Routing Protocols

Overview

- Routing in WSNs is challenging due to distinguish from other wireless networks like mobile ad hoc networks or cellular networks.
  - First, it is not possible to build a global addressing scheme for a large number of sensor nodes. Thus, traditional IP-based protocols may not be applied to WSNs. In WSNs, sometimes getting the data is more important than knowing the IDs of which nodes sent the data.
  - Second, in contrast to typical communication networks, almost all applications of sensor networks require the flow of sensed data from multiple sources to a particular BS.

Overview (cont.)

- Third, sensor nodes are tightly constrained in terms of energy, processing, and storage capacities. Thus, they require carefully resource management.
- Fourth, in most application scenarios, nodes in WSNs are generally stationary after deployment except for, may be, a few mobile nodes.
- Fifth, sensor networks are application specific, i.e., design requirements of a sensor network change with application.
- Sixth, position awareness of sensor nodes is important since data collection is normally based on the location.
- Finally, data collected by many sensors in WSNs is typically based on common phenomena, hence there is a high probability that this data has some redundancy.

Overview (cont.)

- The task of finding and maintaining routes in WSNs is nontrivial since energy restrictions and sudden changes in node status (e.g., failure) cause frequent and unpredictable topological changes.
- To minimize energy consumption, routing techniques proposed for WSNs employ some well-known routing strategies, e.g., data aggregation and in-network processing, clustering, different node role assignment, and data-centric methods were employed.

Outline

- 4.1 Routing Challenges and Design Issues in WSNs
- 4.2 Flat Routing
- 4.3 Hierarchical Routing
- 4.4 Location Based Routing
- 4.5 QoS Based Routing
- 4.6 Data Aggregation and Convergecast
- 4.7 Data Centric Networking
- 4.8 ZigBee
- 4.9 Conclusions

Routing Challenges and Design Issues in WSNs

Overview

- The design of routing protocols in WSNs is influenced by many challenging factors. These factors must be overcome before efficient communication can be achieved in WSNs.
  - **Node deployment**
  - **Energy considerations**
  - **Data delivery model**
  - **Node/link heterogeneity**
  - **Fault tolerance**
  - **Scalability**
  - **Network dynamics**
  - **Transmission media**
  - **Connectivity**
  - **Coverage**
  - **Data aggregation/convergecast**
  - **Quality of service**

Node Deployment

- Node deployment in WSNs is application dependent and affects the performance of the routing protocol.
- The deployment can be either deterministic or randomized.
- In deterministic deployment, the sensors are manually placed and data is routed through pre-determined paths.
- In random node deployment, the sensor nodes are scattered randomly creating an infrastructure in an ad hoc manner.

Energy Considerations

- Sensor nodes can use up their limited supply of energy performing computations and transmitting information in a wireless environment. Energy conserving forms of communication and computation are essential.
- In a multi-hop WSN, each node plays a dual role as data sender and data router. The malfunctioning of some sensor nodes due to power failure can cause significant topological changes and might require rerouting of packets and reorganization of the network.

Data Delivery Model

- Time-driven (continuous)
  - Suitable for applications that require periodic data monitoring
- Event-driven
  - React immediately to sudden and drastic changes
- Query-driven
  - Respond to a query generated by the BS or another node in the network
- Hybrid
- The routing protocol is highly influenced by the data reporting method

Node/Link Heterogeneity

- Depending on the application, a sensor node can have a different role or capability.
- The existence of a heterogeneous set of sensors raises many technical issues related to data routing.
- Even data reading and reporting can be generated from these sensors at different rates, subject to diverse QoS constraints, and can follow multiple data reporting models.

Fault Tolerance

- Some sensor nodes may fail or be blocked due to lack of power, physical damage, or environmental interferences
- It may require actively adjusting transmission powers and signaling rates on the existing links to reduce energy consumption, or rerouting packets through regions of the network where more energy is available

Scalability

- The number of sensor nodes deployed in the sensing area may be on the order of hundreds or thousands, or more.
- Any routing scheme must be able to work with this huge number of sensor nodes.
- In addition, sensor network routing protocols should be scalable enough to respond to events in the environment.

Network Dynamics

- Routing messages from or to moving nodes is more challenging since route and topology stability become important issues
- Moreover, the phenomenon can be mobile (e.g., a target detection/ tracking application).

Transmission Media

- In general, the required bandwidth of sensor data will be low, on the order of 1-100 kb/s. Related to the transmission media is the design of MAC.
  - TDMA (time-division multiple access)
  - CSMA (carrier sense multiple access)

Connectivity

- High node density in sensor networks precludes them from being completely isolated from each other.
- However, may not prevent the network topology from being variable and the network size from shrinking due to sensor node failures.
- In addition, connectivity depends on the possibly random distribution of nodes.

Coverage

- In WSNs, each sensor node obtains a certain view of the environment.
- A given sensor's view of the environment is limited in both range and accuracy.
- It can only cover a limited physical area of the environment.

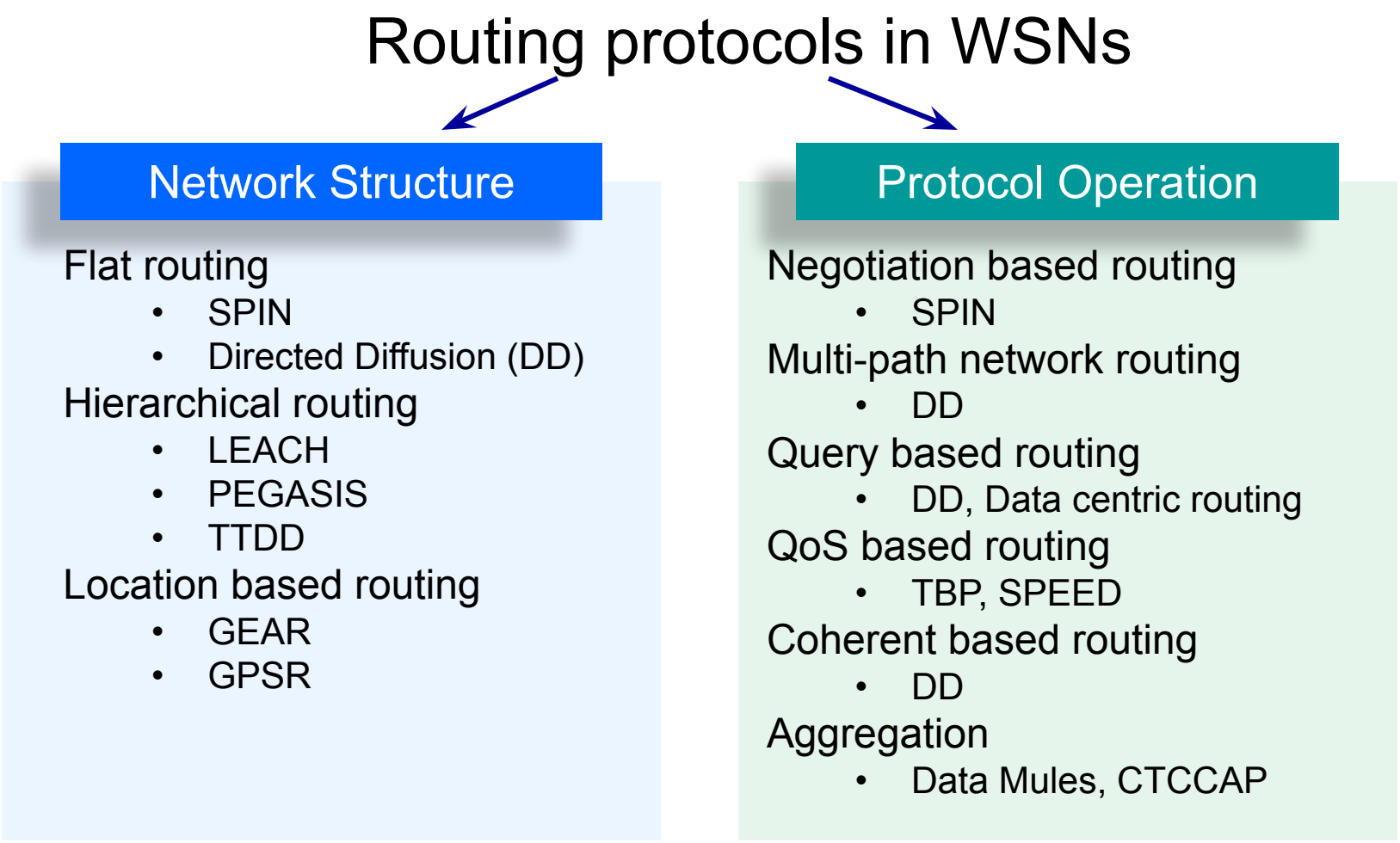
Data Aggregation/Convergecast

- Since sensor nodes may generate significant redundant data, similar packets from multiple nodes can be aggregated to reduce the number of transmissions.
- Data aggregation is the combination of data from different sources according to a certain aggregation function.
- Convergecasting is collecting information “upwards” from the spanning tree after a broadcast.

Quality of Service

- In many applications, conservation of energy, which is directly related to network lifetime.
- As energy is depleted, the network may be required to reduce the quality of results in order to reduce energy dissipation in the nodes and hence lengthen the total network lifetime.

Routing Protocols in WSNs: A taxonomy



Reference

- J. N. Al-Karaki and A. E. Kamal, “Routing techniques in wireless sensor networks: a survey,” IEEE Wireless Communications, vol. 11, no. 6, pp. 6-28, Dec. 2004.

Chapter 4.2
Flat Routing

Overview

- In flat network, each node typically plays the same role and sensor nodes collaborate together to perform the sensing task.
- Due to the large number of such nodes, it is not feasible to assign a global identifier to each node. This consideration has led to data centric routing, where the BS sends queries to certain regions and waits for data from the sensors located in the selected regions. Since data is being requested through queries, attribute-based naming is necessary to specify the properties of data.
- Prior works on data centric routing, e.g., SPIN and Directed Diffusion, were shown to save energy through data negotiation and elimination of redundant.

4.2.1
SPIN
Sensor Protocols for Information via Negotiation

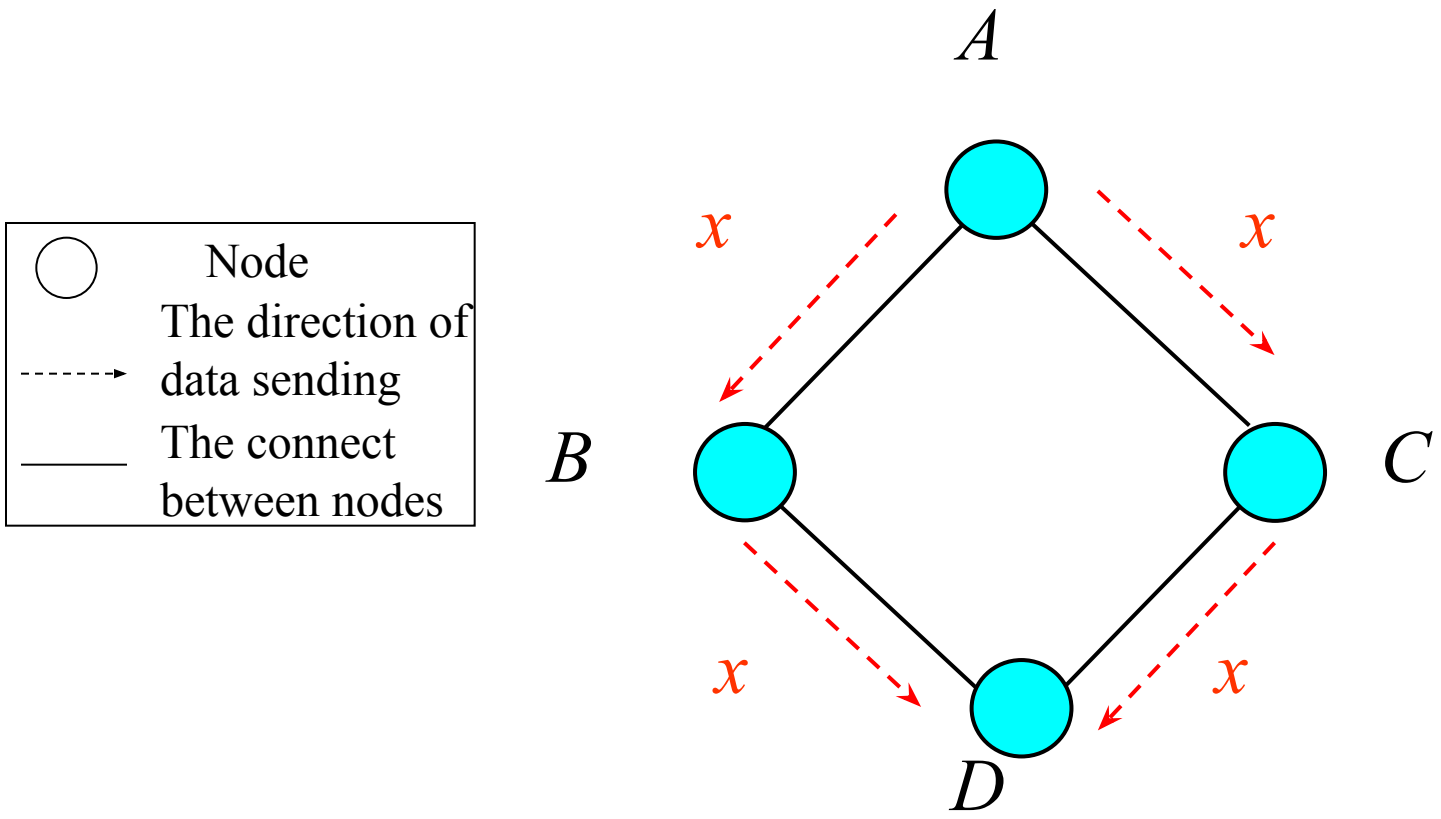
SPIN -Motivation

- Sensor Protocols for Information via Negotiation, SPIN
  - A Negotiation-Based Protocols for Disseminating Information in Wireless Sensor Networks.
- Dissemination is the process of distributing individual sensor observations to the whole network, treating all sensors as sink nodes
  - Replicate complete view of the environment
  - Enhance fault tolerance
  - Broadcast critical piece of information

