Extracting a mobility model from real user traces

Minkyong Kim @ Dartmouth
David Kotz @ Dartmouth
Songkuk Kim @ Xerox

INFOCOM
April 26, 2006

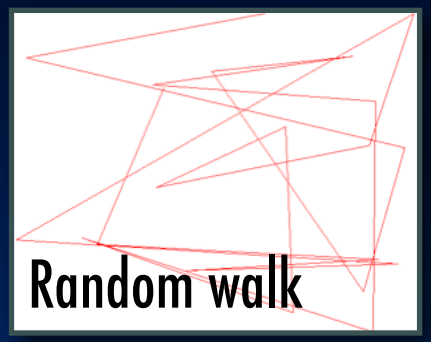
Simulating user mobility

- Wireless-network usage is increasing
  - Mobile systems or applications need be aware of people's mobility
  - Not feasible to test in real environment
  - Thus, resort to simulations

- To simulate people's movements
  - Trace-driven: limited parameter space
  - Model-based: no realistic models

Need a new mobility model

- Current models aren't realistic
  - Variation of random-walk models
  - Based on intuition of designer



- Goal: Develop mobility model using real traces
- Mobility traces
  - Physical mobility traces aren't available
  - Use network mobility traces: syslog
  - Why syslog? Easy to collect, readily available

Syslog traces

- Dartmouth has campus-wide wireless network
  - Around 560 access points, on 1km^2 main campus
- Access points (APs) collect syslog traces
  - Record client events (associate, authenticate,...)
  - Each entry: time stamp, AP, client, event type
- Two types of models can be developed
  - Model of AP-association patterns
  - Model of **physical mobility** ← ***Our goal***

Focus on always-on devices

- On-and-off devices
  - Laptops
- Always-on devices
  - Vocera communicators
  - Cisco VoIP phones

Vocera

Cisco

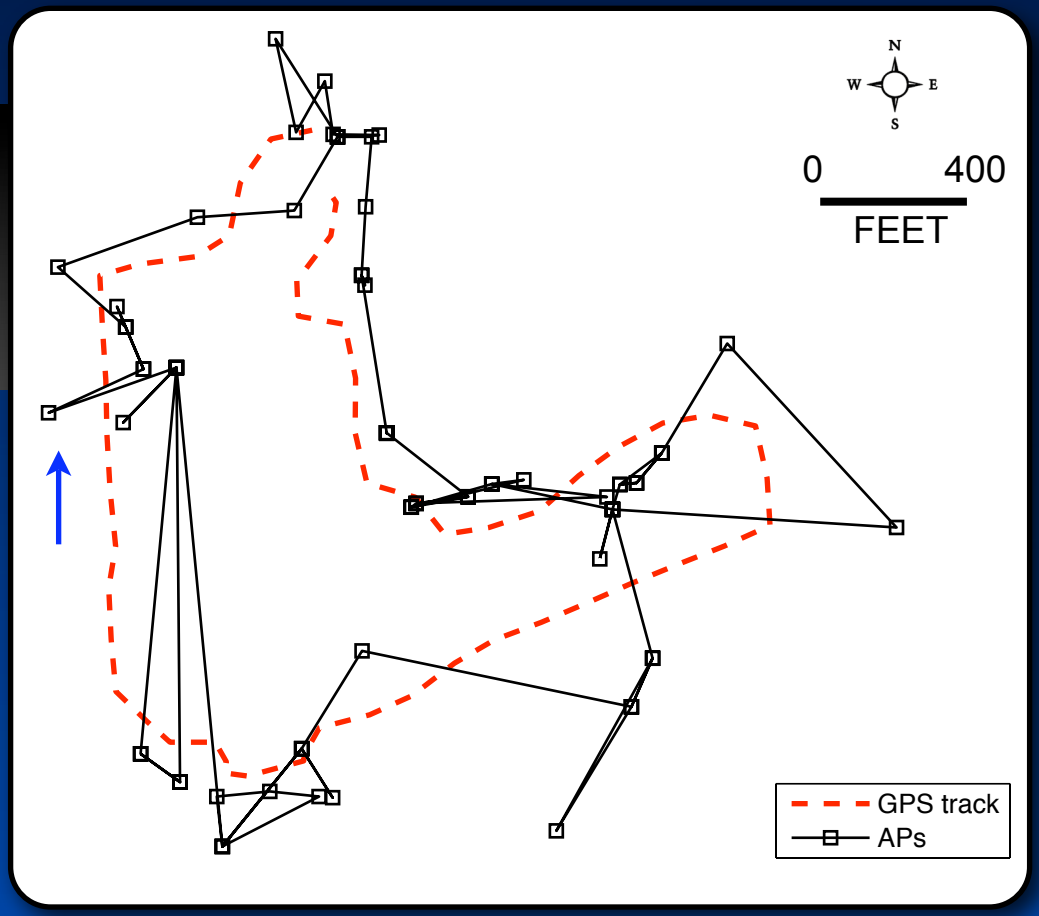
- Usage patterns are different
  - Not enough path information for laptops
  - Focus on **always-on devices**

Estimating physical location

- *Problem*: Syslogs don't contain users' physical location, only sequence of AP locations
- *Challenge*: How to estimate physical location?

