cov. area
 - Tests have good GPS accuracy
 - Mean drive length: 30-35 mins
- Leave-one-out evaluation
 - Train on all but test, evaluate on test drive

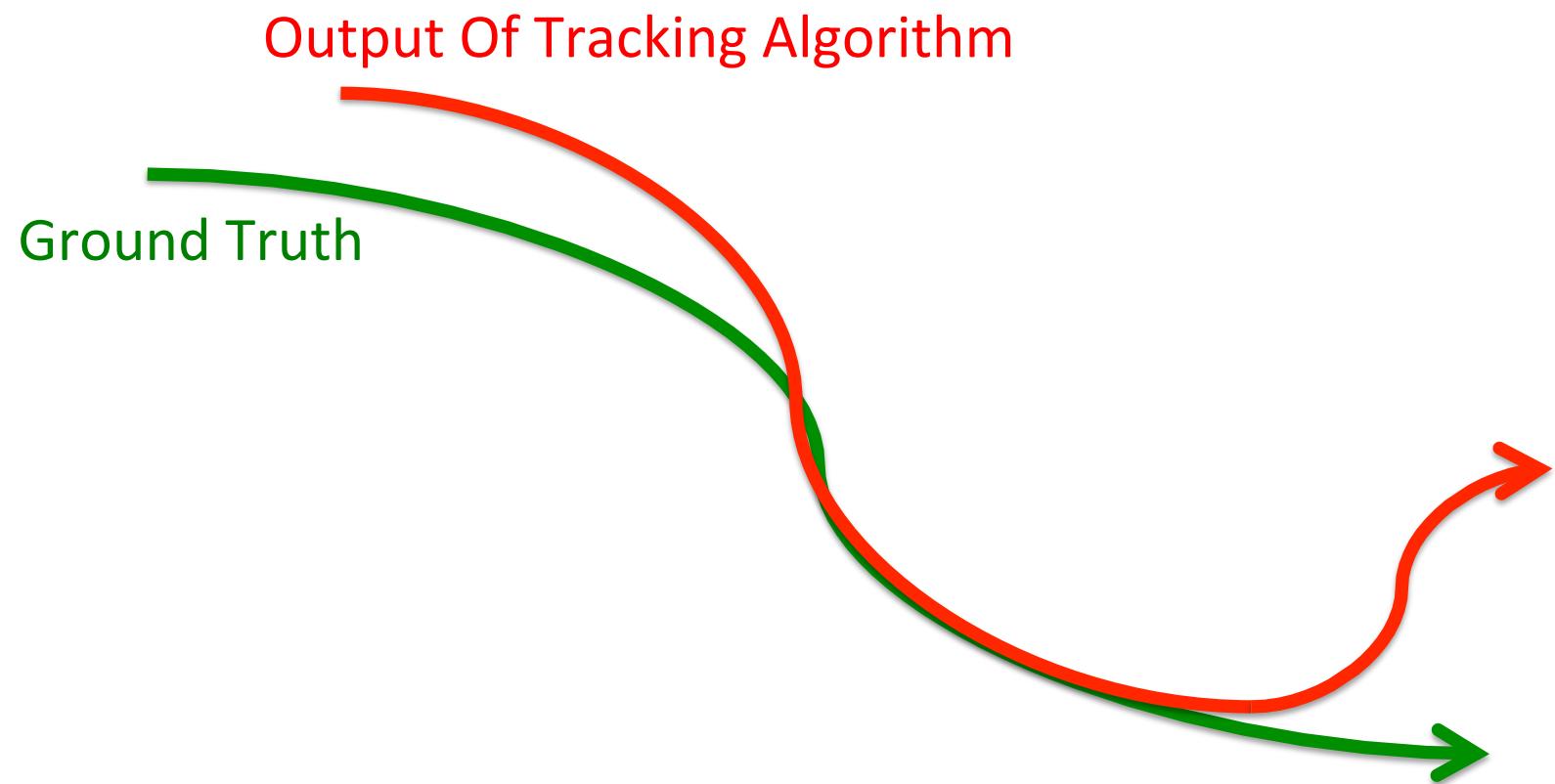


We Compared CTrack To...

