Tutorial: Wireless Sensor Networks

November 2005

Krishna M. Sivalingam, Associate Professor
Dept. of CSEE
University of Maryland, Baltimore County (UMBC)
Baltimore, MD 21250
krishna@umbc.edu
www.cs.umbc.edu/~krishna; dawn.cs.umbc.edu

General Overview

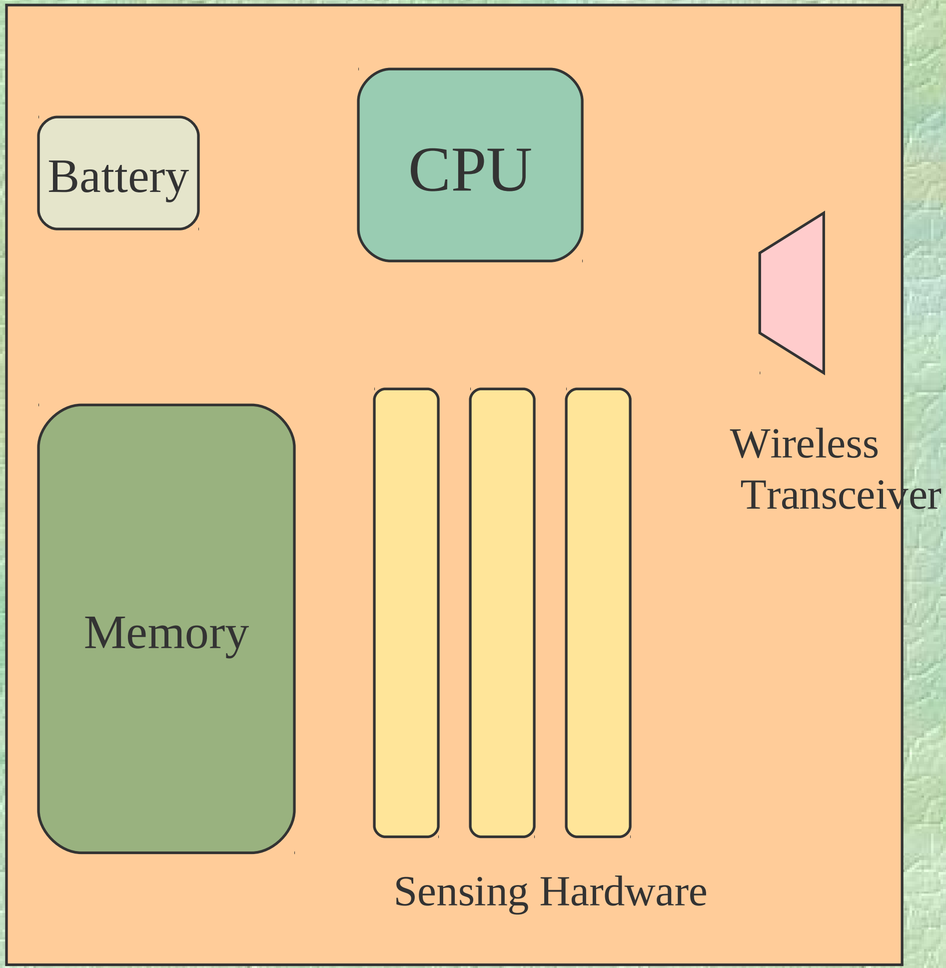
- Introduction to Wireless Sensor Networks
- Data Dissemination and Routing Protocols
- Data Gathering
- Medium Access Control Protocols
- Locationing and Coverage
- Testbeds/Applications
- Security in Wireless Sensor Networks
- Summary & Discussion

Motivation

- GOAL: Deeply Networked Systems or Pervasive Networking
- 98% of all processors are not in traditional desktop computer systems, but in house-hold appliances, vehicles, and machines on factory floors
- Add reliable wireless communications and sensing functions to the billions of physically embedded computing devices to support ubiquitous networked computing
- Distributed Wireless Sensor Networks is a collection of embedded sensor devices with networking capabilities

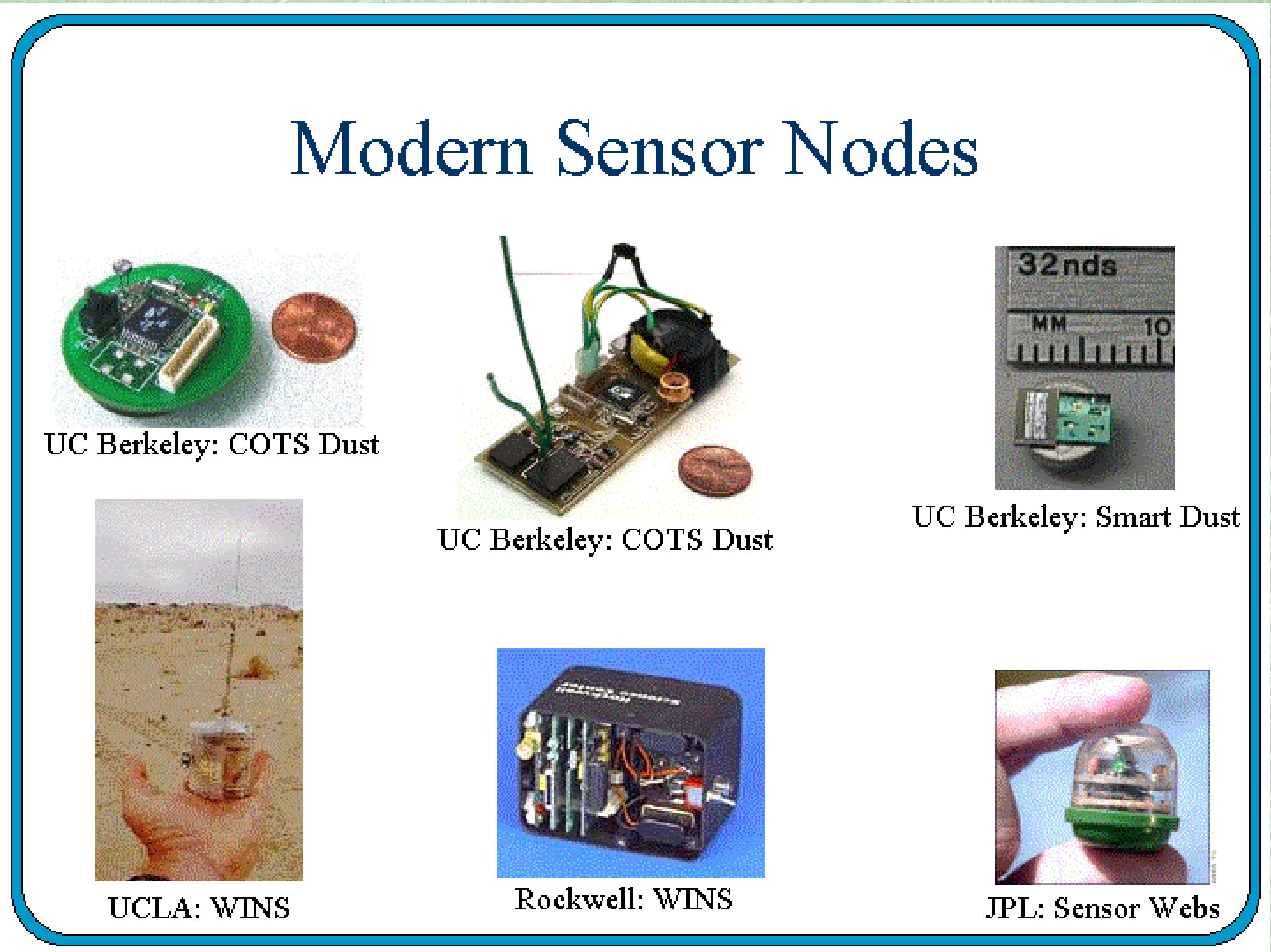
Introduction to WSN

Background , contd.

