Algorithmic Foundations of Ad Hoc Networks: Part II

Rajmohan Rajaraman, Northeastern U.
www.ccs.neu.edu/home/rraj/AdHocTutorial.ppt
(Part II of a joint tutorial with Andrea Richa, Arizona State U.)

July 2004
Many Thanks to...

- Roger Wattenhofer and organizers of the summer school
- ETH Zurich
- All the researchers whose contributions will be discussed in this tutorial
Outline

1. Medium Access Control
2. Power Control
3. Topology Control
4. Fundamental limits of ad hoc networks
5. Sensor Network Protocols

Application

Routing
What’s Not Covered?

• Frequency (channel) assignment
  – Arises in cellular networks
  – Modeled as coloring problems

• Ad Hoc Network Security
  – Challenges due to the low-power, wireless, and distributed characteristics
  – Authentication, key sharing,…
  – Anonymous routing

• Smart antenna:
  – Beam-forming (directional) antenna
  – MIMO systems

• Many physical layer issues

• …
Medium Access Control
Medium Access Control Protocols

- **Schedule-based:** Establish transmission schedules statically or dynamically
  - TDMA: Assign channel to station for a fixed amount of time
  - FDMA: Assign a certain frequency to each station
  - CDMA: Encode the individual transmissions over the entire spectrum

- **Contention-based:**
  - Let the stations contend for the channel
  - Random access protocols
Contention Resolution Protocols

- **CSMA** (Carrier-sense multiple access)
  - Ethernet
  - Aloha
- **MACA** [Kar90] (Multiple access collision avoidance)
- **MACAW** [BDSZ94]
- **CSMA/CA** and **IEEE 802.11**
- Other protocols:
  - Bluetooth
  - Later, MAC protocols for sensor networks
Ingredients of MAC Protocols

• Carrier sense (CS)
  – Hardware capable of sensing whether transmission taking place in vicinity

• Collision detection (CD)
  – Hardware capable of detecting collisions

• Collision avoidance (CA)
  – Protocol for avoiding collisions

• Acknowledgments
  – When collision detection not possible, link-layer mechanism for identifying failed transmissions

• Backoff mechanism
  – Method for estimating contention and deferring transmissions
Carrier Sense Multiple Access

- Every station senses the carrier before transmitting
- If channel appears free
  - Transmit (with a certain probability)
- Otherwise, wait for some time and try again
- Different CSMA protocols:
  - Sending probabilities
  - Retransmission mechanisms
Slotted Aloha

- Proposed for packet radio environments where every node can hear every other node
- Assume collision detection
- Stations transmit at the beginning of a slot
- If collision occurs, then each station waits a random number of slots and retries
  - Random wait time chosen has a geometric distribution
  - Independent of the number of retransmissions
- Analysis in standard texts on networking theory [BG92]
Ethernet

- CSMA with collision detection (CSMA/CD)
- If the adaptor has a frame and the line is idle: transmit
- Otherwise wait until idle line then transmit
- If a collision occurs:
  - **Binary exponential** backoff: wait for a random number $\in [0, 2^i-1]$ of slots before transmitting
  - After ten collisions the randomization interval is frozen to max 1023
  - After 16 collisions the controller throws away the frame
CSMA for Multihop Networks

- In CSMA, sender decides to transmit based on carrier strength in *its vicinity*
- Collisions occur at *the receiver*
- Carrier strengths at sender and receiver may be different:

  **Hidden Terminal**

![Diagram showing a hidden terminal in a multihop network with nodes A, B, and C. Node A and node C transmit simultaneously, causing a collision at node B.]
CSMA for Multihop Networks

- In CSMA, sender decides to transmit based on carrier strength in its vicinity.
- Collisions occur at the receiver.
- Carrier strengths at sender and receiver may be different:

![Exposed Terminal Diagram]

---

CSMA for Multihop Networks

- In CSMA, sender decides to transmit based on carrier strength in its vicinity.
- Collisions occur at the receiver.
- Carrier strengths at sender and receiver may be different:
Multiple Access Collision Avoidance

- No carrier sense
- Collision avoidance using **RTS/CTS** handshake
  - Sender sends Request-to-Send (RTS)
    - Contains length of transmission
  - If receiver hears RTS and not currently deferring, sends Clear-to-Send (CTS)
    - Also contains length of transmission
  - On receiving CTS, sender starts DATA transmission
- Any station overhearing an RTS defers until a CTS would have finished
- Any station overhearing a CTS defers until the expected length of the DATA packet
MACA in Action

• If C also transmits RTS, collision at B
**MACA in Action**

- C knows the expected DATA length from CTS

![Diagram](image)

A → CTS → B → C

Defers until DATA completion
MACA in Action

- Avoids the hidden terminal problem
**MACA in Action**

- CTS packets have fixed size

![Diagram showing MACA in Action](image)

- RTS packets have fixed size
- Defers until CTS
MACA in Action

• C does not hear a CTS
MACA in Action

- C is free to send to D; no exposed terminal
MACA in Action

• Is C really free to send to D?
MACA in Action

• In fact, C increases its backoff counter!
The CSMA/CA Approach

- Add carrier sense; C will sense B’s transmission and refrain from sending RTS
False Blocking

- F sends RTS to E; D sends RTS to C
- E is *falsely* blocked [Bha98, RCS03]
False Blocking

Show that false blocking may lead to temporary deadlocks
Alternative Approach: MACAW

- [BDSZ94]
- No carrier sense, no collision detection
- Collision avoidance:
  - Sender sends RTS
  - Receiver sends CTS
  - Sender sends DS
  - Sender sends DATA
  - Receiver sends ACK
  - Stations hearing DS defer until end of data transmission
- Backoff mechanism:
  - Exponential backoff with significant changes for improving fairness and throughput
The IEEE 802.11 Protocol

- Two medium access schemes
- Point Coordination Function (PCF)
  - Centralized
  - For infrastructure mode
- Distributed Coordination Function (DCF)
  - For ad hoc mode
  - CSMA/CA
  - Exponential backoff
CSMA/CA with Exponential Backoff

- Begin
  - Busy?
    - Yes: Transmit frame
    - No: Max window?
      - Yes: Max attempt?
        - Yes: Increment attempt
        - No: Discard packet
      - No: Double window
  - No: Wait inter-frame period
    - Increment attempt
    - Wait U[0,W]
Performance Analysis of 802.11

- Markov chain models for DCF
- Throughput:
  - Saturation throughput: maximum load that the system can carry in stable conditions
- Fairness:
  - Long-term fairness
  - Short-term fairness
- Focus on collision avoidance and backoff algorithms
Analysis of Saturation Throughput

• Model assumptions [Bia00]:
  - No hidden terminal: all users can hear one another
  - No packet capture: all receive powers are identical
  - Saturation conditions: queue of each station is always nonempty

• Parameters:
  - Packet lengths (headers, control and data)
  - Times: slots, timeouts, interframe space
A Stochastic Model for Backoff

- Let $b(t)$ denote the backoff time counter for a given node at slot $t$
  - Slot: constant time period $\sigma$ if the channel is idle, and the packet transmission period, otherwise
  - Note that $t$ is not the same as system time
- The variable $b(t)$ is non-Markovian
  - Its transitions from a given value depend on the number of retransmissions
A Stochastic Model for Backoff

• Let $s(t)$ denote the backoff stage at slot $t$
  - In the set $\{0,...,m\}$, where $m$ is the maximum number of backoffs
• Is $(s(t),b(t))$ Markovian?
• Unfortunately, no!
  - The transition probabilities are determined by collision probabilities
  - The collision probability may in turn depend on the number of retransmissions suffered
• Independence Assumption:
  - Collision probability is constant and independent of number of retransmissions
Markov Chain Model

Bianchi 00
Steady State Analysis

• Two probabilities:
  – Transmission probability $\tau$
  – Collision probability $p$

• Analyzing the Markov chain yields an equation for $\tau$ in terms of $p$.