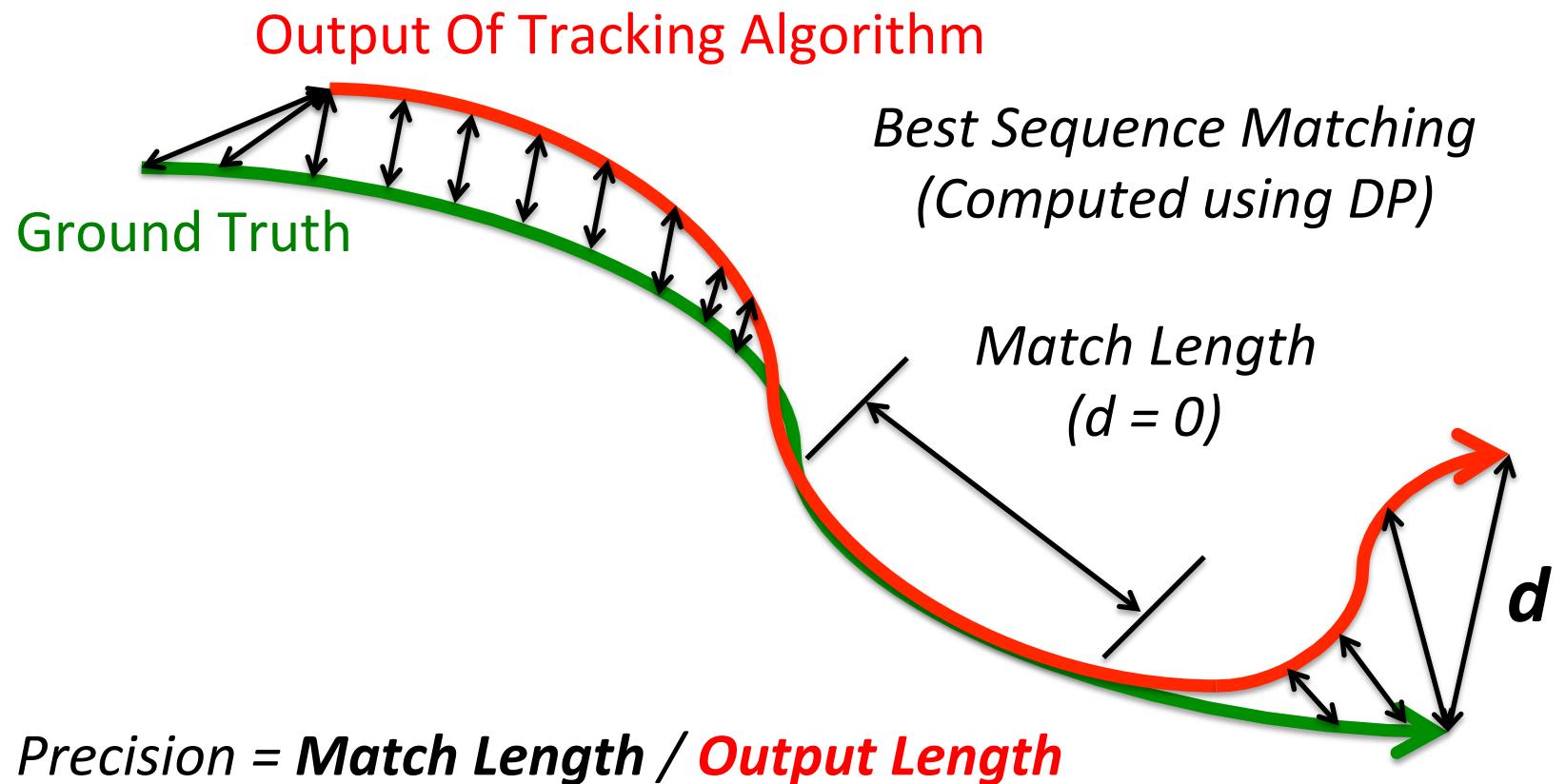
- Placelab + VTrack
 - Look up single best matching GSM fingerprint for each time
 - Match (lat, lon) using VTrack
- GPS k + VTrack
 - Get a GPS sample every k secs
 - Match (lat, lon) using VTrack
 - $k = 4$ min is *energy-equivalent* to CTrack



