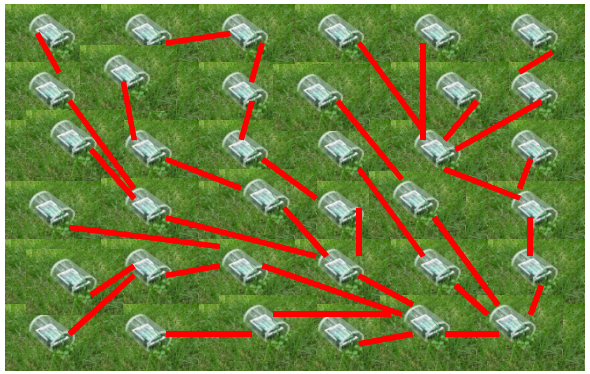
Wireless Sensor Networks
Routing & Security

General Presentation

"While the last 50 years have been dominated by a march to ever more complex computers, the next few decades will see the rise of simple sensors – by the billions."



Wireless Sensor Networks is one of the top **10 Technologies** that will change the World in 21st Century

Digital sensors and actuators

Very inexpensive and can be integrated into silicon

Wireless

Low power inexpensive RF

Silicon integration

Sensor, DSP, CPU, FPGA, wireless, actuators

Huge embedded software on a chip/device

Moving to Smart Dust

(Micro)sensors

Low power, cheap sensors

Sensor module (e.g., acoustic, seismic, image)

A digital processor for signal processing and network protocol functions

Radio for communication

Battery-operated

Sensors monitor environment

Cameras, microphones, physiological sensors, etc.

Gather data for some purpose

Microsensor data limited in range and accuracy

Each node can only gather data from a limited physical area of the environment

Data may be noisy

Data aggregation enables higher quality (less noisy) data to be obtained that gives information about a larger physical area than any individual data signal

Hundreds or thousands of nodes scattered throughout an environment

New wireless networking paradigm

Requires **autonomous operation**

Highly **dynamic** environments

Sensor nodes added/fail

Events in the environment

Distributed computation and communication protocols required

Mica Mote:

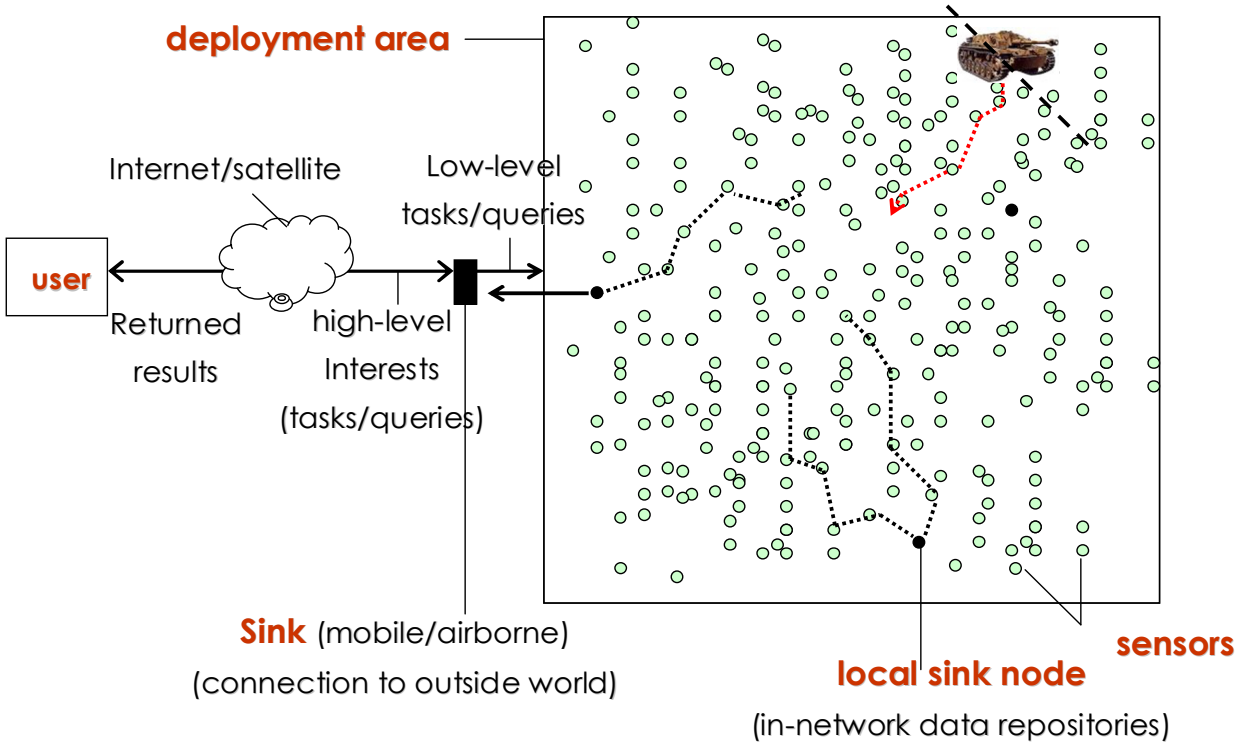
Processor: 4Mhz
Memory: 128KB Flash and 4KB RAM
Radio: 916Mhz and 40Kbits/second.
Transmission range: 100 Feet