• However, we also have

$$p = 1 - (1 - \tau)^{n-1}$$

• Solve for $\tau$ and $p$.
Saturation Throughput Calculation

• Probability of at least one transmission

\[ P_{tr} = 1 - (1 - \tau)^n \]

• Probability of a successful slot

\[ P_s = \frac{n\tau(1 - \tau)^{n-1}}{1 - (1 - \tau)^n} \]

• Throughput: (packet length \( L \))

\[ P_s P_{tr} L \]

\[ (1 - P_{tr})\sigma + P_{tr} L \]
Analysis vs. Simulations

Fig. 6. Saturation Throughput: analysis versus simulation.

<table>
<thead>
<tr>
<th>n</th>
<th>BAS</th>
<th>RTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.8473</td>
<td>0.846 ± 0.001</td>
</tr>
<tr>
<td>2</td>
<td>0.8198</td>
<td>0.817 ± 0.001</td>
</tr>
<tr>
<td>3</td>
<td>0.8368</td>
<td>0.835 ± 0.001</td>
</tr>
<tr>
<td>3</td>
<td>0.8279</td>
<td>0.823 ± 0.001</td>
</tr>
</tbody>
</table>

Bianchi 00
Fairness Analysis

• How is the throughput distributed among the users?

• Long-term:
  – Steady-state share of the throughput

• Short-term:
  – Sliding window measurements
  – Renewal reward theory based on Markov chain modeling
Long-Term Fairness

- Basic binary exponential backoff:
  - Steady-state throughput equal for all nodes
  - However, *constant probability* (> 0) that one node will capture the channel

Consider two nodes running CSMA with basic exponential backoff on a shared slotted channel. Assume that both nodes have an infinite set of packets to send. Prove that there is a *constant (> 0) probability* that one node will have *O(1)* throughput, while the other will be unable to send even a single packet.
Long-Term Fairness

• Basic binary exponential backoff:
  – Steady-state throughput equal for all nodes
  – However, *constant probability* (> 0) that one node will capture the channel

• Bounded binary exponential backoff:
  – After a certain number of retransmissions, backoff stage set to zero and packet retried

• MACAW: All nodes have the same backoff stage
Short-Term Fairness

• Since focus on successful transmissions, need not worry about collision probabilities
• The CSMA/CA and Aloha protocols can both be captured as Markov chains
• CSMA/CA has higher throughput, low short-term fairness
  – The capture effect results in low fairness
• Slotted Aloha has low throughput, higher short-term fairness
• [KKB00]
Backoff in MACAW

• Refinement of exponential backoff to improve fairness and throughput

• Fairness:
  – Nodes contending for the same channel have the same backoff counter
  – Packet header contains value of backoff counter
  – Whenever a station hears a packet, it copies the value into its backoff counter

• Throughput:
  – Sharing backoff counter across channels causes false congestion
  – Separate backoff counter for different streams (destinations)
Open Problems in Contention Resolution

- Throughput and fairness analysis for multihop networks
  - Dependencies carry over hops
  - In the “single hop” case nodes get synchronized since every node is listening to the same channel
  - Channels that a node can communicate on differ in the multihop case
  - Even the simplest case when only one node cannot hear all nodes is hard

- Fairness analysis of MACAW
  - All nodes contending for a channel use same backoff number; similar fairness as slotted Aloha?
  - Different backoff numbers for different channels
Transmission and Sensing Ranges

Transmission range

Sensing/interference range

250m

550m
Effect on RTS/CTS Mechanism
Effect on RTS/CTS Mechanism
Effect on RTS/CTS Mechanism
Effect on RTS/CTS Mechanism
Effect on RTS/CTS Mechanism
Implications of Differing Ranges

- Carrier sense does not completely eliminate the hidden terminal problem
- The unit disk graph model, by itself, is not a precise model
- The differing range model itself is also simplistic
  - Radios have power control capabilities
  - Whether a transmission is received depends on the signal-to-interference ratio
  - Protocol model for interference [GK00]
Power Control
What and Why

• The ability of a mobile wireless station to control its energy consumption:
  – Switching between idle/on/off states
  – Controlling transmission power

• Throughput:
  – Interference determined by transmission powers and distances
  – Power control may reduce interference allowing more spatial reuse

• Energy:
  – Power control could offer significant energy savings and enhance network lifetime
The Attenuation Model

• Path loss:
  – Ratio of received power to transmitted power
  – Function of medium properties and propagation distance
• If $P_R$ is received power, $P_T$ is the transmitted power, and $d$ is distance

$$P_R = O\left(\frac{P_T}{d^\alpha}\right)$$

• where $\alpha$ ranges from 2 to 4
Interference Models

• In addition to path loss, bit-error rate of a received transmission depends on:
  – Noise power
  – Transmission powers and distances of other transmitters in the receiver’s vicinity

• Two models:
  – Physical model
  – Protocol model
The Physical Model

- Let \( \{X_i\} \) denote set of nodes that are simultaneously transmitting.
- Let \( P_i \) be the transmission power of node \( X_i \).
- Transmission of \( X_i \) is successfully received by \( Y \) if:

\[
\frac{P_i}{d(X_i, Y)^\alpha} \geq \beta \\
N + \sum_{k \neq i} \frac{P_k}{d(X_k, Y)^\alpha}
\]

- \( \beta \) is the min signal-interference ratio (SIR).
The Protocol Model

- Transmission of $X_i$ is successfully received by $Y$ if for all $k$

$$\frac{P_i}{d(X_i, Y)^\alpha} \geq (1 + \Delta) \frac{P_k}{d(X_k, Y)^\alpha}$$

- where $\Delta$ is a protocol-specified guard-zone to prevent interference
Scenarios for Power Control

- **Individual transmissions:**
  - Each node decides on a power level on the basis of contention and power levels of neighbors

- **Network-wide task:**
  - Broadcast
  - Multicast

- **Static:**
  - Assign fixed (set of) power level(s) to each node
  - Topology control
Review of Proposed Schemes