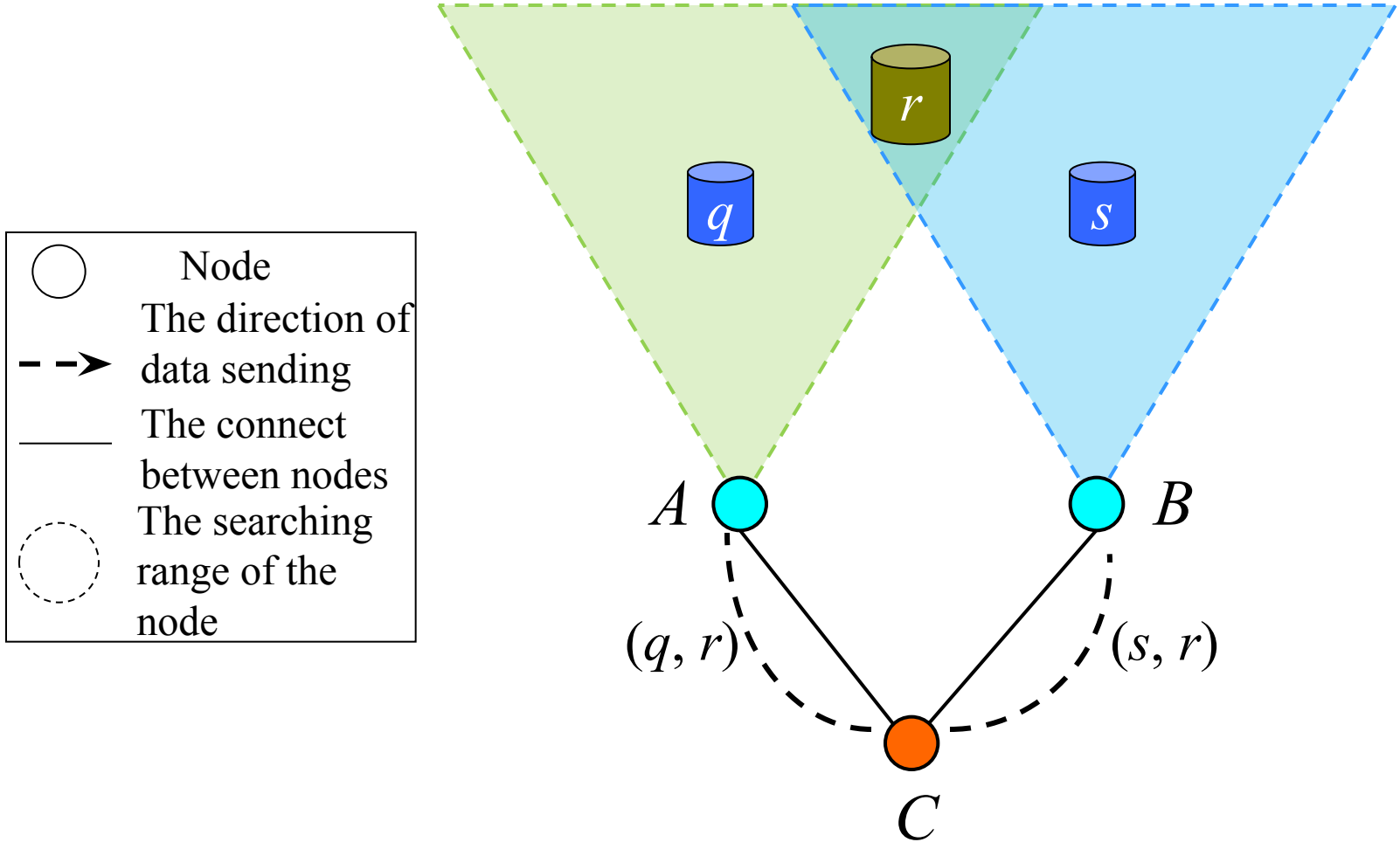
SPIN (cont.)- Motivation

- Flooding is the classic approach for dissemination
- Source node sends data to all neighbors
- Receiving node stores and sends data to all its neighbors
- Disseminate data quickly
- Deficiencies
  - Implosion
  - Overlap
  - Resource blindness

SPIN (cont.)-Implosion



SPIN (cont.)- Overlap



SPIN (cont.)- Resource blindness

- In flooding, nodes do not modify their activities based on the amount of energy available to them.

- A network of embedded sensors can be resource-aware and adapt its communication and computation to the state of its energy resource.

SPIN (cont.)

- Negotiation
  - Before transmitting data, nodes negotiate with each other to overcome implosion and overlap
  - Only useful information will be transferred
  - Observed data must be described by meta-data
- Resource adaptation
  - Each sensor node has resource manager
  - Applications probe manager before transmitting or processing data
  - Sensors may reduce certain activities when energy is low

SPIN (cont.)- Meta-Data

- Completely describe the data
  - Must be smaller than the actual data for SPIN to be beneficial
  - If you need to distinguish pieces of data, their meta-data should differ
- Meta-Data is application specific
  - Sensors may use their geographic location or unique node ID
  - Camera sensor may use coordinate and orientation

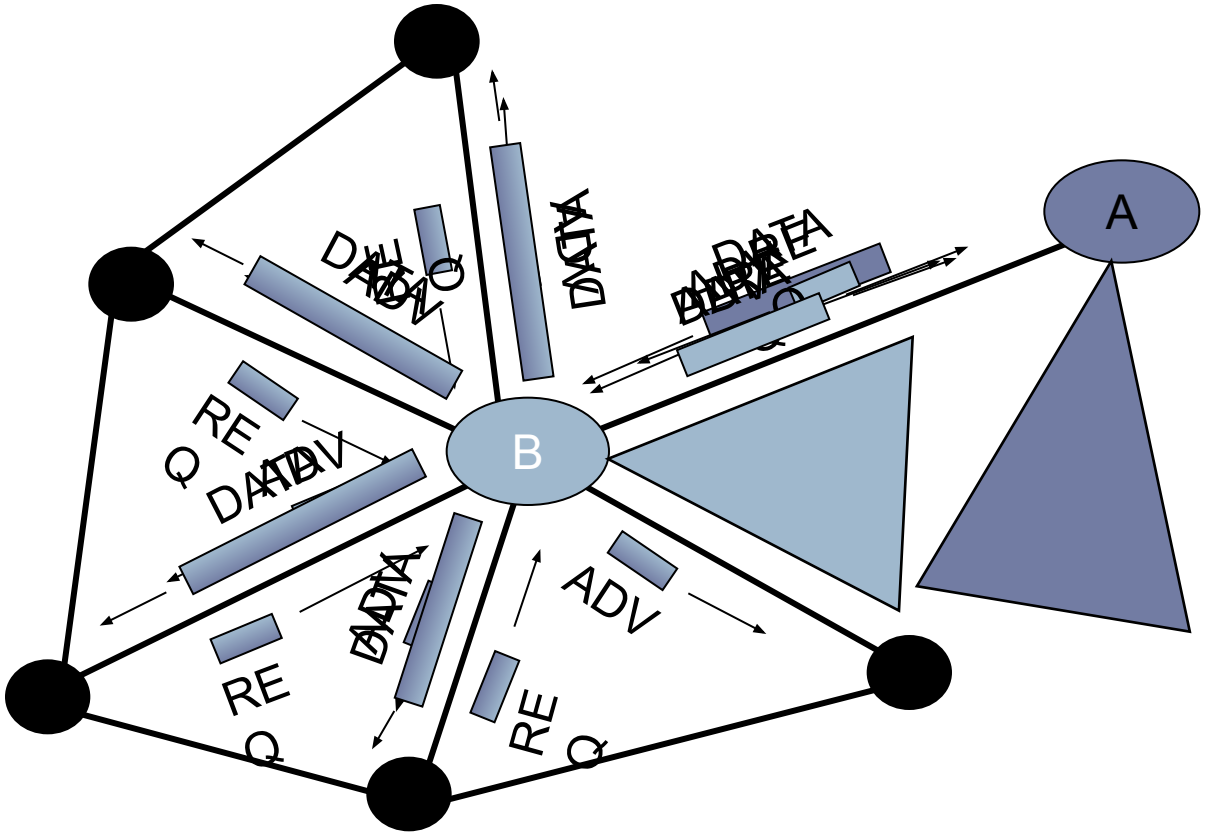
SPIN (cont.)- SPIN family

- Protocols of the SPIN family
  - SPIN-PP
    - It is designed for a point to point communication, i.e., hop-by-hop routing
  - SPIN-EC
    - It works similar to SPIN-PP, but, with an energy heuristic added to it
  - SPIN-BC
    - It is designed for broadcast channels
  - SPIN-RL
    - When a channel is lossy, a protocol called SPIN-RL is used where adjustments are added to the SPIN-PP protocol to account for the lossy channel.

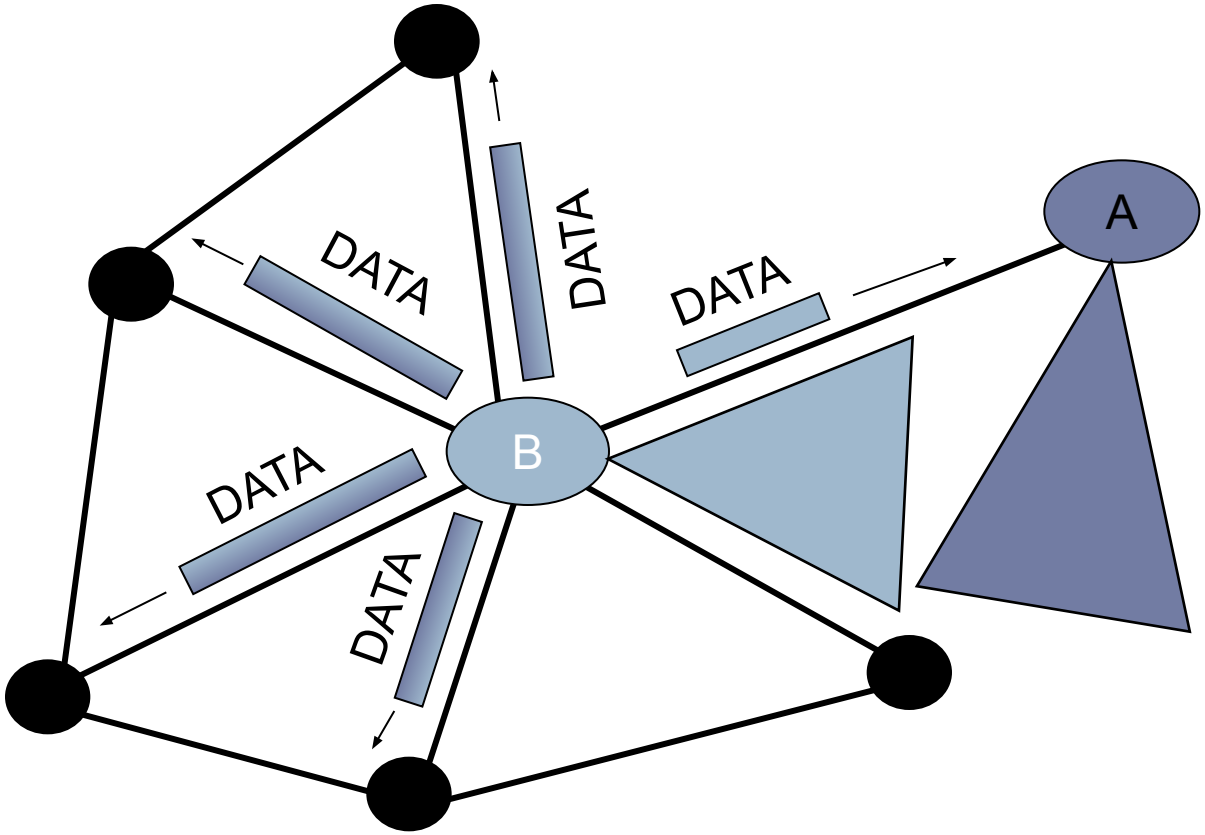
SPIN (cont.)- Three-stage handshake protocol

- SPIN-PP: A three-stage handshake protocol for point-to-point media
  - ADV – data advertisement
    - Node that has data to share can advertise this by transmitting an ADV with meta-data attached
  - REQ – request for data
    - Node sends a request when it wishes to receive some actual data
  - DATA – data message
    - Contain actual sensor data with a meta-data header
    - Usually much bigger than ADV or REQ messages

SPIN (3-Step Protocol)

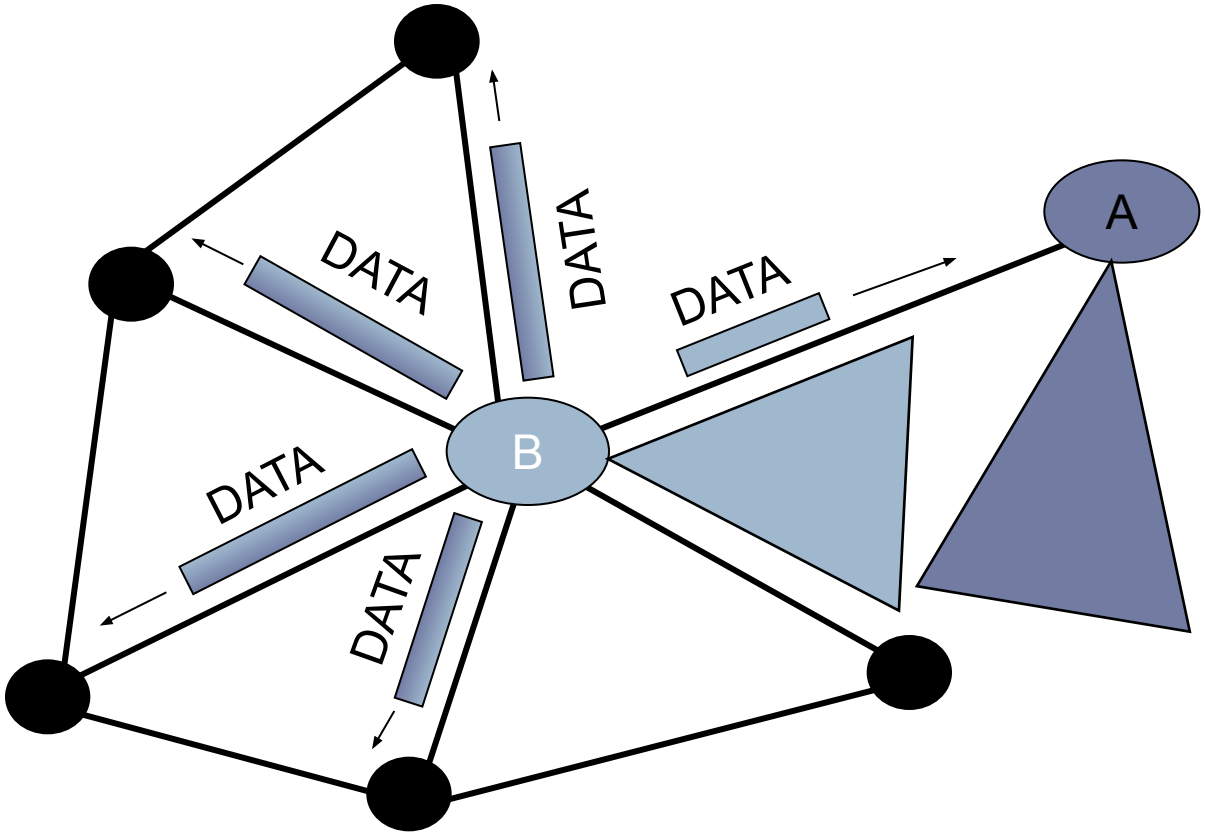


SPIN (3-Step Protocol)



