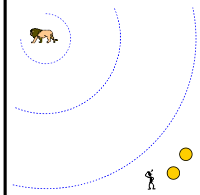
CS551
Routing in Sensor Networks
[Intanagonwiwat00a]

Bill Cheng

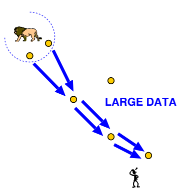
http://merlot.usc.edu/cs551-f12

Remote Sensing



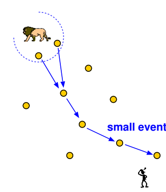
- Remote approach
  - few, large, expensive sensors are far from phenomena
  - they use complex algorithms to factor out noise
  - e.g. satellite-based sensing
- Problem:
  - SNR decreases rapidly with distance
    - noise limits performance (resolution)

Sensor Arrays



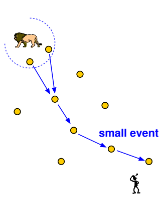
- Centralized approach
  - some, cheap?, dumb sensors are close to phenomena
  - collected data is sent to process at smart, expensive *central node* (or nodes)
- Problem:
  - bandwidth requirements high
    - can't use wireless
  - difficult to deploy

Future Wireless Sensor Networks



- Distributed approach
  - many small, smart, cheap sensors close to phenomena
  - nodes will have processing capability
- Why wireless?
  - deployment is trivial
  - also enables large numbers of nodes
    - dense sensing small event

Future Wireless Sensor Networks

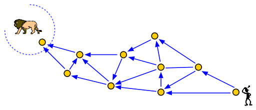


- Challenges
  - for ease of deployment, must also be battery operated
  - energy management now becomes an important issue
  - energy cost of communication outweighs other costs in the system
    - energy required to transmit 1 bit 10m is same as energy for 1000 processor operations
- How do we overcome this?
  - *process data within the network*
    - data must be *self-describing*
    - user *names data*, not node

Directed Diffusion

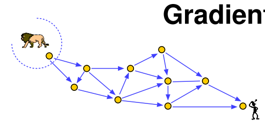
- Users express interest in data (becoming *sink*)
  - specified by *attributes*, not IP address
- Sink sends out *interests*
  - by default: flooded through network
  - could use attributes for help (geography)
  - could use cached old routes
- *Sources* reply to interests with data
  - first, send *exploratory (low rate)* data
  - flooded on return paths
- *Sink* reinforces a path
  - sets up *reinforced* path
  - *non-exploratory (high rate)* data only follows reinforced paths

Interest Propagation



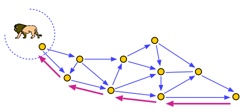
- Initial interest specifies low data rate as exploratory
  - The desired data rate will be achieved by reinforcement
- After receiving an interest, the node creates states and re-sends to a subset of its neighbors
  - Flood the interest
  - Direct interest or limit scope using GPS info
  - Direct interest using route history

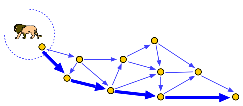
Exploratory Data Propagation and Gradient Establishment



- Sensor's first data is exploratory (low-rate data)
- Sent throughout network, establishing gradients
  - map attributes to next hop at each node in network
  - nodes have multiple gradients

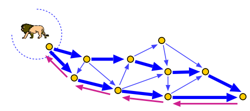
Reinforcement

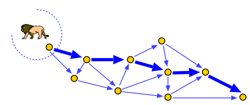




- Sink reinforces some path to get high rate or non-exploratory data
- Each hop propagates reinforcement back to sources
- Which link to reinforce?
  - default: lowest latency
  - alternatives: maximum remaining energy, or greedy tree

Negative Reinforcement



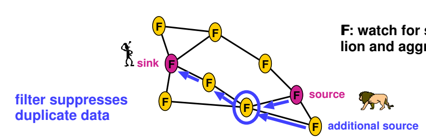


- Should detect and prune unnecessary paths
  - (paths that send the same info)
- Negative reinforcement
  - implicit negative reinforcement (just let gradient time out)
  - explicit negative reinforcement

Naming and attributes

- IP has node address and ports and DNS and URLs
  - needs resource discovery
    - humans use search engines
    - embedded systems use something like Jini
- Directed diffusion uses *attribute-based naming*
  - sinks subscribe to sensor EQ acoustic; target IS lions; lat GT 100; lat LT 101; long GT 43; long LT 44
  - sensors publish sensor IS acoustic; target EQ *; lat IS 100.5; long IS 43.05

Filters



F: watch for sensor data type lion and aggregate it

- Support *app-specific*, *in-network processing*
  - duplicate suppression
  - aggregation
  - collaborative signal processing
  - caching, etc.
- Mechanism:
  - assume filters are pre-deployed in net
  - match on attributes
  - filter can take any action (send new msgs, suppress messages, etc.)

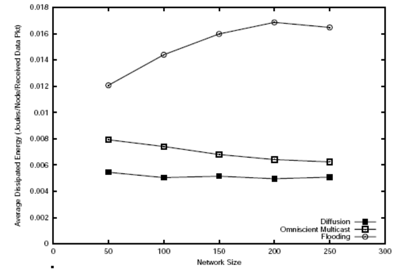
Differences from Traditional Networking

- Neighbor-to-neighbor communication (not end-to-end)
- Localized algorithms
  - no globally unique IDs
  - no explicit global information (routing tables)
- Data and queries are named independently from their producers
- In-network processing
- Application-specific
  - net-wide attributes (like sensor type or latitude/longitude)
  - app-specific data aggregation

Energy Scaling

- Good performance even as number of nodes grows

[Intanagonwiwat00a, figure 4a]



- Diffusion uses less energy than omniscient multicast (optimal)
  - how?
    - duplicate suppression (cannot do it in IP networks)
    - diffusion does in-network aggregation

In-Network Processing

[Intanagonwiwat00a figure 6b]

- Duplicate suppression is critical to diffusion
- Shows the importance of app-specific in-network processing

Critique

- Looking at sensor networks
  - 100s of embedded, unattended, small devices
- Multi-hop communication
  - coordinate communication between sensors and users
- *Data-centric communication*
  - uses ***in-network processing*** (ex. aggregation)
    - not end-to-end
  - uses ***application-specific routing*** (mixes routing layer and application)
  - uses ***attribute-based names*** (rather than addresses)

Discussion

- Really, a radically new networking architecture
- ...motivated by a new technology
- Articulates the rationale behind this architecture well
  - in-network processing
- Routing scheme a bit too complex