Chapter 14

Sensor Networks

Outline

- Introduction
- Fixed Wireless Sensor Networks
- Wireless Sensor Networks
- Sensor Deployment
- Network Characteristics
- Design Issues in Sensor Networks
- Secured Communication
- Summary

Introduction

- These are a new class of ad hoc networks
- Enable reliable monitoring and analysis of unknown and untested environments
- These are a collection of tiny disposable and low power devices
- A sensor node is a device that converts a sensed attribute (e.g., temperature, vibration) into a form understandable by users
- Includes a sensing module, a communication module, memory, and typically a small battery
- These have been used for years for a number of applications
- The number of sensors can be large to cover as much area as desirable
- Sensor networks are usually unattended and some degree of fault-tolerance needed

Characteristics of Wireless Sensor Networks

Advantages:

- Ease of deployment
- Extended range
- Fault tolerance
- Mobility (some)

Limitations:

- Low-bandwidth
- Error-prone transmissions
- Need for collision-free channel access
- Limited amount of energy available
- Derive energy from personal batteries
- Usually sensors placed where it is difficult to replace their batteries

Characteristics of Wireless Sensor Networks

Parameters that change as per applications:

- **Power availability**
- **Position (if nodes are mobile)**
- **Reachibility**
- **Type of task (attributes the nodes need to operate on)**

Unique challenges in designing protocol:

- In traditional wired and wireless networks, each node is identified by a unique ID used for routing. This cannot be used effectively in sensor networks; since these networks are data centric, routing to and from specific nodes is not required
- Adjacent nodes may have similar data and it is desirable to aggregate this data and send it
- The requirements of the network change with application, hence it is application specific

Protocols for Sensor Networks

- Protocols ought to be application specific
- Yet generic enough to be data centric, capable of data aggregation and minimizing energy consumption
- Ought to have following additional features:
  - Attribute-based addresses are composed of a series of attribute-value pairs that specify certain physical parameters to be sensed- temperature =100◦ C
  - Location awareness is important- GPS devices can be deployed (expensive)
  - Sensor should react immediately to drastic changes in the environment.
  - Query handling from BS to provide requested data- reliable mechanism to transmit the query to appropriate sensor nodes

Response to Sensor Queries

- Data from various nodes need to be aggregated to reduce the traffic
- Queries that monitor the system are mostly duration-based queries
- Time-critical queries should reach the user immediately
- Some queries just require a snapshot view of the network at that instant

- User queries can be broadly categorized into three types:
  - Historical queries: Used for analysis of historical data stored at the BS, e.g., “What was the temperature 2 hours back in the northwest quadrant?”
  - One time query: Gives a snapshot of the network, e.g. “What is the current temperature in the northwest quadrant?”
  - Persistent query: Used to monitor the network over a time interval with respect to some parameters, e.g. “Report the temperature for the next 2 hours”

Inherent Dynamic Nature of Sensors

- The nodes on deployment should create and assemble a network, dynamically adapt to device failure, and degradation, manage mobility of sensors, and react to changes in task and requirements (self-organizing)
- Some sensors may detect an event that trigger a big unit, like a camera generating heavy traffic
- Final control could be turned off or on, should be capable of dynamically trading precision for energy or scope for convergence