Notice the color of the data packets sent by node B

SPIN (3-Step Protocol)



SPIN effective when DATA sizes are large :
REQ, ADV overhead gets amortized

SPIN (cont.)- SPIN-EC (Energy-Conserve)

- Add simple energy-conservation heuristic to SPIN-PP
  - SPIN-EC: SPIN-PP with a low-energy threshold
- Incorporate low-energy-threshold
- Works as SPIN-PP when energy level is high
- Reduce participation of nodes when approaching low-energy-threshold
  - When node receives data, it only initiates protocol if it can participate in all three stages with all neighbor nodes
  - When node receives advertisement, it does not request the data
- Node still exhausts energy below threshold by receiving ADV or REQ messages

SPIN (cont.)- Conclusion

- SPIN protocols hold the promise of achieving high performance at a low cost in terms of complexity, energy, computation, and communication
- Pros
  - Each node only needs to know its one-hop neighbors
  - Significantly reduce energy consumption compared to flooding
- Cons
  - Data advertisement cannot guarantee the delivery of data
    - If the node interested in the data are far from the source, data will not be delivered
    - Not good for applications requiring reliable data delivery, e.g., intrusion detection

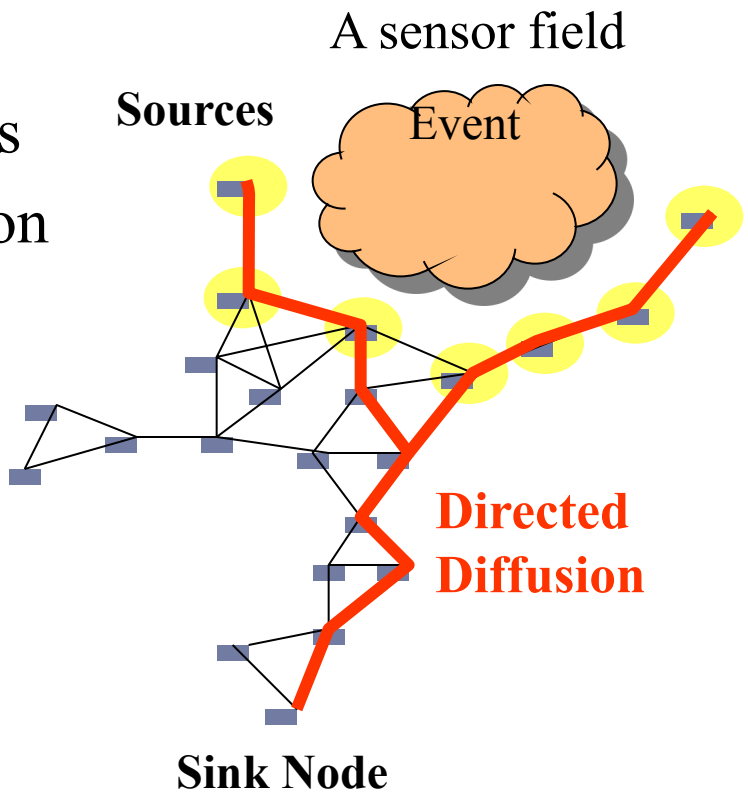
SPIN (cont.)- Reference

- J. Kulik, W.R. Heinzelman, and H. Balakrishnan, "Negotiation-based protocols for disseminating information in wireless sensor networks," *Wireless Networks*, Vol. 8, pp. 169-185, 2002.

4.2.2
Directed Diffusion
A Scalable and Robust Communication Paradigm for Sensor Networks

Overview

- Data-centric communication
  - Data is named by attribute-value pairs
  - Different form IP-style communication
    - End-to-end delivery service
  - e.g.
    - How many pedestrians do you observe in the geographical region X?



Overview (cont.)

- Data-centric communication (cont.)
  - Human operator's query (task) is *diffused*
  - Sensors begin collecting information about query
  - Information returns along the reverse path
  - Intermediate nodes *aggregate* the data
    - Combing reports from sensors
- Directed Diffusion is an important milestone in the data centric routing research of sensor networks

Directed Diffusion

- Typical IP based networks
  - Requires unique host ID addressing
  - Application is end-to-end

- Directed diffusion – use publish/subscribe
  - Inquirer expresses an interest, *I*, using attribute values
  - Sensor sources that can service *I*, reply with data

Directed Diffusion (cont.)

- Directed diffusion consists of
  - Interest - Query which specifies what a user wants
  - Data - Collected information
  - Gradient
    - Direction and data-rate
    - Events start flowing towards the originators of interests
  - Reinforcement
    - After the sink starts receiving events, it reinforces at least one neighbor to draw down higher quality events

Data Naming

- Expressing an Interest
  - Using attribute-value pairs
  - e.g.,

```
Type = Wheeled vehicle      // detect vehicle location
Interval = 20 ms            // send events every 20ms
Duration = 10 s             // Send for next 10 s
Rect = [-100,100, 200,400]  // from sensors in this area
```

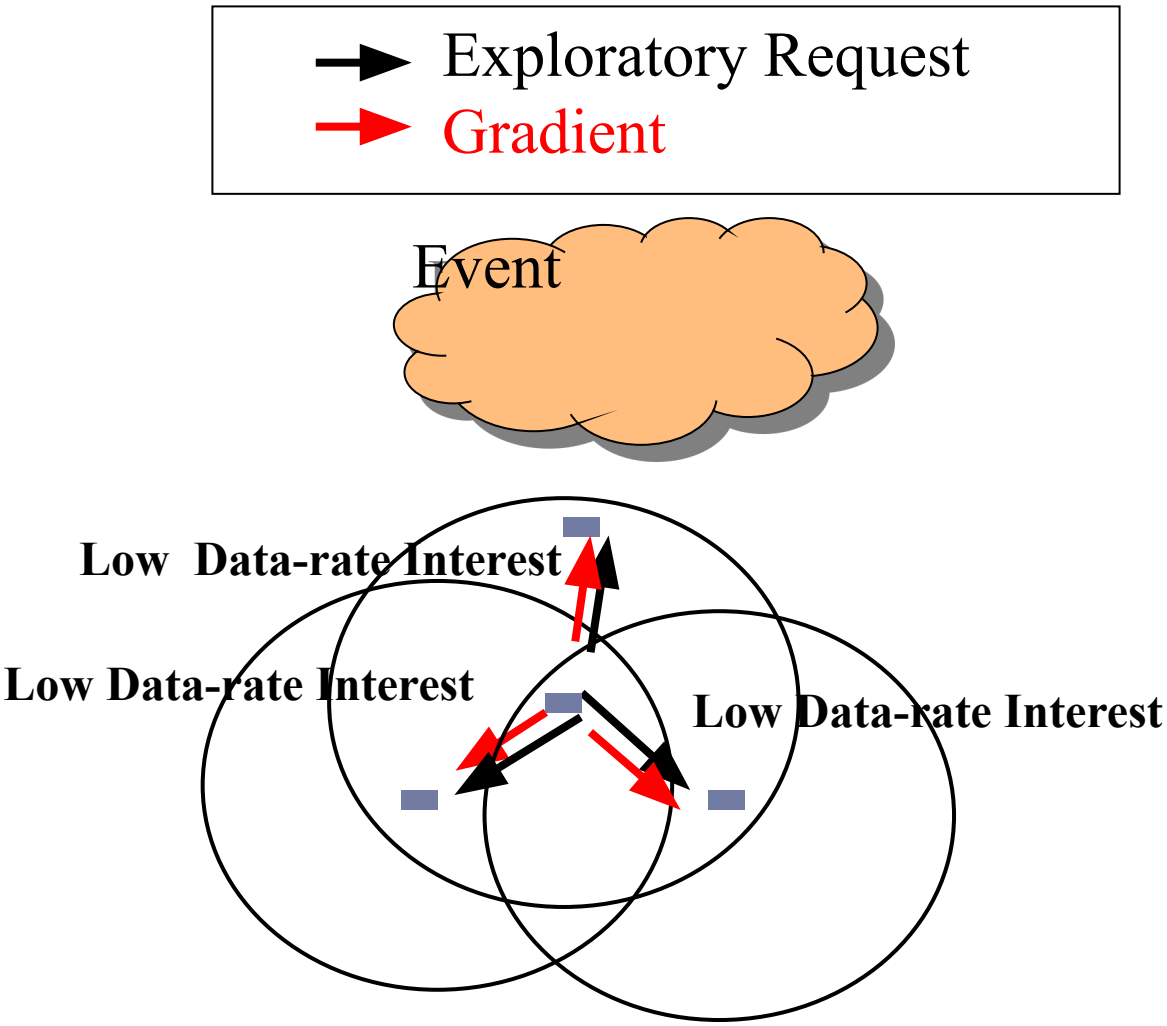
Interests and Gradients

- Interest propagation
  - The sink broadcasts an interest
  - Exploratory interest with low data-rate
  - Neighbors update interest-cache and forwards it
    - Flooding
    - Geographic routing
    - Use cached data to direct interests
- Gradient establishment
  - Gradient set up to upstream neighbor
  - Low data-rate gradient
    - Few packets per unit time needed

Gradient Set Up

- Inquirer (sink) broadcasts exploratory interest, ***i*1**
  - Intended to discover routes between source and sink
- Neighbors update interest-cache and forwards ***i*1**
- Gradient for ***i*1** set up to upstream neighbor
  - No source routes
  - Gradient – a weighted reverse link
  - Low gradient Few packets per unit time needed

Exploratory Gradient

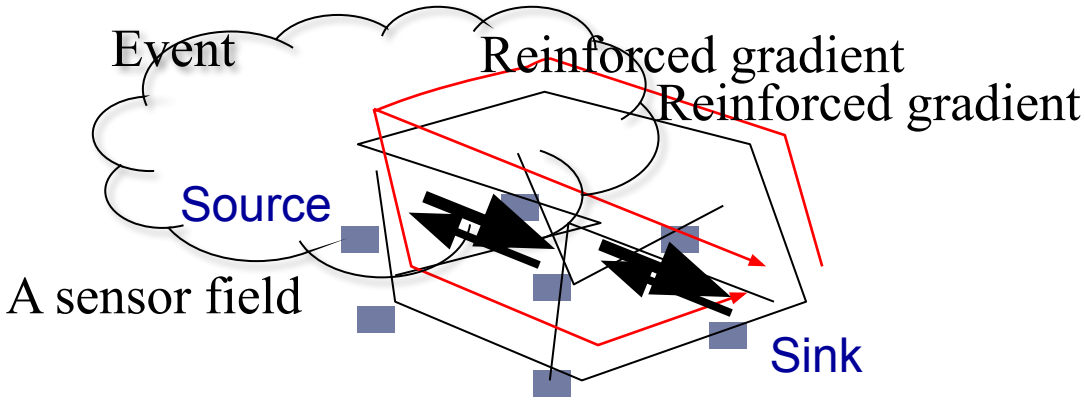


Data Propagation

- A sensor node that detects a target
  - Search its interest cache
  - Compute the highest requested data-rate among all its outgoing gradients
  - Data message is unicast individually
- A node that receives a data message
  - Find a matching interest entry in its cache
  - Check the data cache for loop prevention
  - Re-send the data to neighbors

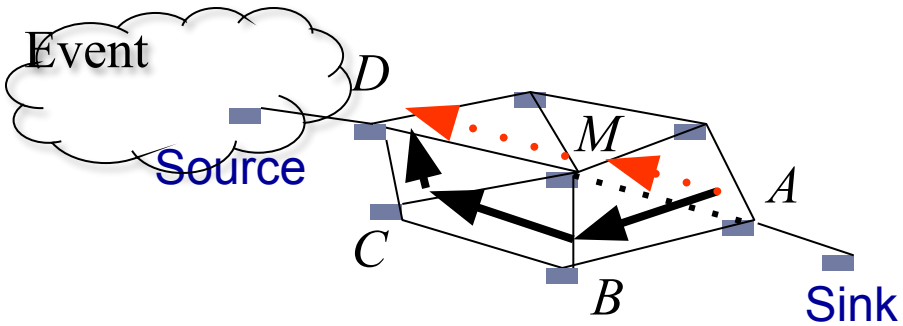
Reinforcement (1/4)

- Positive reinforcement
  - Sink selects the neighboring node
  - Original interest message but with high data-rate
  - Neighboring node must also reinforce at least one neighbor
  - Low-delay path is selected
  - Exploratory gradients still exist: useful for faults



Reinforcement (2/4)

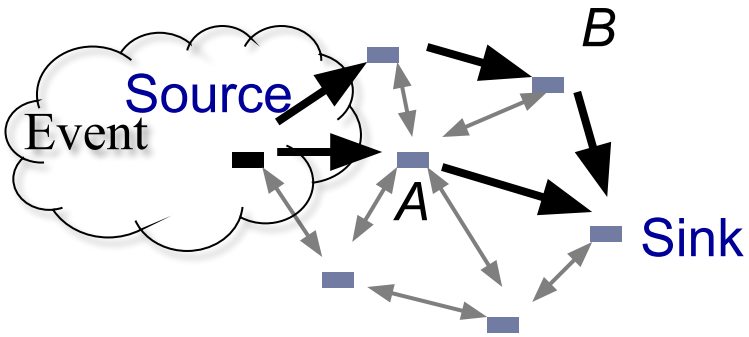
- Path failure and recovery
  - Link failure detected by reduced rate, data loss
    - Choose next best link (i.e., compare links based on infrequent exploratory downloads)
  - Negatively reinforce lossy link
    - Either send interest with base (exploratory) data rate or allow neighbor's cache to expire over time



