15-446 Distributed Systems
Spring 2009

L-22 Sensor Networks

Ad Hoc Routing

- Goal: Communication between wireless nodes
  - No external setup (self-configuring)
  - Often need multiple hops to reach dst

Overview

- Ad hoc routing
- Sensor Networks
- Directed Diffusion
- Aggregation
  - TAG
  - Synopsis Diffusion

Ad Hoc Routing

- Create multi-hop connectivity among set of wireless, possibly moving, nodes
- Mobile, wireless hosts act as forwarding nodes as well as end systems
- Need routing protocol to find multi-hop paths
  - Needs to be dynamic to adapt to new routes, movement
  - Interesting challenges related to interference and power limitations
  - Low consumption of memory, bandwidth, power
  - Scalable with numbers of nodes
  - Localized effects of link failure
Challenges and Variants

- Poorly-defined “links”
  - Probabilistic delivery, etc. Kind of $n^2$ links
- Time-varying link characteristics
- No oracle for configuration (no ground truth configuration file of connectivity)
- Low bandwidth (relative to wired)
- Possibly mobile
- Possibly power-constrained

Problems Using DV or LS

- DV protocols may form loops
  - Very wasteful in wireless: bandwidth, power
  - Loop avoidance sometimes complex
- LS protocols: high storage and communication overhead
- More links in wireless (e.g., clusters) - may be redundant $\rightarrow$ higher protocol overhead

Problems Using DV or LS

- Periodic updates waste power
  - Tx sends portion of battery power into air
  - Reception requires less power, but periodic updates prevent mobile from “sleeping”
- Convergence may be slower in conventional networks but must be fast in ad-hoc networks and be done without frequent updates

Proposed Protocols

- Destination-Sequenced Distance Vector (DSDV)
  - DV protocol, destinations advertise sequence number to avoid loops, not on demand
- Temporally-Ordered Routing Algorithm (TORA)
  - On demand creation of hbh routes based on link-reversal
- Dynamic Source Routing (DSR)
  - On demand source route discovery
- Ad Hoc On-Demand Distance Vector (AODV)
  - Combination of DSR and DSDV: on demand route discovery with hbh routing
DSR Concepts

- Source routing
  - No need to maintain up-to-date info at intermediate nodes
- On-demand route discovery
  - No need for periodic route advertisements

DSR Components

- Route discovery
  - The mechanism by which a sending node obtains a route to destination
- Route maintenance
  - The mechanism by which a sending node detects that the network topology has changed and its route to destination is no longer valid

DSR Route Discovery

- Route discovery - basic idea
  - Source broadcasts route-request to Destination
  - Each node forwards request by adding own address and re-broadcasting
  - Requests propagate outward until:
    - Target is found, or
    - A node that has a route to Destination is found

C Broadcasts Route Request to F

[Diagram showing broadcast route request from source C to destination F]
C Broadcasts Route Request to F

C Transmits a Packet to F

H Responds to Route Request

Forwarding Route Requests

- A request is forwarded if:
  - Node is not the destination
  - Node not already listed in recorded source route
  - Node has not seen request with same sequence number
  - IP TTL field may be used to limit scope
- Destination copies route into a Route-reply packet and sends it back to Source
**Route Cache**

- All source routes learned by a node are kept in Route Cache
  - Reduces cost of route discovery
- If intermediate node receives RR for destination and has entry for destination in route cache, it responds to RR and does not propagate RR further
- Nodes overhearing RR/RP may insert routes in cache

**Sending Data**

- Check cache for route to destination
  - If route exists then
    - If reachable in one hop
      - Send packet
    - Else insert routing header to destination and send
  - If route does not exist, buffer packet and initiate route discovery

**Discussion**

- Source routing is good for on demand routes instead of a priori distribution
- Route discovery protocol used to obtain routes on demand
  - Caching used to minimize use of discovery
- Periodic messages avoided

**Overview**

- Ad Hoc Routing
- **Sensor Networks**
- Directed Diffusion
- Aggregation
  - TAG
  - Synopsis Diffusion
Smart-Dust/Motes

- First introduced in late 90’s by groups at UCB/UCLA/USC
  - Published at Mobicom/SOSP conferences
- Small, resource limited devices
  - CPU, disk, power, bandwidth, etc.
- Simple scalar sensors – temperature, motion
- Single domain of deployment (e.g. farm, battlefield, etc.) for a targeted task (find the tanks)
- Ad-hoc wireless network