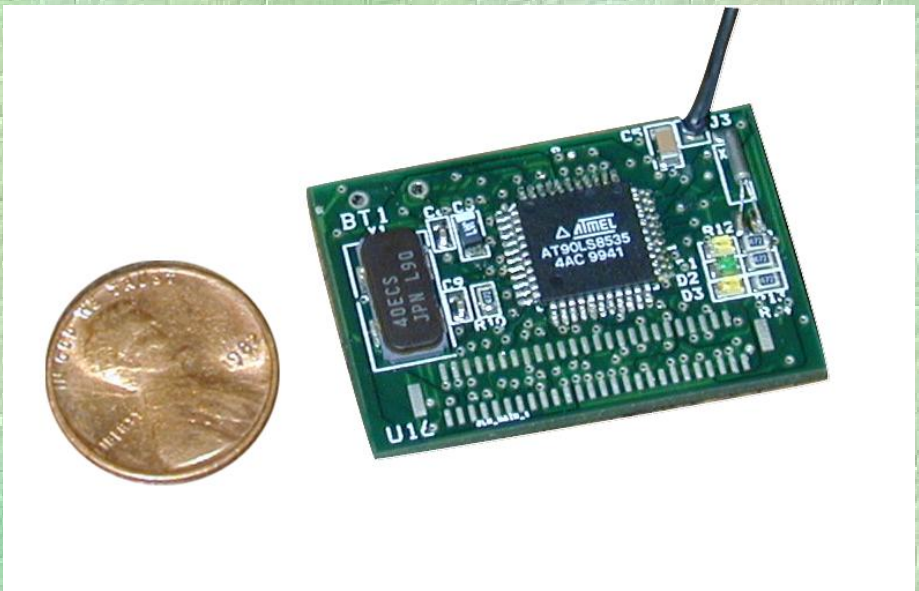


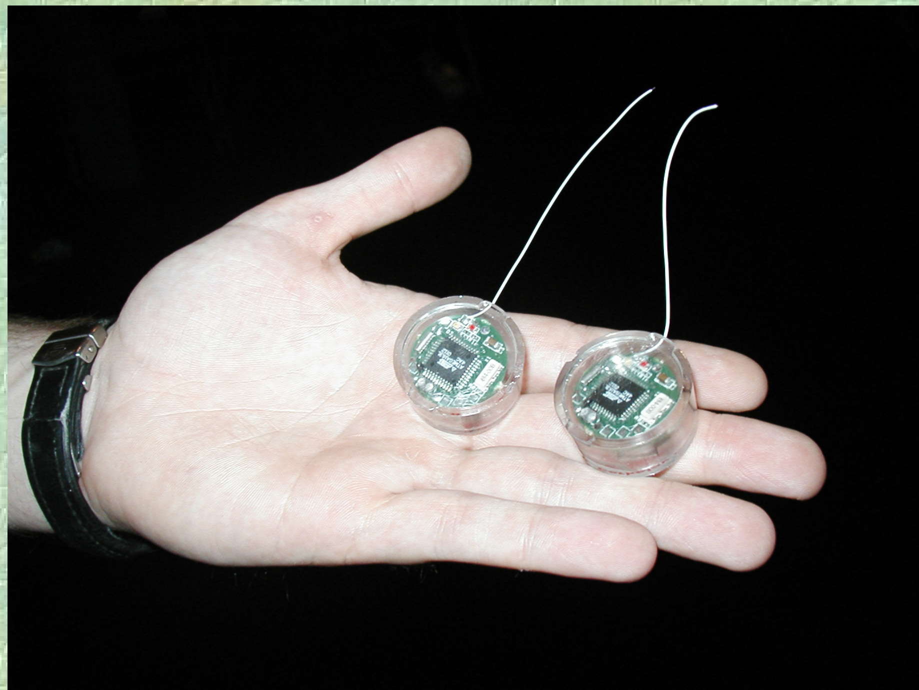
- Sensors
  - Enabled by recent advances in MEMS technology
  - Integrated Wireless Transceiver
  - Limited in
    - Energy
    - Computation
    - Storage
    - Transmission range
    - Bandwidth

Background, contd.



Sensor Nodes, contd.





Sensors (contd.)

- The overall architecture of a sensor node consists of:
  - The sensor node processing subsystem running on sensor node main CPU
  - The sensor subsystem and
  - The communication subsystem
- The processor and radio board includes:
  - TI MSP430 microcontroller with 10kB RAM
  - 16-bit RISC with 48K Program Flash
  - IEEE 802.15.4 compliant radio at 250 Mbps
  - 1MB external data flash
  - Runs TinyOS 1.1.10 or higher
  - Two AA batteries or USB
  - 1.8 mA (active); 5.1uA (sleep)

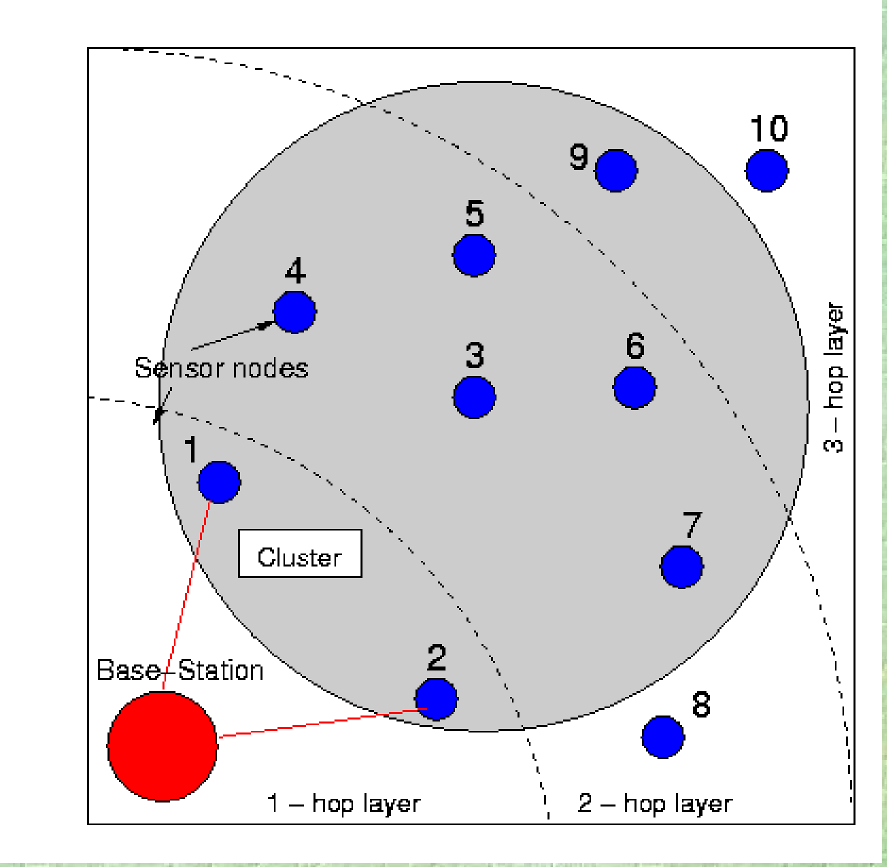
Crossbow Mote

TPR2400CA-TelosB

Overall Architecture of a sensor node

Wireless Sensor Networks (WSN)