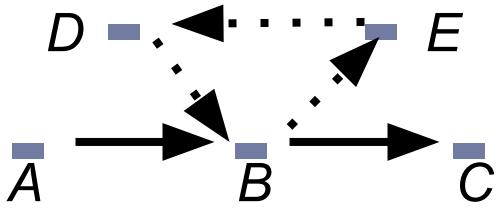
Link *A*-*M* lossy
A reinforces *B*
B reinforces *C*
C reinforces *D*
or
A negative reinforces *M*
M negative reinforces *D*

Reinforcement (3/4)

- Multipath routing
  - Consider each gradient's link quality
- Using negative reinforcement
  - Path Truncation
  - Loop removal
    - For resource saving
  - Ex:
    - *B* gets same data from both *A* and *D*, but *D* always delivers late due to looping
    - *B* negative reinforces *D*, *D* negative reinforces *E*, *E* negative reinforces *B*
    - Loop $B \rightarrow E \rightarrow D \rightarrow B$ eliminated
  - Conservative negative reinforces useful for fault resilience



Multiple paths



A removable loop

Design Considerations

- Design Space for Diffusion

Diffusion element	Design Choices
Interest Propagation	•Flooding •Constrained or directional flooding based on location •Directional propagation based on previously cached data
Data Propagation	•Reinforcement to single path delivery •Multipath delivery with selective quality along different paths • Multipath delivery with probabilistic forwarding
Data caching and aggregation	•For robust data delivery in the face of node failure •For coordinated sensing and data reduction • For directing interests
Reinforcement	•Rules for deciding when to reinforce •Rules for how many neighbors to reinforce •Negative reinforcement mechanisms and rules

Directed Diffusion: Pros & Cons

- Different from SPIN in terms of on-demand data querying mechanism
  - Sink floods interests only if necessary (lots of energy savings)
  - In SPIN, sensors advertise the availability of data
- Pros
  - Data centric: All communications are neighbor to neighbor with no need for a node addressing mechanism
  - Each node can do aggregation & caching
- Cons
  - On-demand, query-driven: Inappropriate for applications requiring continuous data delivery, e.g., environmental monitoring
  - Attribute-based naming scheme is application dependent
    - For each application it should be defined a priori
    - Extra processing overhead at sensor nodes

Conclusions

- Directed diffusion, a paradigm proposed for event monitoring sensor networks
- Directed Diffusion has some novel features - data-centric dissemination, reinforcement-based adaptation to the empirically best path, and in-network data aggregation and caching.
- Notion of gradient (exploratory and reinforced)
- Energy efficiency achievable
- Diffusion mechanism resilient to fault tolerance
  - Conservative negative reinforcements proves useful

References

- C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks,” *in the Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks* (*MobiCom’00*), August 2000.
- C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, “Directed Diffusion for Wireless Sensor Networking,” IEEE/ACM Transactions on Networking, Vol. 11, No. 1, Feb. 2003.

Chapter 4.3
Hierarchical Routing

Overview

- In a hierarchical architecture, higher energy nodes can be used to process and send the information while low energy nodes can be used to perform the sensing of the target.
- Hierarchical routing is mainly two-layer routing where one layer is used to select cluster heads and the other layer is used for routing.
- Hierarchical routing (or cluster-based routing), e.g., LEACH, PEGASIS, TTDD, is an efficient way to lower energy consumption within a cluster and by performing data aggregation and fusion in order to decrease the number of transmitted messages to the base stations.

4.3.1
LEACH
Low-Energy Adaptive Clustering Hierarchy

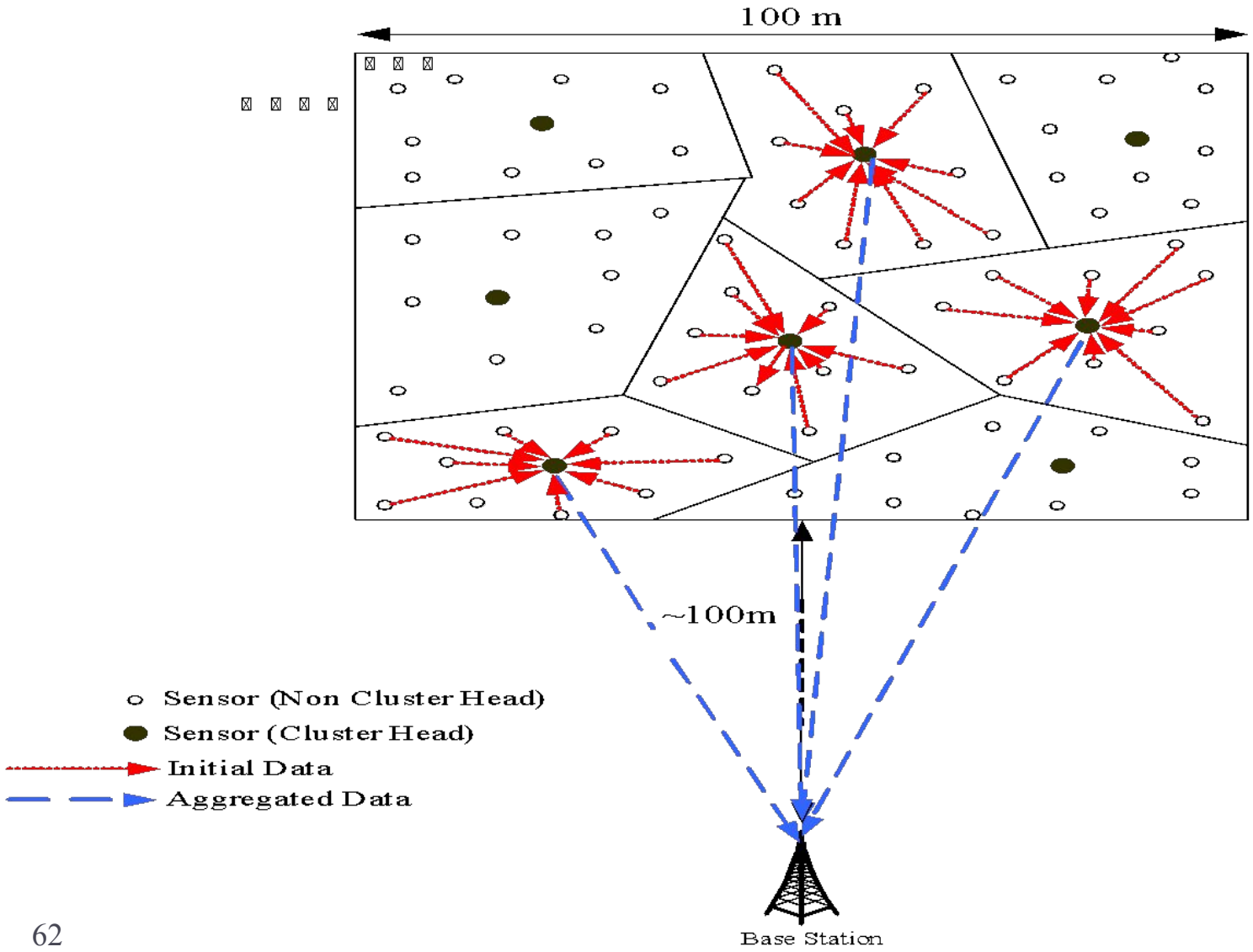
LEACH

- LEACH (Low-Energy Adaptive Clustering Hierarchy), a clustering-based protocol that minimizes energy dissipation in sensor networks.
- LEACH outperforms classical clustering algorithms by using adaptive clusters and rotating cluster-heads, allowing the energy requirements of the system to be distributed among all the sensors.
- LEACH is able to perform local computation in each cluster to reduce the amount of data that must be transmitted to the base station.
- LEACH uses a CDMA/TDMA MAC to reduce inter-cluster and intra-cluster collisions.

LEACH (cont.)

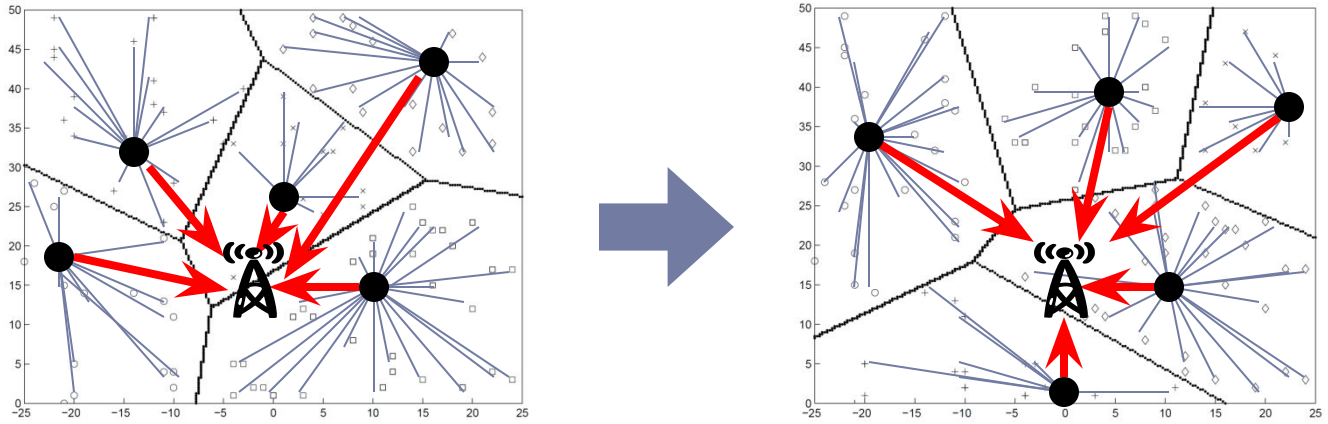
- Sensors elect themselves to be local cluster-heads at any given time with a certain probability.
- Each sensor node joins a cluster-head that requires the minimum communication energy.
- Once all the nodes are organized into clusters, each cluster-head creates a transmission schedule for the nodes in its cluster.
- In order to balance the energy consumption, the cluster-head nodes are not fixed; rather, this position is self-elected at different time intervals.

LEACH


100 m
~100m
Base Station
Sensor (Non Cluster Head)
Sensor (Cluster Head)
Initial Data
Aggregated Data

LEACH: Adaptive Clustering

- Periodic independent self-election
  - Probabilistic
- CSMA MAC used to advertise
- Nodes select advertisement with strongest signal strength
- Dynamic TDMA cycles

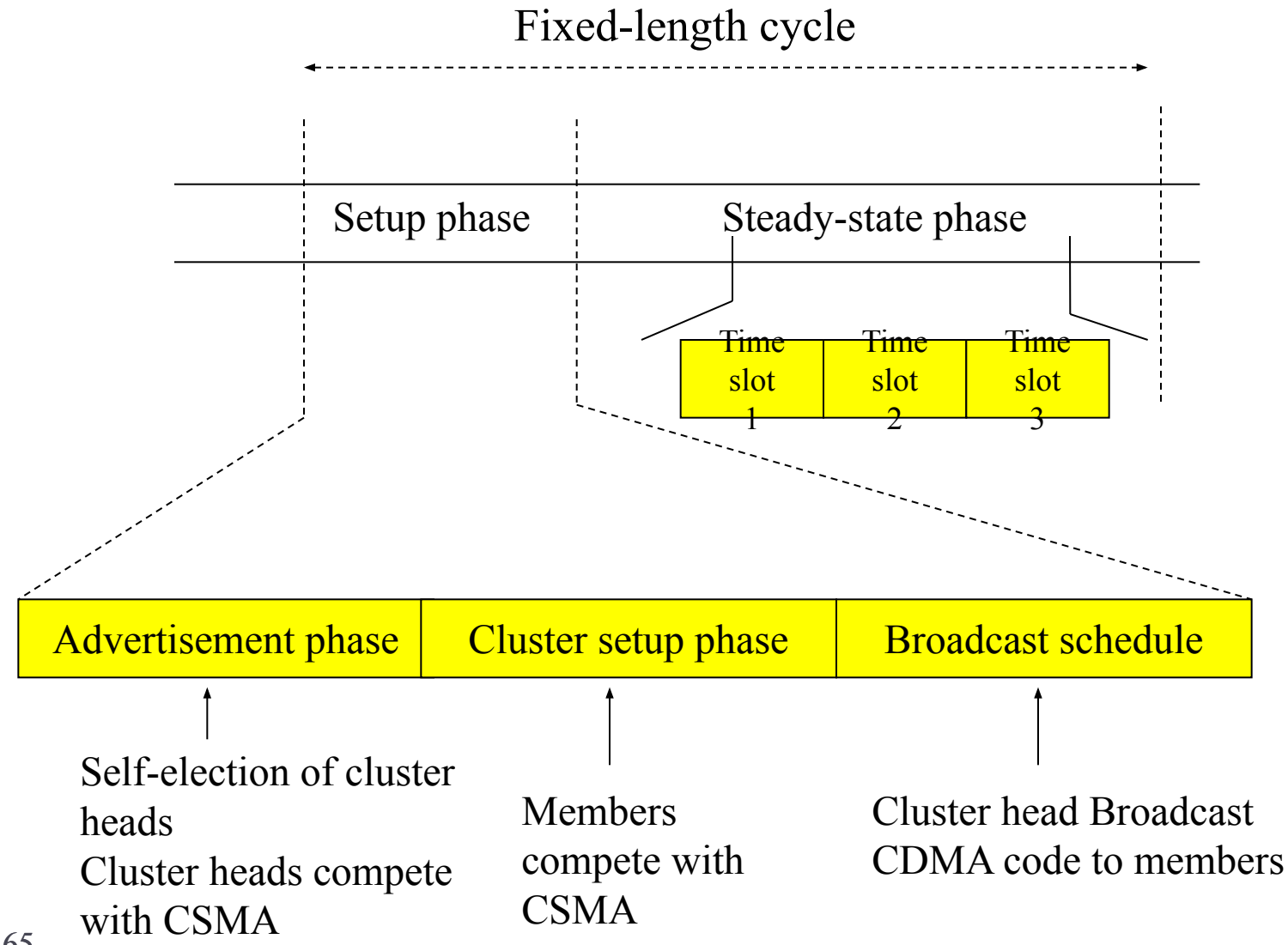


All nodes marked with a given symbol belong to the same cluster, and the cluster head nodes are marked with a •.

Algorithm

- Periodic process
- Two phases per round:
  - Setup phase
    - Advertisement: Execute election algorithm
    - Members join to cluster
    - Cluster-head broadcasts schedule
  - Steady-State phase
    - Data transmission to cluster-head using TDMA
    - Cluster-head transfers data to BS (Base Station)

Algorithm (cont.)