Berkeley Motes

- Devices that incorporate communications, processing, sensors, and batteries into a small package
- Atmel microcontroller with sensors and a communication unit
  - RF transceiver, laser module, or a corner cube reflector
  - Temperature, light, humidity, pressure, 3 axis magnetometers, 3 axis accelerometers

Smart-Dust/Motes

- Hardware
  - UCB motes
- Programming
  - TinyOS
- Query processing
  - TinyDB
  - Directed diffusion
  - Geographic hash tables
- Power management
  - MAC protocols
  - Adaptive topologies

Berkeley Motes (Levis & Culler, ASPLOS 02)

- Devices that incorporate communications, processing, sensors, and batteries into a small package
  - Atmel microcontroller with sensors and a communication unit
    - RF transceiver, laser module, or a corner cube reflector
    - Temperature, light, humidity, pressure, 3 axis magnetometers, 3 axis accelerometers
Sensor Net Sample Apps

- Habitat Monitoring: Storm petrels on great duck island, microclimates on James Reserve.
- Earthquake monitoring in shake-test sites.
- Vehicle detection: sensors along a road, collect data about passing vehicles.
- Traditional monitoring apparatus.

Metric: Communication

- Lifetime from one pair of AA batteries
  - 2-3 days at full power
  - 6 months at 2% duty cycle
- Communication dominates cost
  - < few mS to compute
  - 30mS to send message

Communication In Sensor Nets

- Radio communication has high link-level losses
  - typically about 20% @ 5m
- Ad-hoc neighbor discovery
- Tree-based routing

Overview

- Ad Hoc Routing
- Sensor Networks
- Directed Diffusion
- Aggregation
  - TAG
  - Synopsis Diffusion
The long term goal

Embed numerous distributed devices to monitor and interact with physical world: in work-spaces, hospitals, homes, vehicles, and "the environment" (water, soil, air...)

Network these devices so that they can coordinate to perform higher-level tasks.

Requires robust distributed systems of tens of thousands of devices.

Motivation

- Properties of Sensor Networks
  - Data centric, but not node centric
  - Have no notion of central authority
  - Are often resource constrained
- Nodes are tied to physical locations, but:
  - They may not know the topology
  - They may fail or move arbitrarily
- Problem: How can we get data from the sensors?

Directed Diffusion

- Data centric – nodes are unimportant
- Request driven:
  - Sinks place requests as interests
  - Sources are eventually found and satisfy interests
  - Intermediate nodes route data toward sinks
- Localized repair and reinforcement
- Multi-path delivery for multiple sources, sinks, and queries

Motivating Example

- Sensor nodes are monitoring a flat space for animals
- We are interested in receiving data for all 4-legged creatures seen in a rectangle
- We want to specify the data rate
**Interest and Event Naming**

- **Query/interest:**
  1. Type=four-legged animal
  2. Interval=20ms (event data rate)
  3. Duration=10 seconds (time to cache)
  4. Rect=[-100, 100, 200, 400]
- **Reply:**
  1. Type=four-legged animal
  2. Instance = elephant
  3. Location = [125, 220]
  4. Intensity = 0.6
  5. Confidence = 0.85
  6. Timestamp = 01:20:40
- **Attribute-Value pairs, no advanced naming scheme**

**Diffusion (High Level)**

- Sinks broadcast interest to neighbors
- Interests are cached by neighbors
- Gradients are set up pointing back to where interests came from at low data rate
- Once a sensor receives an interest, it routes measurements along gradients

**Illustrating Directed Diffusion**

- Setting up gradients
- Sending data
- Reinforcing stable path
- Recovering from node failure

**Summary**

- **Data Centric**
  - Sensors net is queried for specific data
  - Source of data is irrelevant
  - No sensor-specific query
- **Application Specific**
  - In-sensor processing to reduce data transmitted
  - In-sensor caching
- **Localized Algorithms**
  - Maintain minimum local connectivity – save energy
  - Achieve global objective through local coordination
- Its gains due to aggregation and duplicate suppression may make it more viable than ad-hoc routing in sensor networks
Overview

- Ad Hoc Routing
- Sensor Networks
- Directed Diffusion
- Aggregation
  - TAG
  - Synopsis Diffusion

TAG Introduction

- Programming sensor nets is hard!
- Declarative queries are easy
  - Tiny Aggregation (TAG): In-network processing via declarative queries
- In-network processing of aggregates
  - Common data analysis operation
  - Communication reducing
  - Operator dependent benefit
- Across nodes during same epoch
- Exploit semantics improve efficiency!
- Example:
  - Vehicle tracking application: 2 weeks for 2 students
  - Vehicle tracking query: took 2 minutes to write, worked just as well!

