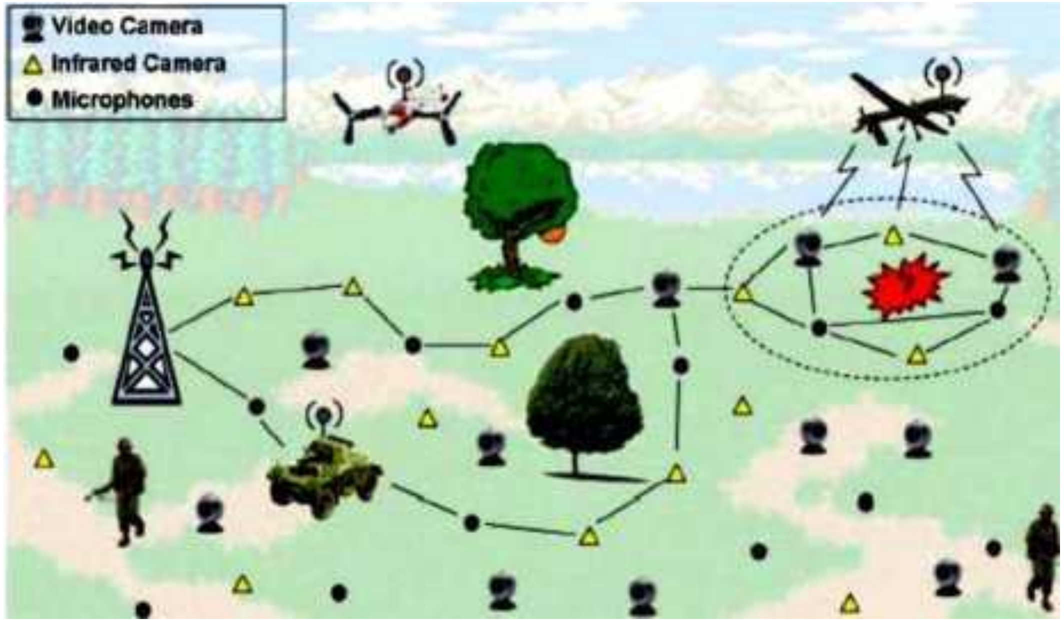
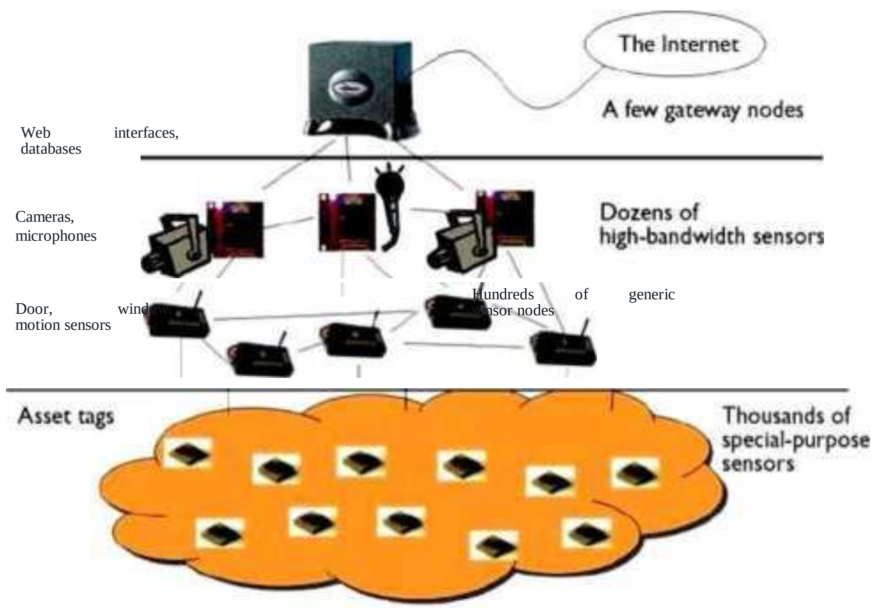
INFRASTRUCTURE ESTABLISHMENT

INTRODUCTION

- When a sensor network is first activated, various tasks must be performed to establish the necessary infrastructure
- In particular, each node must discover which other nodes it can directly communicate with
- Nodes near one another may wish to organize themselves into clusters.