Algorithm Summary

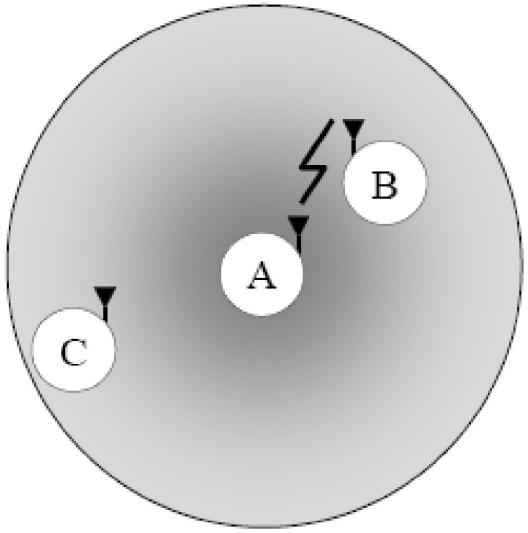
- Set-up phase
  - Node *n* choosing a random number *m* between 0 and 1
  - If $m < T(n)$ for node *n*, the node becomes a cluster-head where

$$T(n) = \begin{cases} \dfrac{P}{1 - P[r * \bmod(1/P)]} & if \ \ n \in G \\ 0 & otherwise, \end{cases}$$

  - where P = the desired percentage of cluster heads (e.g., P= 0.05), r=the current round, and G is the set of nodes that have not been cluster-heads in the last $1/P$ rounds. Using this threshold, each node will be a cluster-head at some point within 1/P rounds. During round 0 (r=0), each node has a probability P of becoming a cluster-head.

Algorithm Summary (cont.)

- Set-up phase
  - Cluster heads assign a TDMA schedule for their members where each node is assigned a time slot when it can transmit.
  - Each cluster communications using different CDMA codes to reduce interference from nodes belonging to other clusters.
- TDMA intra-cluster
- CDMA inter-cluster
  - Spreading codes determined randomly
  - Broadcast during advertisement phase

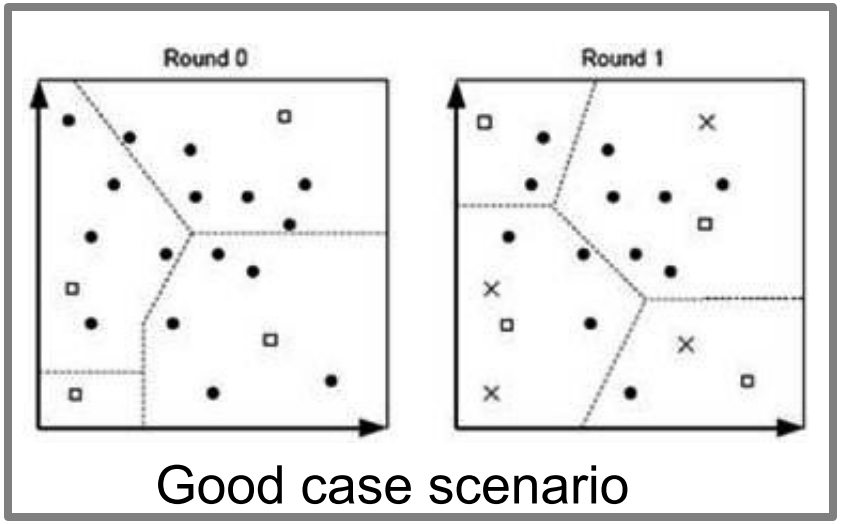


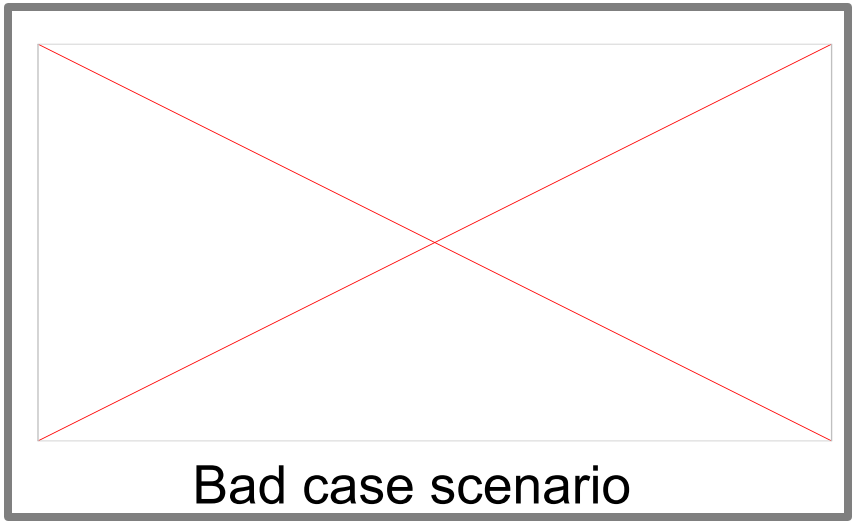
Algorithm Summary (cont.)

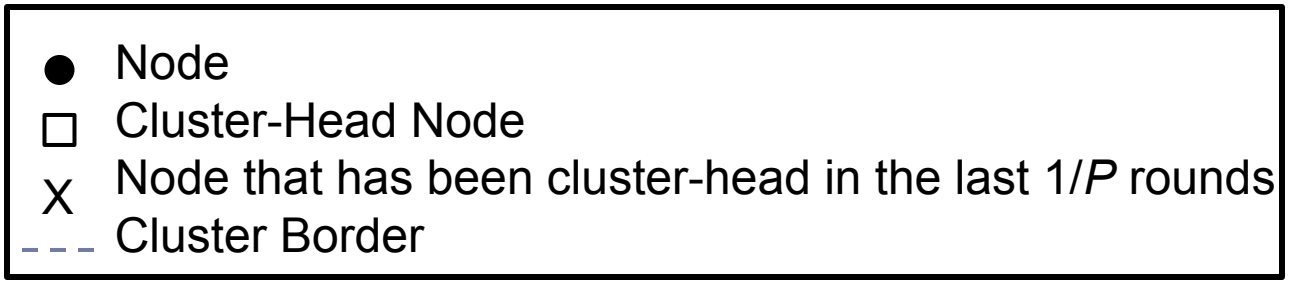
- Steady-state phase
  - All source nodes send their data to their cluster heads
  - Cluster heads perform data aggregation/fusion through local transmission
  - Cluster heads send aggregated data back to the BS using a single direct transmission

An Example of a LEACH Network

- While neither of these diagrams is the optimum scenario, the second is better because the cluster-heads are spaced out and the network is more properly sectioned







Conclusions

- Advantages
  - Increases the lifetime of the network
  - Even drain of energy
  - Distributed, no global knowledge required
  - Energy saving due to aggregation by CHs
- Disadvantages
  - LEACH assumes all nodes can transmit with enough power to reach BS if necessary (e.g., elected as CHs)
  - Each node should support both TDMA & CDMA
  - Need to do time synchronization
  - Nodes use single-hop communication

Reference

- W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless sensor networks,” *Proceedings of the 33rd Hawaii International Conference on System Sciences*, January 2000.

Chapter 4.4
Location Based Routing

Overview

- Sensor nodes are addressed by means of their locations.
  - The distance between neighboring nodes can be estimated on the basis of incoming signal strengths.
  - Relative coordinates of neighboring nodes can be obtained by exchanging such information between neighbors.
- To save energy, some location based schemes demand that nodes should go to sleep if there is no activity.
- More energy savings can be obtained by having as many sleeping nodes in the network as possible.
- Hereby, two important location based routing protocols, GEAR and GPSR, are introduced.
  - Geographical and Energy Aware Routing (GEAR)
  - Greedy Perimeter Stateless Routing (GPSR)

4.4.1
GEAR
Geographical and Energy Aware Routing

Geographical and Energy Aware Routing (GEAR)

- The protocol, called Geographic and Energy Aware Routing (GEAR), uses energy aware and geographically-informed neighbor selection heuristics to route a packet towards the destination region.
- The key idea is to restrict the number of interests in directed diffusion by only considering a certain region rather than sending the interests to the whole network. By doing this, GEAR can conserve more energy than directed diffusion.
- The basic concept comprises of two main parts
  - Route packets towards a target region through geographical and energy aware neighbor selection
  - Disseminate the packet within the region

Energy Aware Neighbor Computation

- Each node N maintains state $h(N, R)$ which is called learned cost to region R, where R is the target region
- Each node infrequently updates neighbor of its cost
- When a node wants to send a packet, it checks the learned cost to that region of all its neighbors
- If a node does not have the learned cost of a neighbor to a region, the estimated cost is computed as follows:

  $c(N_i, R) = \alpha d(N_i, R) + (1-\alpha)e(N_i)$

  where

  α = tunable weight, from 0 to 1.

  $d(N_i, R)$ = normalized the largest distance among neighbors of N

  $e(N_i)$ = normalized the largest consumed energy among neighbors of N

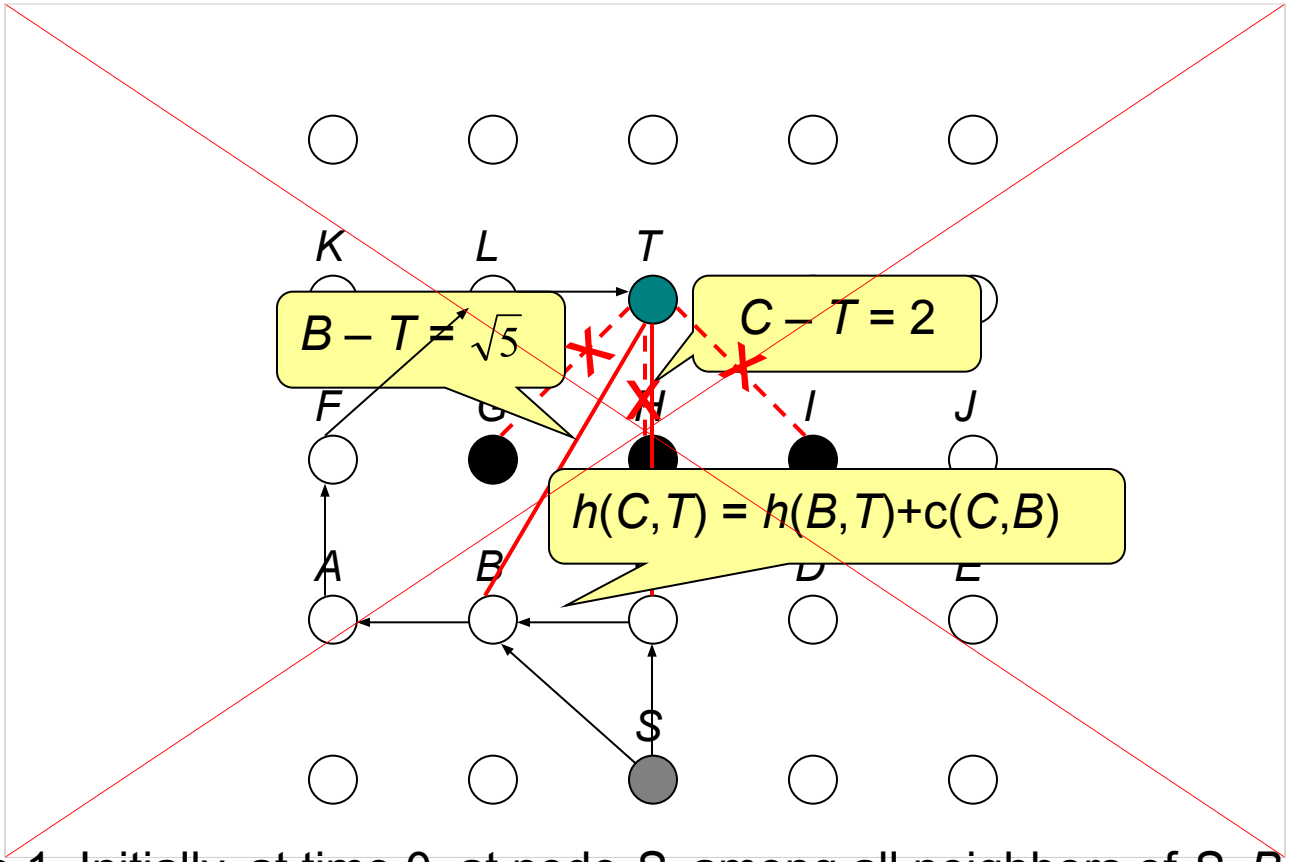
Energy Aware Neighbor Computation (cont.)

- When a node wants to forward a packet to a destination, it checks to see if it has any neighbor closer to destination than itself
- In case of multiple choices, it aims to minimize the learned cost $h(N_{min}, R)$
- It then sets its own cost to:

$$h(N, R) = h(N_{min}, R) + c(N, N_{min})$$

$c(N, N_{min})$ = the transmission cost from N and N_{min}

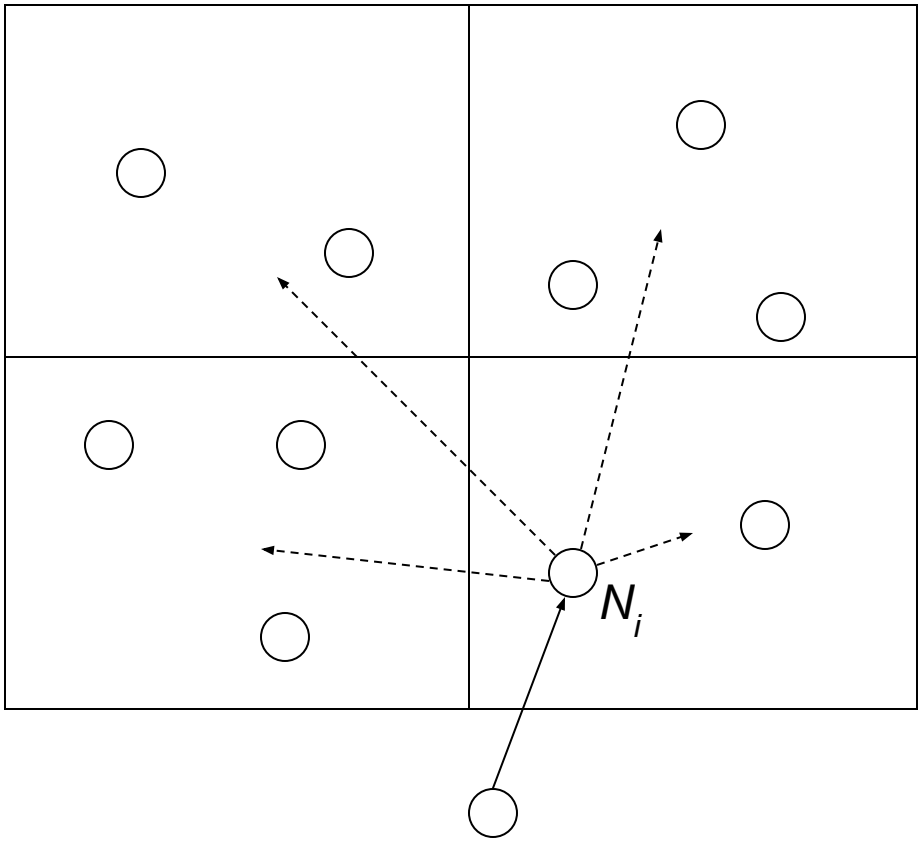
Forwarding Around Holes



α is set to 1. Initially, at time 0, at node S, among all neighbors of S, B, C, D are closer to T than S. $h(B,T)=c(B,T)=\sqrt{5}$, $h(C,T)=c(C,T)=2$, $h(D,T)=c(D,T)=\sqrt{5}$. After that , $h(C, T) = h(B, T) + 1$