Illustration: Aggregation

- In each epoch:
  - Each node samples local sensors once
  - Generates **partial state record** (PSR)
    - local readings
    - readings from children
  - Outputs PSR during its comm. slot.
- At end of epoch, PSR for whole network output at root
  - (In paper: pipelining, grouping)
Types of Aggregates

- SQL supports MIN, MAX, SUM, COUNT, AVERAGE
- Any function can be computed via TAG
- In network benefit for many operations
  - E.g. Standard deviation, top/bottom N, spatial union/intersection, histograms, etc.
  - Compactness of PSR

Taxonomy of Aggregates

- TAG insight: classify aggregates according to various functional properties
  - Yields a general set of optimizations that can automatically be applied

<table>
<thead>
<tr>
<th>Property</th>
<th>Examples</th>
<th>Affects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Partial State</td>
<td>MEDIAN: unbounded, MAX: 1 record</td>
<td>Effectiveness of TAG</td>
</tr>
<tr>
<td>Duplicate</td>
<td>MIN: dup. insensitive, AVG: dup. sensitive</td>
<td>Routing Redundancy</td>
</tr>
<tr>
<td>Sensitivity</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exemplary vs. Summary</td>
<td>MAX: exemplary COUNT: summary</td>
<td>Applicability of Sampling, Effect of Loss</td>
</tr>
<tr>
<td>Monotonic</td>
<td>COUNT: monotonic AVG: non-monotonic</td>
<td>Hypothesis Testing, Snooping</td>
</tr>
</tbody>
</table>

Benefit of In-Network Processing

Simulation Results
2500 Nodes
50x50 Grid
Depth = ~10
Neighbors = ~20

Optimization: Channel Sharing ("Snooping")

- Insight: Shared channel enables optimizations
- Suppress messages that won’t affect aggregate
  - E.g., MAX
  - Applies to all exemplary, monotonic aggregates
Optimization: Hypothesis Testing

- **Insight**: Guess from root can be used for suppression
  - E.g. ‘MIN < 50’
  - Works for monotonic & exemplary aggregates
  - Also summary, if imprecision allowed

- **How is hypothesis computed?**
  - Blind or statistically informed guess
  - Observation over network subset

Optimization: Use Multiple Parents

- For duplicate insensitive aggregates
- Or aggregates that can be expressed as a linear combination of parts
  - Send (part of) aggregate to all parents
  - In just one message, via broadcast
  - Decreases variance

Overview

- Ad Hoc Routing
- Sensor Networks
- Directed Diffusion
- Aggregation
  - TAG
  - Synopsis Diffusion

Aggregation in Wireless Sensors

Aggregate data is often more important

**In-network aggregation over tree with unreliable communication**

Used by current systems,
TinyDB [Madden et al. OSDI’02]
Cougar [Bonnet et al. MDM’01]

Not robust against node- or link-failures
### Traditional Approach

- **Reliable communication**
  - E.g., RMST over Directed Diffusion [Stann'03]
- **High resource overhead**
  - 3x more energy consumption
  - 3x more latency
  - 25% less channel capacity
- **Not suitable for resource constrained sensors**

### Exploiting Broadcast Medium

- **Robust multi-path**
- **Energy-efficient**
  - Double-counting
  - Different ordering

Challenge:
- **Order and duplicate insensitivity (ODI)**

### A Naïve ODI Algorithm

- **Goal:** count the live sensors in the network

- 

### Synopsis Diffusion (SenSys’04)

- **Goal:** count the live sensors in the network

- 

- **Synopsis should be small**

- **Approximate COUNT algorithm:** logarithmic size bit vector
Synopsis Diffusion over Rings

- A node is in ring $i$ if it is $i$ hops away from the base-station.
- Broadcasts by nodes in ring $i$ are received by neighbors in ring $i-1$.
- Each node transmits once = optimal energy cost (same as Tree).

Evaluation

Approximate COUNT with Synopsis Diffusion

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Energy (mJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tree</td>
<td>41.8 mJ</td>
</tr>
<tr>
<td>Syn. Diff.</td>
<td>42.1 mJ</td>
</tr>
</tbody>
</table>

More robust than Tree

Almost as energy efficient as Tree