Video Camera
Infrared Camera
Microphones


The Internet
A few gateway nodes
Web interfaces, databases
Cameras, microphones
Dozens of high-bandwidth sensors
Door,
motion sensors
Hundreds of generic
Asset tags
Thousands of special-purpose sensors

- we survey some common techniques used to establish such infrastructure

  -> topology control

  -> clustering, time

  -> synchronization and

  -> localization

Topology Control

- A sensor network node that first wakes up executes a protocol to discover which other nodes it can communicate with bidirectionally
- This set of neighbors is determined by the radio power of the nodes
- The problem of *topology control for a sensor network is how to set the radio range* for each node so as to minimize energy usage.

same transmission range:

- This *homogeneous topology control setting defines the critical transmitting range (CTR) problem: compute the minimum common transmitting* range *r such that the network is connected.*
- The solution to the CTR problem depends on information about the physical placement of the nodes.

- If the node locations are known a priori, or determined using the techniques.

  ->then the CTR problem has a simple answer.
- the critical transmitting range is the length of the longest edge of the nodes.

- This easily follows from the property that the MST contains the shortest edge across any partition of the nodes.
- The CTR problem has also been studied in a probabilistic context,where the node positions are not known but their locations come from a known distribution.

geometric random graphs

- The probabilistic theory best suited to the analysis of CTR is the theory of *geometric random graphs (GRG)*
- In the GRG setting, *n points are* distributed into a region according to some distribution, and then some aspect of the node placement is investigated.

- For example,it is shown that if *n points are randomly and uniformly* distributed in the unit square, then the critical transmission range is, with high probability,

$$r = c\sqrt{\frac{\log n}{n}},$$

- for some constant *c >0. Such asymptotic results can help a node* designer set the transmission range in advance,

- If nodes can have different transmission ranges, then the goal becomes to minimize

$$\sum_{1}^{n} r_i^{\alpha},$$

- where *ri denotes the range assigned to node i and α is the exponent* describing the power consumption law for the system.
- This is the *range assignment problem*

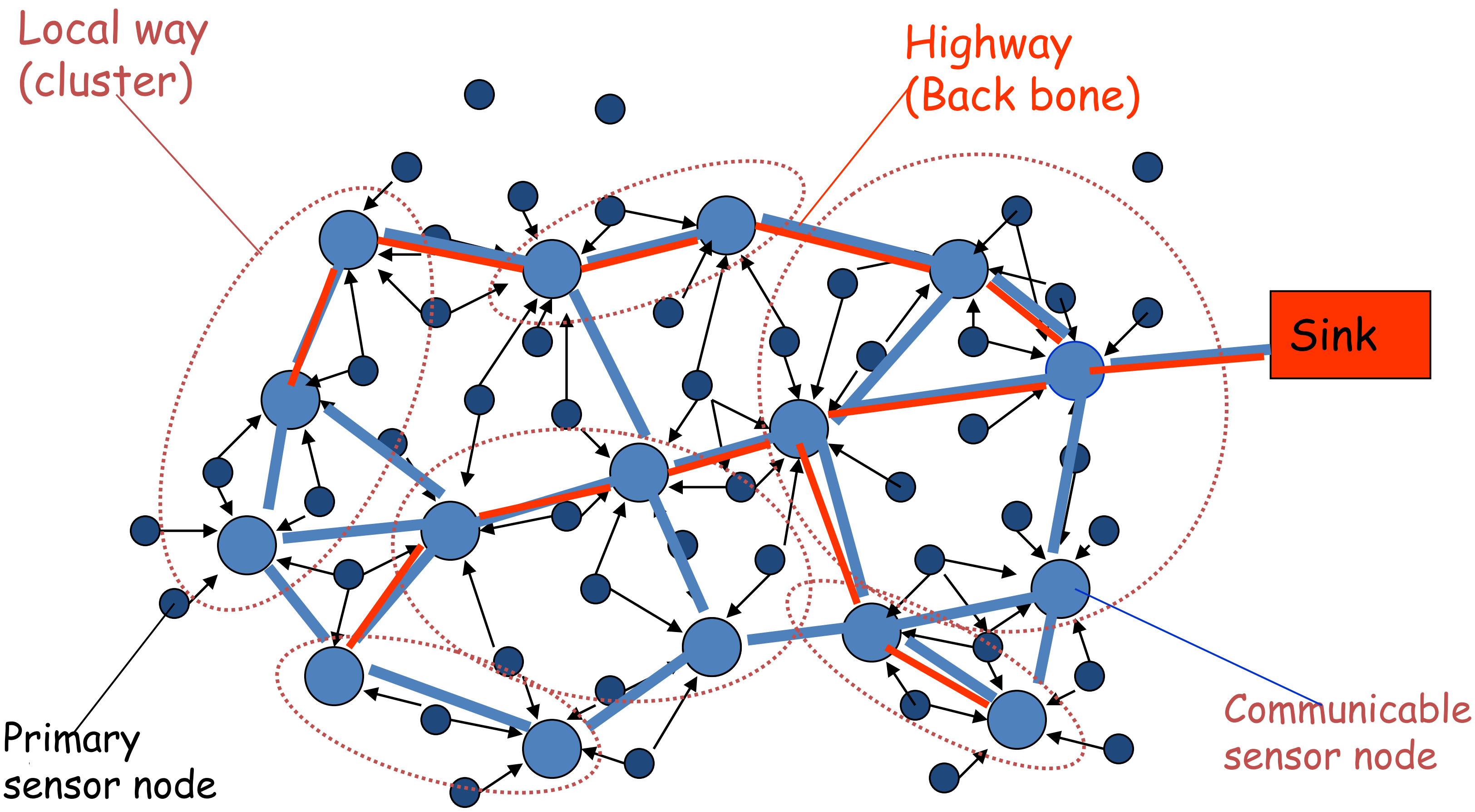
- Several protocols have been proposed that attempt to directly solve the CTR problem in a distributed way.
- For example, the COMPOW protocol of computes routing tables for each node at different power levels; a node selects the minimum transmit power so that its routing

