Supporting Energy-Efficient Uploading Strategies for Continuous Sensing Applications on Mobile Phones

M. Musolesi, M. Piraccini, K. Fodor, A. Corradio, and A. Campbell
University of St Andrews, University of Bologna, Ericsson Research, Dartmouth College
Pervasive 2010

Optimizing Sensor Data Acquisition for Energy-Efficient Smartphone-based Continuous Event Processing

A. Misra and L. Lim
Singapore Management University, University of Hawaii
MDM 2011

Energy-Efficient Collaborative Sensing with Mobile Phones

X. Sheng, J. Tang, and W. Zhang
Syracuse University, NY
INFOCOM 2012

Supporting Energy-Efficient Uploading Strategies for Continuous Sensing Applications on Mobile Phones

M. Musolesi, M. Piraccini, K. Fodor, A. Corradio, and A. Campbell
University of St Andrews, University of Bologna, Ericsson Research, Dartmouth College
Pervasive 2010

Optimizing Sensor Data Acquisition for Energy-Efficient Smartphone-based Continuous Event Processing

A. Misra and L. Lim
Singapore Management University, University of Hawaii
MDM 2011

Energy-Efficient Collaborative Sensing with Mobile Phones

X. Sheng, J. Tang, and W. Zhang
Syracuse University, NY
INFOCOM 2012

Continuous Sensing Applications

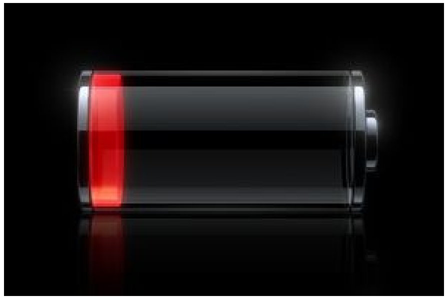
- Sensor-enabled mobile phones
- Continuously make inferences about people and environment
- Transmit data in real-time to a server

User activities inferred by CenceMe

Avatar refreshed in a consistent way

Communication Cost

- Continuous sensing applications have significant communication costs
- Battery life time lasts only for a few hours
- Financial cost for data transmission



Intelligent Data Uploading

- Trade-off between information availability and accuracy
- Guarantee satisfactory user experience
- Scenarios:
  - Connectivity always available
  - Connectivity intermittently available
  - GPS information available

Dataset Description

- Collected during the deployment of the CenceMe application
- 20 Nokia N95 phones
- High-level activities inferred by the CenceMe classifier, GPS location coordinates
- Data from two weeks

Used as ground-truth for the experiments

Optimizing User State Uploading

- High-level states inferred from processing the raw sensor data
- Set of possible activities S = {Sitting, Standing, Walking, Running}
- Two cases:

  - Online strategies: Connectivity always available

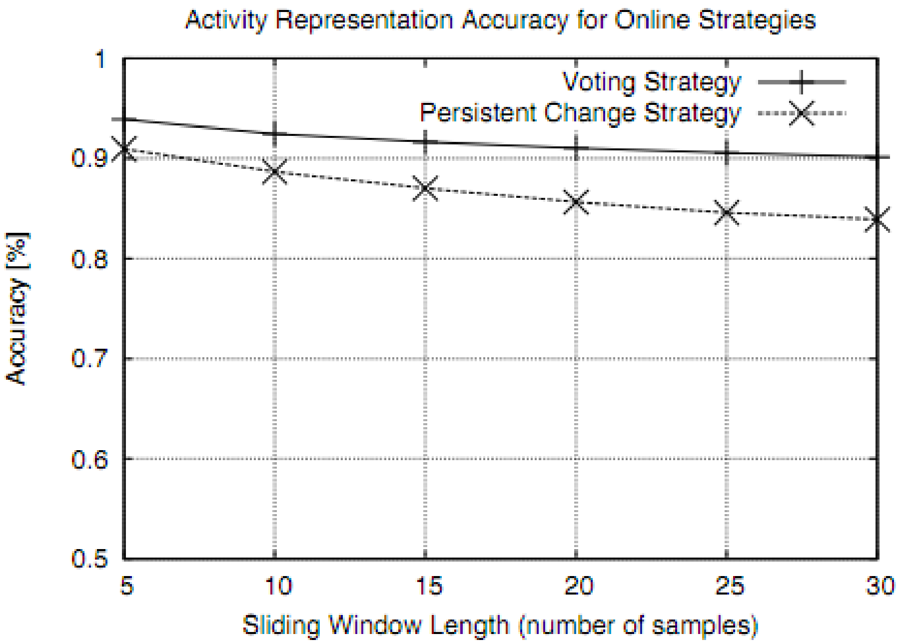
  - Offline strategies: Connectivity intermittently available