Recursive Geographic Forwarding

- Once the target region is reached, the packets are disseminated within the region by recursive geographic forwarding
- Forwarding stops when a node is the only one in a sub-region

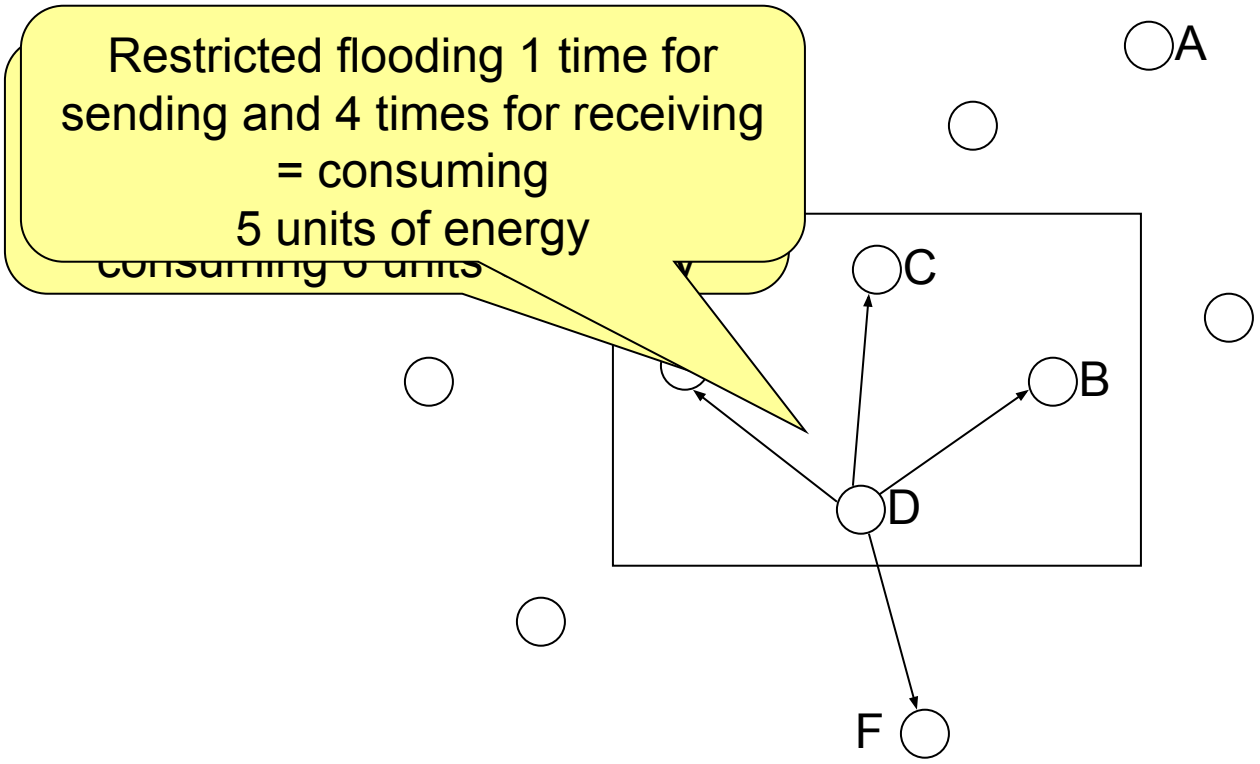


Recursive Geographic Forwarding (cont.)

- When network density is low recursive geographic forwarding is subject to two pathologies: inefficient transmissions and non-termination

Recursive Geographic Forwarding (cont.)

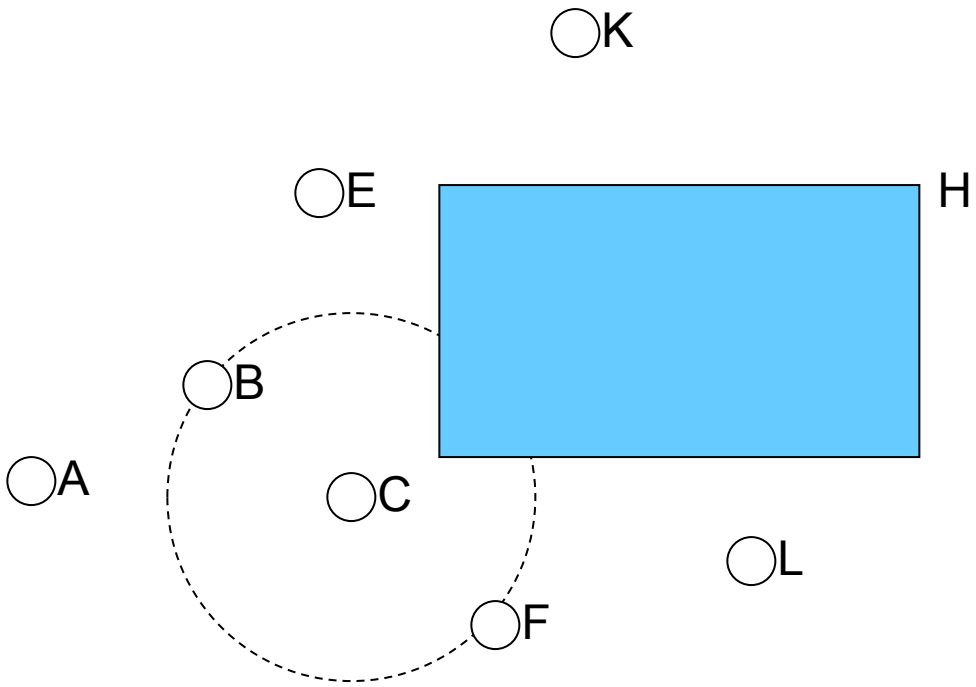
- Inefficient Transmission
  - Recursive geographic forwarding vs. Restricted flooding



Recursive Geographic Forwarding (cont.)

Pathologies

- Non-Termination
  - In the recursive geographic forwarding protocol, packet forwarding terminates when *the target subregion is empty.*



Recursive Geographic Forwarding (cont.)

Proposed solution for pathologies

- Solution:
- Node degree is used as a criteria to differentiate low density networks from high density ones
- Choice of restricted flooding over recursive geographic forwarding if the receiver's node degree is below a threshold value

Conclusion

- GEAR strategy attempts to balance energy consumption and thereby increase network lifetime
- GEAR performs better in terms of connectivity after initial partition

References

- Y. Yu, D. Estrin, and R. Govindan, “Geographical and Energy-Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks”, *UCLA Computer Science Department Technical Report*, UCLA-CSD TR-01-0023, May 2001.
- Nirupama Bulusu, John Heidemann, and Deborah Estrin. “Gps-less low cost outdoor localization for very small devices”. *IEEE Personal Communications Magazine*, 7(5):28-34, October 2000.
- L. Girod and D. Estrin. “Robust range estimation using acoustic and multimodal sensing”. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001)*, Maui, Hawaii, October 2001.
- Nissanka B. Priyantha, Anit Chakraborty, and Hari Balakrishnan. “The cricket location-support system”. In *Proc. ACM Mobicom*, Boston, MA, 2000.
- Andreas Savvides, Chih-Chieh Han, and Mani B. Strivastava. “Dynamic fine-grained localization in adhoc networks of sensors”. In *Proc. ACM Mobicom*, 2001.

4.4.2
GPSR
Greedy Perimeter Stateless Routing

Greedy Perimeter Stateless Routing (GPSR)

- Greedy Perimeter Stateless Routing (GPSR) proposes the aggressive use of geography to achieve scalability
- GEAR was compared to a similar non-energy-aware routing protocol GPSR, which is one of the earlier works in geographic routing that uses planar graphs to solve the problem of holes
- In case of GPSR, the packets follow the perimeter of the planar graph to find their routes
- Although the GPSR approach reduces the number of states a node should keep, it has been designed for general mobile ad hoc networks and requires a location service to map locations and node identifiers.

Algorithm & Example

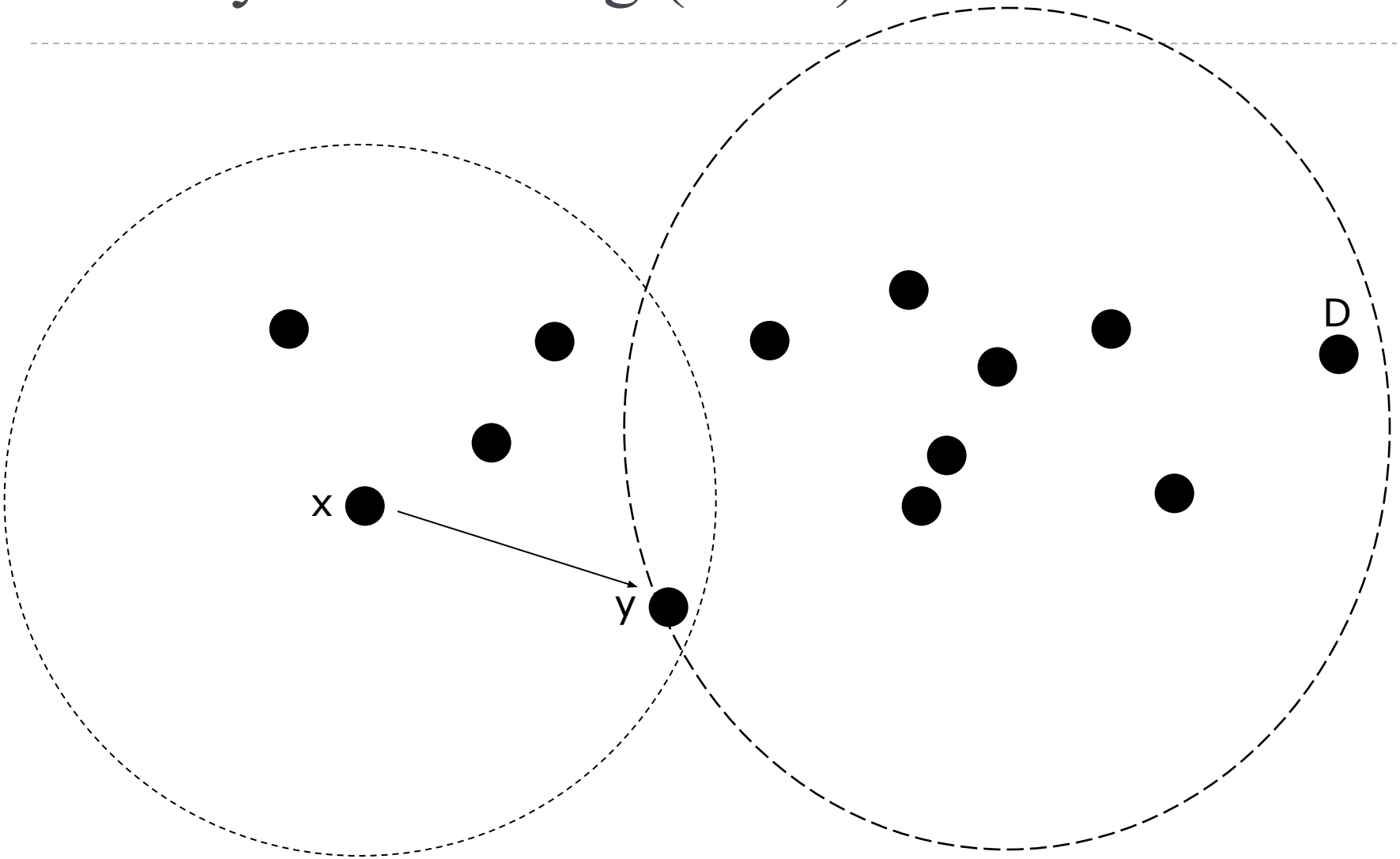
- The algorithm consists of two methods:
  greedy forwarding + perimeter forwarding

- Greedy forwarding, which is used wherever possible, and perimeter forwarding, which is used in the regions greedy forwarding cannot be done.

Greedy Forwarding (cont.)

- Under GPSR, packets are marked by their originator with their destinations' locations
- As a result, a forwarding node can make a locally optimal, greedy choice in choosing a packet's next hop
- Specifically, if a node knows its radio neighbors' positions, the locally optimal choice of next hop is the neighbor geographically closest to the packet's destination
- Forwarding in this scheme follows successively closer geographic hops, until the destination is reached

Greedy Forwarding (cont.)