Sample walk

User walked for 20 min, carrying GPS and Vocera

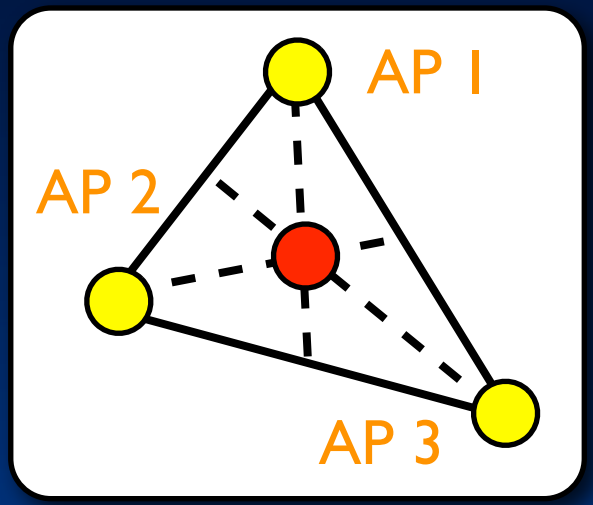


Estimating physical location

- Estimate physical location using filters
  - Centroid filters

    Triangle centroid
    use three associations

    Time-based centroid
    use associations within 60s



  - Kalman filter: estimate position given knowledge on system's behavior and measurements with noise
- Kalman filter performed the best

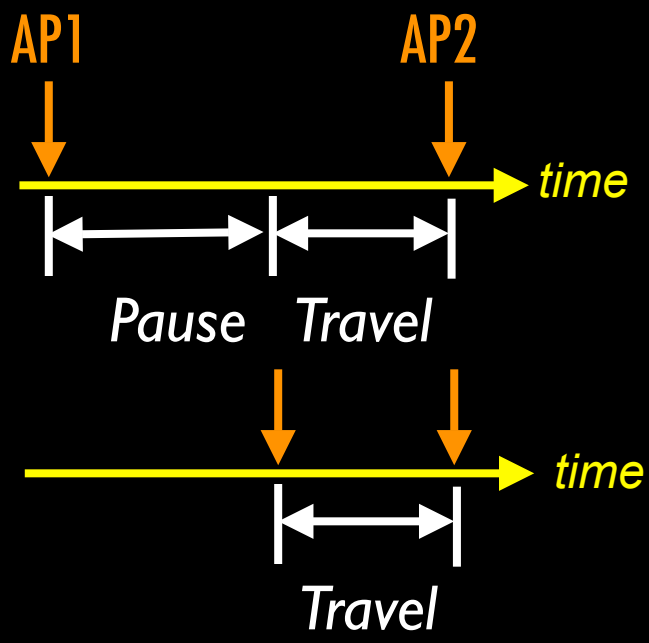
Extracting pause time

- *Problem*: Syslogs have only association time stamps
- *Challenge*: Separate time into travel and pause
- For given distance, expected travel time is known
  - If elapsed time is longer than expected, user probably paused and then moved

Let s=distance/elapsed

If s < 0.5m/s
user paused and then moved

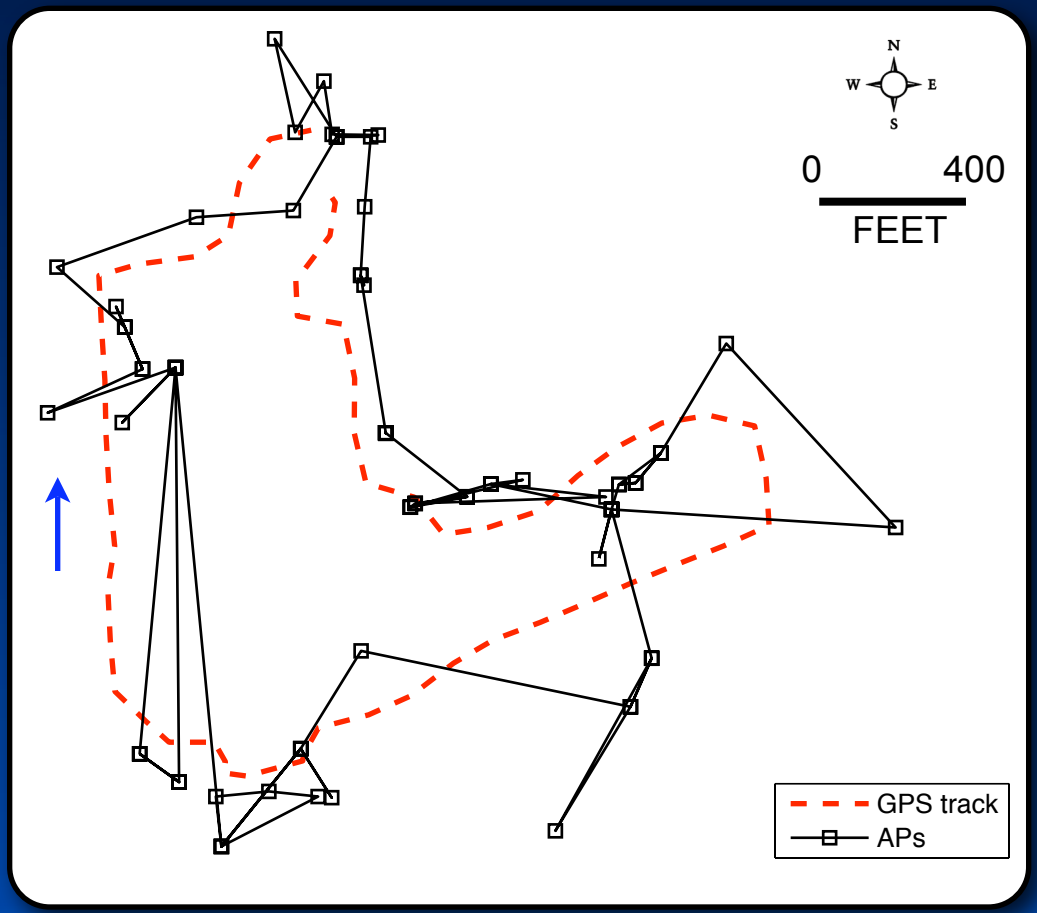
If s is in normal range [0.5-10m/s]
user didn't pause