TinyOS: operating System: small, open source and energy efficient

Wireless Sensor Networks (WSN)

Numerous sensors, deployed at high density in regions requiring surveillance and monitoring
Sensors are memory as well as energy constrained
Nodes are unattended and resource-constrained; their energy cannot be replenished and network topology is unknown

Paradigm shift:
Functionalities not from the individual nodes but from the network



Possible applications for WSNs

Battle ground surveillance
- Enemy movement (tanks, soldiers, etc)

Environmental monitoring
- Habitat monitoring
- Forrest fire monitoring

Hospital tracking systems
- Tracking patients, doctors, drug administrators

Sensor Network Differences

Traditional wireless networks - Users can update and maintain devices (e.g., each computer maintained by a human)
Wireless sensor networks - May be impossible to update or maintain sensor nodes, due to sheer numbers as well as deployment locations

Traditional wireless networks - Communication between two specific end-users
Wireless sensor networks - Communication data-centric
End-user does not care that the data came from node X, only what the data describes

Traditional wireless networks - Goal: providing high QoS bandwidth efficiency
Wireless sensor networks - Goal: prolonging lifetime of the network
Requires energy conservation
Willing to give up performance in terms of QoS or bandwidth efficiency

Traditional wireless networks - Data are important
Wireless sensor networks - End user does not require all the data
Data from neighboring nodes are highly correlated, making the data redundant
End user typically cares about a higher-level description of events occurring in the environment nodes are monitoring
Network quality often based on quality of aggregate data set rather than individual signals
Traditional wireless networks - Intermediate nodes do not care what the data are
Wireless sensor networks - Application-specific routing to improve Performance

Traditional wireless networks - Nodes operating (mostly) independently
Wireless sensor networks - Sensor network application computation
May need to be distributed throughout network (e.g., localized algorithms that achieve desired global result)
May require hierarchical structure
Enables computation / communication tradeoff
Three processing levels: node, local, and global

Traditional wireless networks - Operate in (mostly) benign environments
Wireless sensor networks - May be deployed in hostile or dangerous territory

Routing in WSN

- Internet (TCP/IP)
  - Routing tables often large
  - Can be updated frequently
- WSN
  - Frequent topology changes
  - Modest local storage
  - Expensive to update frequently
  - => Need local, stateless algorithms where nodes know only immediate neighbors

  Consider the following:
  - The fundamental difference between classical routing and routing for sensor networks is that the separation between address and content of packet **no longer viable**

What does it mean?

  - Network is a system, individual nodes come and go, information sensed by one node can be sensed by another close by

- Data-centric view
  - Routing decision as based not on destination address, but rather on destination attributes and relation to attribute of packet content
  - Information providers and information seekers must be matched using data attributes and not (hard) network address

Examples of Attributes

- Node location
- Types of sensor connected to a node
  - Send a control packet to all nodes that have a light sensor connected
- Certain range of values in certain type of sensed data
  - Get max, min temperature values in from the sensor network
- Pull model
  - Network is queried similar to a database
- Push model
  - Network can initiate flow of information based on events

Routing protocol design issues:

Reactive vs. proactive routing approach:

reactive (on-demand) → a route is not identified unless it is required;

proactive → periodically exchange control messages and provide the required routes instantly.