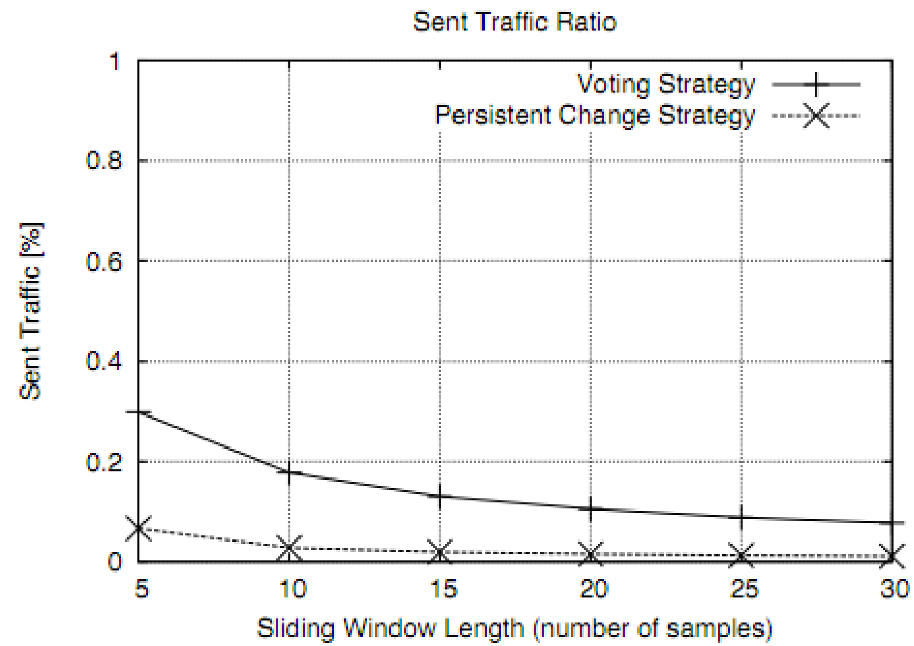
Online Strategies

- Always upload
- Upload in presence of changes
- Upload in presence of persistent changes (change is not isolated)
- Voting based uploading (state with highest frequency is uploaded)

Accuracy and transmission overhead of all techniques with respect to “upload in presence of changes”