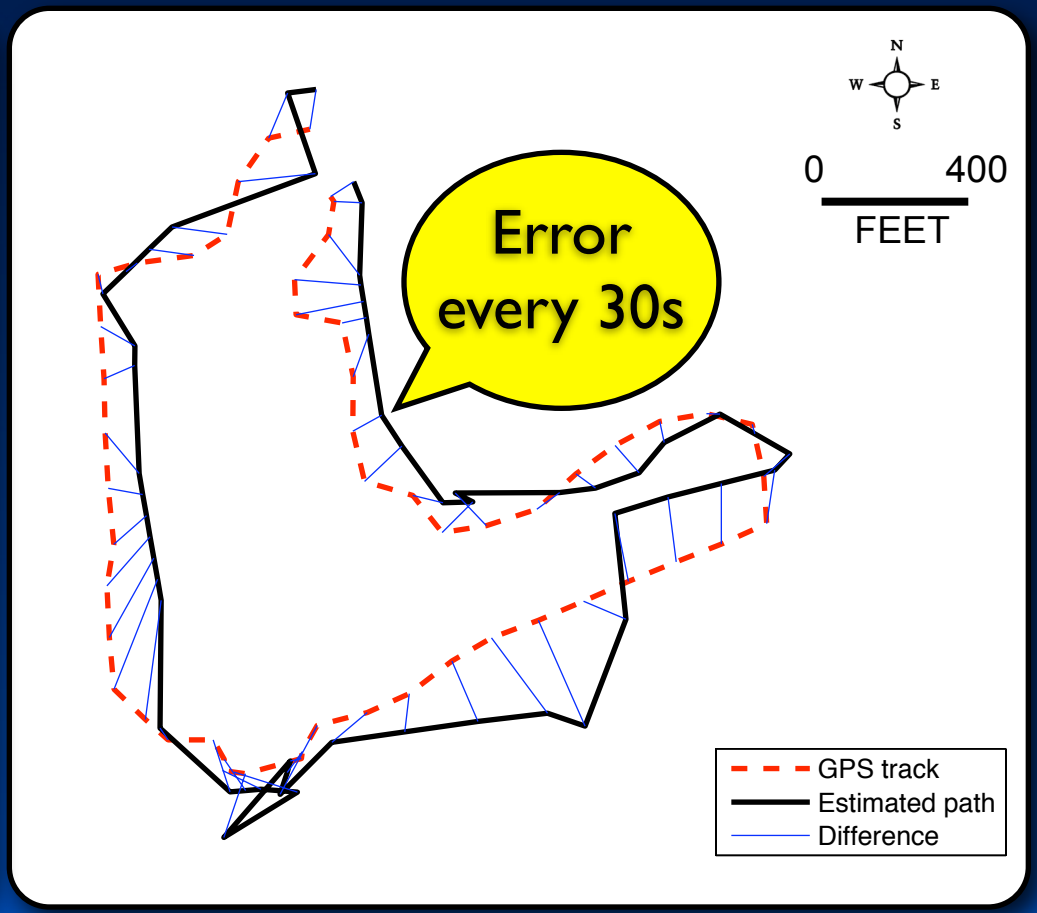
Evaluation method

- Four people walked, carrying GPS, Vocera, and Cisco phone
  - Each walk lasted 30 min with a 10 min pause
  - Four Vocera traces, four Cisco traces