- Distributed collection of networked sensors



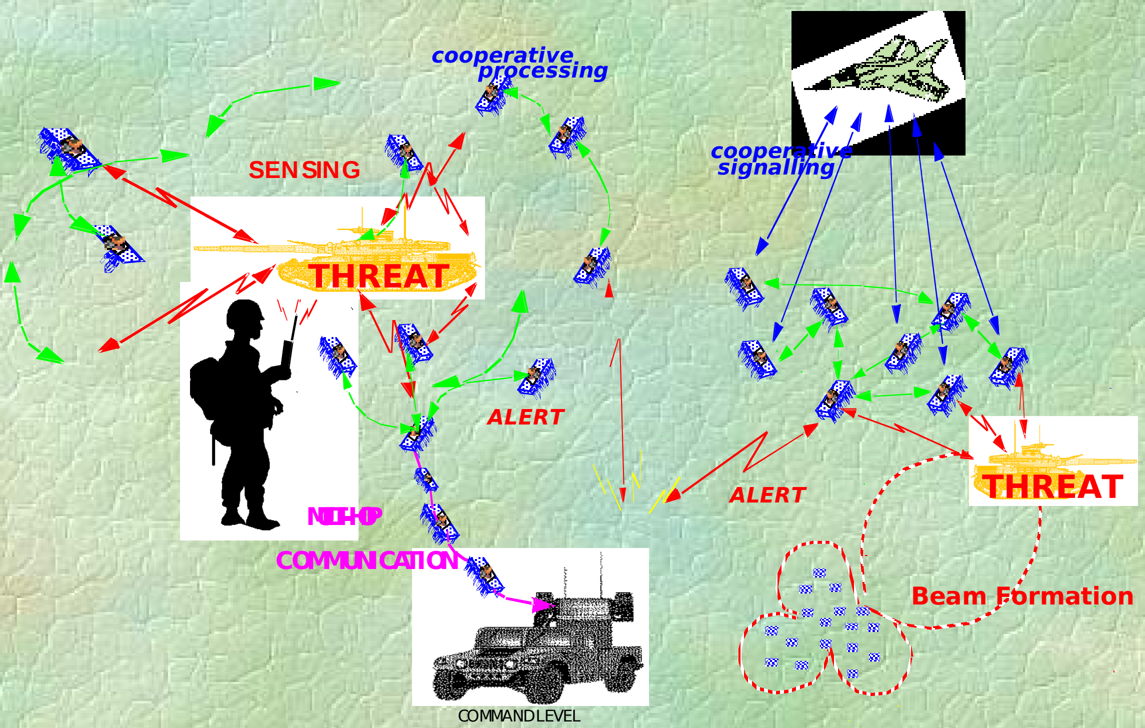
Networked vs. individual sensors

- Extended range of sensing:
  - Cover a wider area of operation
- Redundancy:
  - Multiple nodes close to each other increase fault tolerance
- Improved accuracy:
  - Sensor nodes collaborate and combine their data to increase the accuracy of sensed data
- Extended functionality:
  - Sensor nodes can not only perform sensing functionality, but also provide forwarding service.

Applications of sensor networks

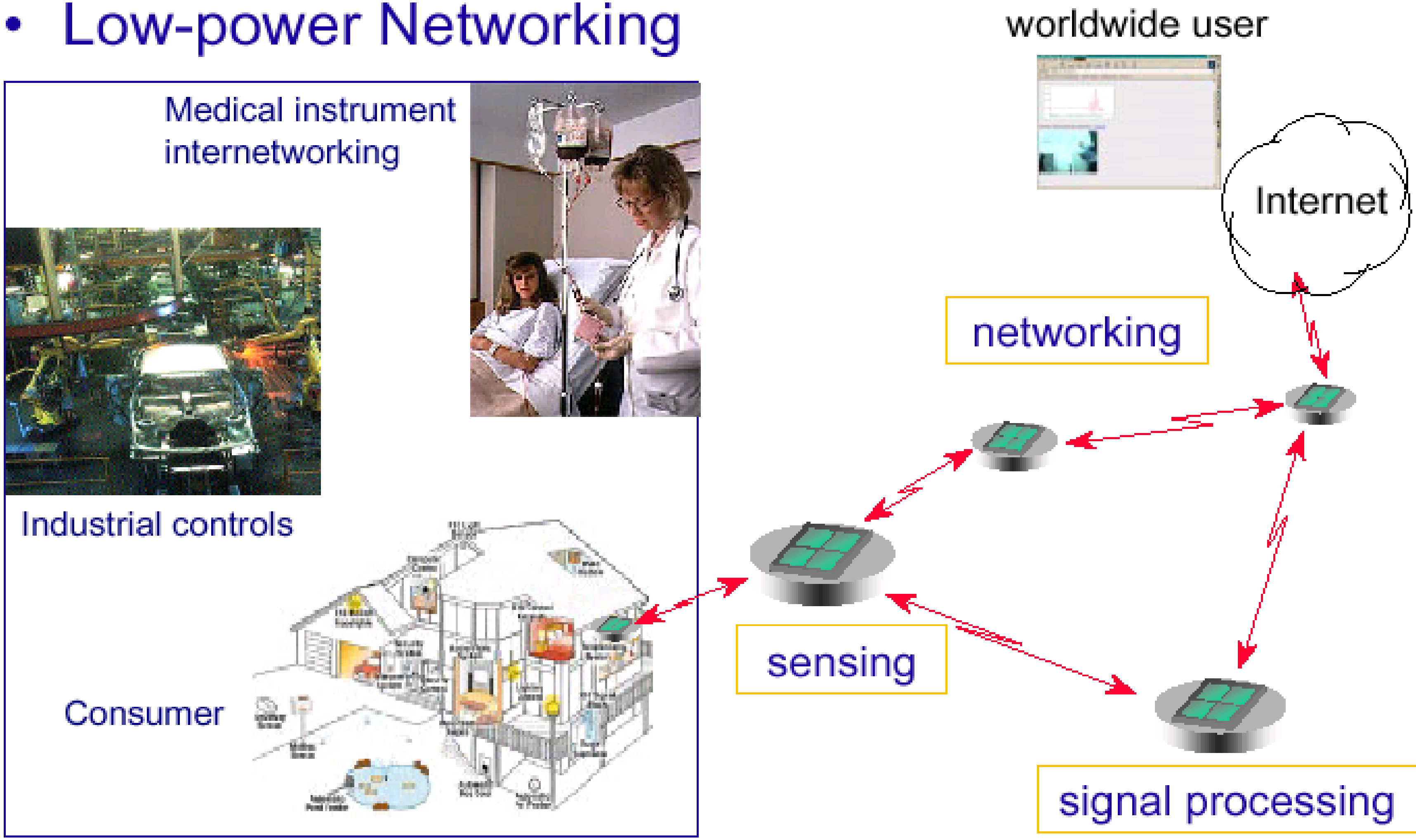
- Physical security for military operations
- Indoor/Outdoor Environmental monitoring
- Seismic and structural monitoring
- Industrial automation
- Bio-medical applications
- Health and Wellness Monitoring
- Inventory Location Awareness
- Future consumer applications, including smart homes.

Applications, contd.



Applications, contd.

- Low-power Networking



Characteristics and challenges

- Deeply distributed architecture: localized coordination to reach entire system goals, no infrastructure with no central control support
- Autonomous operation: self-organization, self-configuration, adaptation, exception-free
  - TCP/IP is open, widely implemented, supports multiple physical network, relatively efficient and light weight, but requires manual intervention to configure and to use.
- Energy conservation: physical, MAC, link, route, application
- Scalability: scale with node density, number and kinds of networks
- Data centric network: address free route, named data, reinforcement-based adaptation, in-network data aggregation

Challenges, contd.

- Challenges
  - Limited battery power
  - Limited storage and computation
  - Lower bandwidth and high error rates
  - Scalability to 1000s of nodes
- Network Protocol Design Goals
  - Operate in self-configured mode (no infrastructure network support)
  - Limit memory footprint of protocols
  - Limit computation needs of protocols -> simple, yet efficient protocols
  - Conserve battery power in all ways possible

WSN vs. MANET

- Wireless sensor networks may be considered a subset of Mobile Ad-hoc NETworks (MANET).
- WSN nodes have less power, computation and communication compared to MANET nodes.
- MANETs have high degree of mobility, while sensor networks are mostly stationary.
  - Freq. node failures in WSN -> topology changes
- Routing protocols tend to be complex in MANET, but need to be simple in sensor networks.
- Low-power operation is even more critical in WSN.
- MANET is address centric, WSN is data centric.

Why not port Ad Hoc Protocols?