- Basic power control scheme
  - PCM
- POWMAC
- $\delta$-PCS
- PCMA
- PCDC

Energy

Throughput and energy

Dual channel schemes
The Basic Power Control Scheme

• The IEEE 802.11 does not employ power control
  – Every transmission is at the maximum possible power level $P_{\text{max}}$
• Transmit RTS/CTS at $P_{\text{max}}$
• In the process, determine minimum power level $P$ needed to transmit:
  – Function of sender-receiver distance $d$
• DATA and ACK are sent at level $P$
Deficiency of the Basic Scheme
Deficiency of the Basic Scheme
Power Control MAC (PCM)

- RTS/CTS at $P_{\text{max}}$
- For DATA packets:
  - Send at the minimum power $P$ needed, as in the basic scheme
  - Periodically send at $P_{\text{max}}$, to maintain the collision avoidance feature of 802.11
- ACK sent at power level $P$
- Throughput comparable to 802.11
- Significant energy savings [JV02]
POWMAC

- Access window for RTS/CTS exchanges
- Multiple concurrent DATA packet transmissions following RTS/CTS
- Collision avoidance information attached in CTS to bound transmission power of potentially interfering nodes
- Aimed at increasing throughput as well as reducing energy consumption
- [MK04]
$\delta$-PCS

- IEEE 802.11
- Basic power control scheme
- $\delta$-PCS: $0 \leq \delta \leq \alpha$ \[\text{[JLNRR04]}\]

Simulations indicate:
- $\delta$ in the range 2-3 provides best performance
- 30-40% increase in throughput and 3-fold improvement in energy consumption
- Fair over varying distance ranges
Dual-Channel Schemes

• Use a separate control channel
• PCMA [MBH01]:
  – Receiver sends busy tone pulses advertising its interference margin
• PCDC [MK03]:
  – RTS/CTS on control channel
• Signal strength of busy tones used to determine transmission power for data
Open Problems in Power Control

- Develop an analytical model for measuring the performance of power control protocols
  - Model for node locations
  - Model for source and destination selections: effect of transmission distances
  - Interaction with routing
  - Performance measures: throughput, energy, and fairness
Topology Control
Connectivity

- Given a set of nodes in the plane
- Goal: Minimize the maximum power needed for connectivity
- Let $p : V \rightarrow \mathbb{N}$ denote the power function
- Induced graph contains edge $(u, v)$ if

\[ p(u), p(v) \geq d(u, v)^\alpha \]
Connectivity

- To obtain a given topology $H$, need
  \[ p(u) = \max_{(u,v) \in H} d(u,v)^\alpha \]

- Goal: Minimize the maximum edge length

- MST!
  - MST also minimizes the weight of the max-weight edge

- Find MST $T$ and set
  \[ p(u) = \max_{(u,v) \in T} d(u,v)^\alpha \]
Connectivity: Distributed Heuristics

• Motivated by need to address mobility [RRH00]
• Initially, every node has maximum power
• Nodes continually monitor routing updates to track connectivity

• Neighbor Reduction Protocol:
  – Each node attempts to maintain degree within a range, close to a desired degree
  – Adjusts power depending on current degree
  – Magnitude of change dependent on difference between current and desired degree

• Neighbor Addition Protocol:
  – Triggered if node recognizes graph not connected
  – Sets power to maximum level
Connectivity: Total Power Cost

Given a set of nodes in the plane, determine an assignment of power levels that achieves connectivity at minimum total power cost.
Bounded-Hops Connectivity

- **Goal:** Minimize the total power cost needed to obtain a topology that has a diameter of at most $h$ hops [CPS99, CPS00]
- **Assume** $\alpha = 2$
- **Lower bound:**
  - If minimum distance is $\delta$, then total power cost is at least
    \[
    \Omega(\delta^\alpha n^{1+1/h})
    \]
- **Upper bound:**
  - If maximum distance is $D$, then total power cost is at most
    \[
    O(D^\alpha n^{1/h})
    \]
K-Connectivity

- Goal: Minimize the maximum transmission power to obtain a k-connected topology
- Critical transmission radius
  - Smallest radius $r$ such that if every node sets its range to $r$ then the topology is k-connected
- Critical neighbor number $[WY04]$
  - Smallest number $l$ such that if every node sets its transmission range to the distance to the $l$th nearest neighbor then the topology is k-connected
- Characterization of the critical transmission radius and critical neighbor number for random node placements $[WY04]$
Energy-Efficient Topologies

• Goal: Construct a topology that contains energy-efficient paths
  – For any pair of nodes, there exists a path nearly as energy-efficient as possible

• Constraints:
  – Sparseness
  – Constant degree
  – Distributed construction
Formalizing Energy-Efficiency

• Given a subgraph $H$ of $G$, the complete graph over the $n$ nodes:
  - Define energy-stretch of $H$ as the maximum, for all $u$ and $v$, of the ratio of the least energy path between $u$ and $v$ in $H$ to that in $G$

$$\max_{u,v} \frac{\text{optimal-energy}_H(u,v)}{\text{optimal-energy}_G(u,v)}$$

• Variant of distance-stretch

$$\max_{u,v} \frac{\text{optimal-distance}_H(u,v)}{\text{optimal-distance}_G(u,v)}$$

• Since $\alpha > 1$, a topology of distance-stretch $O(1)$ also has energy-stretch $O(1)$
Spanners

- Spanners are topologies with $O(1)$ distance stretch
- Extensively studied in the graph algorithms and graph theory literature [Epp96]
- (Distance)-spanners are also energy-spanners
- Spanners for Euclidean space based on proximity graphs:
  - Delaunay triangulation
  - The Yao graph
The Yao Graph

- Each node divides the space into sectors of angle
- Fixes an edge with the nearest neighbor in each sector.
- Sparse: each node fixes at most $\frac{2\pi}{\theta}$ edges
- Stretch is at most $\frac{1}{1 - 2\sin(\theta/2)}$
The Yao Graph

• Each node divides the space into sectors of angle
• Fixes an edge with the nearest neighbor in each sector.
• Sparse: each node fixes at most $\frac{2\pi}{\theta}$ edges
• Stretch is at most $\frac{1}{1 - 2\sin(\theta/2)}$
• Degree could be $\Omega(n)$
Variants of the Yao Graph

• Can derive a constant-degree subgraph by a phase of edge removal [WLBW00, LHB⁺01]
  - Increases stretch by a constant factor
  - Need to process edges in a coordinated order

• Locally computable variant of the Yao graph [LWWF02, WL02]
  1. Each node divides the space into sectors of angle $\theta$.
  2. Each node computes a neighbor set which consists of each nearest neighbor in all its sectors.
  3. $(u,v)$ is selected if $v$ is in $u$’s neighbor set and $u$ is the nearest among those that selected $v$ in its neighbor set.
Local Postprocessing of Yao Graph

1. Each node divides the space into sectors of angle $\theta$
Local Postprocessing of Yao Graph

2. Each node computes a neighbor set which consists of each nearest neighbor in all its sectors.
2. Each node computes a neighbor set which consists of each nearest neighbor in all its sectors.
Local Postprocessing of Yao Graph