Raw AP associations



Filtered user path



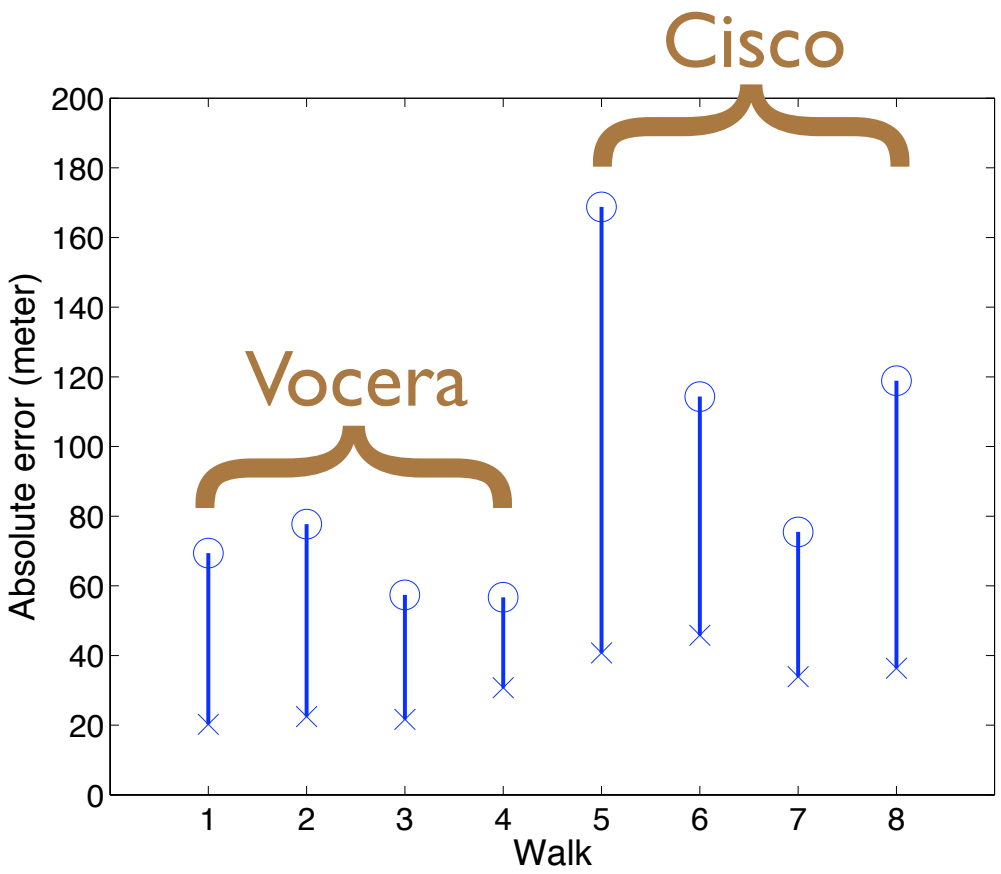
Evaluation result

Path extractor (Kalman filter)

Error computed every 30s per walk

Ground truth: GPS

Median error: 20m - 46m

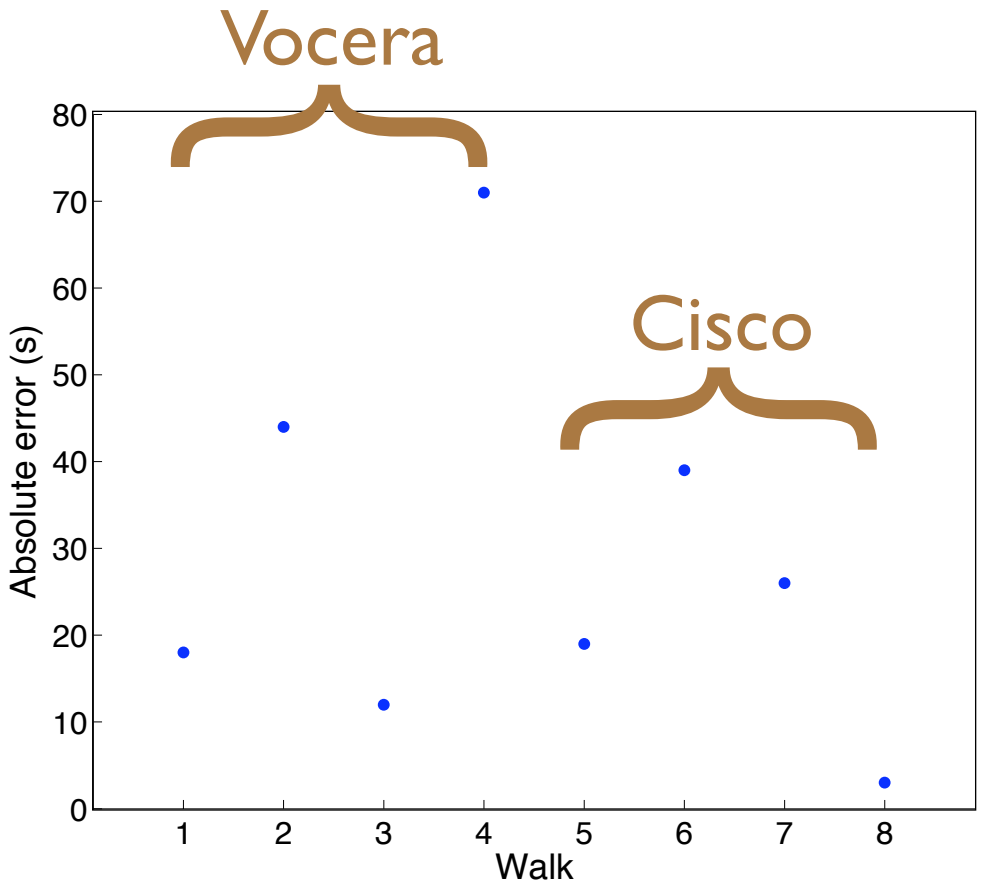


Pause extractor

Error computed for a 600s pause

Ground truth: user recorded

Error: 3s - 71s



Network traces

- June 2003 - June 2004
- 198 always-on devices (existing users)
- To remove diurnal effects, considered 8am-6pm
- Divide workday traces based on
  - *Diameter*: maximum distance between any two APs visited by user during workday