- Ad Hoc networks require significant amount of routing data storage and computation
  - Sensor nodes are limited in memory and CPU
- Topology changes due to node mobility are infrequent as in most applications sensor nodes are stationary
  - Topology changes when nodes die in the network due to energy dissipation
- Scalability with several hundred to a few thousand nodes not well established
- GOAL: Simple, scalable, energy-efficient protocols

Focus: Radio Transceiver Usage

- The wireless radio transceiver is typically in three modes:
  - Transmit – Maximum power consumption
  - Receive
  - Idle
  - Turned off – Least power consumption
- Sensor node exists in three modes: Active, standby, and battery dead
- Turnaround time: Time to change from one mode to another (esp. important is time from sleep to wakeup and vice-versa)
- Protocol design attempts to place node in these different modes depending upon several factors
- Sample power consumption from 2 sensor nodes shown next

Rockwell Node (SA-1100 proc)

MCU Mode	Sensor Mode	Radio Mode	Power(mW)
Active	On	Tx(36.3mW)	1080.5
		Tx(13.8mW)	942.6
		Tx(0.30mW)	773.9
Active	On	Rx	751.6
Active	On	Idle	727.5
Active	On	Sleep	416.3
Active	On	Removed	383.3
Active	Removed	Removed	360.0
Sleep	On	Removed	64.0

UCLA Medusa node (ATMEL CPU)

MCU Mode	Sensor	Radio(mW)	Data rate	Power(mW)
Active	On	Tx(0.74,OOK)	2.4Kbps	24.58
		Tx(0.74,OOK)	19.2Kbps	25.37
		Tx(0.10,OOK)	2.4Kbps	19.24
		Tx(0.74,OOK)	19.2Kbps	20.05
		Tx(0.74,ASK)	19.2Kbps	27.46
		Tx(0.10,ASK)	2.4Kbps	21.26
Active	On	Rx	-	22.20
Active	On	Idle	-	22.06
Active	On	Off	-	9.72
Idle	On	Off	-	5.92

Energy conservation

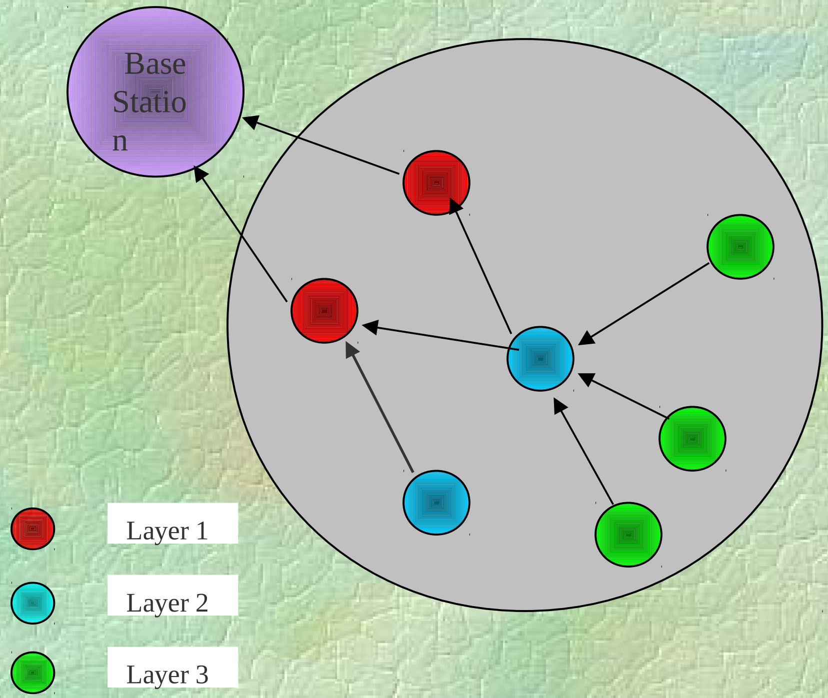
Physical layer	• Low power circuit(CMOS, ASIC) design • Optimum hardware/software function division • Energy effective waveform/code design • Adaptive RF power control
MAC sub-layer	• Energy effective MAC protocol • Collision free, reduce retransmission and transceiver on-times • Intermittent, synchronized operation • Rendezvous protocols
Link layer	• FEC versus ARQ schemes; Link packet length adapt.
Network layer	• Multi-hop route determination • Energy aware route algorithm • Route cache, directed diffusion
Application layer	• Video applications: compression and frame-dropping • In-network data aggregation and fusion

See Jones, Sivalingam, Agrawal, and Chen survey article in ACM WINET, July 2001;
See Lindsey, Sivalingam, and Raghavendra book chapter in Wiley Handbook of Mobile Computing, Ivan Stojmenovic, Editor, 2002.

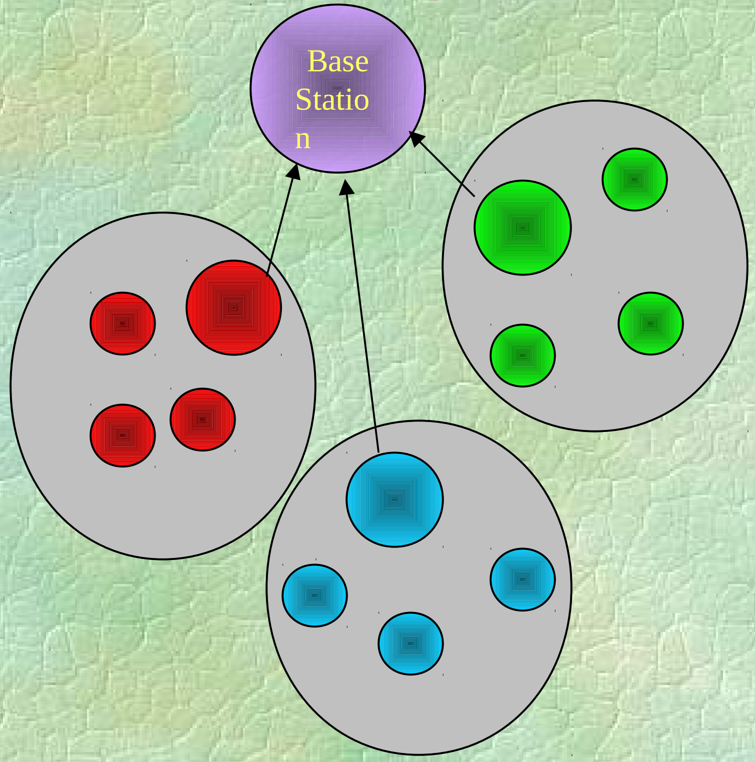
Network Architectures

Network Architectures

Layered Architecture



Clustered Architecture



Larger Nodes denote Cluster Heads

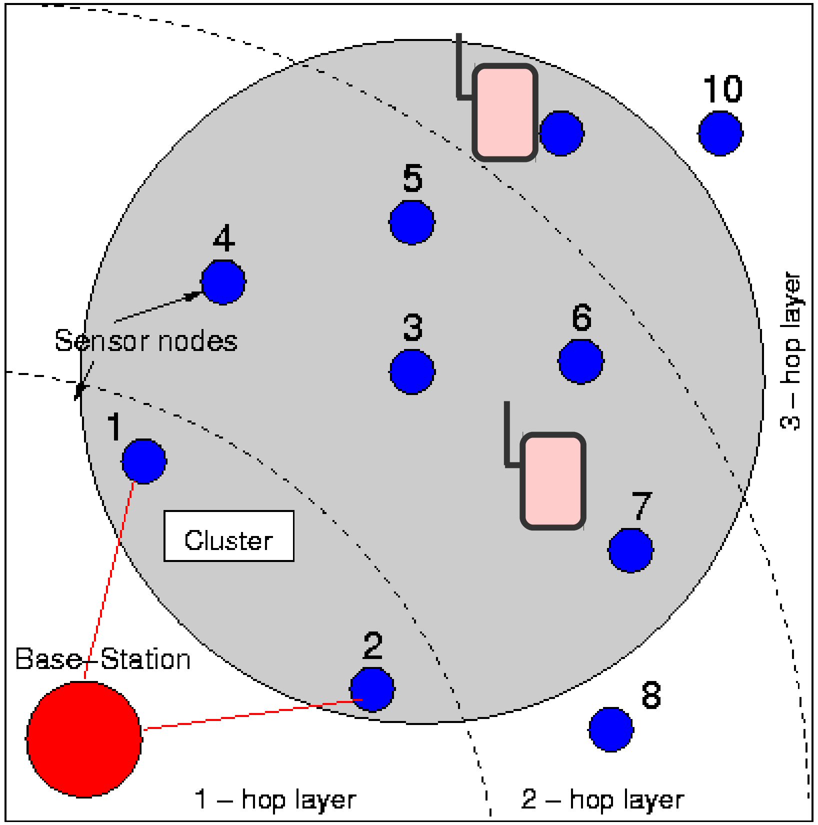
Clustered network architecture

- Sensor nodes autonomously form a group called clusters.
- The clustering process is applied recursively to form a hierarchy of clusters.