Tradeoff between the latency in finding the route and the control traffic overhead

Centralized vs. distributed approaches: no centralized control → a distributed routing protocol is preferred

Optimal route: number of hops on the path or the overall link cost

Scalability: routing protocol should scale well in large wireless ad hoc and sensor networks, with rapid topology changes and link failures

Control traffic overhead: must minimize the control traffic overhead required to discover the routes

Efficiency: routes affect the performance of the network → delay, throughput, and energy efficiency

WSN Routing

- Geographic routing (more traditional view)
  - Greedy distance
  - Compass
  - Convex perimeter routing
  - Routing on a curve
  - Energy-minimizing broadcast
- Attribute-based routing (data-centric view)
  - Directed diffusion
  - Rumor routing
  - Geographic hash tables

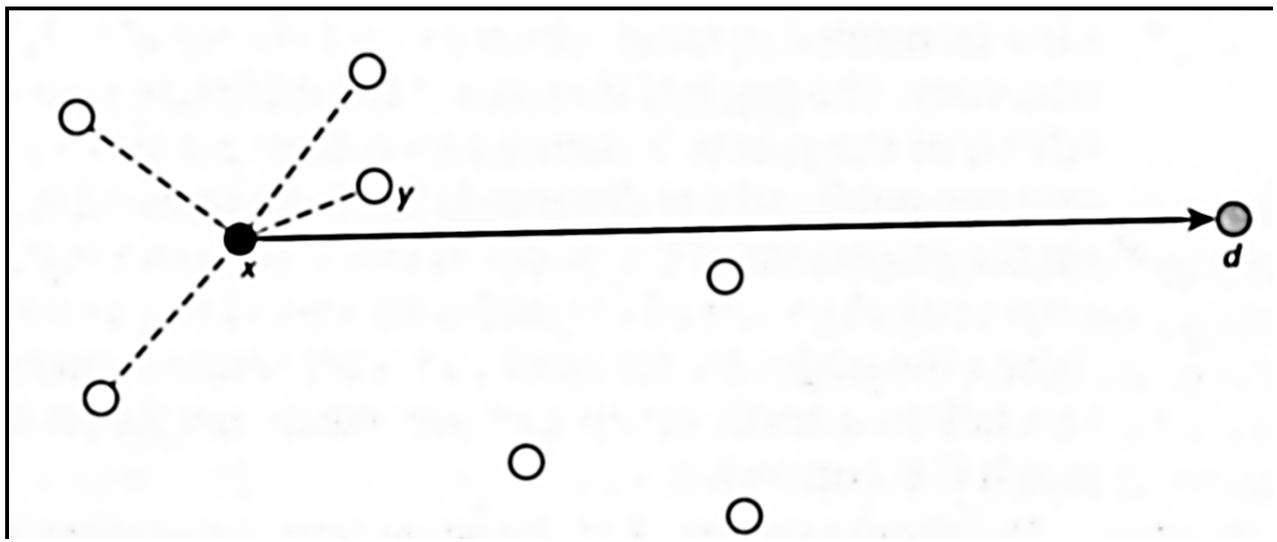
Greedy Distance Routing

Among the neighbors y of x closer to d than x
Pick the one closest to d

Compass Routing

Among the neighbors y of x that make an angle $\angle dxy < \pi$,
Pick one that minimizes the angle $\angle dxy$

The case that Greedy Distance Routing may not always possible:
All neighbors of x are farther from d than x itself
Packet from s →d, encounter a "void" or "hole" in the network
Message stuck at x
If allow the packet to backtrack from x, we may force the packet to oscillate between only two nodes



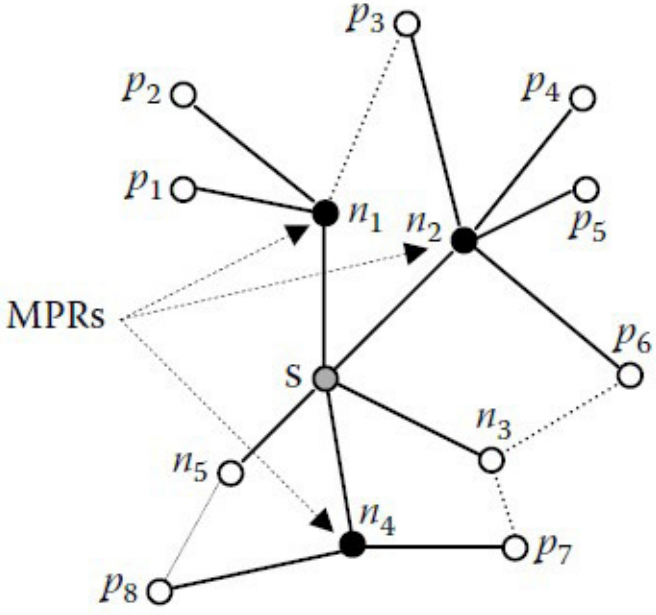
Routing on A Curve

- Specify a curve a packet should follow
- Analytical description of a curve carried by the packet
- Curves may correspond to natural features of the environment where the network is deployed
- Can be implemented in a local greedy fashion that requires no global knowledge
- Curve specified in parametric form C(t)=(x(t),y(t))
  - t – time parameter – could be just relative time
- Each node makes use of nodes trajectory information and neighbor positions to decide the next hop for the packet
- Also called *trajectory-based* routing