Set	Diameter	Workdays
Mobile	>= 100 m	3,252 (46%)
Stationary	< 100 m	3,876

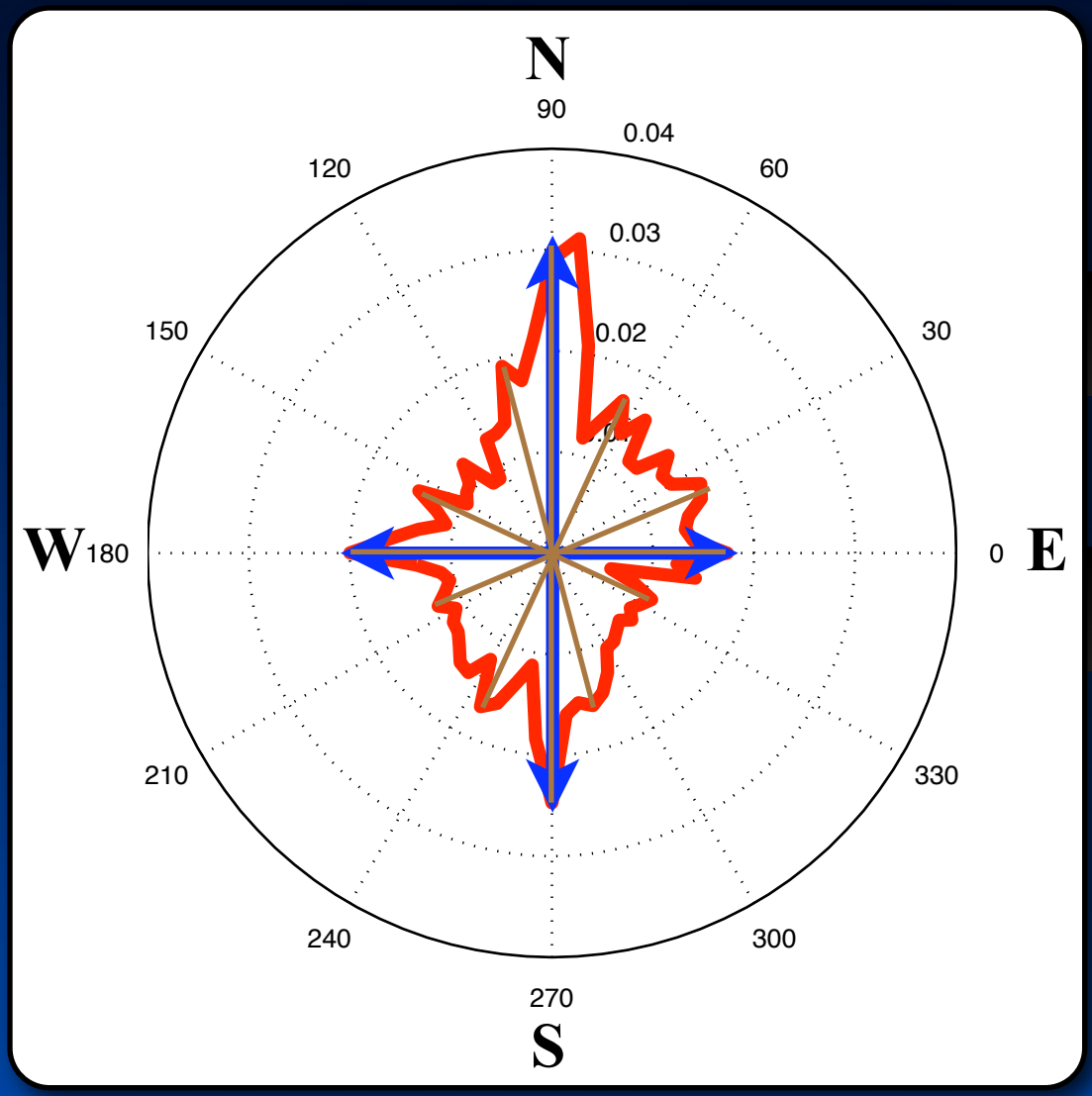
Temporal characteristics

Characteristic	Distribution	Mean	Median
Pause	log-normal	1,677s	163s
Speed	log-normal	1.65m/s	1.26m/s
Start time *08:00-*	exponential	09:54	09:06
End time *-18:00*	exponential	16:30	17:06