Cluster architecture (contd.)

- Example - LEACH protocol
  - It uses two-tier hierarchy clustering architecture.
  - It uses distributed algorithm to organize the sensor nodes into clusters.
  - The cluster-head nodes create TDMA schedules.
  - Nodes transmit data during their assigned slots.
  - The energy efficiency of the LEACH is mainly due to data fusion.

Layered Network Architecture



A 10 node sensor network depicting cluster of node 3; there are 2 mobile nodes

- A few hundred sensor nodes (half/full duplex)
- A single powerful base-station
- Network nodes are organized into concentric ***Layers***
- ***Layer:*** Set of nodes that have the same hop-count to the base-station
- Additional Mobile Nodes traversing the network
- Wireless Multi-Hop Infrastructure Network Architecture (MINA)

MINA, contd.

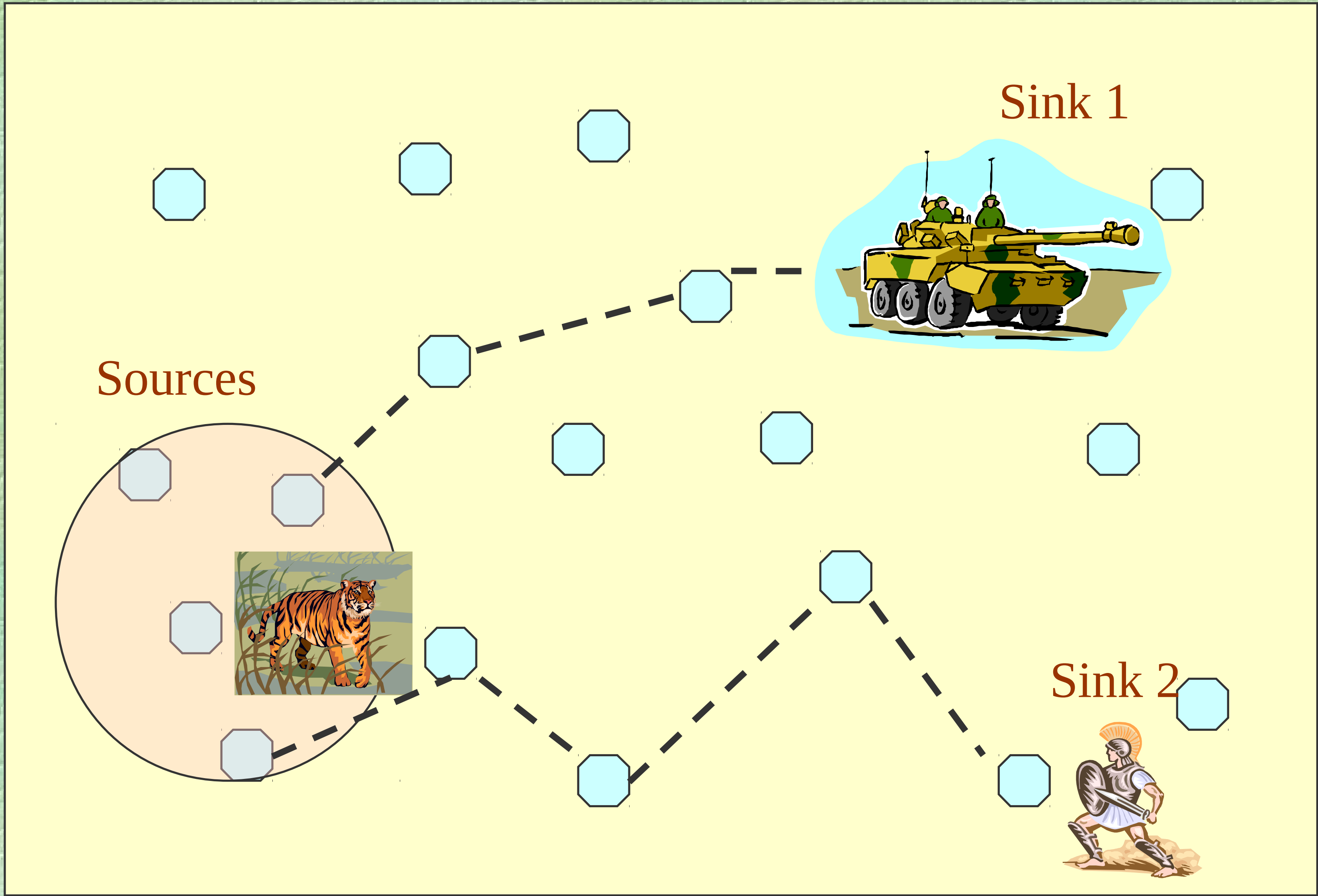
- Set of wireless sensor nodes create an infrastructure – provide sensing and data forwarding functionality
- Mobile soldiers with hand-held units access the sensors for data and also to communicate with a remote BS
- BS is data gathering, processing entity and communication link to larger network
- Shorter-range, low-power transmissions preferred for covert operations and to conserve power

Data Dissemination Architectures and Protocols

Data Dissemination

- In ad hoc networks, traffic is peer-to-peer
  - Multi-hop routing is used to communicate data
- In WSN, other traffic models are possible:
  - Data Collection Model
  - Data Diffusion Model
- Data Collection Model: Source sends data to a collection entity (e.g. gateway): periodically or on-demand
- Data Diffusion Model:
  - *Source*: A sensor node that generates data, based on its sensing mechanisms' observations
  - Event: Something that needs to be reported, e.g. in target detection; some abnormal activity
  - Sink: A node, randomly located in the field, that is interested in events and seeks such information

Data Diffusion: Concept



Diffusion: Basics

- Data-centric vs. address centric architecture
  - Individual network address is not critical; Data is important and is accessed as needed
- User can pose a specific task, that could be executed by sensor nodes
- Concept of Named Data: (Attribute, Value) Pair
- Sink node requests data by sending “interests” for data
- Interests are propagated through the network, setting up gradients in the network, designed to “draw” data
- Data matching the interest is then transmitted towards the sink, over multiple paths (obtained by the gradients
- The sink can then reinforce some of these paths to optimize

Diffusion Basics, contd.