Optimized Link State Routing (OLSR) Protocol

Based on classical **link state** routing protocols, adapted for ad hoc mobile networks (group of mobile nodes forming a temporary network, without given infrastructure or centralized administration)
Concept of **MPR** (Multipoint Relay Node) or **RN** (relay Node): selected node for forwarding messages (usually flooding messages)
Selection of MPR: each node (source node: s) from network selects a set of nodes from the total set of adjacent nodes (one hop neighbor), such a manner **to cover as much as possible** the set of two hop neighbors; repeat process for covering all nodes

Start node : s
One hop neighbors: nodes n
Two Hop neighbors: nodes p



Optimized Link State Routing (OLSR) Protocol

Protocol OLSR uses:
Hello messages for neighbor discovery
TC (Topology Control) messages for transport of MPR information

Each MPR advertises info about nodes have selected it
Each node getting a TC message, stores important info about topology (topology table) and the MPR nodes forward info to advertise all nodes from network
Based on this info, routing table is generated, using the least-cost basic algorithm, with hop number as metric

Optimized Energy-Delay Routing (OEDR) Protocol

Proactive distributed routing protocol

Definitions:

- N = set of nodes in the network
- s = source node
- N(s) = one-hop neighbor of s
- $N^2(s)$ = two-hop neighbor of s
- MPR(s) = selected multi-point relay (MPR) set of nodes s
- RT(s) = routing table of node s
- $C_{x,y}$ = Energy(x→y)*Delay(x→y) = cost of link between nodes x and y
- E_x = energy level of node x
- $C^{MPR}_{s, n1, n2} = C_{s,n1}+C_{n1,n2}+(1/E_{n1})$ = Cost for MPR selection of node s, to reach the two-hop n2 neighbor , with one-hop n1 neighbor as the intermediate node
- $Cost_{s,d}$ = Energy-delay (cost) of the entire path between a source s and the destination d

Main steps of protocol to follow:

OEDR 1: Network sensing and Energy-delay (cost) of the entire path between a source and the destination

- Each node in the network periodically generates HELLO messages and transmits to all the one-hop neighbors → fields: transmission time, transmission energy, and the energy level, list of one-hop neighbors, link costs between source and neighbors
- When a HELLO packet is received by a node, it can calculate the transmission delay of the packet as the difference between the transmission time stamped in the packet at the source and the received time at the destination
- The energy used can be calculated as the difference between the transmission energy stamped in the packet and the received energy
- The energy-delay product is computed by each node = cost; recorded in the one-hop neighbor table; the neighbors of the HELLO message originator node (one-hop neighbor) are recorded as two-hop neighbors along with the costs of their links

➔ After receiving a HELLO message each node has: link costs to all the one-hop neighbors, energy levels of the one-hop neighbors, lists of the two-hop neighbors that can be reached from each of the one-hop neighbors, and their link costs

OEDR2: Multipoint Relay (MPR) Selection

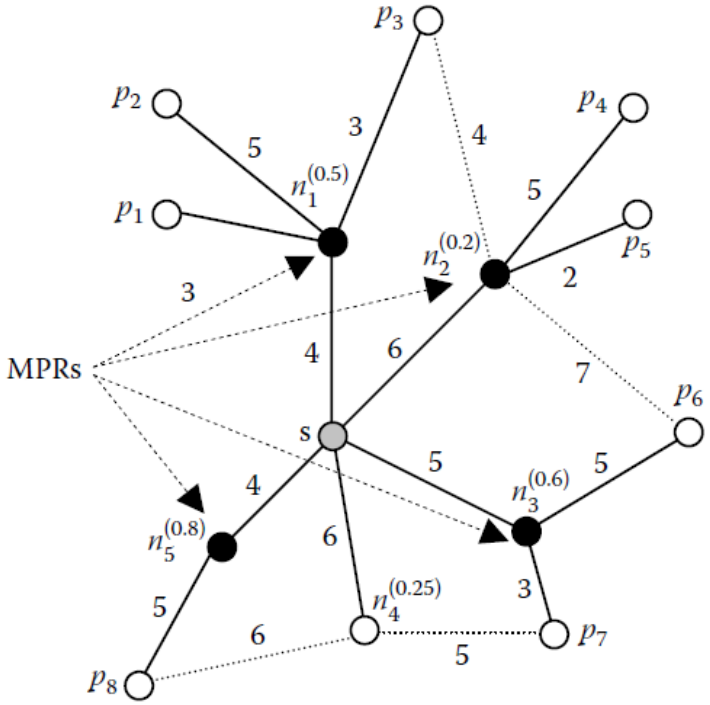
1. Start with an empty MPR set MPR(s) of node s: MPR(s)={}
2. First identify those two-hop neighbor nodes in $N^2(s)$ which have only one neighbor in the one-hop neighbor set N(s). Add these nodes of N(s) to the MPR set MPR(s) if they are not already in MPR(s).