✳ *Human walking speed: 3mile/h = 1.34m/s*

Movement direction

- Histogram with 5° bin weighted by duration of movement



NS, EW are popular

Symmetry across 180°

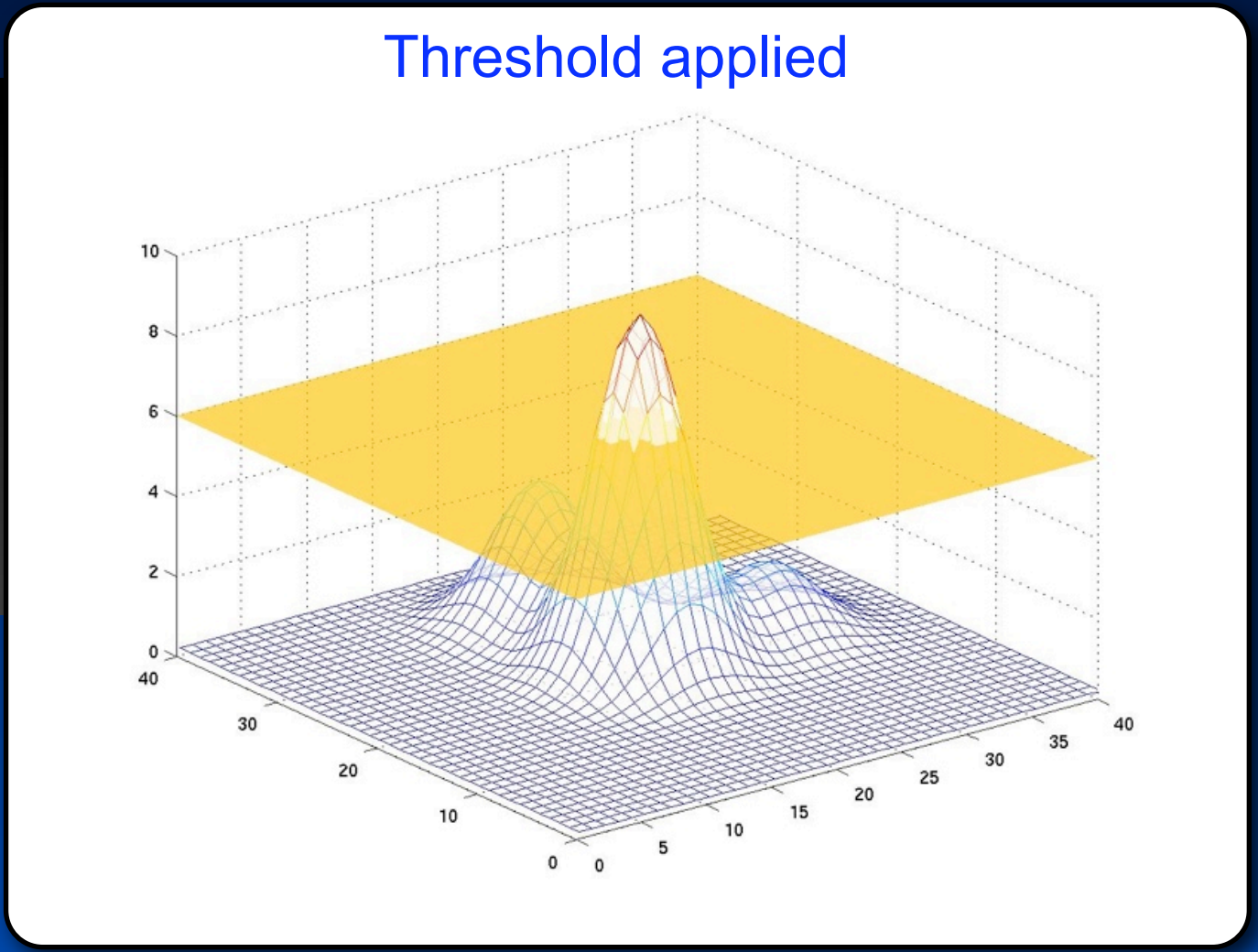
Hotspots on campus map

- *Problem*: Estimated user location, with error
- *Challenge*: How to define popular regions?