Accuracy and transmission overhead





90% accuracy achieved, 80% data traffic saved

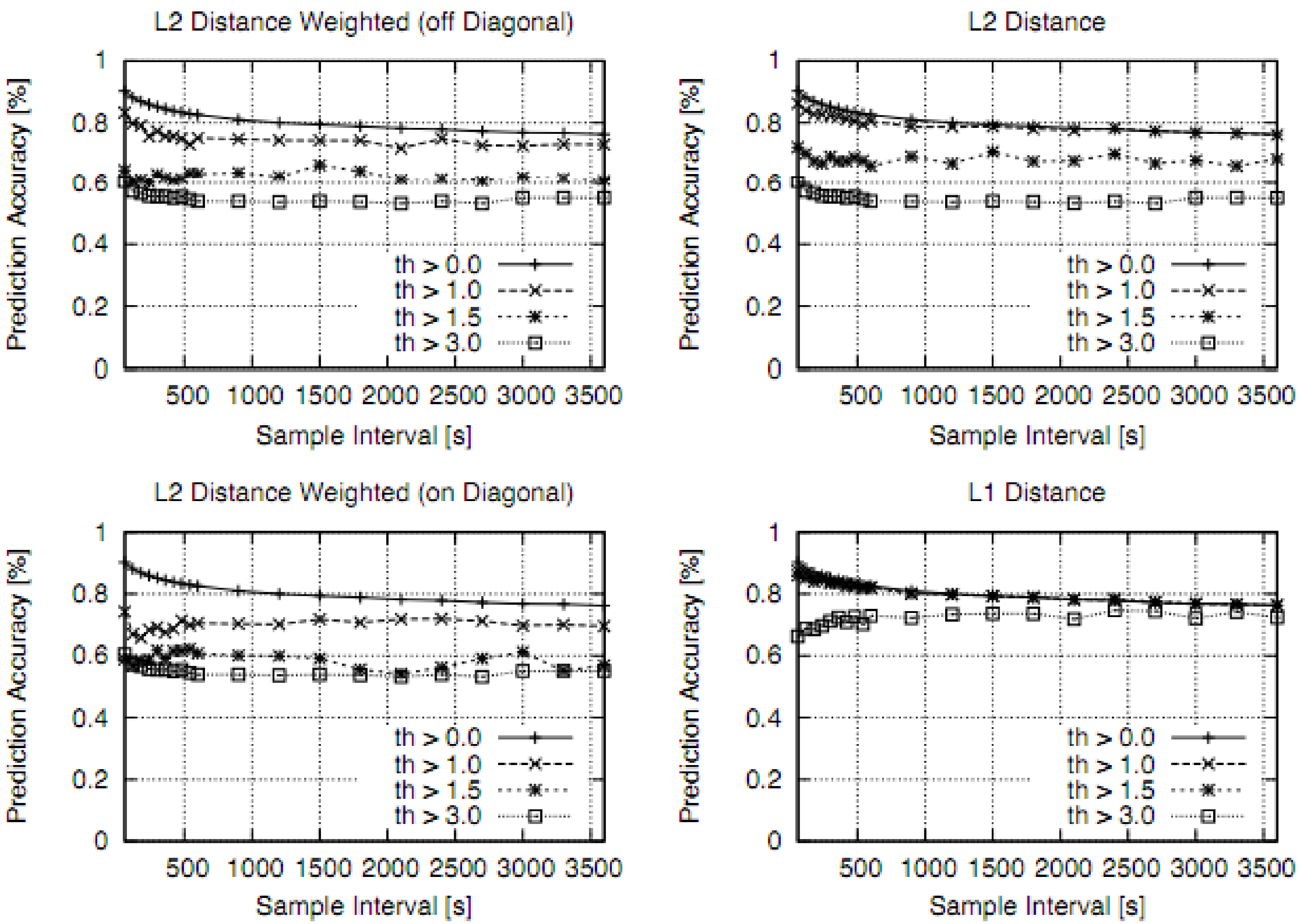
Offline Strategies

- Forecast next state during a disconnection
- Markov chain based prediction

M_{phone}	M_{server}	M_{server}
Run-time matrix	Server-side matrix	Server-side matrix
0.7 0.2 0.2 0.1	0.6 0.1 0.2 0.1	0.6 0.1 0.2 0.1
0.2 0.7 0.1 0.0	0.2 0.7 0.1 0.0	0.2 0.7 0.1 0.0
0.1 0.2 0.5 0.2	0.0 0.3 0.5 0.2	0.0 0.3 0.5 0.2
0.1 0.1 0.2 0.6	0.1 0.1 0.3 0.5	0.1 0.1 0.3 0.5

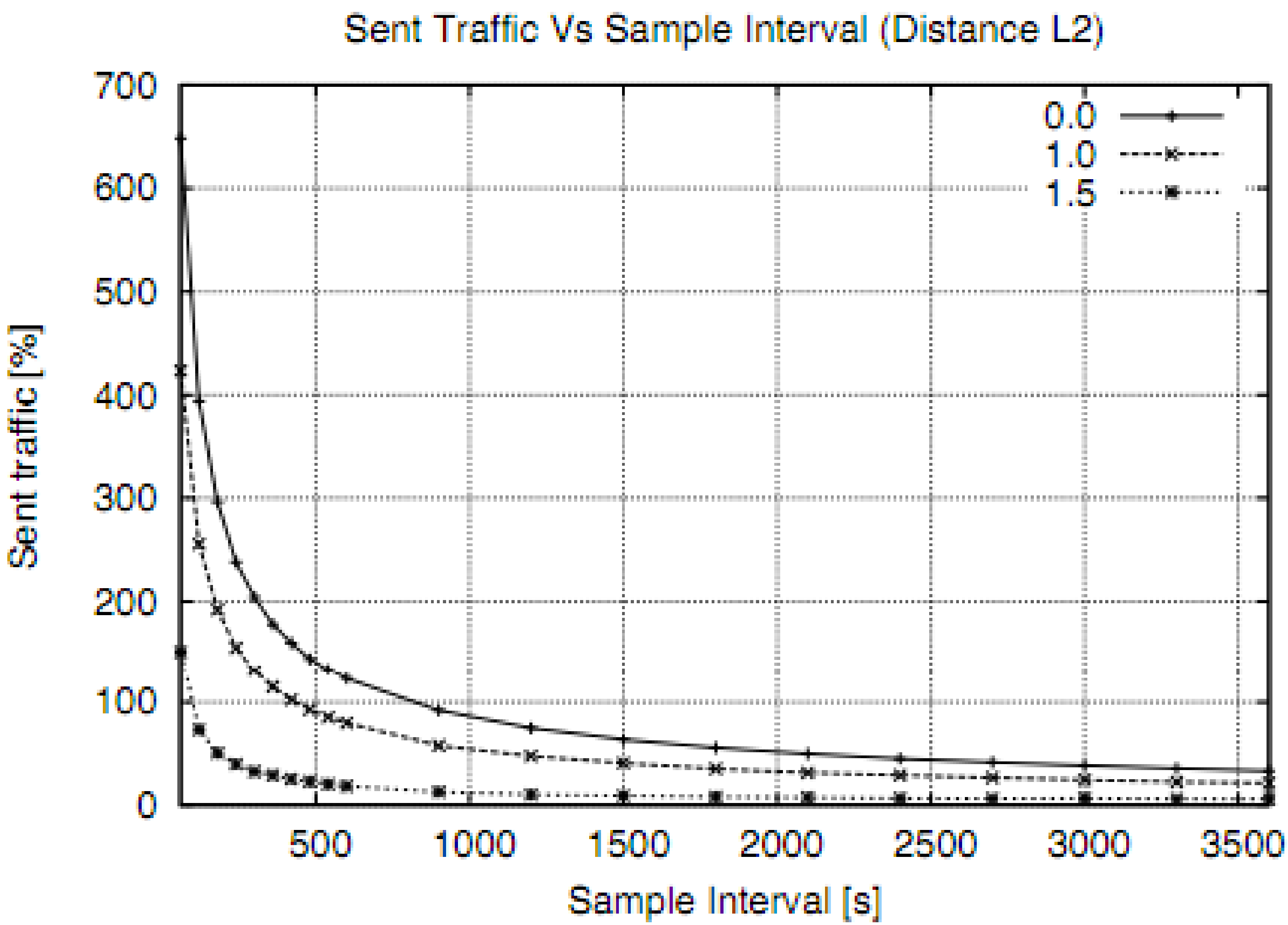
Transition matrix models sequence of state changes