- Design Issues:
  - How does a sink express its interest in one or more events?
  - How do sensor nodes keep track of existing interests from multiple sinks?
  - When an event occurs, how does data get propagated from source(s) to sink(s)?
  - Can in-network data processing (e.g. data fusion), data aggregation and data caching help improve performance?
- [Intanagonwiwat et. al.; ACM MobiCom 2000]

Diffusion Basics, contd

- Example Task

  {Type = Animal; Interval = 20ms; Time = 10s;

  Region = [-100, 100, 200, 400] }
- The above task instructs a sensor node in the specified region to track for animals; If animal is tracked/detected, then send observations every 20 ms for 10s
- The above task is sent via interest messages and all sensor nodes register this task.
- When a node detects an event, it then constructs a Data Event message

Diffusion: Basics, contd

- Data Event Example:

  {Type = Animal; Instance = Tiger;

  Location = [101, 201]; Intensity = 0.4;

  Confidence = 0.8; Timestamp = 2:51:00}

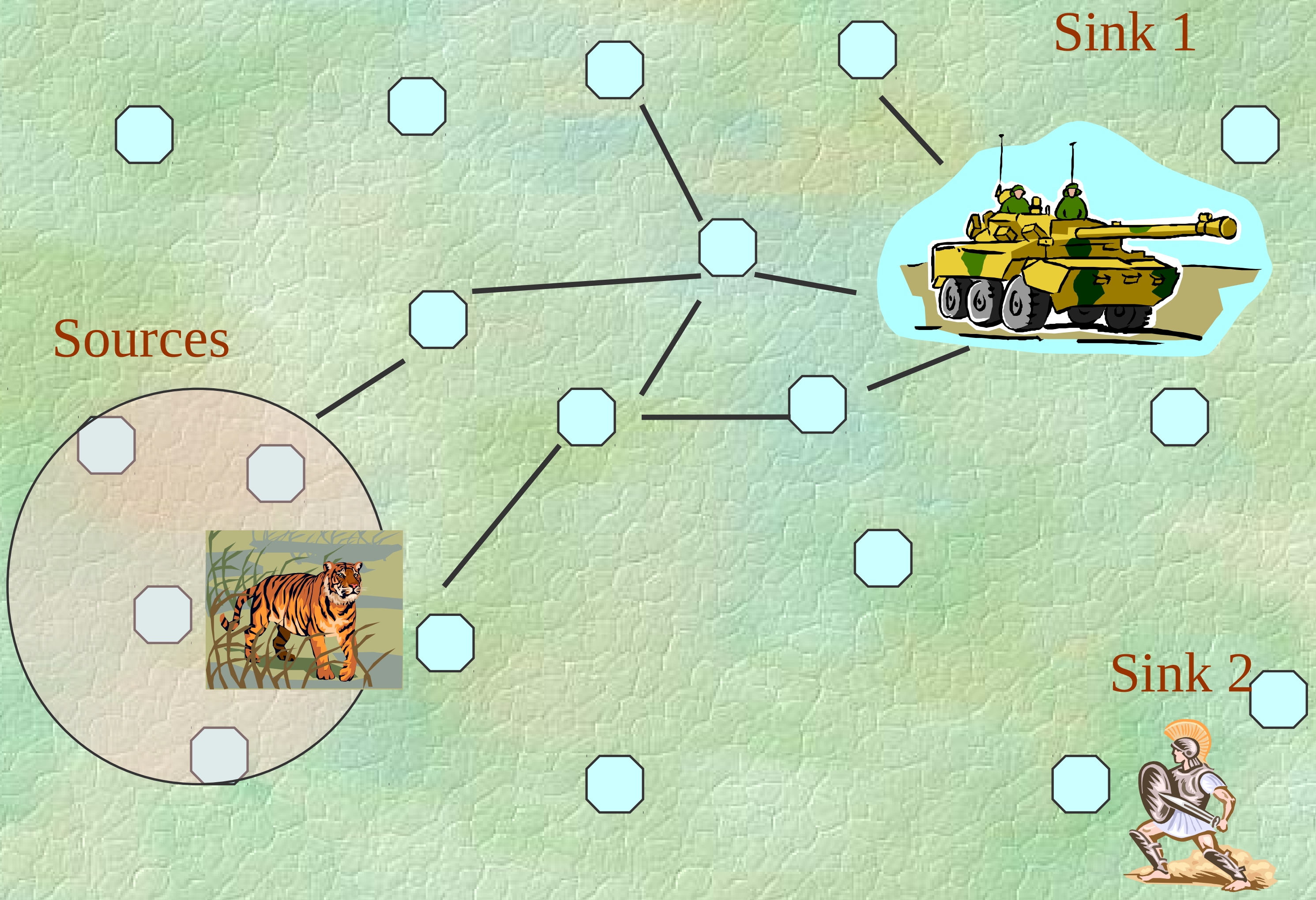
Interests and Gradients:

- For each active task that a sink is interested in:
  - Sink broadcasts interest to its neighbors
    - Initially, to explore, it could set large interval (e.g 1s)
  - Sink refreshes each interest, using timestamps
- Each sensor node maintains an *interest cache*
  - Interest aggregation is possible

Diffusion: Interests

- When a node receives an interest, it:
  - Checks cache to see if an entry is present.
  - If no entry, creates an entry with a single gradient to neighbor who sent this interest
  - Gradient specifies the direction and data rate.
- Resend interest to a subset of its neighbors
  - This is essentially flooding-based approach
  - Other probabilistic, location-based and other intelligent forwarding approaches possible
- Similar to multicast tree formation, at sink instead of at source

Diffusion: Interest Propagation



Diffusion: Data Propagation

- When a sensor node detects a target, it:
  - Searches interest cache for matching entry
  - If found, computes highest requested event rate among its gradients
  - Instructs sensor sub-system to generate data at this rate
  - Sends data to neighbors on its gradient list
- Intermediate nodes maintain a *data cache*
  - Caches recently received events
  - Forwards event data to neighbors on its gradient list, at original rate or reduced rate (intelligently)

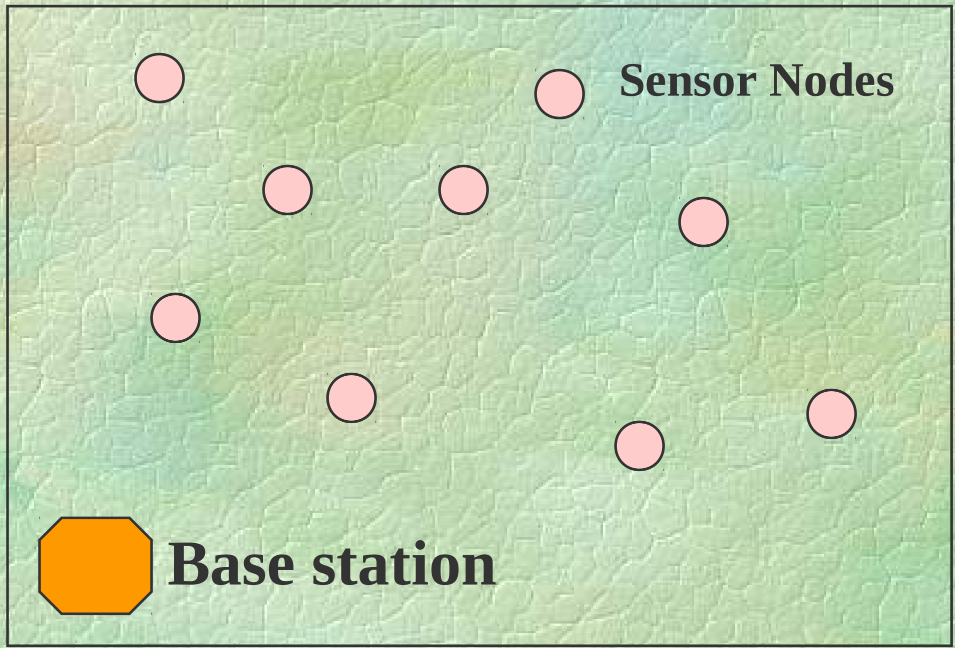
Diffusion: Reinforcement

- When sink gets an event notification, it:
  - Picks a suitable set of neighbor(s) (best link, low delay, etc.) and sends a refresh interest message, with higher notification rate (e.g. every 10 ms instead of every 1s)
    - This will prune some of its neighbors (since interests in a node's cache will expire)
  - Each selected neighbor forwards this new interest to a subset of its neighbors; selecting a smaller set of paths
- Negative reinforcement also necessary to de-select weaker paths if a better path found.