1. Align the center of unit Gaussian with visit location

2. At each location, sum up Gaussian distributions

3. Regions above threshold considered as hotspots

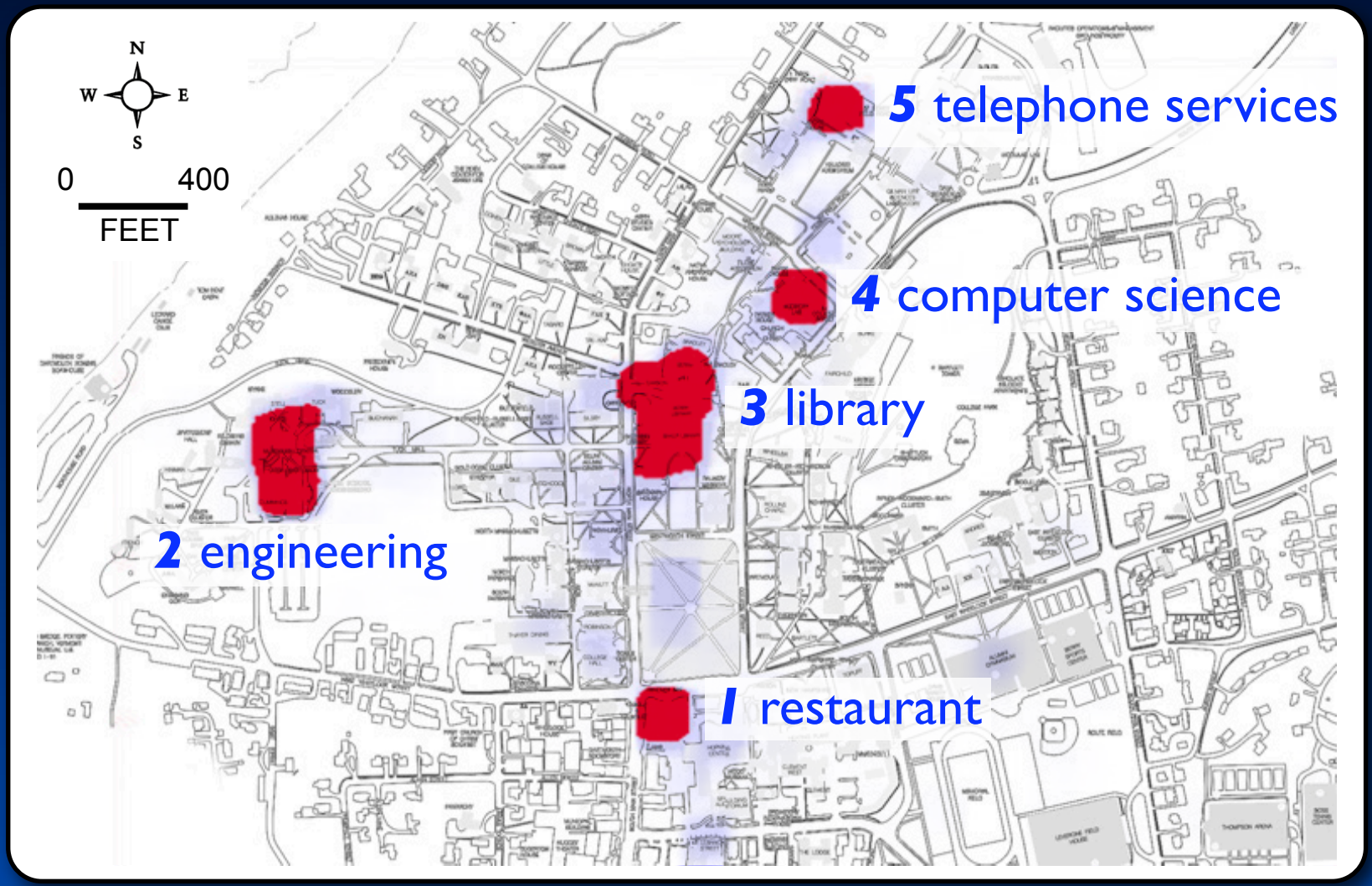


Gaussian applied

Before cut

Gaussian applied

After cut

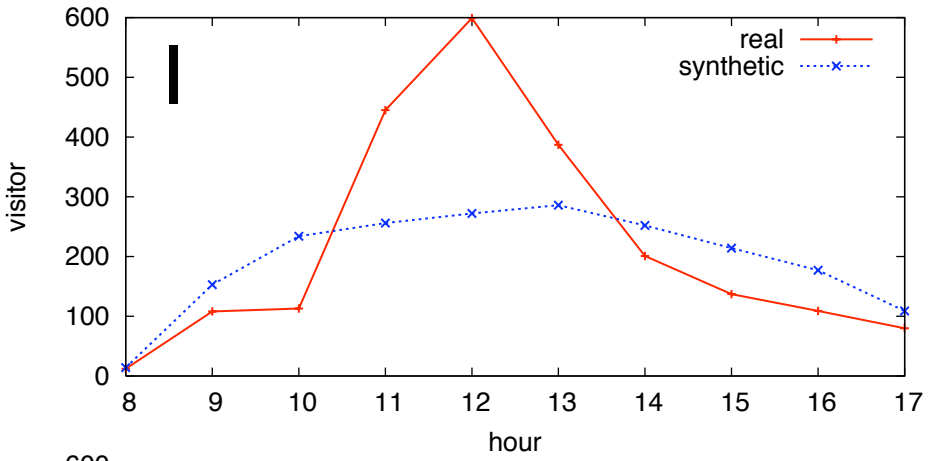


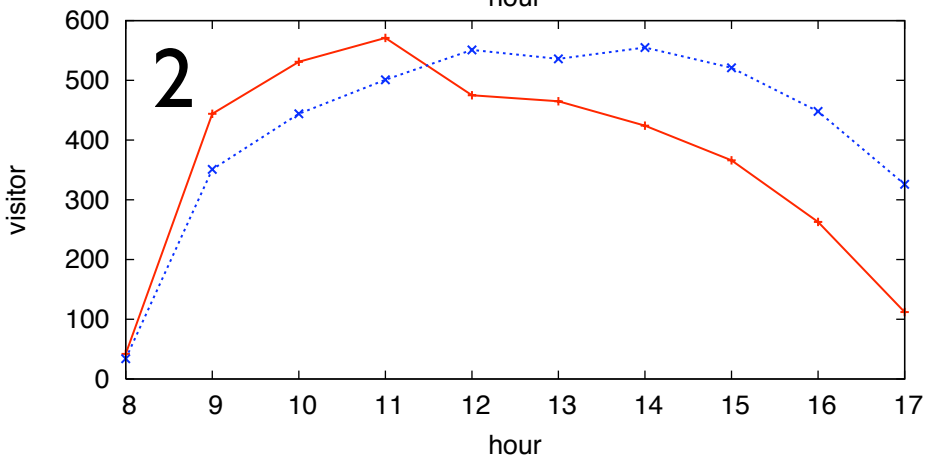
Mobility model

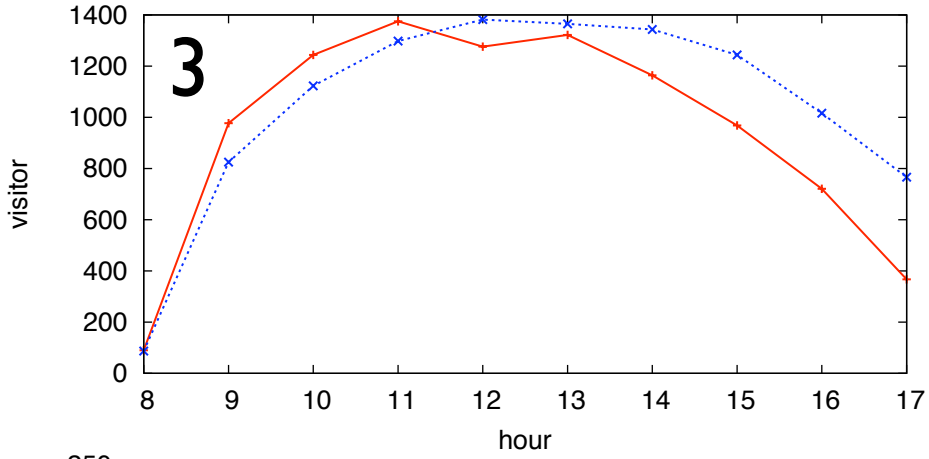
- Model describes how users move between regions
  - 5 hotspots, 1 coldspot, 1 offstate
- For each user
  - Insert a new user using **start time** distribution
  - Choose start location using **initial region** distribution
- For each movement
  - Choose destination region using **transition probability** matrix