3. \((u,v)\) is selected if \(v\) is in \(u\)’s neighbor set and \(u\) is the nearest among those that selected \(v\) into its nearest neighbor.
Local Postprocessing of Yao Graph

3. \((u,v)\) is selected if \(v\) is in \(u\)’s neighbor set and \(u\) is the nearest among those that selected \(v\) into its nearest neighbor.
Properties of the Topology

- By definition, constant-degree
- For $\theta$ sufficiently small, the topology has constant energy stretch for arbitrary point sets \[JRS03\]
  - Challenge: Unlike for the Yao graph, the min-cost path from $u$ to $v$ may traverse nodes that are farther from $u$ than $v$
- Does the algorithm yield a distance-spanner?
  - Can establish claim for specialized node distributions \[JRS03\]
  - Weak spanner property holds \[GLSV02\]
Other Recent Work

• Energy-efficient planar topologies:
  – Combination of localized Delaunay triangulation and Yao structures
  – Planar, degree-bounded, and energy-spanner [WL03, SWL04]
Topology Control and Interference

- Focus thus far on energy-efficiency and connectivity
- Previous interference models (physical and protocol models) for individual transmissions
- How to measure the “interference quotient” of a topology?
  - **Edge interference number**: What is the maximum number of edges that an edge interferes with?
  - **Node interference number**: What is the maximum number of nodes that an edge interferes with?
Edge Interference Number

- Defined by [MadHSV02]
- When does an edge interfere with another edge?
  - The lune of the edge contains either endpoint of the other edge

\[ L(e) = \text{lune of } e \]
\[ I(e) = \left| \{(u,v) \in T : L(e) \cap \{u,v\} \neq \emptyset \} \right| - 1 \]
\[ I(T) = \max_{e \in T} I(e) \]
Node Interference Number

- Defined by [BvRWZ04]
- When does an edge interfere with another node?
  - The lune of the edge contains the node

\[ L(e) = \text{lune of } e \]
\[ I(e) = |L(e) - \{u, v\}| \]
\[ I(T) = \max_{e \in T} I(e) \]
Minimizing NIM

- Goal: Determine connected topology that minimizes NIM
- $I(e)$ is independent of the topology

\[ L(e) = \text{lune of } e \]
\[ I(e) = \left| L(e) - \{u, v\} \right| \]
\[ I(T) = \max_{e \in T} I(e) \]
Minimizing NIM

- Set weight of $e$ to be $I(e)$
- Find spanning subgraph that minimizes maximum weight
  - MST!
- Calculating $L(e)$ possible using local communication
- Computing an MST difficult to do locally
- In general, minimizing NIM hard to do locally
Spariness and Interference

Prove that for a random distribution of nodes on the plane, the Yao graph has an NIM (or EIM) of $O(\log n)$ with high probability.
Sparseness and Interference

- Does sparseness necessarily imply low interference?
- No! [BvRWZ04]
- Performance of topologies based on proximity graphs (e.g., Yao graph) may be bad

Nearest neighbor forest

\[ \text{NIM} = \Omega(n) \]

Optimal

\[ \text{NIM} = O(1) \]
Low-Interference Spanners

- Goal: Determine a topology that has distance-stretch of at most $t$, and has minimum NIM among all such topologies [BvRWZ04]
- Let $T$, initially empty, be current topology
- Process edges in decreasing order of $I(\cdot)$
- For current edge $e = (u,v)$:
  - Until stretch-$t$ path between $u$ and $v$ in $T$, repeatedly add edge with least $I(\cdot)$ to $T$

- NIM-optimal
- Amenable to a distributed implementation:
  - $L(e)$ computable locally
  - Existence of stretch-$t$ path can be determined by a search within a local neighborhood
Minimum Energy Broadcast Routing

- Given a set of nodes in the plane
- Goal: Broadcast from a source to all nodes
- In a single step, a node may broadcast within a range by appropriately adjusting transmit power
Minimum Energy Broadcast Routing

- Energy consumed by a broadcast over range $r$ is proportional to $r^\alpha$
- Problem: Compute the sequence of broadcast steps that consume minimum total energy
- Centralized solutions
- NP-complete [ZHE02]
Three Greedy Heuristics

- In each tree, power for each node proportional to $\alpha$th exponent of distance to farthest child in tree
- **Shortest Paths Tree (SPT)** [WNE00]
- **Minimum Spanning Tree (MST)** [WNE00]
- **Broadcasting Incremental Power (BIP)** [WNE00]
  - “Node” version of Dijkstra’s SPT algorithm
  - Maintains an arborescence rooted at source
  - In each step, add a node that can be reached with minimum increment in total cost
- SPT is $\Omega(n)$-approximate, MST and BIP have approximation ratio of at most 12 [WCLF01]
Lower Bound on SPT

- Assume \((n - 1)/2\) nodes per ring
- Total energy of SPT: 
  \( (n - 1)(\varepsilon^\alpha + (1 - \varepsilon)^\alpha)/2 \)
- Optimal solution:
  - Broadcast to all nodes
  - Cost 1
- Approximation ratio 
  \( \Omega(n) \)
Performance of the MST Heuristic

• Weight of an edge \((u,v)\) equals \(d(u,v)\alpha\)
• MST for these weights same as Euclidean MST
  – Weight is an increasing function of distance
  – Follows from correctness of Prim’s algorithm
• Upper bound on total MST weight
• Lower bound on optimal broadcast tree
Weight of Euclidean MST

- What is the best upper bound on the weight of an MST of points located in a unit disk?
  - In [6,12]!

- Dependence on $\alpha$
  - $\alpha < 2$ : in the limit $\infty$
  - $\alpha \geq 2$ : bounded
Structural Properties of MST

- Empty
- Disjoint

- $\geq 60^\circ$
- $\leq$ radius
Upper Bound on Weight of MST

- Assume $ \alpha = 2$
- For each edge $e$, its diamond accounts for an area of at least
  \[ \frac{|e|^2}{2\sqrt{3}} \]
- Total area accounted for is at most $\pi(2/\sqrt{3})^2 = 4\pi/3$
- MST cost equals $\sum_e |e|^2$
- Claim also applies for $\alpha > 2$

\[ \sum_e \frac{|e|^2}{2\sqrt{3}} \leq \frac{4\pi}{3} \approx 14.51 \]
Lower Bound on Optimal

- For a non-leaf node \( u \), let \( r_u \) denote the distance to farthest child.
- Total cost is \( \sum_u r_u^\alpha \).
- Replace each star by an MST of the points.
- Cost of resultant graph at most \( 12 \sum_u r_u^\alpha \).

MST has cost at most 12 times optimal.
Performance of the BIP Heuristic

• Let $v_1, v_2, ..., v_n$ be the nodes added in order by BIP.