   For our example: nodes p1, p2, p4 and p5 have one neighbor in N(s). Add n1 and n2 in MPR(s)
3. If there exists a node in $N^2(s)$ for which MPR node is not selected, do the following:

   • For each node in $N^2(s)$, with multiple neighbors from N(s) , select a neighbor from N(s) as MPR node which results in **minimum cost from s to the node in $N^2(s)$,**

   $C^{MPR}_{s, N(s), N2(s)}$

   • Add that node of N(s) in MPR(s) if it is not already in MPR(s)



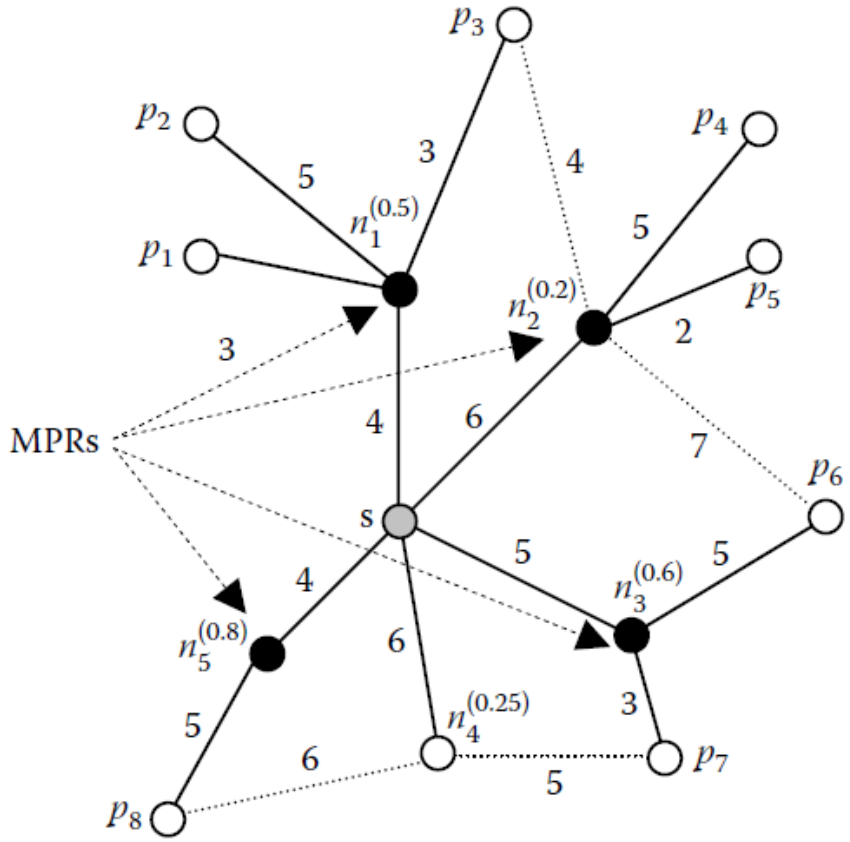
For each node from $N^2(s)$ having no MPR selected (nodes p3, p6, p7,p8), calculate metrics:
For p3: $C^{MPR}_{s, n1,p3} = 4+3+1/0.5 = 9$; $C^{MPR}_{s, n2, p3}=6+4+1/0.2 = 15$
so n1 selected as MPR, but already in set MPR(s)

For p6: n3 will be MPR, so need to add it in MPR(s) = {n1, n2, n3}
For p7: n3 chosen as MPR, already in
For p8: n5 chosen as MPR, add to MPR(s) = {n1, n2, n3, n5}

Compared with OLSR, choosing MPR with OEDR will cover better all nodes in term of cost; example:

For p3: $C^{MPR}_{s, n1,p3} = 4+3+1/0.5 = 9$
as in OEDR,
but with OLSR where
n2 is MPR, the cost is:

$C^{MPR}_{s, n2, p3}=6+4+1/0.2 = 15$



OEDR 3: MPR and Energy-Delay Information Declaration

- Each node in the network that is selected as MPR, by at least one of its neighbors, transmits a TC message periodically →information about the MPR node's selector set, which is a subset of the one-hop neighbors that have selected the sender node as a MPR, link costs (energy- delay) between the MPR node and its selectors.
- TC messages are forwarded as usual broadcast messages, except that only the MPR nodes forward the message to the next hop
- Each node in the network maintains a "topology table" (from the TC messages), and the link costs.
- Whenever a node receives a TC message, it records the information as entries into the topology table with the addresses in the MPR selector set as the possible destinations, the originator of TC message as the last-hop toward destinations, and the corresponding link costs. Based on this information, the routing table is calculated.
- An entry in the topology table consists of the address of a destination (an MPR selector in the received TC message), address of the last-hop node to that destination (originator of the TC message), and the cost of the link between the destination and its last-hop → the destination node can be reached in the last hop through this last-hop node at the given cost

OEDR 4: Routing table calculation

- Each node maintains a “routing table” → to route packets for other destinations in the network
- Routing table entries = the destination address, next-hop address used for reaching that destination, estimated distance to destination, cost of the path from the source to the destination
- Each destination has an entry in the routing table, for which a route is known from the given node to the destination
- Route determination → least cost spanning tree algorithm, based on a cost given by the energy-delay product:

1. Clear all entries in the routing table, RT(s) , of node s
2. Record the new entries in the RT(s) , starting with one-hop neighbors in N(s) as destination nodes. For each neighbor entry in the neighbor table, record a new route entry in the routing table, where destination and next-hop addresses are both set to the address of the neighbor; distance is set to 1, and the cost of the route is set to the cost of the link (from the neighbor table)
3. Then record the new entries for destination nodes i+1 hops away in the routing table. The following procedure is executed for each value of i, starting with i=1 and incrementing it by 1 each time. The execution will be stopped if no new entry is recorded in an iteration.

• For each topology entry in the topology table, if the last-hop address corresponds to the destination address of a route entry with distance equal to i, then check to see if a route entry already exists for this destination address

a. If the destination address of the topology entry does not correspond to the destination address of any route entry in the routing table, then a new route entry is recorded in the routing table, where:

• Destination is set to destination address in topology table

• Next-hop is set to next-hop of the route entry whose destination is equal to previously mentioned last-hop address

• Distance is set to i+1

• Cost of the route is set to the sum: cost of the route entry in the RT(s) with the last-hop as its destination address + cost of the link between the destination and its last-hop from the topology table

b. Else, if there exists a route entry in RT(s) whose destination address corresponds to the destination address of the topology entry, then compute the cost of the new route, and compare with the cost of the old route in the routing table, corresponding to the same destination address

• If the new cost is less than the old cost, then erase the old entry in the RT(s) , and record a new route entry in the RT(s), with fields for the entry computed similar to step a.

Attribute-Based Routing

Data-centric routing

- Cannot assume either the network address or the geographic location of the node for communication
- Enables a good communication path (matching) between
  - Nodes (desiring certain type of info), and
  - Node (having the info)
- Data is described as attribute-value pairs

Example - attribute value-pairs

```
type = animal              // named record type
instance = horse           // instance of this type
location = [89, 154]       // location of horse
rect = [0, 200, 0, 200] // a spatial range of interest
time =02:45:23// time of detection
```

Routing Methods

- Directed Diffusion
- Rumor Routing (http://www.tcs.hut.fi/Studies/T-79.194/slides/ahtiainen_050209.pdf; http://www.tcs.hut.fi/Studies/T-79.194/papers/ahtiainen_050209.pdf)
- Geographic Hashing Tables (http://www.cs.ucl.ac.uk/staff/b.karp/ght-wsna2002.pdf)

Directed Diffusion

A very general approach: attribute-based routing

Sink nodes

- Request info (interests) – initially flooding
- Rebroadcast its interest message – periodically
- Frequency of update desired
- TimeStamp
- Neighbor (direction)

Source nodes – generate info

Interests

- Records indicating a desire for certain type of info
- Propagate across network (interest diffusion) – looking for nodes with matching event records
- Assume persistent interests
  - Measured data from source will be needed for a period of time
  - Allow the Directed Diffusion protocol to learn the optimal paths
- Fields (attributes) in the Interest Records
  - Interval field - indicating the frequency with which the sink wishes to receive information about objects matching the other record attributes
  - Duration field – the period of validity of an interest
  - TimeStamp Field – for differentiate the repeated broadcasts from earlier version