  - Choose **speed** and **pause** from distribution

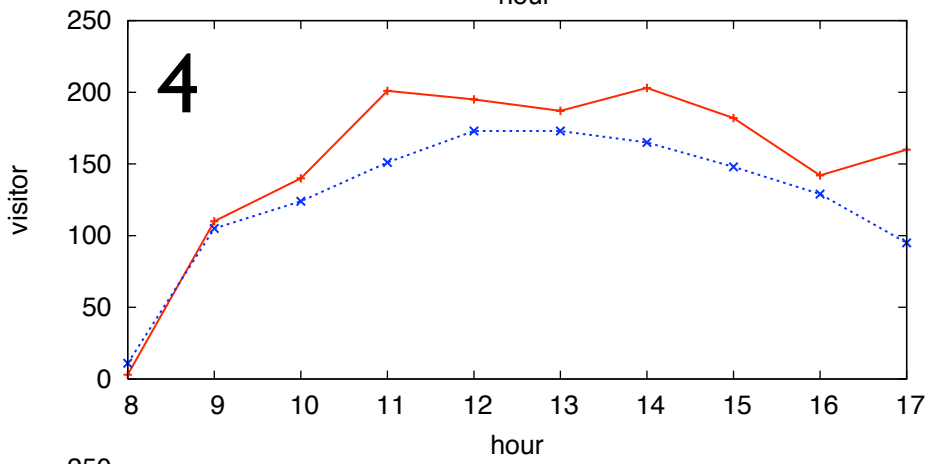
Model validation

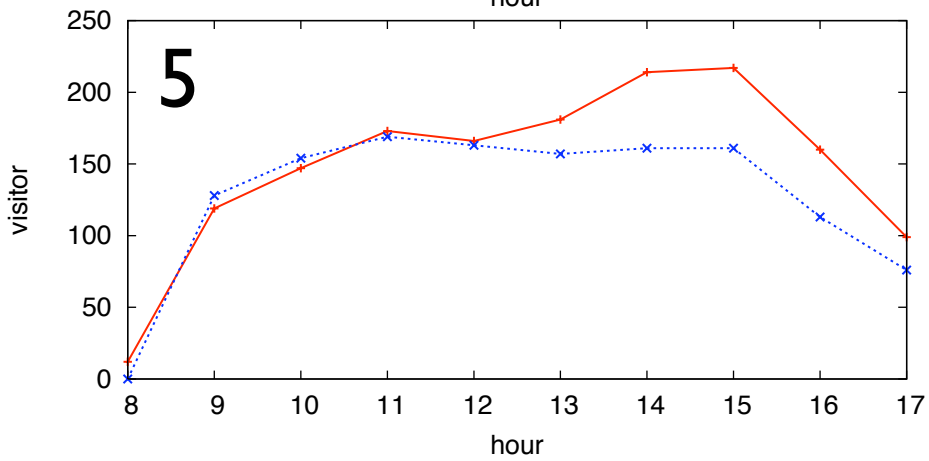
Hourly visitors











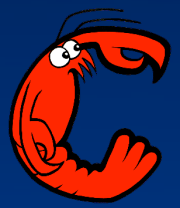
Conclusion

- In this work, we found that...
  - We can effectively extracted physical paths from syslog traces using Kalman filter
  - Commonly used mobility assumptions are incorrect
  - Our model generates realistic movements
- In the future, we plan to work on...
  - Time variation over a course of day
  - Metric for mobility models

Thank you

For related papers and more info

http://www.cs.dartmouth.edu/~minkyong/



For traces used in this paper
http://crawdad.cs.dartmouth.edu/

Center for Mobile Computing

http://cmc.cs.dartmouth.edu/