Sensor Network Services for Mobile Users

Chenyang Lu
Department of Computer Science and Engineering
Washington University in St. Louis

Hint on Question 5.2 in HW2

• Compute the maximum processor utilization that can be assigned to a Deferrable Server to guarantee the task set (periodic tasks).

\[
U_p + U_s \leq U_{s} + \alpha \left[ \frac{U_s + 2}{2U_s + 1} \right]^{1/\alpha} - 1 \Rightarrow U_p \leq \left[ \frac{U_s + 2}{2U_s + 1} \right]^{1/\alpha} - 1
\]

Utilization Bound with DS

- Under RMS

\[
U_s = U_s + n \left( \frac{U_s + 2}{2U_s + 1} \right)^{1/\alpha} - 1
\]

- As \( n \to \infty \):

\[
U_s = U_s + \ln \left( \frac{U_s + 2}{2U_s + 1} \right)
\]
- When \( U_s = 0.186 \), \( \min U_s = 0.652 \)

- System is schedulable if

\[
U_p \leq \ln \left( \frac{U_s + 2}{2U_s + 1} \right)
\]

Fire Monitoring

- Fireman in wild fire
  - report temperature within 100m of the moving user every 2s;
  - temperature data must be no more than 1s old

Navigation through Fire

Cargo Tracking
Services for Mobile Users

- **Mobicast**: information dissemination to moving areas.
- **MobiQuery**: spatiotemporal query for mobile users.
- **Roadmap Query**: navigation in dynamic environments.
- **MLDS**: mobile agent directory service for tracking applications.

A Motivating Application

**Fireman in wild fire**
- report temperature within **100m** of the moving user **every 2s**;
- temperature data must be no more than **1s** old

Spatiotemporal Constraints

- **Spatial constraints**
  - Query area moves with the user
  - All and only the sensors in the current query area should respond to the query
- **Temporal constraints**
  - Result must be delivered within the current period
  - Result should not be older than data validity interval

Meeting the constraints are critical to the fireman’s safety!

Key Challenges

- Limited power, memory and bandwidth.
- Low duty cycle for extending network lifetime.
  - Sleep schedule consists of cycles of long sleep period followed by short active period.
  - Lifetime of **450 days** requires < 1% duty cycle. (based on Mica2 motes [J.Polastre et al. Hot Chips 14])

<table>
<thead>
<tr>
<th>Cycle</th>
<th>Active</th>
<th>Sleep</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.15s</td>
<td>14.85s</td>
</tr>
<tr>
<td>2</td>
<td>0.15s</td>
<td>14.85s</td>
</tr>
<tr>
<td>3</td>
<td>0.15s</td>
<td>14.85s</td>
</tr>
<tr>
<td>4</td>
<td>0.15s</td>
<td>14.85s</td>
</tr>
</tbody>
</table>
**Design Goals of MobiQuery**

- Allows a mobile user to periodically collect sensor data from surrounding areas
  - Meet spatiotemporal constraints
  - Deal with severe resource constraints
    - Reduce storage cost
    - Reduce communication overhead
  - Robust against unpredictable user movement

**Prefetching**

- Predict future query areas
  - **Prefetching**
    - Forewarn nodes ahead of time so that they wake up at the right time to deliver the sensor data
    - Query dissemination and data collection
      - Nodes wake up in time and upload fresh data to user

**Wakeup Process**

- Assuming backbone based power management (ex. CCP, GAF, Span)

![Wakeup Process Diagram]

**Prefetching**

- Greedy Prefetching
  - Forewarn nodes in future query areas ASAP
  - Many routing trees set up simultaneously
  - Prediction of far away query areas is likely to be wrong!
- Just-in-time (JIT) Prefetching
  - Forewarn nodes in future query areas at the right time
  - Store and forward strategy
  - Reduce network contention & storage cost
  - More robust to user motion changes

**Directional Tree Creation (DTC)**

- Wake-up sensors ahead of the user
- Create a new query tree in each query area
- Aggregate sensor data and deliver to the user when the user reaches a query area

**MobiQuery Protocols**

- Directional Tree Creation (DTC) [ICDCS’05]
  - High overhead and network contention at high query rates
  - Requires knowledge of user motion profile
- Directional Tree Maintenance (DTM) [IPSN’05]
  - Reduces overhead and contention
  - Requires knowledge of user motion profile
- Omni-directional Tree Creation (OTC) [IPSN’05]
  - Does not require knowledge of user motion profile
**Generation of Motion Profile**

- **Motion prediction**
  - Predict future user path based on movement history
  - Motion profile available **after** actual movement
- **Motion planning**
  - Motion planner plans user path
  - Motion profile available **before** actual movement

**Directional Tree Maintenance (DTM)**

- Maintains a single moving tree rooted at the user
- Local reconfiguration based on geographic location and user motion profile

**Omni-directional Tree Creation (OTC)**

- No motion profile required
  - Assume knowledge of maximum user speed
  - Wake up sensors in a circular wake-up area
  - Encloses sensors in all possible query areas in the next few query periods
  - Create a tree when the user reaches a query area

**Analysis**

- Derive parameters such that sensors in future query areas are woken up
  - early enough to respond to the query in time
  - late enough to reduce storage cost
- **DTM / DTC**
  - Time to forward wake-up/prefetch message
    - Based on user velocity and sensor sleep period
- **OTC**
  - Size of wake-up area
    - Based on knowledge of maximum user velocity and sensor sleep period