All nodes – track the unexpired interests they have seen

Each node maintains an interest Cache

Entry 1: Distinct Interest Type, duration not expired, neighbors who passed the interest, Gradients associates with each interest

Entry 2:

Entry n:

Forwarding mechanism

Gradients

Derived from the frequency with which a sink requests repeated data about an interest

Use it to direct and control info flow back to the sink

Reinforce good info delivery & disable unproductive ones

Multiplicity of gradients

- Not a concern, does not create persistent data delivery loops

- Allows for the quick reestablishment of info delivery paths when nodes or link fail

WSNs Security – Why and Specificity

Protecting confidentiality, integrity, and availability of the communications and computations

Sensor networks are vulnerable to security attacks due to the broadcast nature of transmission
Sensor nodes can be physically captured or destroyed
Sensor Node Constraints: Battery, CPU power, Memory
Networking Constraints and Features: Wireless, Ad hoc, Unattended.

Battery Power Constraints: Computational Energy Consumption => Crypto algorithms: Public key or Symmetric key?
Public key: 1000 times slower than symmetric encryption; Hardware is complicated;
Energy consumption is high
Conclusion is obvious!

Communications Energy Consumption:
Exchange of keys, certificates, etc.
Per-message additions (padding, signatures, authentication tags)

Memory Constraints

Program Storage and Working Memory
- Embedded OS, security functions (Flash)
- Working memory (RAM)

=> 'thin' operating system (TinyOS), specific 'reduced' programming languages (nesC)

Example: Mica Motes:
- 128KB Flash and 4KB RAM

Framework for Security in Sensor-nets

Tradeoff between Security and Performance
- For example between security and connectivity

Interdependent Solutions
- Match weak encryption to strong keying

Complementary Solutions
- Network Integrity vs. Data Integrity

Flexible Solutions

REMEMBER! **Security goals**

Availability: ensures the survivability of network services despite denial-of-service (DoS) attacks

Confidentiality: ensures that information is not disclosed to unauthorized entities

Integrity: guarantees that a message being transferred is never corrupted

Authentication: enables a node to ensure the identity of the peer node with which it communicates

Non-repudiation: ensures that the origin of a message cannot deny having sent the message

Anonymity: hide sources, destinations and routes (important issue for WSNs)

WSN security characteristics

It is not possible to avoid unauthorized devices to reach the network area

Any device within reach of radio-frequency signals can get access to data being transmitted; thus, attacks of interruption and interception of data are likely

What can be done: spread spectrum increases the difficulty for
signal interruption or eavesdropping

It is important to understand that wireless communications affect only the physical, data link and network layers of the OSI stack

In particular, all methods of cryptography developed at transport layer and above remain valid: can we afford them here?

Major insecurities in sensor networks

Problems arising from lack of individual IDs
- authentication is hard
- non-repudiation is hard to enforce
- node impersonation is easy

Problems arising from sleep-awake cycles and system longevity
- trust relationships hard to establish

Eavesdropping: may give an adversary access to secret information violating confidentiality

Sensors run the risk of being compromised
- by infiltration
- by tampering

Vulnerabilities and threats to a WSN

Outages due to equipment breakdown and power failures
Non-deliberate damage from environmental factors, physical tampering, and information gathering

Threats to a WSN

Passive Information Gathering

Communications are in the clear: intruder can passively pick off data stream

Subversion of a Node

Sensor node is captured, may be tampered with, electronically interrogated and perhaps compromised (cryptographic system disclosed)

Secure sensor nodes, therefore, must be designed to be tamper proof and should react to tampering in a *fail complete manner*

False Node:

An intruder might .add. a node to a system and feed false data or block the passage of true data

Traffic Analysis, Message corruption, Node malfunction / outage

Denial of Service:

*M*ay consist of a jamming the radio link

Could exhaust resources or misroute data

WSN Operational paradigms and correspondent vulnerabilities

Simple Collection and Transmittal

The sensors take periodic measurements and transmit the associated data directly to the collection point (immediately following data collection or scheduled at some periodic interval); nodes only concerned with its transmission (no routing or co-operation among nodes)

Vulnerability: *denial of service* (jamming radio frequency or collision induction), *spoofing* (broadcasting of spurious information), *physical attacks* (node capture, subversion)