• Let $H$ be the complete graph over the same nodes with the following weights:
  - Weight of edge $(v_{i-1}, v_i)$ equals incremental power needed to connect $v_i$.
  - Weight of remaining edges same as in original graph $G$.

• MST of $H$ same as BIP tree $B$.

\[
\text{Cost}_G(B) = \text{Cost}_H(B) \\
\leq \text{Cost}_H(T) \\
\leq \text{Cost}_G(T)
\]
Spanning Trees in Ad Hoc Networks

- Forms a backbone for routing
- Forms the basis for certain network partitioning techniques
- Subtrees of a spanning tree may be useful during the construction of local structures
- Provides a communication framework for global computation and broadcasts
Arbitrary Spanning Trees

• A designated node starts the “flooding” process
• When a node receives a message, it forwards it to its neighbors the first time
• Maintain sequence numbers to differentiate between different ST computations
• Nodes can operate asynchronously
• Number of messages is $O(m)$; worst-case time, for synchronous control, is $O(Diam(G))$
Minimum Spanning Trees

- The basic algorithm [GHS83]
  - $O(m + n \log n)$ messages and $O(n \log n)$ time
- Improved time and/or message complexity [CT85, Gaf85, Awe87]
- First sub-linear time algorithm [GKP98]
  - $O(D + n^{0.61} \log^* n)$
- Improved to $O(D + \sqrt{n} \log^* n)$
- Taxonomy and experimental analysis [FM96]
- $\Omega(D + \sqrt{n} / \log n)$ lower bound [PR00]
The Basic Algorithm

- Distributed implementation of Borouvka’s algorithm from 1926
- Each node is initially a fragment
- Fragment $F_1$ repeatedly finds a min-weight edge leaving it and attempts to merge with the neighboring fragment, say $F_2$
  - If fragment $F_2$ also chooses the same edge, then merge
  - Otherwise, we have a sequence of fragments, which together form a fragment
Subtleties in the Basic Algorithm

• All nodes operate asynchronously
• When two fragments are merged, we should “relabel” the smaller fragment.
• Maintain a level for each fragment and ensure that fragment with smaller level is relabeled:
  – When fragments of same level merge, level increases; otherwise, level equals larger of the two levels
• Inefficiency: A large fragment of small level may merge with many small fragments of larger levels
Asymptotic Improvements to the Basic Algorithm

- The fragment level is set to log of the fragment size \([CT85, Gaf85]\)
  - Reduces running time to \(O(n \log^* n)\)
- Improved by ensuring that computation in level \(\ell\) fragment is blocked for \(O(2^\ell)\) time
  - Reduces running time to \(O(n)\)
A Sublinear Time Distributed Algorithm

- All previous algorithms perform computation over fragments of MST, which may have $\Omega(n)$ diameter
- Two phase approach [GKP98]
  - Controlled execution of the basic algorithm, stopping when fragment diameter reaches a certain size
  - Execute an edge elimination process that requires processing at the central node of a BFS tree
- Running time is $O(Diam(G) + \sqrt{n}\log^* n)$
- Requires a fair amount of synchronization
Open Problems in Topology Control

• Connectivity:
  – Energy-optimal bounded-hops topology
  – Is the energy-spanner variant of the Yao graph a spanner?

• Interference number:
  – What is the complexity of optimizing the edge interference number?

• Minimum energy broadcast routing:
  – Best upper bound on the cost of an MST in Euclidean space
  – Local algorithms

• Tradeoffs among congestion, dilation, and energy consumption [MadHSVVG02]
Capacity of Ad Hoc Networks
The Attenuation Model

- Path loss:
  - Ratio of received power to transmitted power
  - Function of medium properties and propagation distance
- If $P_R$ is received power, $P_T$ is the transmitted power, and $d$ is distance

\[ P_R = O\left(\frac{P_T}{d^\alpha}\right) \]

- where $\alpha$ ranges from 2 to 4
Interference Models

• In addition to path loss, bit-error rate of a received transmission depends on:
  – Noise power
  – Transmission powers and distances of other transmitters in the receiver’s vicinity

• Two models [GK00]:
  – Physical model
  – Protocol model
The Physical Model

• Let \( \{X_i\} \) denote set of nodes that are simultaneously transmitting

• Let \( P_i \) be the transmission power of node \( X_i \)

• Transmission of \( X_i \) is successfully received by \( Y \) if:

\[
\frac{P_i}{d(X_i, Y)^\alpha} \geq \beta
\]

\[
N + \sum_{k \neq i} \frac{P_k}{d(X_k, Y)^\alpha}
\]

• \( \beta \) is the min signal-interference ratio (SIR)
The Protocol Model

- Transmission of $X_i$ is successfully received by $Y$ if for all $k$

$$\frac{P_i}{d(X_i,Y)^\alpha} \geq (1 + \Delta) \frac{P_k}{d(X_k,Y)^\alpha}$$

- where $\Delta$ is a protocol-specified guard-zone to prevent interference
Measures for Network Capacity

• **Throughput capacity [GK00]:**
  - Number of successful packets delivered per second
  - Dependent on the traffic pattern
  - *What is the maximum achievable, over all protocols, for a random node distribution and a random destination for each source?*

• **Transport capacity [GK00]:**
  - Network transports one *bit-meter* when one bit has been transported a distance of one meter
  - Number of bit-meters transported per second
  - *What is the maximum achievable, over all node locations, and all traffic patterns, and all protocols?*
Transport Capacity: Assumptions

• $n$ nodes are arbitrarily located in a unit disk
• We adopt the protocol model
  – Each node transmits with same power
  – Condition for successful transmission from $X_i$ to $Y$: for any $k$

\[ d(X_i,Y) \geq (1+\delta)d(X_k,Y) \]

• Transmissions are in synchronized slots
Transport Capacity: Lower Bound

• What configuration and traffic pattern will yield the highest transport capacity?
• Distribute $n/2$ senders uniformly in the unit disk
• Place $n/2$ receivers just close enough to senders so as to satisfy threshold $2/n^2$
Transport Capacity: Lower Bound

sender
receiver
Transport Capacity: Lower Bound

- Sender-receiver distance is $\Omega(1/\sqrt{n})$
- Assuming channel bandwidth $W$, transport capacity is $\Omega(W\sqrt{n})$
- Thus, transport capacity per node is $\Omega\left(\frac{W}{\sqrt{n}}\right)$
Transport Capacity: Upper Bound

- For any slot, we will upper bound the total bit-meters transported.
- For a receiver $j$, let $r_j$ denote the distance from its sender.
- If channel capacity is $W$, then bit-meters transported per second is

$$\leq W \left( \sum_{\text{receiver } j} r_j \right)$$
Transport Capacity: Upper Bound

• Consider two successful transmissions in a slot: $i \rightarrow j$ and $k \rightarrow \ell$

\[
\begin{align*}
    d(j, \ell) \geq (1 + \delta)d(i, j) - d(\ell, k) \\
    d(\ell, j) \geq (1 + \delta)d(k, \ell) - d(i, j) \\
    d(\ell, j) \geq \frac{\delta}{2}(d(i, j) + d(k, \ell))
\end{align*}
\]
Transport Capacity: Upper Bound