table contains all other nodes.

Clustering

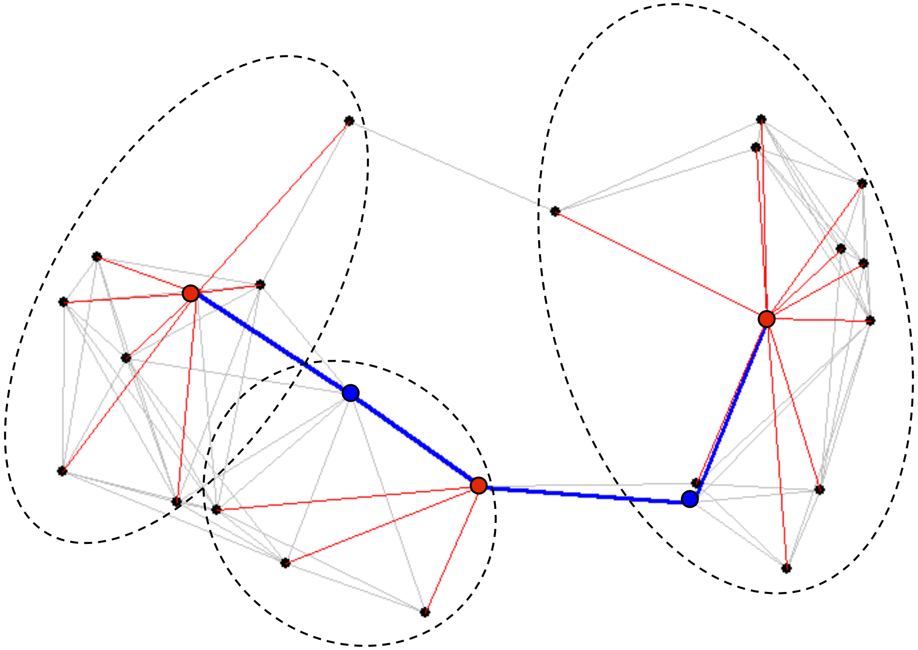
- The nodes in a sensor network often need to organize themselves into clusters.
- Clustering allows hierarchical structures to be built on the nodes and enables more efficient use of scarce resources, such as frequency spectrum, bandwidth, and power.
- These more capable nodes can naturally play the role of *cluster-heads.*