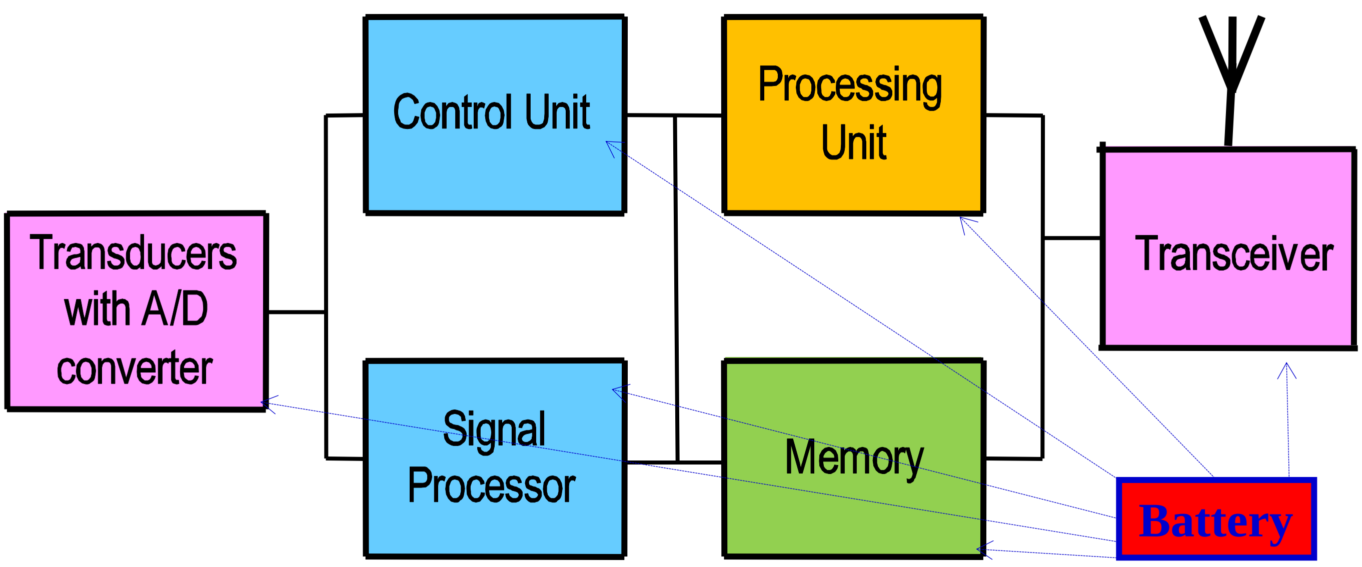
- DARPA SensIT program focused on adaptive fidelity, dynamically adjusting the overall fidelity of sensing in response to task dynamics
- Adaptive self-configuring sensor network topologies (ASCENT)- which nodes should join the routing infrastructure to adapt to an environment and terrain conditions

Other Applications

- In a warfare, sensors can be used to determine
  - Enemy's activities, identify troops movement, type of tanks and armaments, area of activities, etc.
- A similar scheme can be utilized to monitor:
  - A disaster area due to natural or man-made cause such as earthquake, flooding, wild-fire, tsunami, volcanic eruption, hurricane, storms, explosions, etc.
- Many other civilian applications have been reported in the literature, including
  - Habitat monitoring
  - Plant growth checking
  - Environmental monitoring for harmful gases and health hazard pollutants, forecasting
  - Drinking water quality
  - Soil moisture monitoring
  - Health-care
  - Streets, building, bridge and structural monitoring
  - Home/office automation

A general architecture of a fixed Sensor Node

- In defense and disaster, a random sensor deployment using low-flying airplanes or unmanned aerial vehicles (UAVs)
- In most other civilian applications, the terrain is easily reachable and sensors can be placed at any required place



Fixed Wireless Sensor Networks

MicaZ Module

MicaZ Sensor Board

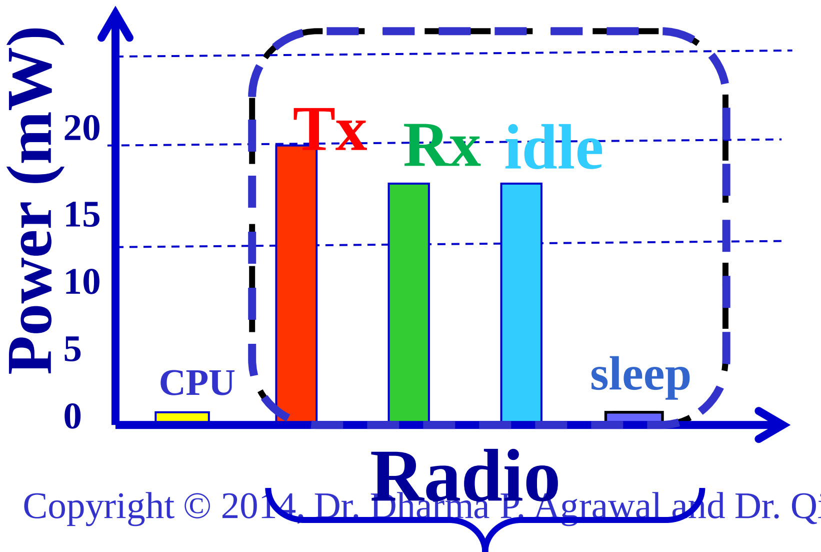
- Sensors can be placed at fixed locations and linking them by a wireless network to perform distributed sensing tasks
- Useful for continuous and regular monitoring, such as facility and environmental sampling, security and surveillance, health care monitoring, and underwater measurements
- Communication between sensors is achieved by LOS infrared beam or conventional wireless radio communications like FDMA or TDMA
- Functional requirements of a sensor node

Parameters of MICA Motes Sensors

Parameters of MICA motes sensor units and power consumption [*Credit : www.xbow.com/Products*]