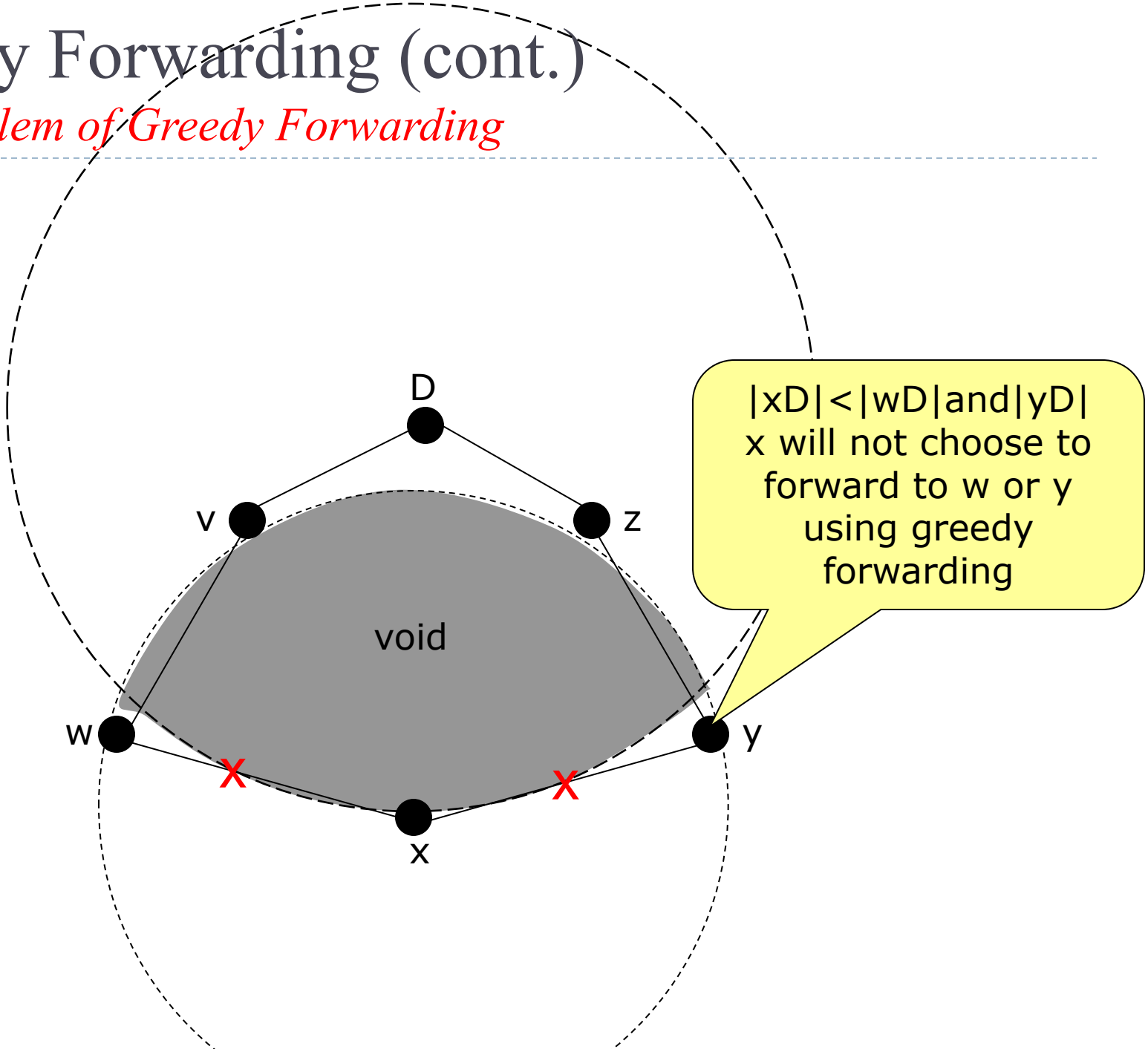


Greedy Forwarding (cont.)

- A simple beaconing algorithm provides all nodes with their neighbors' positions: periodically, each node transmits a beacon to broadcast MAC address, containing its own identifier (e.g., IP address) and position
- Position is encoded as two four-byte floating point quantities, for x and y coordinate values
- Upon not receiving a beacon from a neighbor for longer than timeout interval T, a GPSR router assumes that the neighbor has failed or gone out-of-range, and deletes the neighbor from its neighbor table

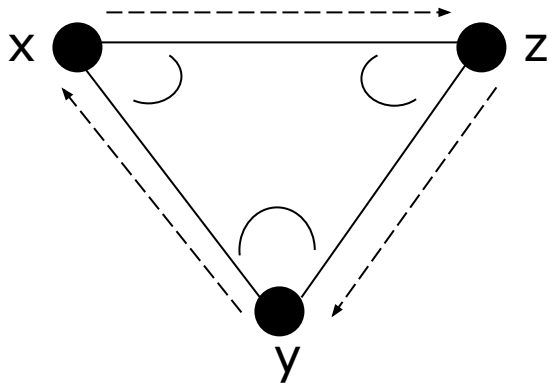
Greedy Forwarding (cont.)

The Problem of Greedy Forwarding

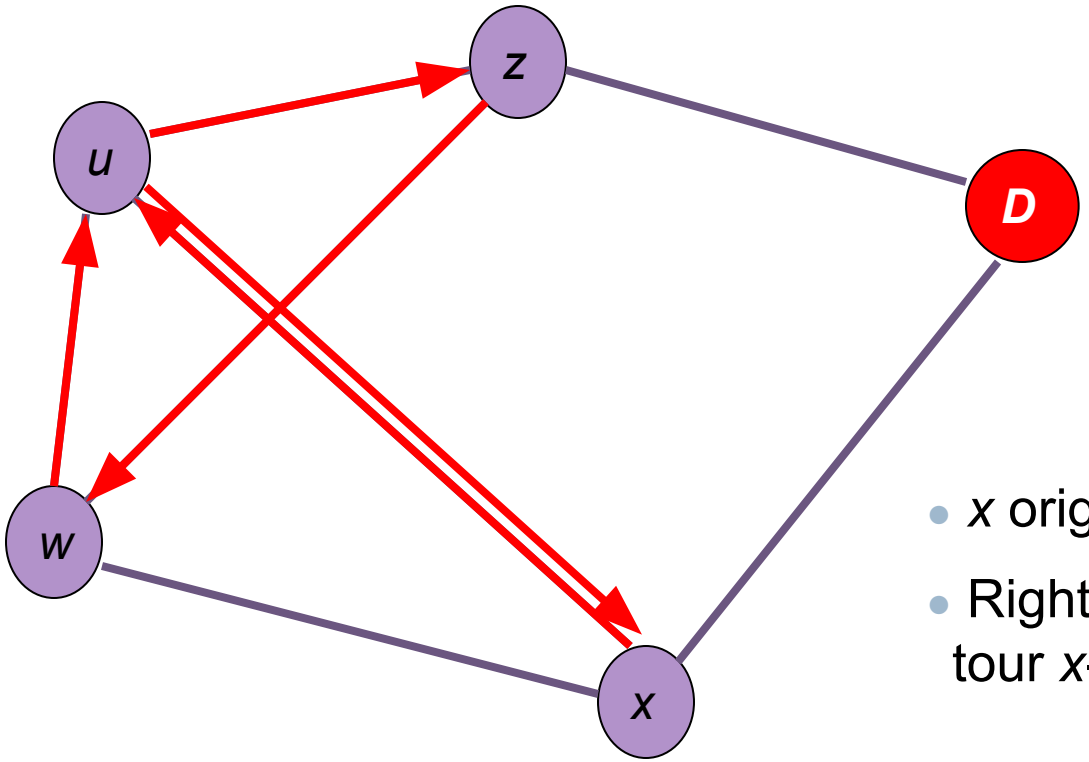


The Right-Hand Rule: Perimeters

- In previous works, use the right-hand rule to map perimeters by sending packets on tours of them. The state accumulated in these packets is cached by nodes, which recover from local maxima in greedy forwarding by routing to a node on a cached perimeter closer to the destination.
- This approach requires a heuristic, the no-crossing heuristic, to force the right-hand rule to find perimeters that enclose voids in regions where edges of the graph cross

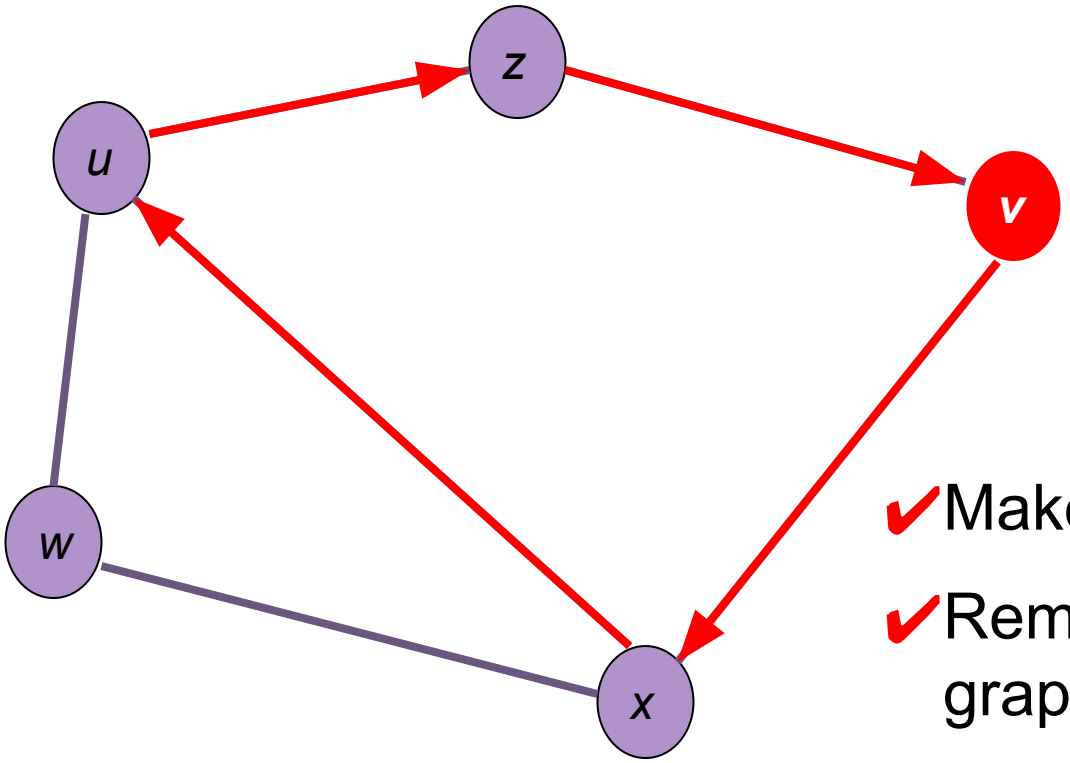


Right-Hand Rule Does Not Work with Cross Edges



- *x* originates a packet to *u*
- Right-hand rule results in the tour *x*-*u*-*z*-*w*-*u*-*x*

Remove Crossing Edge



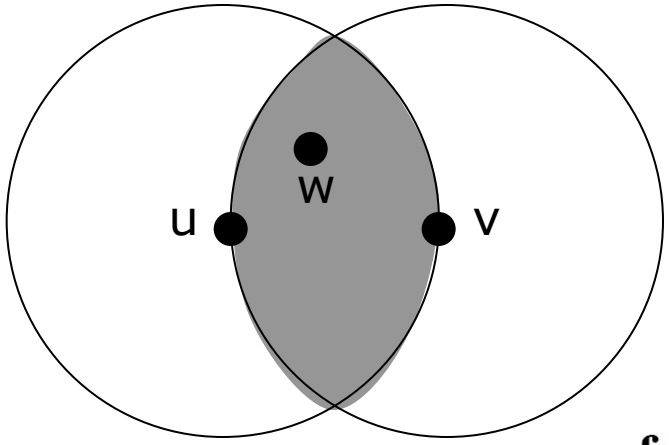
- ✔Make the graph planar
- ✔Remove (*w*,*z*) from the graph
- ✔ Right-hand rule results in the tour *x*-*u*-*z*-v-*x*

Make a Graph Planar

- A graph in which no two edges cross is known as planar. A set of nodes with radios, where all radios have identical, circular radio range r, can be seen as a graph: each node is a vertex, and edge (n, m) exists between nodes n and m if the distance between n and m, $d(n, m) \leqq r$.
- Convert a connectivity graph to planar non-crossing graph by removing "bad" edges
- Ensure the original graph will not be disconnected
- Two types of planar graphs:
  - Relative Neighborhood Graph (RNG)
  - Gabriel Graph (GG)

Planarized Graphs (cont.)

Relative Neighborhood Graph (RNG)

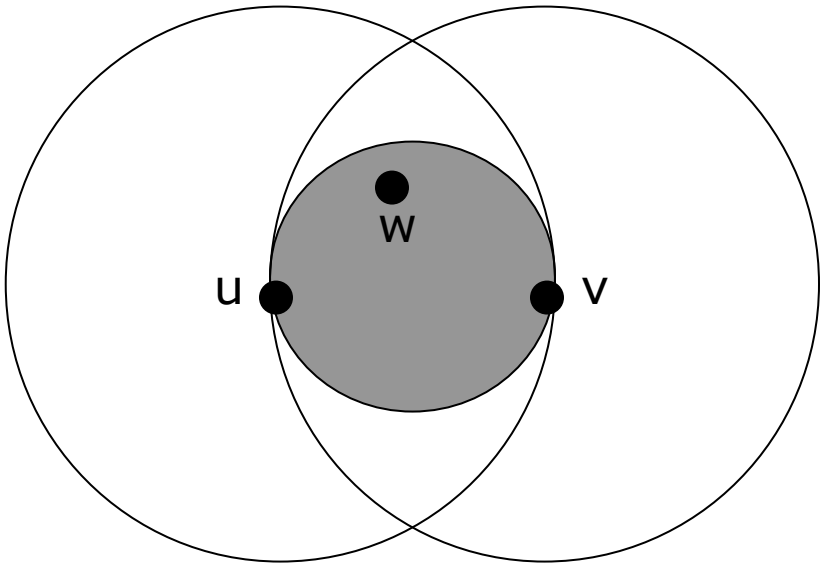


$$\forall w \neq u, v : d(u,v) \leq \max[d(u,w), d(v,w)]$$

for all $v \in N$ **do**
  for all $w \in N$ **do**
    if $w == v$ **then**
      continue
    else if $d(u,v) > \max[d(u,w), d(v,w)]$ **then**
      eliminate edge (u,v)
      break
    end if
  end for
end for

Planarized Graphs (cont.)