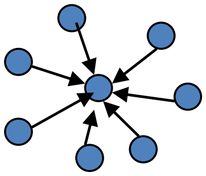
Flat and Structured WSNs: An Example



Flat (unstructured) Network

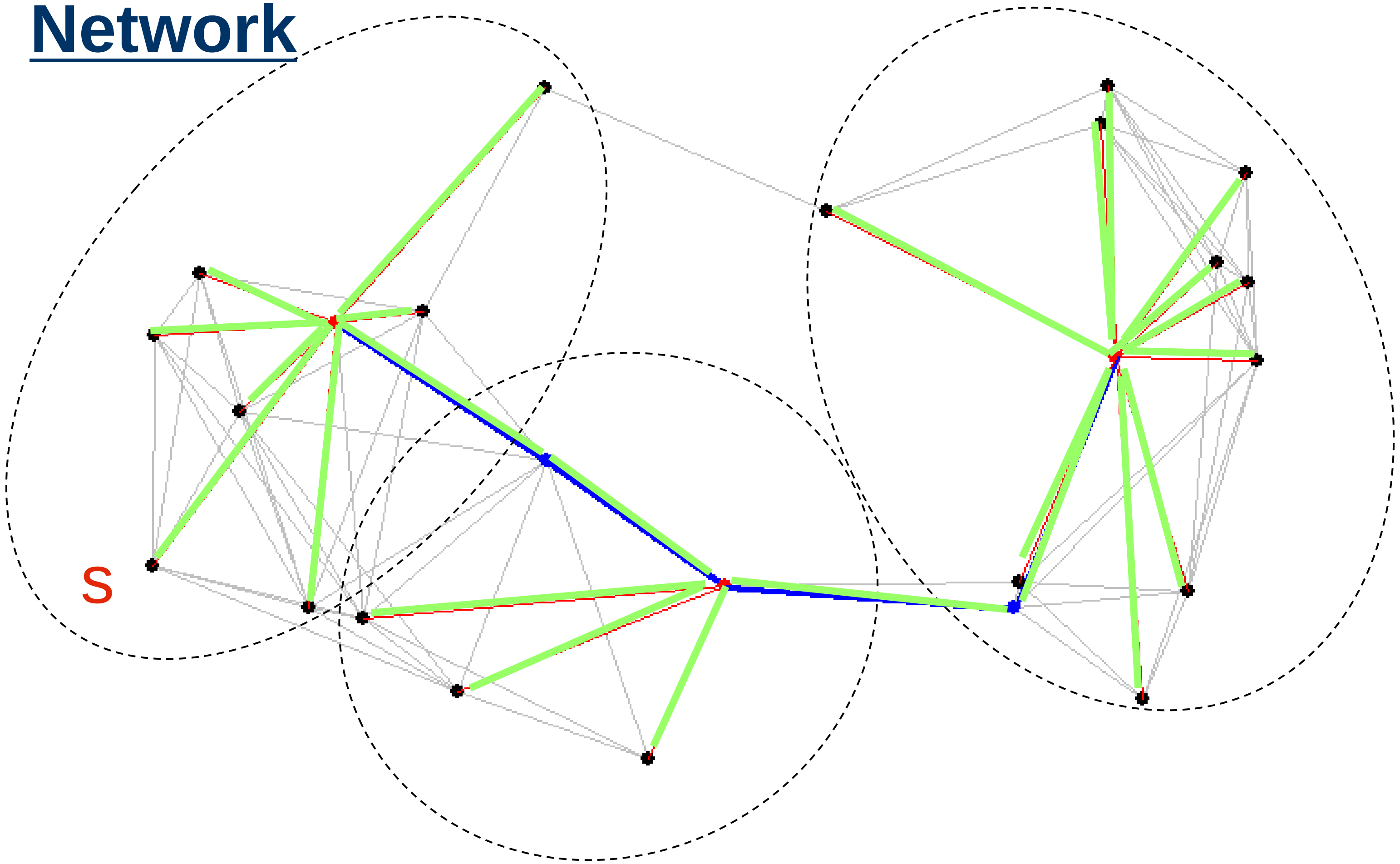
Clustering-based (structured) Network





MAC takes care collision problem but needs time and energy! e.g., Broadcast storm problem

Broadcast on A Cluster-Based Network



Time Synchronization

- Since the nodes in a sensor network operate independently, their clocks may not be, or stay, synchronized with one another.
- This can cause difficulties when trying to integrate and interpret information sensed at different nodes.
- For example, if a moving car is detected at two different times along a road, before we can even tell in what
- direction the car is moving, we have to be able to meaningfully compare the detection times.

- clearly we must be able to transform the two time readings into a common frame of reference before we can estimate the speed of the vehicle.
- Estimating time differences across nodes accurately is also important during node localization
- Solution is:::: TDMA radio schedule

Clocks and Communication Delays

- Computer clocks are based on hardware oscillators which provide a local time for each sensor network node.
- At real time *t the computer* clock indicates time *C(t).*
- For a perfect hardware clock, the derivative *dC(t)/dt should be equal* to 1.
- If this is not the case, we speak of clock *skew (also called drift).*

- The clock skew can actually change over time due to environmental conditions, such as temperature and humidity, but we will assume it stays bound close to 1, so that

$$\boxed{1 - \rho \leq \frac{dC(t)}{dt} \leq 1 + \rho,}$$

- where ρ *denotes the maximum skew. A typical value of ρ for today's* hardware is 10−6.