- Balls of radii $\Theta(r_j)$ around $j$, for all $j$, are disjoint

\[
\begin{align*}
\sum_j r_j^2 &= O(1) \\
(\sum_j r_j)^2 &= O(h) = O(n) \\
\sum_j r_j &= O(\sqrt{n})
\end{align*}
\]

- So bit-meters transported per slot is $O(W\sqrt{n})$
Throughput Capacity of Random Networks

- The throughput capacity of an $n$-node random network is
  \[
  \Theta\left(\frac{W}{\sqrt{n \log n}}\right)
  \]

- There exist constants $c$ and $c'$ such that
  \[
  \lim_{n \to \infty} \Pr\left[c \frac{W}{\sqrt{n \log n}} \text{ is feasible} \right] = 1
  \]
  \[
  \lim_{n \to \infty} \Pr\left[c' \frac{W}{\sqrt{n \log n}} \text{ is feasible} \right] = 0
  \]
Implications of Analysis

- **Transport capacity:**
  - Per node transport capacity decreases as \( \frac{1}{\sqrt{n}} \)
  - Maximized when nodes transmit to neighbors

- **Throughput capacity:**
  - For random networks, decreases as \( \frac{1}{\sqrt{n \log n}} \)
  - Near-optimal when nodes transmit to neighbors

- Designers should focus on small networks and/or local communication
Remarks on Capacity Analysis

- Similar claims hold in the physical model as well
- Results are unchanged even if the channel can be broken into sub-channels
- More general analysis:
  - Power law traffic patterns [LBD+03]
  - Hybrid networks [KT03, LLT03, Tou04]
  - Asymmetric scenarios and cluster networks [Tou04]
Asymmetric Traffic Scenarios

• Number of destinations smaller than number of sources
  - \( n^d \) destinations for \( n \) sources; \( 0 < d \leq 1 \)
  - Each source picks a random destination

• If \( 0 < d < 1/2 \), capacity scales as \( n^d \)
• If \( 1/2 < d \leq 1 \), capacity scales as \( n^{1/2} \)

• [Tou04]
Power Law Traffic Pattern

• Probability that a node communicates with a node \(x\) units away is

\[
p(x) = \frac{x^\alpha}{\int_1^x t^\alpha dt}
\]

  – For large negative \(\alpha\), destinations clustered around sender
  – For large positive \(\alpha\), destinations clustered at periphery

• As \(\alpha\) goes from \(< -2\) to \(> -1\), capacity scaling goes from \(O(1)\) to \(O(1/\sqrt{n})\) \([\text{LBD}^+03]\)
Relay Nodes

• Offer improved capacity:
  – Better spatial reuse
  – Relay nodes do not count in $n$
  – Expensive: addition of $kn$ nodes as pure relays yields less than $\sqrt{k} + 1$-fold increase

• Hybrid networks: $n$ wireless nodes and $n^d$ access points connected by a wired network
  – $0 < d < 1/2$: No asymptotic benefit
  – $1/2 < d \leq 1$: Capacity scaling by a factor of $n^d$
Mobility and Capacity

- A set of $n$ nodes communicating in random source-destination pairs
- Expected number of hops is $\sqrt{n}$
- Necessary $\sqrt{n}$ scaling down of capacity
- Suppose no tight delay constraint
- Strategy: packet exchanged when source and destination are near each other
  - Fraction of time two nodes are near one another is $1/n$
- Refined strategy: Pick random relay node (a la Valiant) as intermediate destination [GT01]
- Constant scaling assuming that stationary distribution of node location is uniform
Open Problems in Capacity Analysis

• Detailed study of impact of mobility
  – [GT01] study is “optimistic”

• Capacity of networks with beam-forming antennas [Ram98]
  – Omnidirectional antennas incur a tradeoff between range and spatial reuse
  – A beam-forming antenna can transmit/receive more energy in preferred transmission and reception directions

• Capacity of MIMO systems
Algorithms for Sensor Networks
Why are Sensor Networks Special?

- Very tiny nodes
  - 4 MHz, 32 KB memory
- More severe power constraints than PDAs, mobile phones, laptops
- Mobility may be limited, but failure rate higher
- Usually under one administrative control
- A sensor network gathers and processes specific kinds of data relevant to application
- Potentially large-scale networks comprising of thousands of tiny sensor nodes
Focus Problems

• Medium-access and power control:
  – Power saving techniques integral to most sensor networks
  – Possibility of greater coordination among sensor nodes to manage channel access

• Synchronization protocols:
  – Many MAC and application level protocols rely on synchronization

• Query and stream processing:
  – Sensor network as a database
  – Queries issued at certain gateway nodes
  – Streams of data being generated at the nodes by their sensors
  – Need effective in-network processing and adequate networking support
MAC Protocols for Sensor Networks

• Contention-Based:
  – Random access protocols
  – IEEE 802.11 with power saving methods

• Scheduling-Based:
  – Assign transmission schedules (sleep/awake patterns) to each node
  – Variants of TDMA

• Hybrid schemes
Proposed MAC Protocols

• PAMAS [SR98]:
  – Contention-based access
  – Powers off nodes that are not receiving or forwarding packets
  – Uses a separate signaling channel

• S-MAC [YHE02]:
  – Contention-based access

• TRAMA [ROGLA03]:
  – Schedule- and contention-based access

• Wave scheduling [TYD+04]:
  – Schedule- and contention-based access

• Collision-minimizing CSMA [TJB]:
  – For bursty event-based traffic patterns
S-MAC

• Identifies sources of energy waste [YHE03]:
  – Collision
  – Overhearing
  – Overhead due to control traffic
  – Idle listening

• Trade off latency and fairness for reducing energy consumption

• Components of S-MAC:
  – A periodic sleep and listen pattern for each node
  – Collision and overhearing avoidance
S-MAC: Sleep and Listen Schedules

• Each node has a sleep and listen schedule and maintains a table of schedules of neighboring nodes

• Before selecting a schedule, node listens for a period of time:
  – If it hears a schedule broadcast, then it adopts that schedule and rebroadcasts it after a random delay
  – Otherwise, it selects a schedule and broadcasts it

• If a node receives a different schedule after selecting its schedule, it adopts both schedules

• Need significant degree of synchronization
S-MAC: Collision and Overhearing Avoidance

• Collision avoidance:
  – Within a listen phase, senders contending to send messages to same receiver use 802.11

• Overhearing avoidance:
  – When a node hears an RTS or CTS packet, then it goes to sleep
  – All neighbors of a sender and the receiver sleep until the current transmission is over
TRAMA

• Traffic-adaptive medium adaptive protocol [ROGLA03]
• Nodes synchronize with one another
  – Need tight synchronization
• For each time slot, each node computes an MD5 hash, that computes its priority

\[ p(u, t) = MD5(u \oplus t) \]