Evaluation Metrics

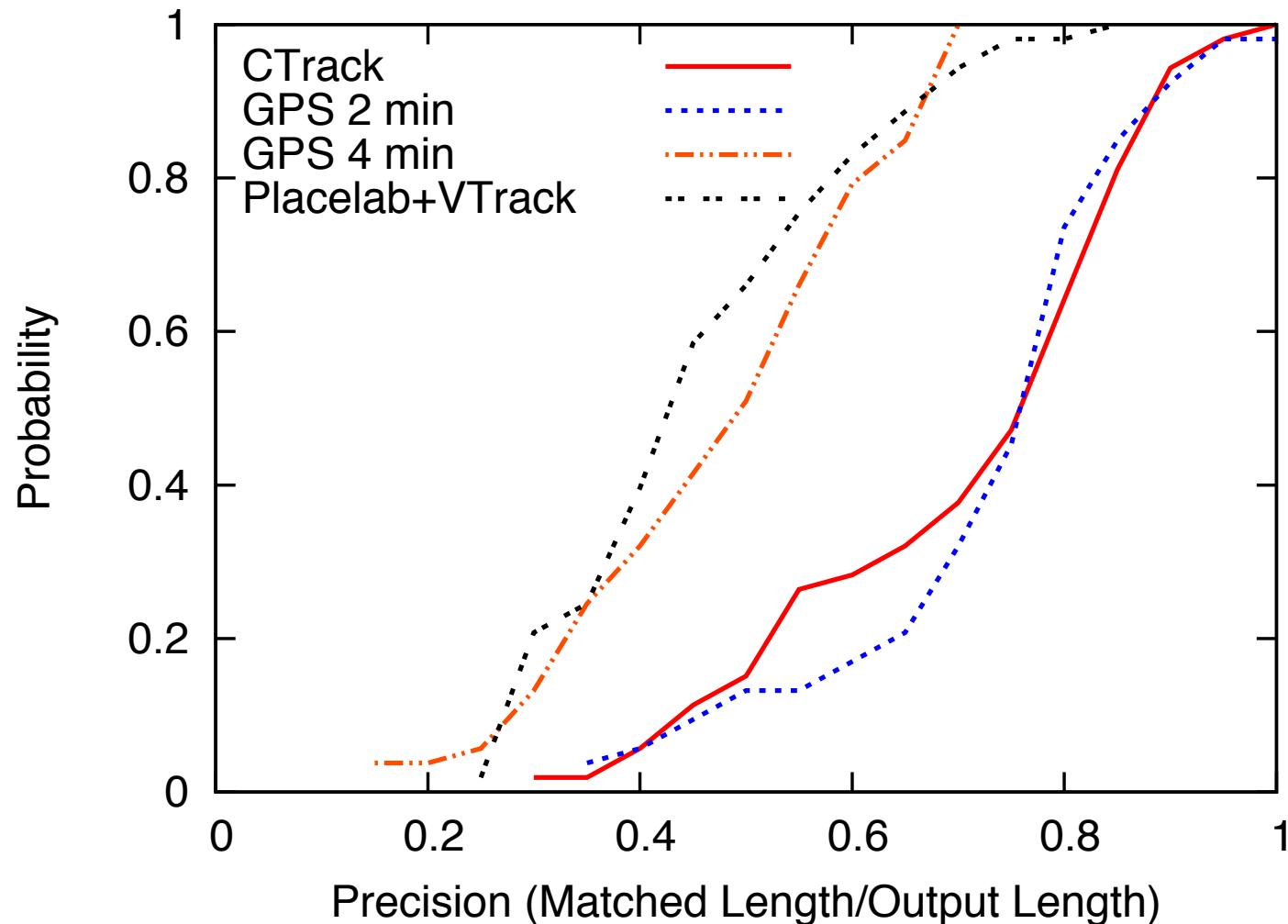


Evaluation Metrics

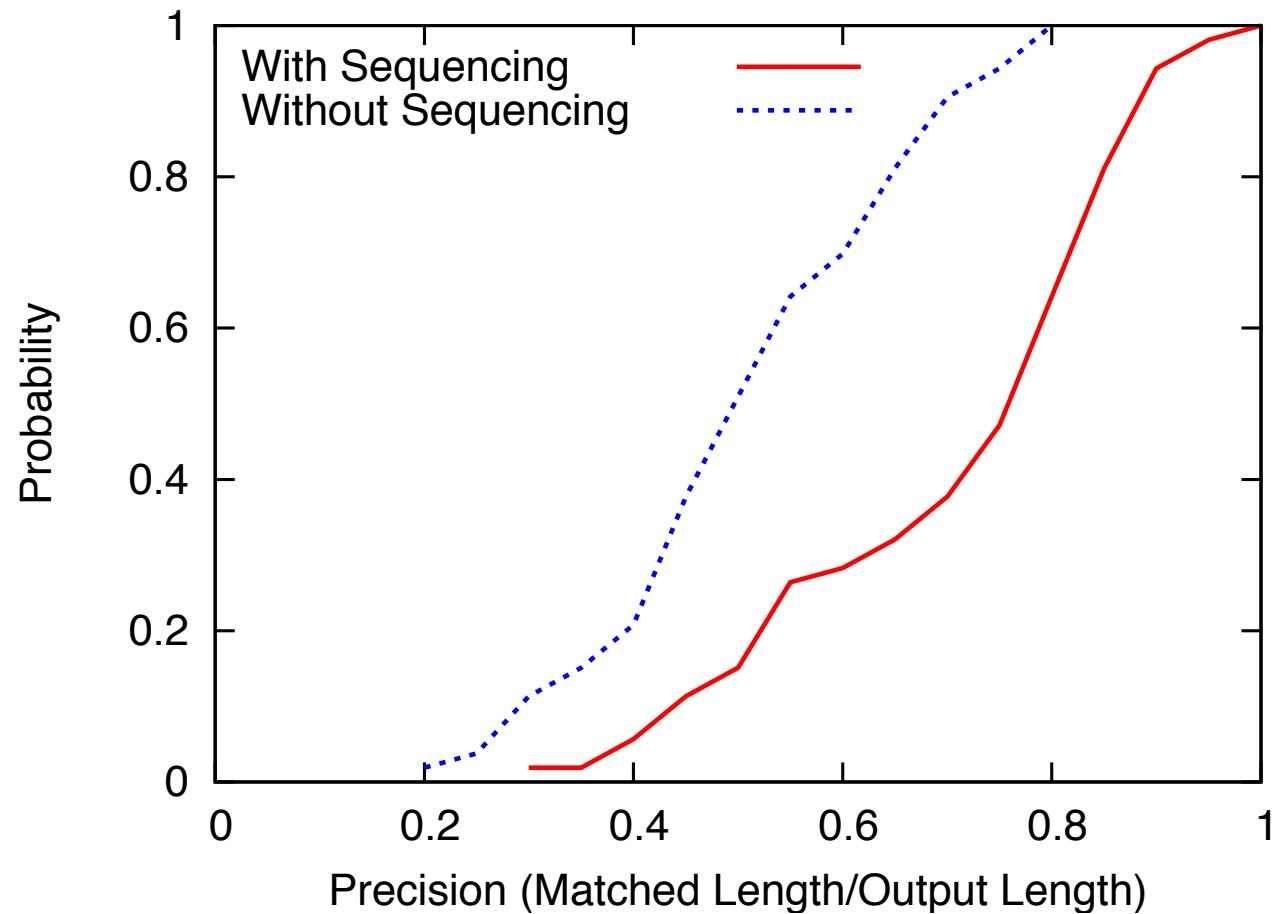


$Recall = \text{Match Length} / \text{Ground Truth Length}$

CTrack Has 75% Precision: 3x Less Error Than Placelab+VTrack, GPS k



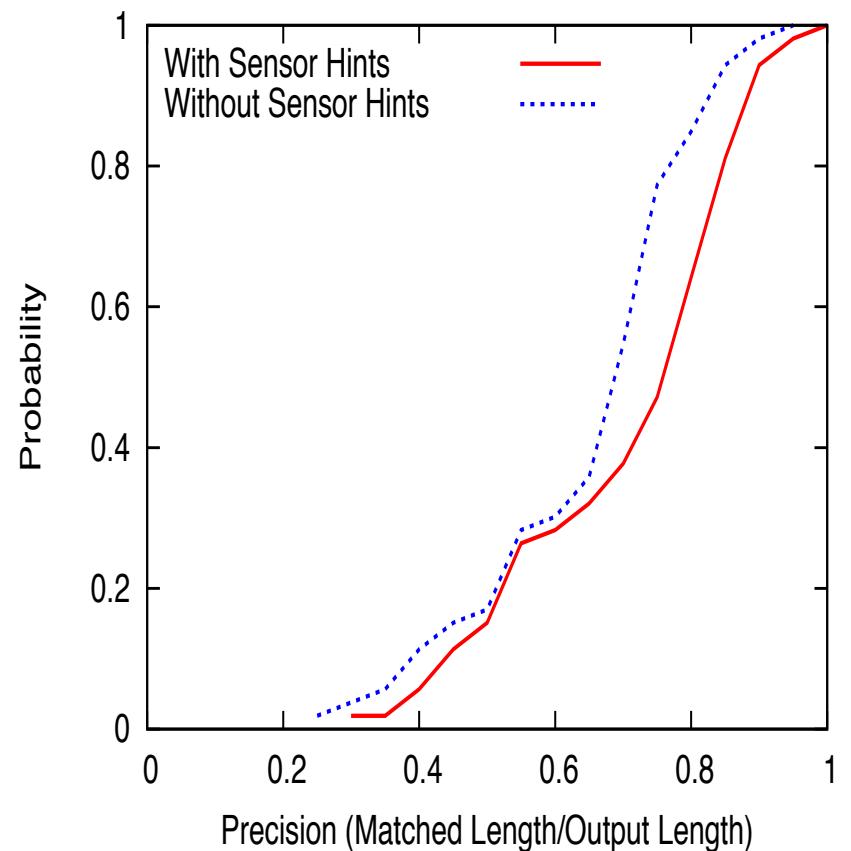
Grid Sequencing Step is Critical



*Sequencing First Before Converting to (Lat, Lon)
Coordinates is Critical*

Impact Of Sensor Hints

- Small in quantitative terms
 - 6% precision, 3% recall
- But help correct some systematic errors
 - Turn hints fix “kinks” in output track
 - Movement hints fix “looping” in GSM signature



*Makes sense to take advantage of hints when available
(i.e. on a smartphone) – they are free in terms of energy!*

Conclusions

- CTrack is a *cellular-only system* that:
 - Can recover over 75% of a user's track
 - Significantly (over 3x) better in energy/accuracy tradeoff than existing approaches
- Broader impact + future work:
 - Make large scale deployment of location-based apps feasible without running into energy barriers
 - Enable devices without GPS (85% of the market) to contribute to and benefit from location-based services