- *Send time: This is the time taken by the sender to construct the* message, including delays introduced by operating system calls, context switching, and data access to the network interface.
- *Access time: This is the delay incurred while waiting for access to* the transmission channel due to contention, collisions, and the like. The details of that are very MAC-specific.

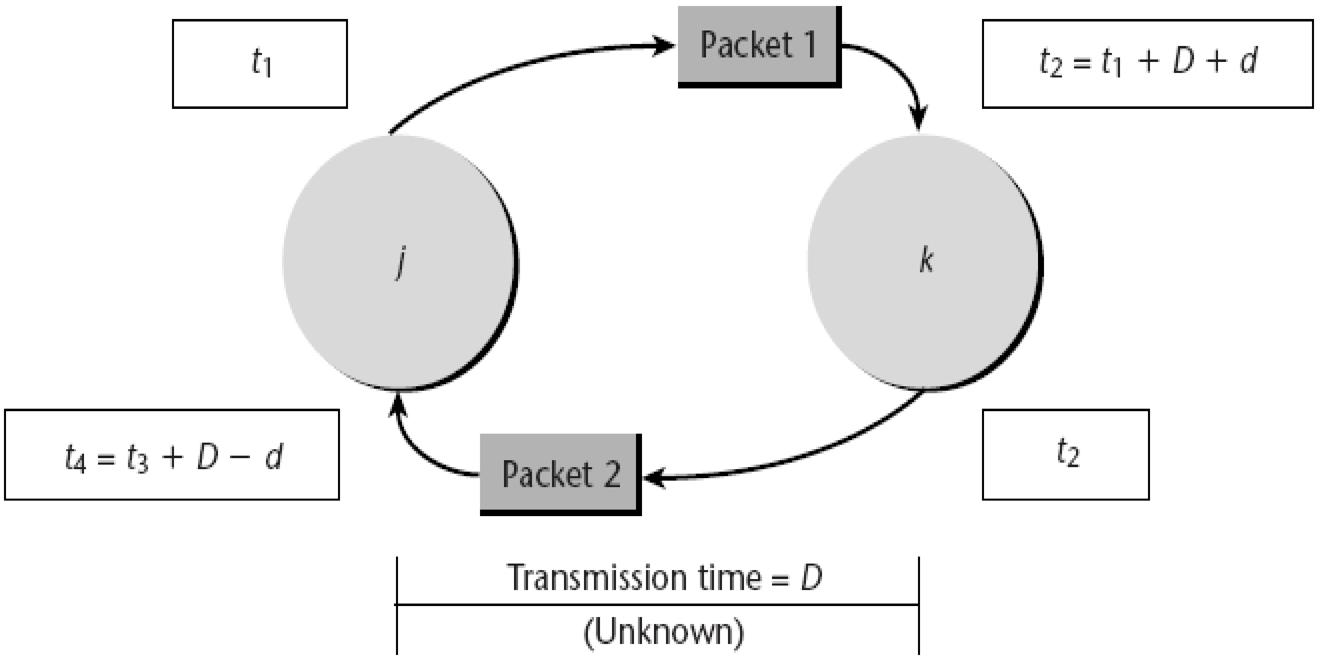
- *Propagation time: This is the time for the message to travel across* the channel to the destination node. It can be highly variable, from negligible for single-hop wireless transmission to very long in multihop wide-area transmissions.
- *Receive time: This is the time for the network interface on the* receiver side to get the message and notify the host of its arrival.This delay can be kept small by time-stamping the incoming packet inside the network driver's interrupt handler.

- If there is no clock skew and the (unknown) delay *D in the channel* between two nodes is constant and perfectly symmetric, then two
- nodes *i and j can estimate their phase difference d with three message* exchanges, as follows.

Condition

- Node *i reads its local clock with time value t1 and sends this in* a packet to node *j.*
- Node *j records the time t2 according to its own clock when the* packet was received. We must have $t2 = t1 + D + d$.
- Node *j, at time t3, sends a packet back to i containing t1, t2, and t3.*
- Node *i receives this packet at time t4. We must have* $t4 = t3+D-d$.Therefore, node *i can eliminate D from the above two equations* and compute $d = (t2 - t1 - t4 + t3)/2$.
- Finally, node *i sends the computed phase difference d back to* node *j.*

Node *j, at time t3, sends a packet back to i containing t1, t2, and t3.*



Clock phase difference estimation, using three message exchanges