**OTC - Algorithm**

**Directional Tree Maintenance (DTM)**
Simulations - Metrics

- Success ratio – fraction of queries that meet deadline for which, the fraction of sensors that respond in a query area is above a threshold
- Communication cost – total number of messages normalized by total number of query results

Simulations - Settings

- Run on ns-2
- Backbone-based Power management: Coverage Configuration Protocol
- 200 sensors in a 450mx450m area
- Query radius = 150m
- Sensor nodes
  - Communication Range – 105m
  - Sensing Range – 50m
  - Bandwidth – 2Mbps

Impact of Sleep Period

Figure 1. Effect on Success Ratio (T_p=0.5s); threshold = 90%

- No prefetching performs extremely poorly
- DTM > OTC >> DTC at long sleep periods

Impact of User Speed

Figure 2. Effect on Success Ratio (T_p=1s, T_s=15s); threshold = 90%

- DTM and OTC successfully adapt to different speed ranges.

Impact of Location Error

Figure 3. Effect on Success Ratio; threshold = 80%

- OTC is not affected by inaccuracy in the motion profile;
- DTM delivers 80% of the query results for location error of 15m.

Impact of Motion Changes

Figure 4. Effect on Success Ratio (location error = 10m); threshold = 80%

- OTC is not affected by user motion changes
- DTM maintains a success ratio greater than 80%
**Communication Cost**

![Graph showing communication cost comparison between OTC, DTC, and DTM under accurate motion profile (Ts=9s).]

**Implementation on Motes**
- Implemented DTC and DTM on 6x3 grid of Mica2 motes
- Acroname PPRK robot carrying stargate was used to emulate the user

**Summary**
- Meet stringent spatiotemporal constraints.
  - DTC - good performance for query periods ≥ 2s
  - DTM - lowest communication overhead.
  - OTC - most robust to location error and motion changes.
- DTM and OTC successfully deliver >80% of results when
  - query period = 1s
  - <1% duty cycle
  - user changes direction every minute
  - location error = 15m

**Roadmap Query for Navigation**

**Problem Addressed**
- Mobile entity *navigation in dynamic environments*
  - Example dynamic environment → fire
- Applications - search and rescue, evacuation
- Find a *safe path* from a start point S to a goal point G, that is clear of fire.
- Safe path - path where the maximum temperature along the path is below $\Delta_T$.

**Example Scenario**

![Example scenario diagram with points A, B, C, S, and G, and temperature values $T > \Delta_T$, $T_{AB} \leq \Delta_T$, and $T_{CG} < \Delta_T$.]
Application Challenges

- Dynamic environments may change quickly.
- Up-to-date information about the environment is required for successful navigation.
- On-board sensors insufficient due to limited sensing range.

Solution?
Obtain up-to-date information about the surrounding area through a sensor network!

New Challenges

Heavy communication workload

Congestion, communication delay and data loss.

Poor safety and navigation performance

Efficient query that can provide required information at minimum communication cost

Assumptions

- Sensor nodes are location aware.
- Mobile user communicates with sensor network through on-board gateway.
- Mobile user gets burnt at $\Delta_{\text{burn}}$
- Nodes have sensing range $R_S$.

TR $T_{\text{burn}}$ $S$

$\Delta_{\text{burn}} - \Delta = R_S$

Roadmap-based Approach

- Roadmap-based navigation algorithm – runs on the mobile entity.
- Roadmap Query (RQ) – collects sensor data from the sensor network.

Roadmap-based Navigation

Roadmap Vertex

Roadmap Edge

Edge Weight = Weighted function of max temperature along edge ($T_{\text{max}}$) and edge length.

If $T_{\text{max}} > \Delta_{\text{burn}}$, edge weight = infinity

Roadmap Query

- Optimized for roadmap-based navigation.
- Collects sensor data only from vicinity of mobile user.
- Selective sampling: collects data only from nodes that lie along roadmap edges.
**RQ - Forwarding Rule**

If received query msg from node $i$, forward query msg if
(a) cover roadmap edge $BC$, AND
(b) closest to $C$ among node $i$’s neighbors.

**RQ - Reply Rule**

Send query reply if
(a) forwarded query msg, OR
(b) cover edge $BC$ and temperature > $\Delta_T$.

**Query Propagation**

**Data Aggregation**

**Implementation**

- Implemented on Agilla - our mobile agent middleware
- Deployed on Mica2 motes.
Simulation Setting

- Simulations done in NS-2
- NIST Fire Dynamics Simulator (FDS) used to obtain realistic fire scenarios.
- Two simulation environments
  - More Dynamic – start time 50s
  - Less Dynamic – start time 200s
- Compared RQ with
  - Dartmouth Algorithm (DA) [Q. Li, et al. 2003]
  - Global Query (GQ)
  - Local Query (LQ) [G. Alankus, et al. 2005]

Success Ratio

![Success Ratio](image)

RQ > LQ > DA > GQ

Communication Cost

![Communication Cost](image)

- RQ has 70% lower communication cost than LQ.
- RQ reduces node participation by 60%

Summary

- Roadmap Query is optimized for navigation in dynamic environments.
  - Uses selective sampling guided by roadmaps.
  - Effectively handles node failures.
- Successfully deployed and tested RQ on the Agilla mobile agent middleware and Mica2 motes.
- Simulation results show that under realistic fire scenarios, Roadmap Query has
  - higher success ratio and
  - lower communication cost.

Optional Readings

- S. Bhattacharya, G. Xing, C. Lu, G.-C. Roman, B. Harris, and O. Chipara, Dynamic Wake-up and Topology Maintenance Protocols with Spatiotemporal Guarantees, IPSN'05.