Accuracy of Offline Strategies



The lower the threshold the higher the accuracy

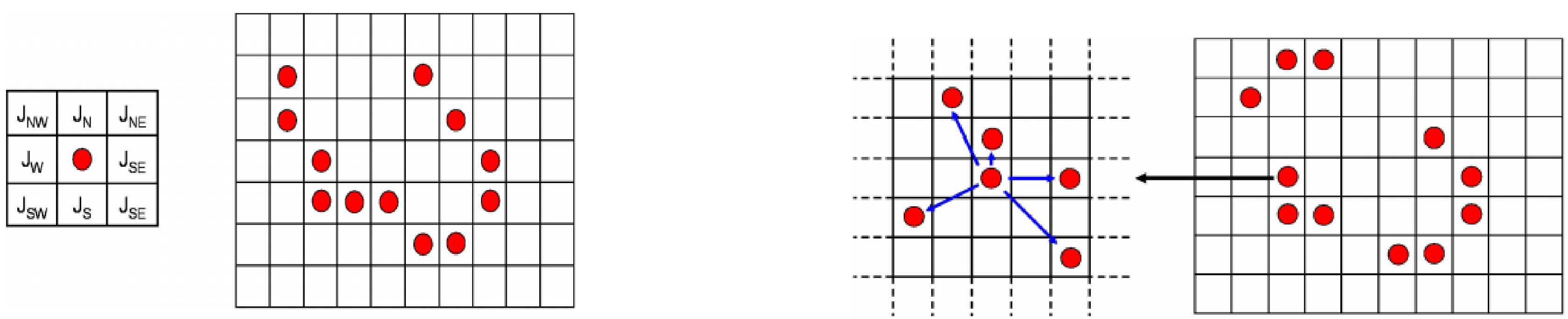
Sent Traffic of Offline Strategies



The sent traffic is up to 7x higher

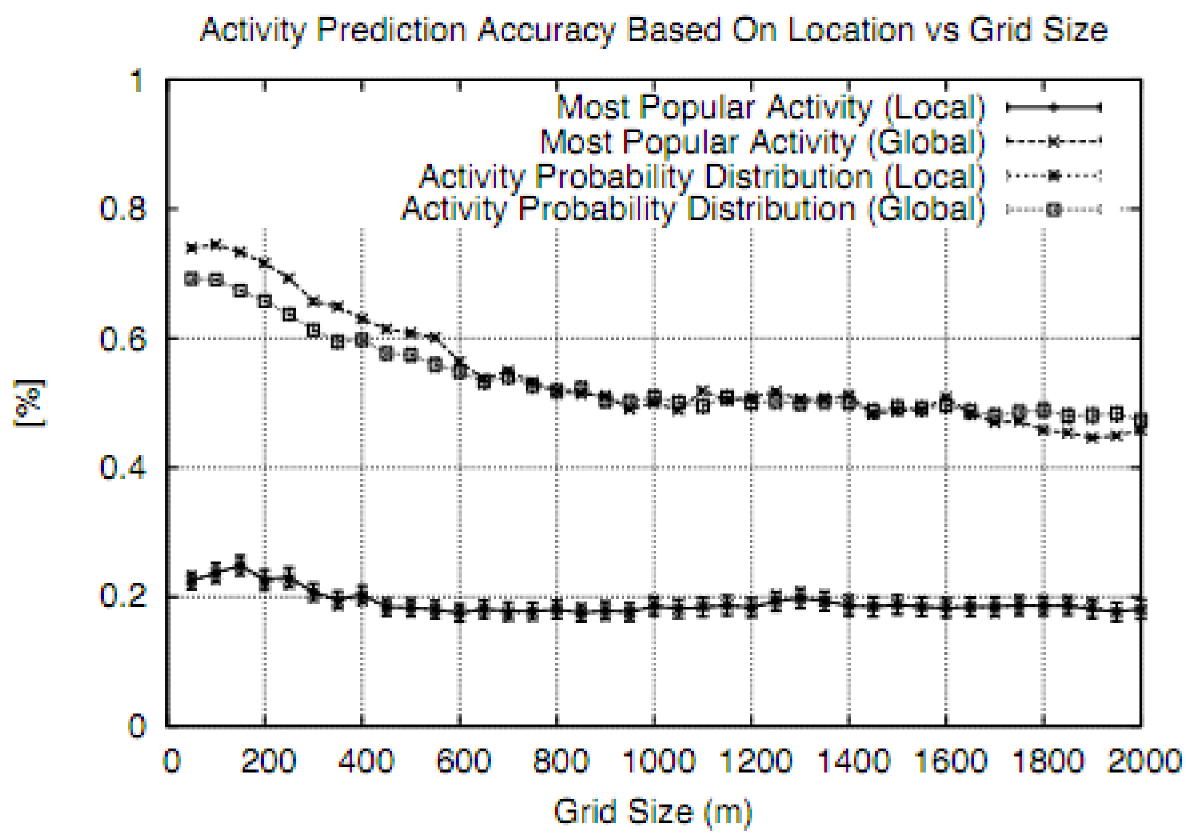
Location-based State Uploading

- Associate state transition matrix to location
- Two-level Markov model: first forecast next location, then predict future activity