Forwarding

Sensors collect and transmit data to one or more neighboring sensors that lie on a path to the controller; routing and collaboration involved

Vulnerability: same as previous, plus: *Black Hole attack* (node drops data instead of forwarding); *Data Corruption* (node change received data); *Resource Exhaustion* (node forward extra data to consume others resources)

Receive and Process Commands

Sensors receive commands from a controller, either directly or via forwarding, and configure or re-configure themselves based on the commands

Vulnerability: same as previous, plus spurious commands to be issued

Self-Organization

Upon deployment WSN self organizes, and a central controller(s) learns the network topology; paradigm may include the use of more powerful sensors that serve as cluster heads for small coalitions within WSN

Vulnerability: same, plus specific attacks (*Induced Routing Loops, Sinkholes, Wormholes* and *HELLO Flooding)*

Data Aggregation

Nodes aggregate data from downstream nodes, incorporating their own data with the incoming data. The composite data is then forwarded to a collection point

Vulnerability: replay attacks because the *authentication* of its downstream peers

Optimization: Flexibility and Adaptation

Predicated upon their own measurements and upon the values of incoming data, this paradigm requires that the sensors in the WSN make decisions

Vulnerabilities: all above

Information assurance in sensor networks

More general topic:

Information operations that protect and defend information and information systems, ensuring their availability, integrity, authentication, confidentiality, and non repudiation. This includes providing for restoration of information systems by incorporating protection, detection, and reaction capabilities

Key components

Network survivability: ability of the network to function in the wake of failures by minimizing their impact

Information availability (information survivability): need for a user to have uninterrupted and secure access to information on the network

Network security: attempts to provide basic security services

Information security: an ongoing process that utilizes software and hardware to help secure information flow

Thus:

Information *assurance* is more inclusive than information *security*

Assurance involves not only *protection and detection* but also *reaction* (mainly survivability and dependability of the system that has been subject to successful attack)

It also includes *proactive* (offensive) information operations, termed *information warfare*, against attackers

A Generic model for WSN Security

Components of the model:

Communication model

Hybrid communication employing both centralized and distributed models; the **centralized model** is used when one or more powerful nodes exist, around which less-powerful sensor nodes can *cluster* and the **distributed model** is employed when no powerful nodes exist. Their coexistence gives a **hierarchical** WSN

Communication security

Mechanisms to secure communication between nodes are also deployed in a hybrid manner:
-case where more powerful nodes exist and clusters can be formed, end-to-end communication security between the designated *clusterhead* and each individual sensor node in the cluster should be used
- case WSN is formed in a distributed manner, it is appropriate to employ pair wise security, but only for a fixed number of pairs

Key management

Sensor nodes in a WSN have limited amount of energy, public-key cryptographic mechanisms, which are expensive in terms of energy consumption, are not suitable

Private-key cryptography, quite applicable to WSNs due to its low energy requirements.

In a hybrid WSN that consists of nodes of varying capabilities and resources, it is feasible to employ both public-key and private key mechanisms for security

Key distribution: mechanisms to solve this problem are: pre-deployed keys (off-line key distribution), group keying and arbitrated keying

Data aggregation

Should be performed either by multiple designated or elected nodes, based on the security mechanism, i.e., end-to-end (if clustering) or pair wise (distributed net)

Self-healing

Co-existence of external attacks on the network, but also breakdowns due to node failure, especially due to energy exhaustion.

Use of both passive and active mechanisms:

Passive mechanisms include data encryption and node authentication
Active mechanisms include key revocation and removing offending nodes from the WSN.

Security countermeasures must exist at every layer:

-spread-spectrum techniques at the link-layer,
- encryption at the network and application layers,
- authentication at the application layer
- aberrant behavior detection at the network and application layers

The ***ideal security model*** would consist of mechanisms to monitor and track the health of all nodes in the network, thereby enabling quick (re)-establishment of secure routes around nodes that provide indications of imminent failure.
Based on the communication model, this may require invoking procedures to distribute keys to neighboring nodes as required

Bibliography

Information Security in Wireless Sensor Networks, Prof. Stephan Olariu
Sensor Network Research Group, Old Dominion University, Norfolk, VA

Security for Sensor Networks, C. Durresi, Computer Science Department Louisiana State University, Baton Rouge, LA

Security for Wireless Sensor Networks, S. Avancha et al., University of Maryland, Baltimore, MD

Wireless Sensor Networks, Hung Le Xuan, Real-time and Multimedia Laboratory, khu.ac.kr

Routing in WSN, A.Kruger, Univ. of Iowa