• Each node is aware of its 2-hop neighborhood
• With this information, each node can compute the slots it has the highest priority within its 2-hop neighborhood
TRAMA: Medium Access

- Alternates between random and scheduled access
- **Random access:**
  - Nodes transmit by selecting a slot randomly
  - Nodes can only join during random access periods
- **Scheduled access:**
  - Each node computes a schedule of slots (and intended receivers) in which will transmit
  - This schedule is broadcast to neighbors
  - A free slot can be taken over by a node that needs extra slots to transmit, based on priority in that slot
  - Each node can determine which slots it needs to stay awake for reception
Wave Scheduling

• Motivation:
  – Trade off latency for reduced energy consumption
  – Focus on static scenarios

• In S-MAC and TRAMA, nodes exchange local schedules

• Instead, adopt a global schedule in which data flows along horizontal and vertical “waves”

• Idea:
  – Organize the nodes according to a grid
  – Within each cell, run a leader election algorithm to periodically elect a representative (e.g., GAF [XHE01])
  – Schedule leaders’ wakeup times according to positions in the grid
Wave Scheduling: A Simple Wave
Wave Scheduling: A Pipelined Wave
Wave Scheduling: Message Delivery

• When an edge is scheduled:
  – Both sender and receiver are awake
  – Sender sends messages for the duration of the awake phase
  – If sender has no messages to send, it sends an NTS message (Nothing-To-Send), and both nodes revert to sleep mode

• Given the global schedule, route selection is easy
  – Depends on optimization measure of interest
  – Minimizing total energy consumption requires use of shortest paths
  – Minimizing latency requires a (slightly) more complex shortest-paths calculation
Collision-Minimizing CSMA

- Focus on bursty event-based traffic [TJB]
  - Room monitoring: A fire triggers a number of redundant temperature and smoke sensors
  - Power-saving: When a node wakes up and polls, all coordinators within range may respond

- **Goal**: To minimize latency

- **Scenario**:
  - N nodes contend for a channel
  - There are K transmission slots
  - Sufficient for any one of them to transmit successfully
  - No collision detection: collisions may be expensive since data packet transmission times may be large

- **Subgoal**: To maximize the probability of a collision-free transmission
Collision-Free Transmission

- Probability of transmission varies over slots
- Probability of successful collision-free transmission in $K$ slots

$$NP_1(1 - p_1)^{N-1} + NP_2(1 - p_1 - p_2)^{N-1} + \ldots + NP_{K-1}(1 - p_1 - p_2 - \ldots - p_{K-1})^{N-1}$$

$$= N \sum_{s=1}^{K-1} p_s (1 - \sum_{r=1}^{s} p_r)^{N-1}$$

- Can calculate probability vector $p^*$ that optimizes above probability
- MAC protocol: CSMA/$p^*$
Synchronization in Sensor Networks
Synchronization in Sensor Networks

- Sensor data fusion
- Localization
- Coordinated actuation
  - Multiple sensors in a local area make a measurement
- At the MAC level:
  - Power-saving duty cycling
  - TDMA scheduling
Synchronization in Distributed Systems

- Well-studied problem in distributed computing
- **Network Time Protocol (NTP)** for Internet clock synchronization [Mil94]
- Differences: For sensor networks
  - Time synchronization requirements more stringent (\(\mu\)s instead of ms)
  - Power limitations constrain resources
  - May not have easy access to synchronized global clocks
Network Time Protocol (NTP)

- **Primary** servers (S1) synchronize to national time standards
  - Satellite, radio, modem
- **Secondary** servers (S2, ...) synchronize to primary servers and other secondary servers
  - Hierarchical subnet

http://www.ntp.org
Measures of Interest

- **Stability**: How well a clock can maintain its frequency
- **Accuracy**: How well it compares with some standard
- **Precision**: How precisely can time be indicated
- **Relative measures**:
  - **Offset**: Difference between times of two clocks
  - **Skew**: Difference between frequencies of two clocks
Synchronization Between Two Nodes

- A sends a message to B; B sends an ack back
- A calculates clock drift and synchronizes accordingly

\[ \Delta = \frac{(T_2 - T_1) - (T_4 - T_3)}{2} \]

\[ d = \frac{(T_2 - T_1) + (T_4 - T_3)}{2} \]

\( \Delta \): Measured offset

\( d \): Propagation delay
Error Analysis

\[
\Delta = \frac{(T_2 - T_1) - (T_4 - T_3)}{2}
\]
\[
d = \frac{(T_2 - T_1) + (T_4 - T_3)}{2}
\]

- $S_A$: Sender time at A
- $R_A$: Receiver time at A
- $P_{A\rightarrow B}$: Prop. time for A→B

$S^{UC}$: $S_A - S_B$
$R^{UC}$: $R_B - R_A$
$P^{UC}$: $P_{A\rightarrow B} - P_{B\rightarrow A}$

Error = \[
\frac{S^{UC} + R^{UC} + P^{UC}}{2}
\]
Sources of Synchronization Error

- **Non-determinism** of processing times
- **Send time**:  
  - Time spent by the sender to construct packet; application to MAC
- **Access time**:  
  - Time taken for the transmitter to acquire the channel and exchange any preamble (RTS/CTS): MAC
- **Transmission time**: MAC to physical
- **Propagation time**: physical
- **Reception time**: Physical to MAC
- **Receive time**:  
  - Time spent by the receiver to reconstruct the packet; MAC to application
Sources of Synchronization Error

- **Sender time** = send time + access time + transmission time
  - Send time variable due to software delays at the application layer
  - Access time variable due to unpredictable contention
- **Receiver time** = receive time + reception time
  - Reception time variable due to software delays at the application layer
- **Propagation time** dependent on sender-receiver distance
  - Absolute value is negligible when compared to other sources of packet delay
  - If node locations are known, these times can be explicitly accounted for
Two Approaches to Synchronization

• Sender-receiver:
  – Classical method, initiated by the sender
  – Sender synchronizes to the receiver
  – Used in NTP
  – **Timing-sync Protocol for Sensor Networks (TPSN)** [GKS03]
• Receiver-based:
  – Takes advantage of broadcast facility
  – Two receivers synchronize with each other based on the reception times of a reference broadcast
  – **Reference Broadcast Synchronization (RBS)** [EGE02]
**TPSN**

- **Time stamping done at the MAC layer**
  - Eliminates send, access, and receive time errors

- **Creates a hierarchical topology**

- **Level discovery**:
  - Each node assigned a level through a broadcast

- **Synchronization**:
  - Level $i$ node synchronizes to a neighboring level $i-1$ node using the sender-receiver procedure
Reference Broadcast Synchronization

• Motivation:
  – Receiver time errors are significantly smaller than sender time errors
  – Propagation time errors are negligible
  – The wireless sensor world allows for broadcast capabilities

• Main idea:
  – A reference source broadcasts to multiple receivers (the nodes that want to synchronize with one another)
  – Eliminates sender time and access time errors
Reference Broadcast Synchronization