Part III: Data Gathering Algorithms

Problem Definition

- Objective: Transmit sensed data from each sensor node to a base station
  - One round = BS collecting data from all nodes
- Goal is to maximize the number of rounds of communication before nodes die and network is inoperable
- Minimize energy AND reduce delay
  - Conflicting requirements



*Energy*Delay metric*

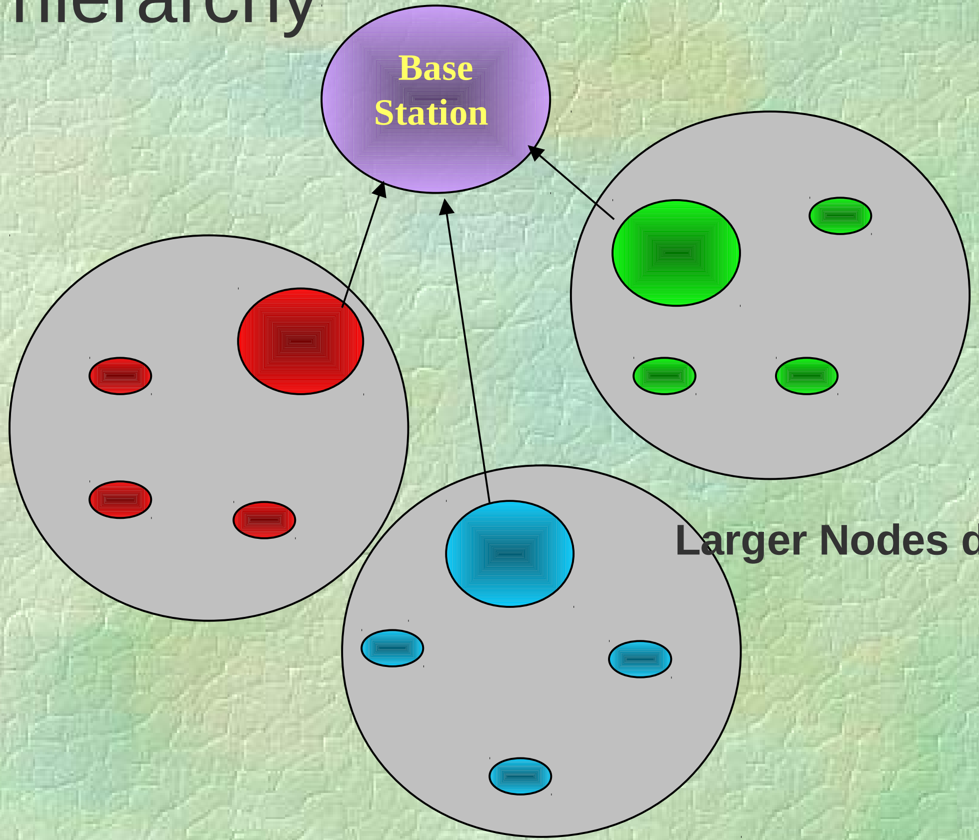
- Why energy * delay metric?
  - Find optimal balance to gather data quickly but in an energy efficient manner
  - Energy = Energy consumed per round
  - Delay = Delay per round (I.e. for all nodes to send packet to BS)
- Why is this metric important?
  - Time critical applications

Direct Transmission

- Direct Transmission
  - All nodes transmit to the base station (BS)
  - Very expensive since BS may be located very far away and nodes need more energy to transmit over longer distances
    - Farther the distance, greater the propagation losses, and hence higher the transmission power
  - All nodes must take turns transmitting to the BS so delay is high (N units for a N-node network)
  - Better scheme is to have fewer nodes transmit this far distance to lower energy costs and more simultaneous transmissions to lower delay

LEACH

- Low Energy Adaptive Clustering Hierarchy
  - Two-level hierarchy



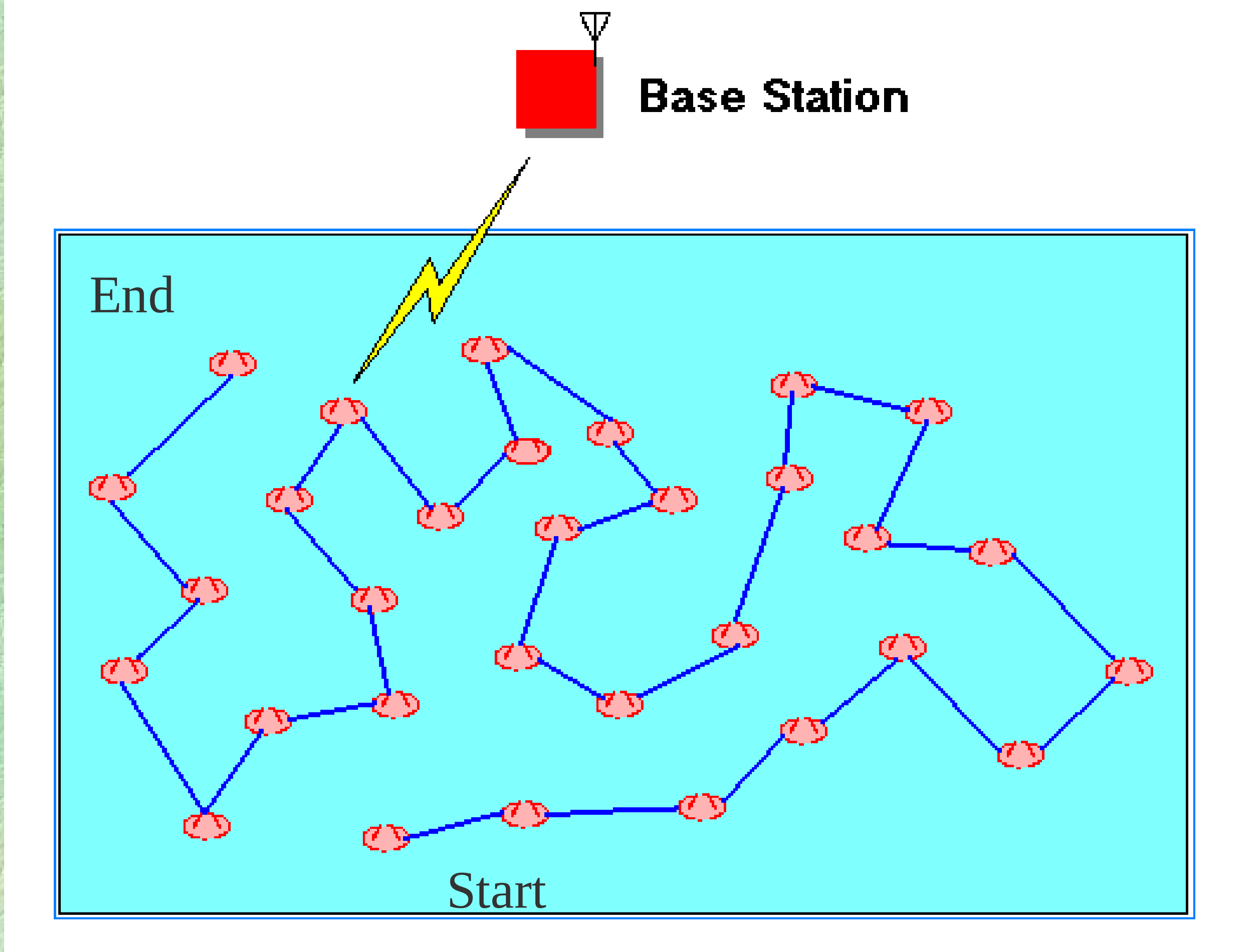
Scheme #1: PEGASIS

- Goals of PEGASIS (Power-Efficient GAthering for Sensor Information Systems)
  - Minimize distance nodes must transmit
  - Minimize number of leaders that transmit to BS
  - Minimize broadcasting overhead
  - Minimize number or messages leader needs to receive
  - Distribute work more equally among all nodes

PEGASIS

- Greedy Chain Algorithm
  - Start with node furthest away from BS
  - Add to chain closest neighbor to this node that has not been visited
  - Repeat until all nodes have been added to chain
  - Constructed before 1st round of communication and then reconstructed when nodes di
- Data fusion at each node (except end nodes)
  - Only one message is passed at every node
- Delay calculation: N units for an N-node network
  - Sequential transmission is assumed

PEGASIS



Scheme #2: Binary Scheme

- Chain-based as described in PEGASIS
- At each level node only transmits to another node
- All nodes receiving at any level rise to the next level
- Delay: O(log2 N)

Step 4:	**c3 → BS**			
Step 3:	**c3 ← c7**			
Step 2:	**c1 → c3**		**c5 → c7**	
Step 1:	**c0 → c1**	**c2 → c3**	**c4 → c5**	**c6 → c7**

MAC Protocols for WSN

MAC Protocols

- What is fundamentally different for MAC Protocol design in WSN?
  - Low-power operation is even more critical
  - Reduced coordination and synchronization is beneficial
  - Resilience to frequent node failures
  - Suitably blend with the network architecture
    - Probably application dependent
  - Scalability to support large number of nodes
    - Thousands of nodes likely
    - Limited bandwidth availability
    - Would the 802.11 family of protocols work?