Gabriel Graph (GG)



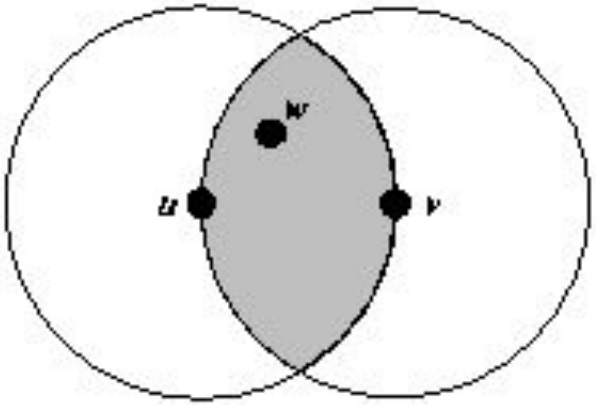
m = midpoint of $\overline{uv}$
for all $v \in N$ **do**
  for all $w \in N$ **do**
    if $w == v$ **then**
      continue
    else if $d(m,w) < d(u,m)$ **then**
      eliminate edge (u,v)
      break
    end if
  end for
end for

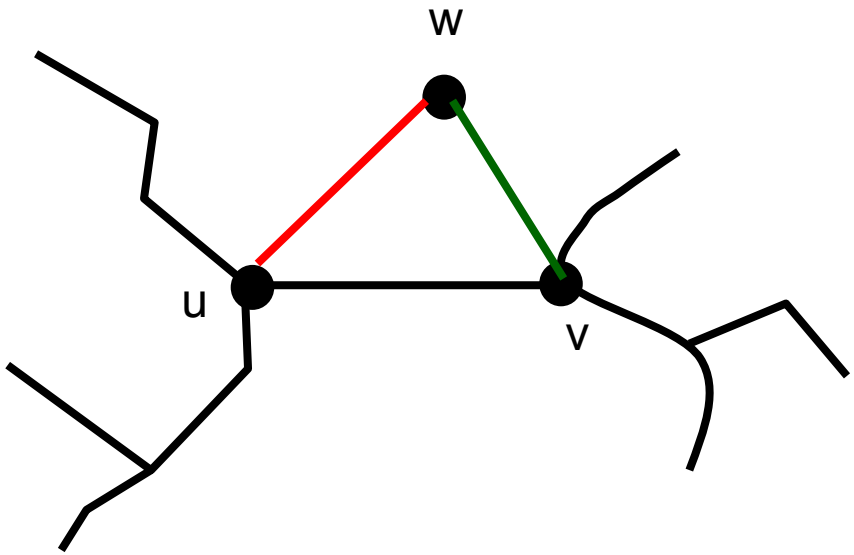
Planarized Graphs (cont.)

- An algorithm for removing edges from the graph that are not part of the RNG or GG would yield a network with no crossing links
- The RNG is a subset of the GG
  - It is because RNG removes more edges
- Hereby, the RNG is used
- If the original graph is connected, RNG is also connected

Connectedness of RNG Graph

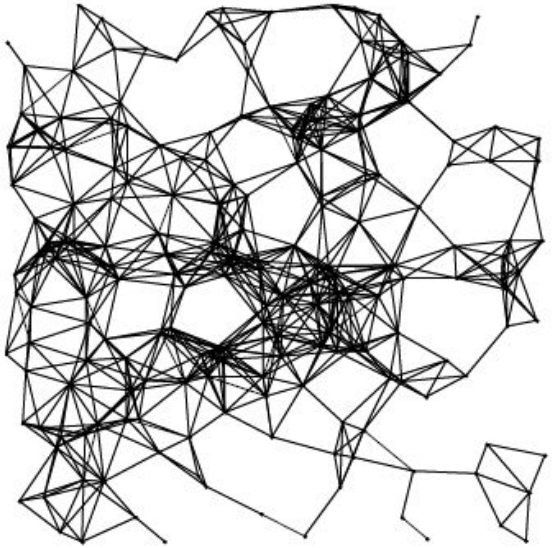
- Key observation
  - Any edge on the minimum spanning tree of the original graph is not removed
  - Proof by contradiction: Assume (u,v) is such an edge but removed in RNG





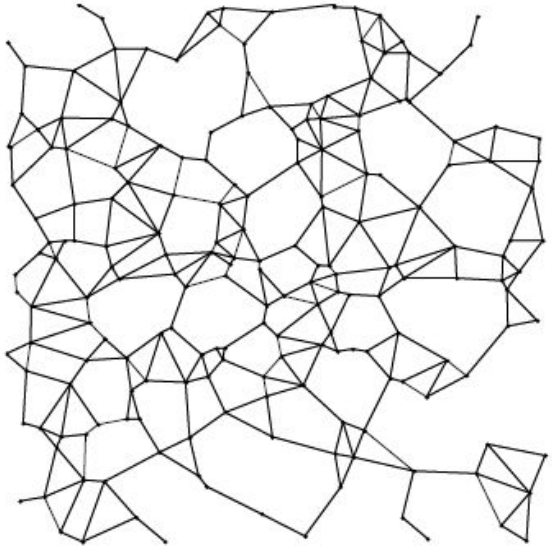
Planarized Graphs (cont.)

Original



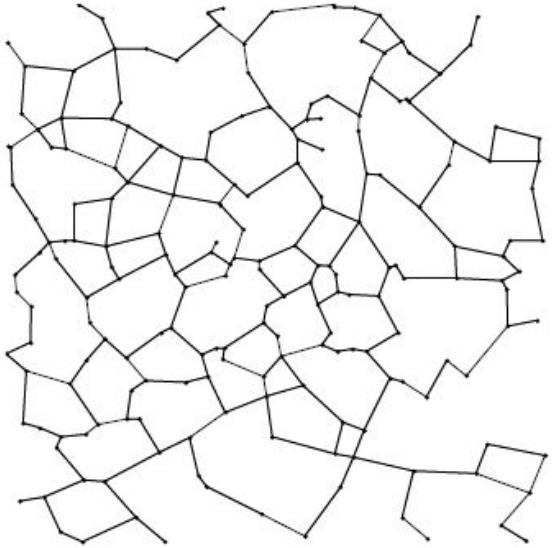
The full graph of a radio network, 200 nodes, uniformly randomly placed on a 2000 x 2000 meter region, with a radio range of 250 m.

Gabriel Graph (GG)



The GG subset of the full graph

Relative Neighborhood Graph (RNG)

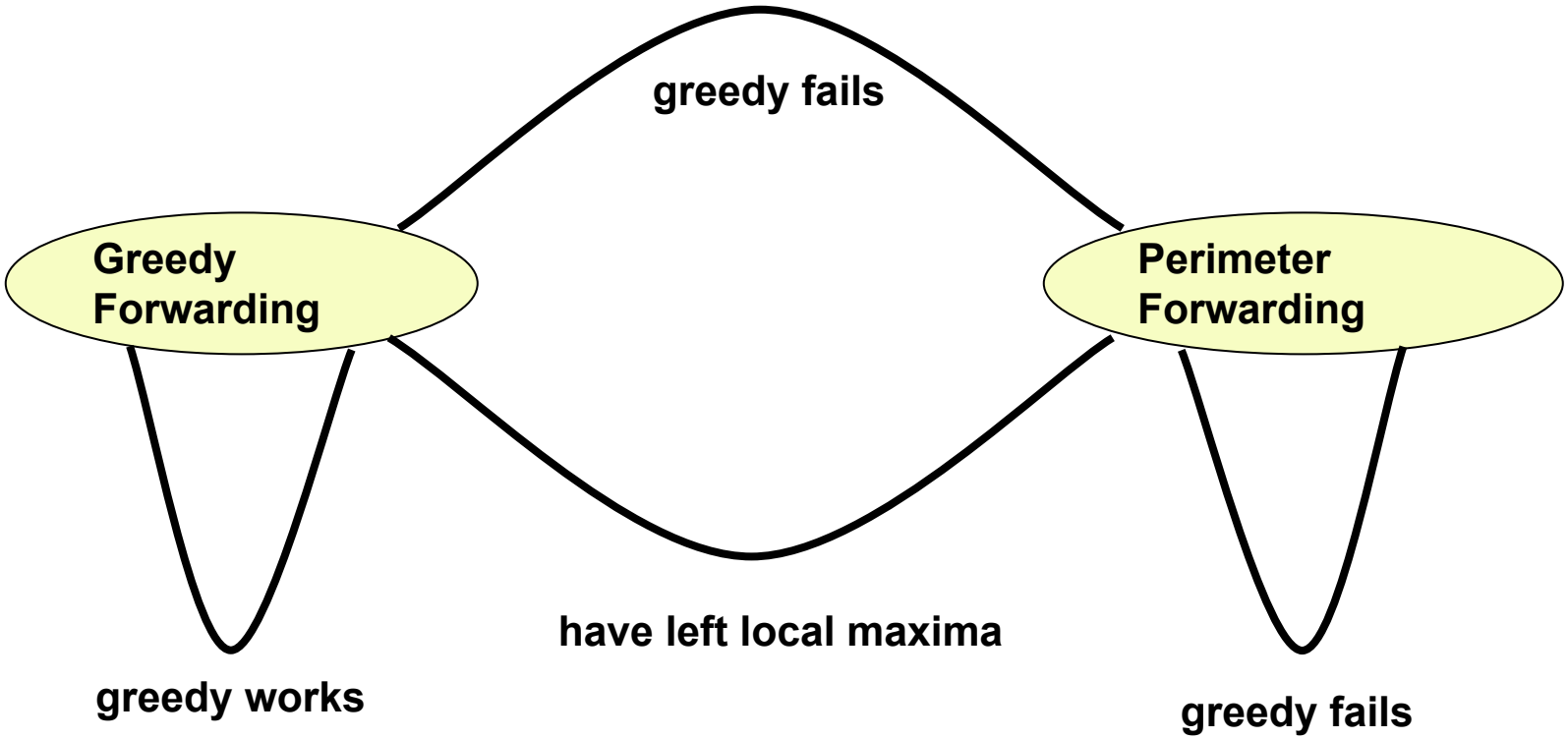


The RNG subset of the full and GG graphs.

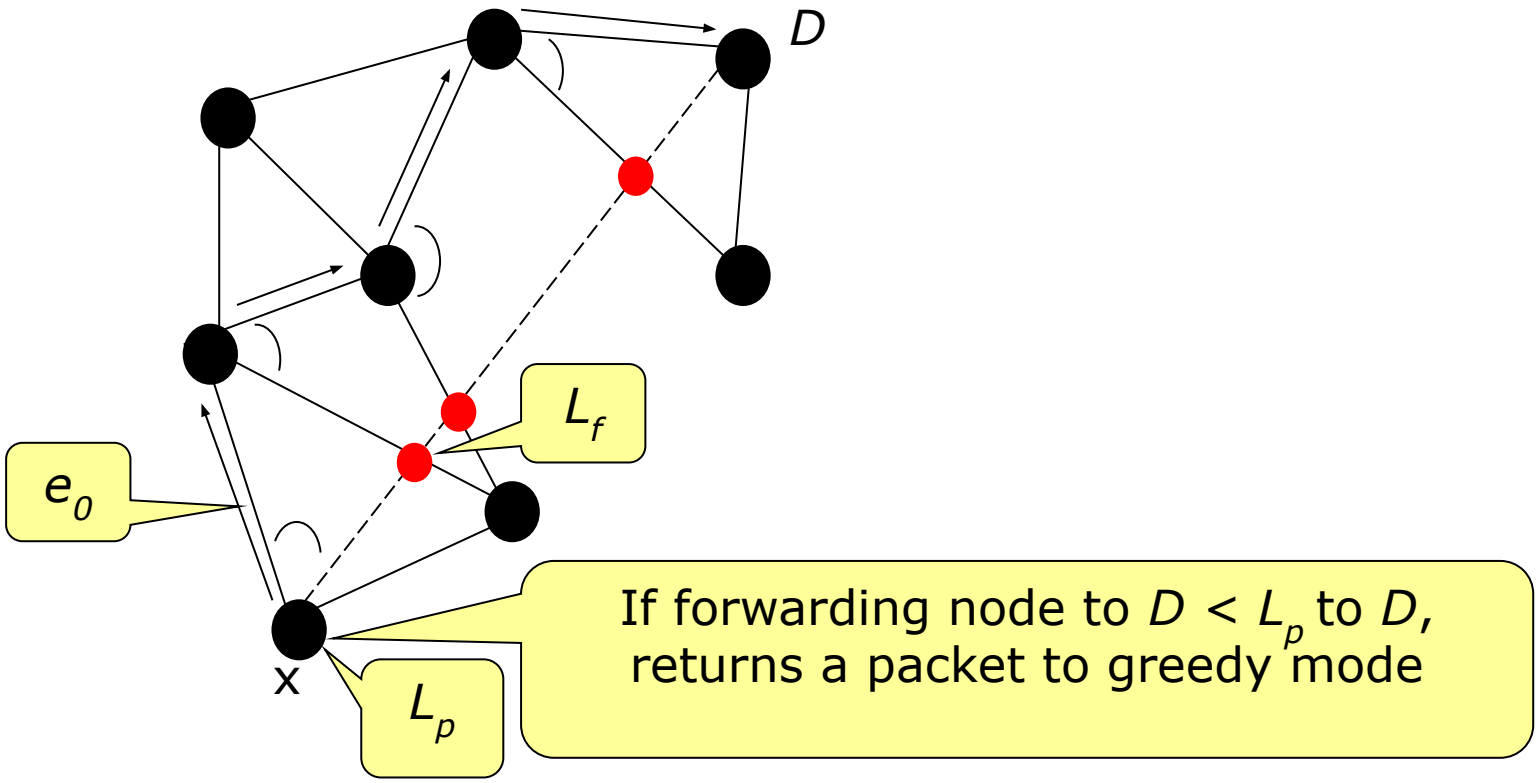
Combining Greedy and Planar Perimeters

- All data packets are marked initially at their originators as greedy mode
- GPSR packet headers include a flag field indicating whether the packet is in greedy mode or perimeter mode
- Packet sources also include the geographic location of the destination in packets
- Only a packet's source sets the location destination field, it is left unchanged as the packet is forwarded through the network
- Upon receiving a greedy-mode packet for forwarding, a node searches its neighbor table for the neighbor geographically closest to the packet's destination
- When no neighbor is closer, the node marks the packet into perimeter mode

GPSR


greedy fails
Greedy Forwarding
Perimeter Forwarding
have left local maxima
greedy works
greedy fails

Combining Greedy and Planar Perimeters (cont.)



Conclusion

- GPSR's benefits all stem from geographic routing's use of only immediate-neighbor information in forwarding decisions.
- GPSR keeps state proportional to the number of its neighbors, while both traffic sources and intermediate DSR routers cache state proportional to the product of the number of routes learned and route length in hops.

References

- B. Karp and H. T. Kung, “Greedy Perimeter Stateless Routing for Wireless Networks”, *Proc. 6th Annual ACM/IEEE Int'l. Conf. Mobile Comp. Net.,* Boston, MA, pp. 243-54, August 2000.
- G. G. Finn, “Routing and addressing problems in large metropolitan-scale internetworks”, Tech. Rep. ISI/RR-87-180, *Information Sciences Institute*, March 1987.
- S. Floyd and V. Jacoboson, “The synchronization of periodic routing messages”, *IEEE/ACM Transactions on Networking*, Vol. 2, pp. 122-136, April 1994.
- B. Karp “Greedy perimeter state routing”, *Invited Seminar at the USC/Information Sciences Institute*, July 1998.
- J. Saltzer, D. P. Reed, and D. Clark, “End-to-end arguments in system design”, *ACM Transactions on Computer Systems*, Vol. 2, No. 4, Pages: 277-288, November 1984.