- Simple form of RBS:
  - A source broadcasts a reference packet to all receivers
  - Each receiver records the time when the packet is received
  - The receivers exchange their observations

- General form:
  - Several executions of the simple form

- For each receiver $j$, receiver $i$ derives an estimate of $\Delta_{ij}$

\[
T_i : \text{Receive time at } i \\
\Delta_{ij} = T_j - T_i \\
\Delta_{ij} = \frac{1}{m} \sum_{k=1}^{m} (T_{kj} - T_{ki})
\]
Reference Broadcast Synchronization

- Clock skew:
  - Averaging assumes $s_{ij}$ equals 1
  - Find the best fit line using least squares linear regression
  - Determines $s_{ij}$ and $\Delta_{ij}$

- Pairwise synchronization in multihop networks:
  - Connect two nodes if they were synchronized by same reference
  - Can add drifts along path
  - But which path to choose?
  - Assign weight equal to root-mean square in regression
  - Select path of min-weight

$t_j = t_i s_{ij} + \Delta_{ij}$
Pairwise and Global Synchronization

• Global consistency:
  – Converting times from $i$ to $j$ and then $j$ to $k$ should be same as converting times from $i$ to $k$

  \[ S_{ik} = S_{ij}S_{jk} \]
  \[ \Delta_{ik} = \Delta_{ij}S_{jk} + \Delta_{jk} \]

• Optimal precision:
  – Find an unbiased estimate for each pair $(s_{ij}, \Delta_{ij})$ with minimum variance

• [KEES03]
Consistency and Optimal Precision

• Min-variance pairwise synchronizations are globally consistent!
• Maximally likely set of offset assignments yield minimum variance synchronizations!
• Flow in resistor networks
  – Bipartite graph connecting the receivers with the sources
  – Resistance of each edge equal to the variance of the error corresponding to that source-receiver pair
  – Min-variance is effective resistance
  – Estimator can be obtained from the current flows
Algorithmic Support for Query Processing in Sensor Networks
The Sensor Network as a Database

• From the point of view of the user, the sensor network generates data of interest to the user
• Need to provide the abstraction of a database
  – High-level interfaces for users to collect and process continuous data streams
• TinyDB [MFHH03], Cougar [YG03]
  – Users specify queries in a declarative language (SQL-like) through a small number of gateways
  – Query flooded to the network nodes
  – Responses from nodes sent to the gateway through a routing tree, to allow in-network processing
  – Especially targeted for aggregation queries
• Directed diffusion [IGE00]
  – Data-centric routing: Queries routed to specific nodes based on nature of data requested
Classification of Queries

• Long-running vs ad hoc
  – Long-running: Issued once and require periodic updates
  – Ad hoc: Require one-time response

• Temporal:
  – Historical
  – Present
  – Future: e.g., trigger queries

• Nature of query operators
  – Aggregation vs. general

• Spatial vs. non-spatial
Processing of Aggregate Queries

• Aggregation query \( q: S \rightarrow \mathbb{R} \)
  - Sum, minimum, median, etc.
• Queries flooded within the network
• An aggregation tree is obtained
• Query results propagated and aggregated up the tree
• Aggregation tree selection
• Multi-query optimization
Multi-Query Optimization

- Given:
  - An aggregation tree
  - Query workload
  - Update probabilities of sensors
- Determine an aggregation procedure that minimizes communication complexity:
- Push vs. pull:
  - When should we proactively send up sensor data?
- Problem space [DGR\textsuperscript{+}03]:
  - Deterministic queries, deterministic updates
  - Deterministic queries, probabilistic updates
  - Probabilistic queries, deterministic updates
  - Probabilistic queries, probabilistic updates
Multi-Query Optimization

- Two queries: A+B and A+C, each with probability $1-\epsilon$
- $\epsilon=0$: Proactively forward each sensor reading up the tree
- $\epsilon$ nearly 1: Let parent pull information
- Intermediate case depends on the ratio of result/query message sizes
Multi-Query Optimization

- $q > 2\varepsilon r$:
  - Push on every edge
- $\varepsilon r < q < 2\varepsilon r$:
  - Pull on $(I,R)$
  - Push on other edges
- $\varepsilon^2 r < q < \varepsilon r$:
  - Push on $(A,I)$
  - Pull on other edges
- $q < \varepsilon^2 r$:
  - Pull on every edge

- Optimizations:
  - Send results of a basis of the projected query set along an edge
Aggregation Tree Selection

• Given:
  – An aggregation procedure for a fixed aggregation tree
  – Query workload: e.g., probability for each query
  – Probability of each sensor update
• Determine an aggregation tree that minimizes the total energy consumption
• Clearly NP-hard
  – Minimum Steiner tree problem is a special case
• Approximation algorithms for interesting special cases
Approximations for Special Cases

- **Individual queries:**
  - Any approximation to minimum Steiner tree suffices
  - MST yields 2-approximation, improved approximations known

- **Universal trees [JLN+04]:**
  - There exists a single tree whose subtree induced by any query is within polylog(n) factor of the optimum
  - Unknown query, deterministic update

- **A single aggregation tree for all concave aggregation functions [GE03]:**
  - All sensor nodes participate
  - The aggregation operator is not known a priori, but satisfies a natural concaveness property
  - There exists a single tree that achieves an $O(\log n)$-approximation
Simultaneous Optimization for Concave Aggregation Functions

\[ f : \mathbb{Z} \mapsto \mathbb{R} \]

\[ f \text{ and } f' \text{ are nondecreasing} \]

- A function that gives the size of the aggregated data given the number of items being aggregated
- Binary aggregation method:
  - Find a min-cost matching
  - For each pair, select one node at random and make it the parent of the other
  - Repeat the procedure with the parents until have exactly one node
Simultaneous Optimization for Concave Aggregation Functions

\[ f : \mathbb{Z} \mapsto \mathbb{R} \]

\( f \) and \( f' \) are nondecreasing

• Independent of the function \( f \)
• Binary aggregation method yields an \( O(\log n) \) approximation for any function
  – \( n \) is the number of nodes
• Can be derandomized to yield the same asymptotic result
Data-Centric Storage and Routing

- Need to ensure the query originator rendezvous with nodes containing matching data
  - Flooding queries is expensive

- Data-centric storage [RKY+02]:
  - Designated collection of nodes storing data items matching a certain predicate
  - These nodes can also perform in-network processing to compute intermediate values

- Data-centric routing [RKY+02]:
  - Gateway determines node(s) storing data matching a particular predicate
  - Routes query to these nodes using unicast or multicast
Open Problems in Sensor Network Algorithms

• Topology control:
  – Aggregation tree selection
  – Scheduling node and edge activations for specific communication patterns

• Multi-query optimization:
  – Need to address general (non-aggregate) queries
  – Related to work in distributed databases; energy consumption a different performance measure
Outline

1. Medium Access Control
2. Power Control
3. Topology Control
4. Fundamental limits of ad hoc networks
5. Sensor Network Protocols

Application

Fundamental limits of ad hoc networks