TDM-Based MAC

- Considered for Clustered architecture
  - Nodes are organized into clusters
  - Each cluster has a *clusterhead*, that communicates directly with gateway or BS node
- TDMA MAC
  - The cluster head knows its members' IDs
  - Creates a simple TDM schedule, allocating time slots to members
  - Broadcasts schedule to members
  - Schedule may be periodically updated
  - Rotating cluster heads possible

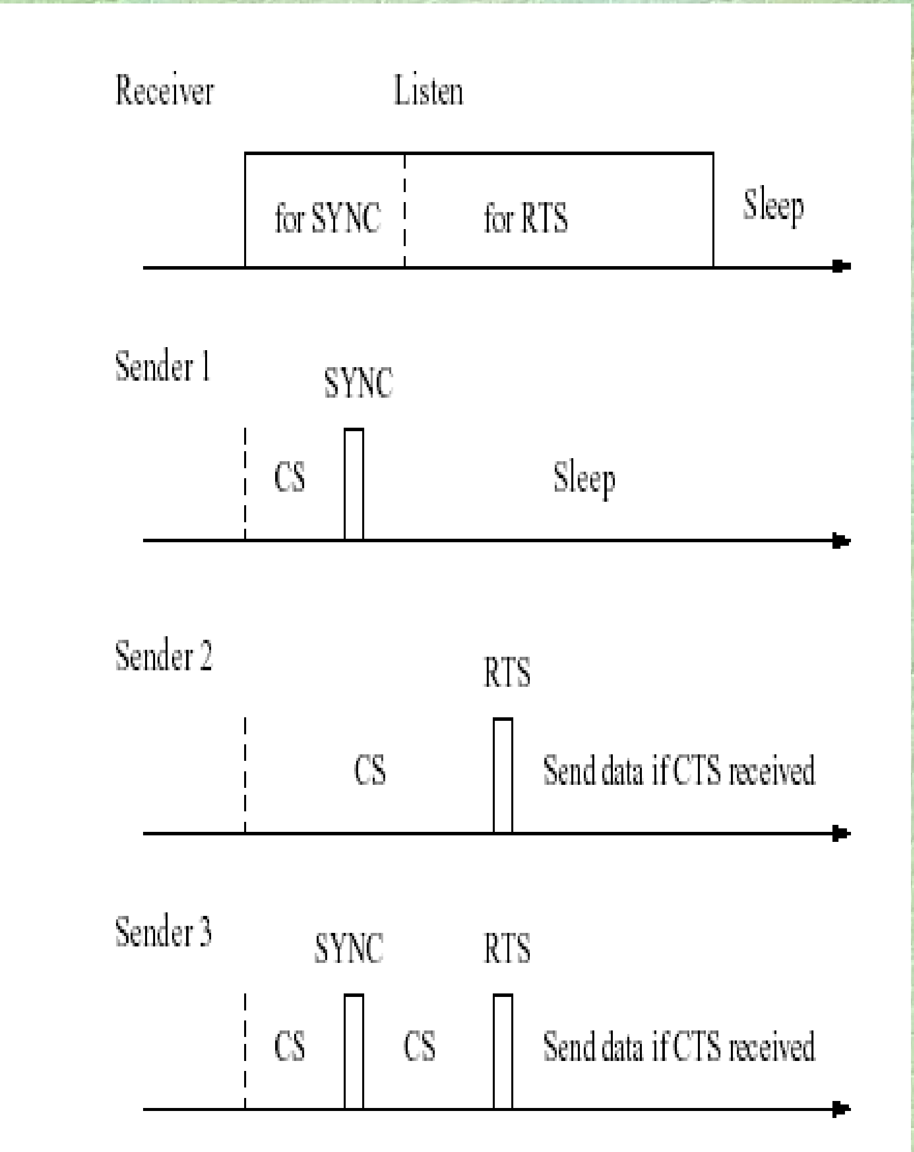
TDM-Based MAC, contd.

- Advantages:
  - Simple to coordinate within cluster
  - No collisions
  - Can be more energy-efficient: members wake up only when they have to send/receive data
- Disadvantages:
  - Adjoining clusters need to coordinate to operate in different channels (or frequencies)
  - TDM is not very scalable to large number of nodes: high delays possible
  - Nodes need to be synchronized within each cluster

S-MAC [Ye et. Al. 2002]

- Sensor-MAC Protocol proposed in 2002
- Assumptions
  - Network consists of several small nodes, deployed in an ad hoc manner
  - Nodes dedicated to a single or few collaborative applications: Per-node fairness is not critical
  - In-network processing assumed: e.g. data fusion, data aggregation, collab signal processing
  - Long idle periods and occasional burst of data: higher latency may be tolerated

S-MAC details, contd.



- Periodic Listen and Sleep Mode of operation
  - Each node sleeps for a while; wakes up and then communicates with its neighbors, as necessary.
  - Periodic synch among neighbors to reduce drift
  - Pair-wise or group-wise node synch
  - Nodes exchange schedule by broadcast
  - MAC is still needed to avoid collisions

Localization (Location Discovery) Algorithms

Location Information

- It is essential, in some applications, for each node to know its location
  - Sensed data coupled with loc. data and sent
- We need a cheap, low-power, low-weight, low form-factor, and reasonably accurate mechanism
- Global Positioning Sys (GPS) is not always feasible
  - GPS cannot work indoors, in dense foliage, etc.
  - GPS power consumption is very high
  - Size of GPS receiver and antenna will increase node form factor

Indoor Localization

- Use a fixed infrastructure
  - Beacon nodes are strategically placed
- Nodes receive beacon signals and measure:
  - Signal Strength
  - Signal Pattern
  - Time of arrival; Time difference of arrival
  - Angle of arrival
- Nodes use measurements from multiple beacons and use different multi-lateration techniques to estimate locations
- Accuracy of estimate depends on correlation between measured entity and distance

Indoor Localization

- Examples of Indoor Loc. Systems
  - RADAR (MSR), Cricket (MIT), BAT (AT&T), etc.
- Some approaches require a priori signal measurement and characterization and database creation
  - Node obtains distance estimate by using database
  - Not always practical to have database loaded in the individual node; only some nodes (e.g. gateway) might carry it.

Sensor Net. Localization

- No fixed infrastructure available
- Prior measurements are not always possible
- Basic idea:
  - Have a few sensor nodes who have known location information
  - These nodes sent periodic beacon signals
  - Other nodes use beacon measurements and triangulation, multi-lateration, etc. to estimate distance
- Following mechanisms presented in Savvides et. al. in ACM MobiCom 2001

Sensor Net. Localization, contd.

- Receiver Signal Strength Indicator (RSSI) was used to determine correlation to distance
  - Suitable for RF signals only
  - Very sensitive to obstacles, multi-path fading, environment factors (rain, etc.)
  - Was not found to have good experimental correlation
  - RF signal had good range, few 10metres
- RF and Ultrasound signals
  - The beacon node transmits an RF and an ultrasound signal to receiver
  - The time difference of arrival between 2 signals is used to measure distance
  - Range of up to 3 m, with 2cm accuracy

Localization algorithms

- Based on the time diff. of arrival
- Atomic Multi-lateration:
  - If a node receives 3 becaons, it can determine its location (similar to GPS)
- Iterative ML:
  - Some nodes not in direct range of beacons
  - Once an unknown node estimates its location, will send out a beacon
  - Multi-hop approach; Errors propagated
- Collaborative ML:
  - When 2+ nodes cannot receive 3 beacons (but can receive say 2), they collaborate