Local and global movement models

Activity Accuracy Prediction



Prediction accuracy decreases for all models with increasing grid sizes

Reviews

- Overall rating: 0.6 (borderline)
- Main concerns:
  - Is 80% accuracy acceptable?
  - Upload strategies and location based prediction are not very original
  - Simulate server's predictions on the phone and use this to decide when to send new matrix
  - Although authors claim to present a general solution, it is not clear for what kind of applications this works
  - Evaluation is based on a very specific data set

Supporting Energy-Efficient Uploading Strategies for Continuous Sensing Applications on Mobile Phones

M. Musolesi, M. Piraccini, K. Fodor, A. Corradio, and A. Campbell
University of St Andrews, University of Bologna, Ericsson Research, Dartmouth College
Pervasive 2010

Optimizing Sensor Data Acquisition for Energy-Efficient Smartphone-based Continuous Event Processing

A. Misra and L. Lim
Singapore Management University, University of Hawaii
MDM 2011

Energy-Efficient Collaborative Sensing with Mobile Phones

X. Sheng, J. Tang, and W. Zhang
Syracuse University, NY
INFOCOM 2012

Data Acquisition

- Smartphones have many on-board sensors
  (e.g., GPS, accelerometer, and compass)
- Smartphones aggregate data from variety of external sensors
  (e.g., medical and environmental sensors)

Data Transmission to Smartphone

- Data transmitted over Personal Area Network (PAN), e.g, Bluetooth, IEEE 802.15.4, and WiFi
- Main goal:

  Reduce data that is transmitted over the PAN interface, without compromising the fidelity of the event processing logic

Continuous Stream Processing

- ACQUA: Acquisition Cost-Aware Query Adaption
- Learns the selectivity properties of different sensor streams
- Optimize sequence in which the smartphone acquires sensor data

Example Query

- Two example episodes to detect:

  - Conjunctive query: walking AND above 25°C AND outside

  - Disjunctive query: walking OR above 25°C OR outside

- Sensors:

  - Accelerometer, Temperature, and GPS

- What is the optimal querying sequence?

  - Conjunctive query: start with the sensor which evaluates to FALSE with high probability

  - Disjunctive query: start with the sensor which evaluates to TRUE with high probability

Functional Requirements

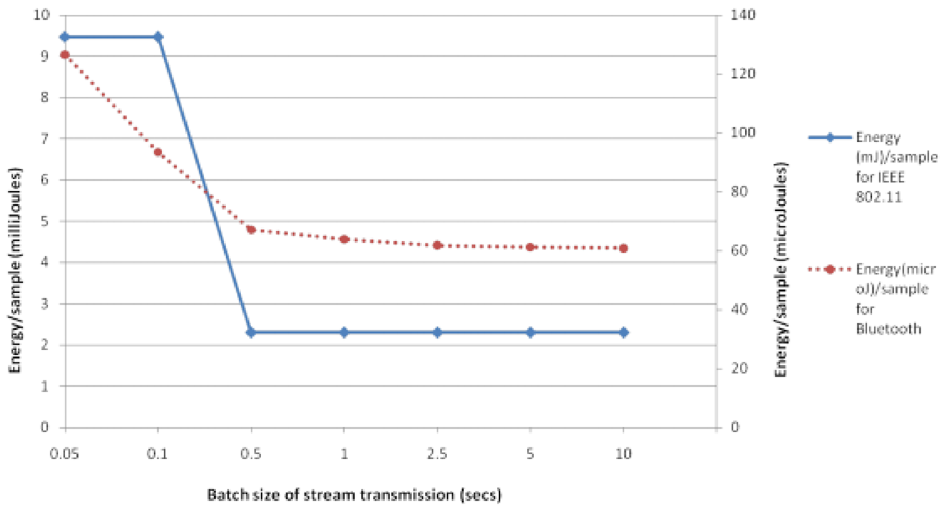
- Heterogeneity in sensor data rates, packet sizes, and radio characteristic
- Adapt to dynamic changes in query selectivity properties
- Take into account other objectives besides energy minimization

Simulation Results

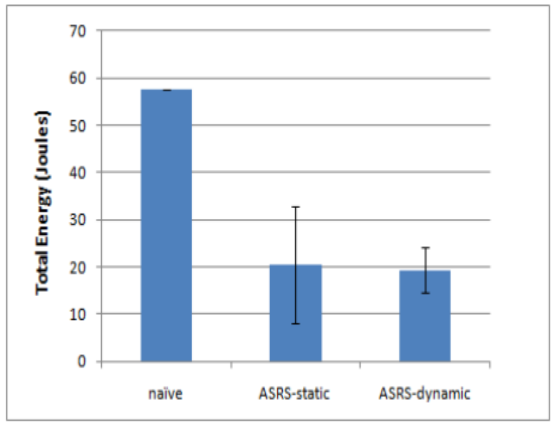
- Evaluate one single query:

  Generate an alert if $((AVG(SP02, 5sec) < 98\%)$ AND
  $((SPREAD(Accel, 10sec) < 2g)$ AND
  $(AVG(HR, 10sec) < 75)))$ OR
  $((AVG(SPO2, 5sec) < 95)$ AND
  $((SPREAD(Accel, 10sec) > 4g)$ AND
  $(AVG(HR, 10sec) > 100)))$.

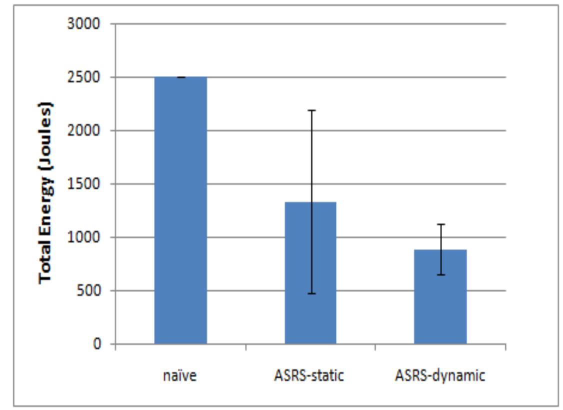
- Two transmission models: WiFi and Bluetooth



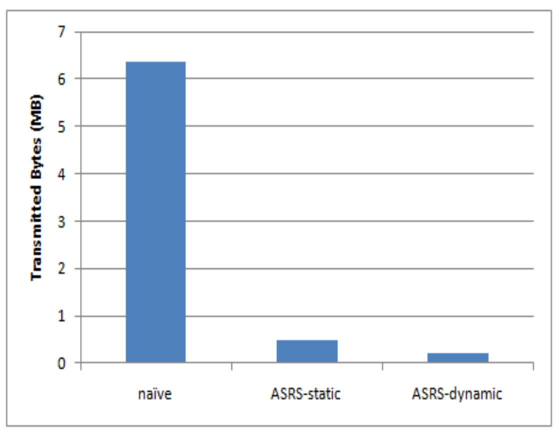
Energy and Data Overhead



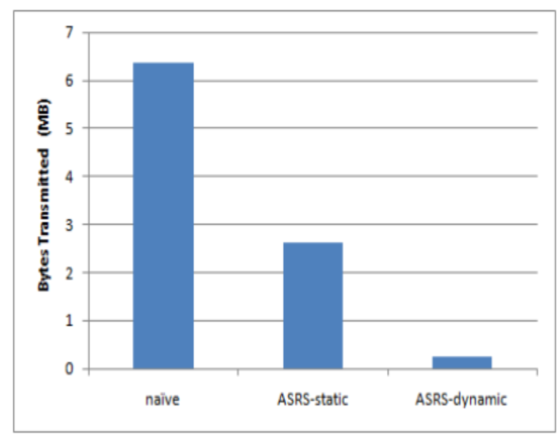
(a) Total Energy (Bluetooth)



(b) Total Energy (WiFi)



(c) Total Bytes of Sensor Data (Bluetooth)



(d) Total Bytes of Sensor Data (WiFi)

~50% and ~70% energy reduction compared to the Naive scheme

Supporting Energy-Efficient Uploading Strategies for Continuous Sensing Applications on Mobile Phones

M. Musolesi, M. Piraccini, K. Fodor, A. Corradio, and A. Campbell
University of St Andrews, University of Bologna, Ericsson Research, Dartmouth College
Pervasive 2010

Optimizing Sensor Data Acquisition for Energy-Efficient Smartphone-based Continuous Event Processing

A. Misra and L. Lim
Singapore Management University, University of Hawaii
MDM 2011

Energy-Efficient Collaborative Sensing with Mobile Phones

X. Sheng, J. Tang, and W. Zhang
Syracuse University, NY
INFOCOM 2012

Collaborative Sensing

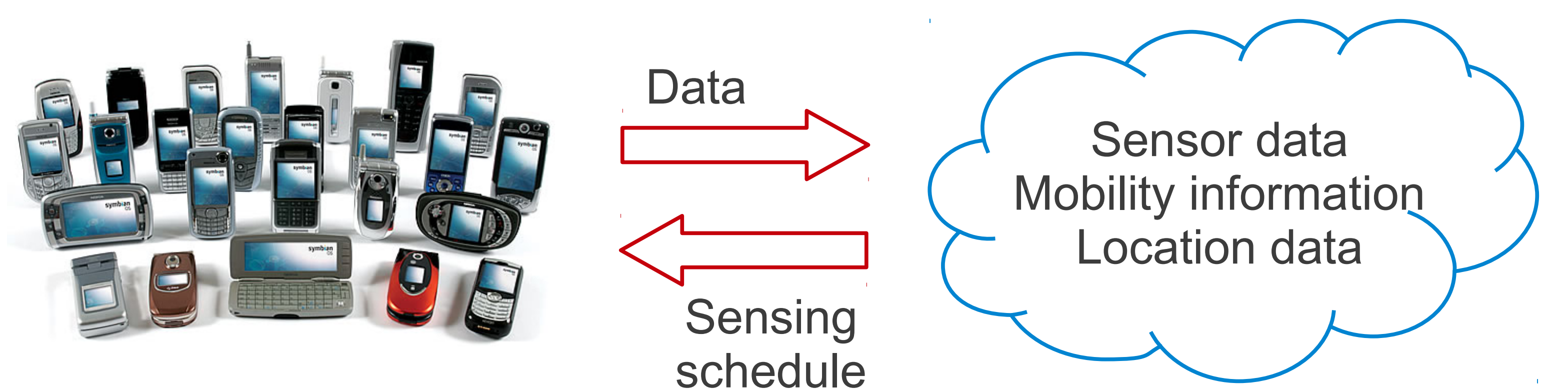
- Participatory Sensing:

  Users actively engage in sensing activity
- Opportunistic Sensing:

  Sensing is fully automated without user involvement

Opportunistic Sensing

- Usually periodic sensing is used
  → not efficient, many redundant data reports
- Control sensing procedure to minimize sensing energy consumption
  → use cloud-assisted collaborative sensing approach



Minimum Energy Collaborative Sensing Schedule (MECSS)

- Input:
  - Region: M roads
  - N mobile users
  - Deadline T
  - Moving trajectory for each user
- Output:
  - Sensing schedule for each user that minimizes total energy consumption and fully covers region

Fair Energy-efficient Collaborative Sensing Schedule (FECSS)

- Input:
  - Region: M roads
  - N mobile users
  - Deadline T
  - Moving trajectory for each user

- Output:
  - *Min-max fair* sensing schedule for each user that minimizes total energy consumption and fully covers region

Collaborative Sensing Algorithms

- Optimal Algorithms (MECSS, FECSS):

  - Moving trajectory from every user required

  These algorithms can be used as benchmarks

- Heuristic Algorithms:

  - Moving trajectory unknown, duty cycled GPS

  - GPS turned on

    - right after initiating sensing task

    - every time user enters new road segment

Simulations

- WiFi signal sensing with three Android phones

TABLE II
ENERGY CONSUMPTION OF A WIFI SCAN

Phone Models	Energy Consumption(μAh)
Google Nexus S	30.99
Samsung S5830	16.25
Samsung I9000	54.08

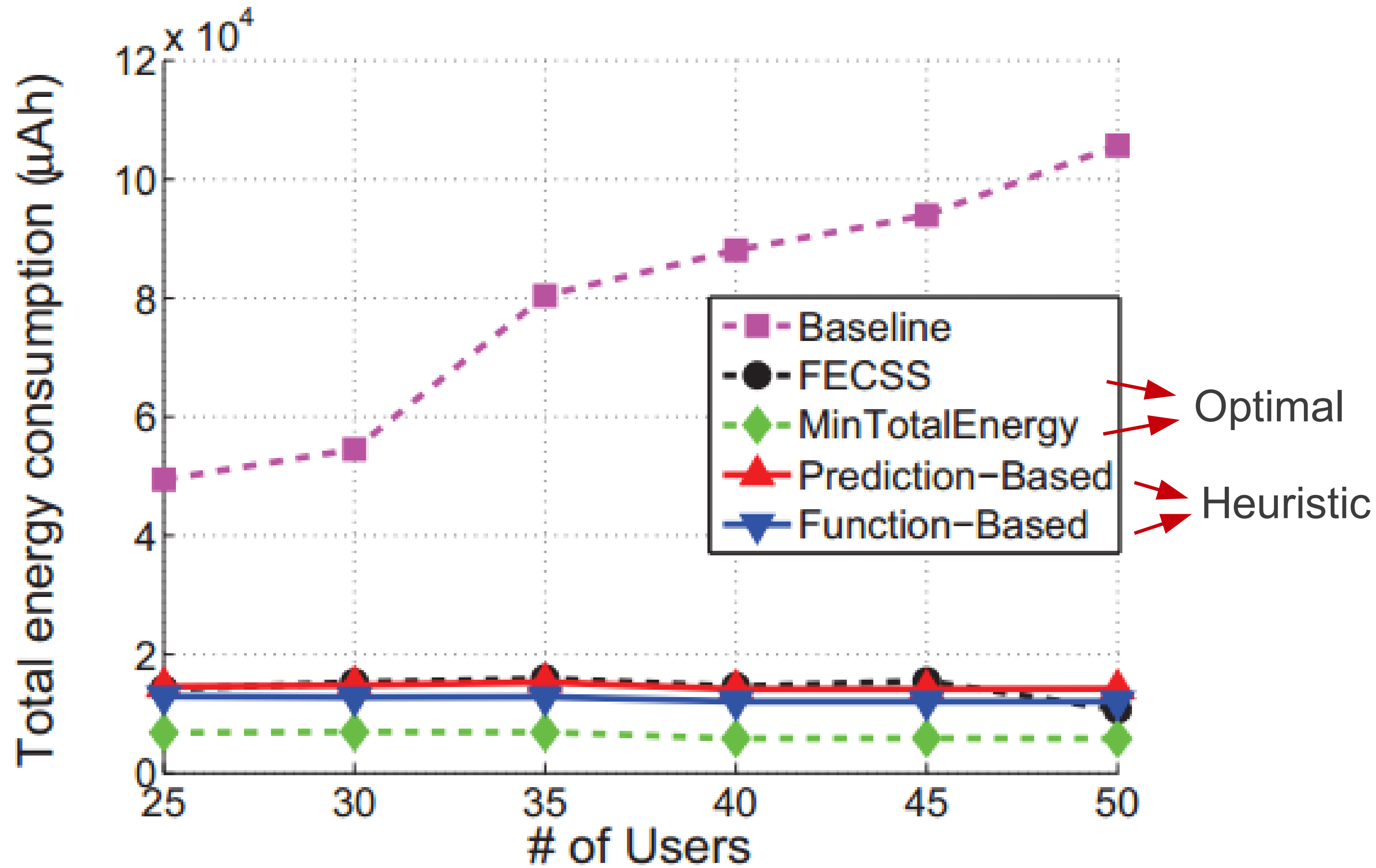
- Target region: 4 blocks in Manhatten, NY
- Mobile users moving trajectory generated with the Manhatten model
- Compare algorithms:

  - Baseline (sampling every 3 seconds)

  - Optimal algorithms: FECSS and MECSS

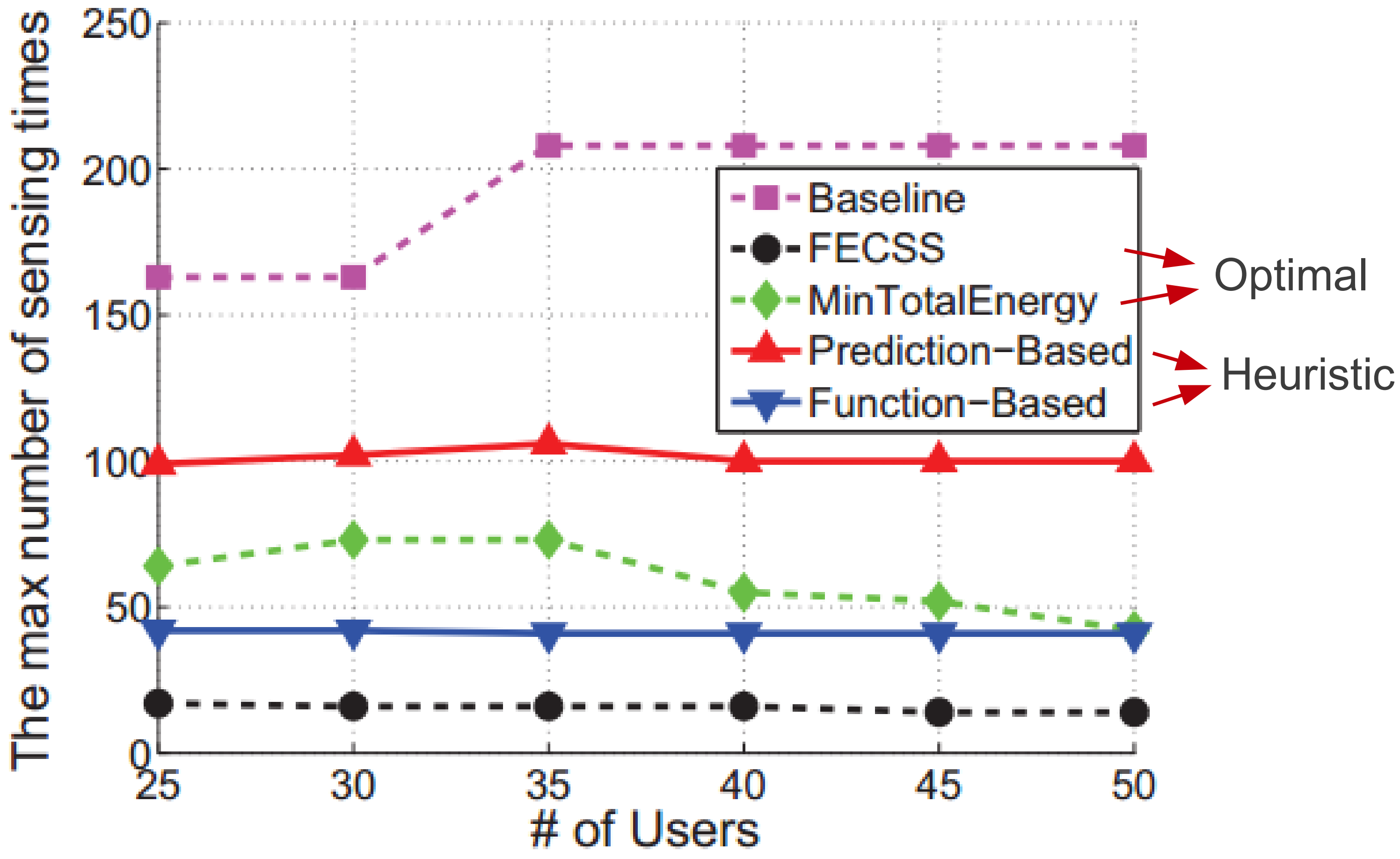
  - Heuristic algorithms: Prediction-based and Function-based

Total Energy Consumption



All algorithms significantly reduce total energy consumption by 80% to 90%

Max. Sensing Times



FECSS guarantees that the max. number of sensing times is minimum among all possible solutions