MICAZ Mote	Characteristics
Processing unit	8 or 16bits; 10-416 MHz in 2.4 GHz range
Serial Flash Storage	512 kbs
RAM	4k-8k bytes
Transceiver	1kbps - 1Mbps in 2.4 GHz; 3m-300m range

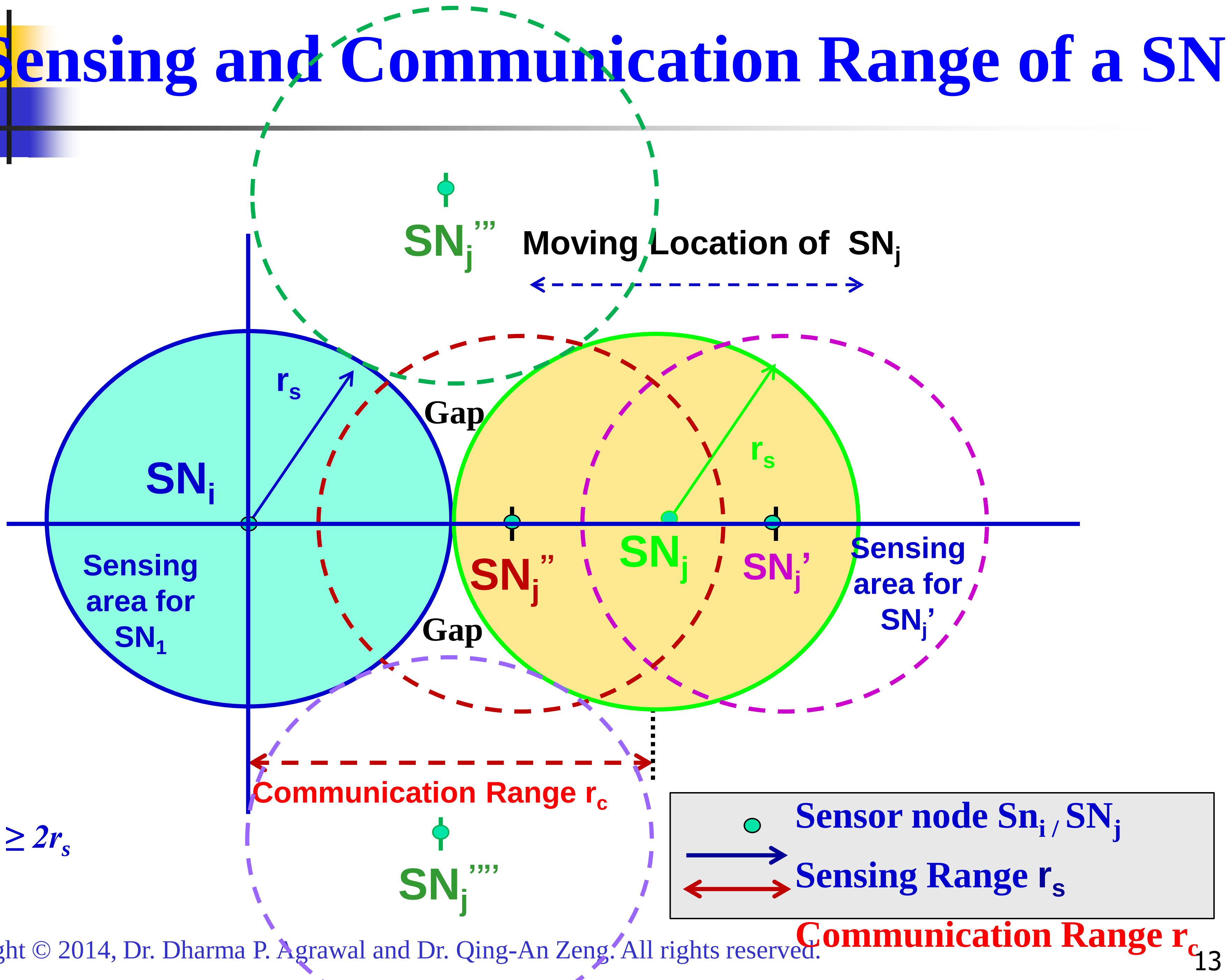
Power consumption of node subsystems



Energy consumption in MICA2 motes sensor unit

Component	Current	Power
Nothing Test	7.5 mA	.02325 mW
Radio Off	4 mA	.0124 mW
Radio Idle	16 mA	.0496 mW
Radio Receiver	16 mA	.0496 mW
Radio Transmit	21 mA	.0651 mW
Computation only	10m A	.0310 mW

Sensing and Communication Range of a SN



$r_t \geq 2r_s$

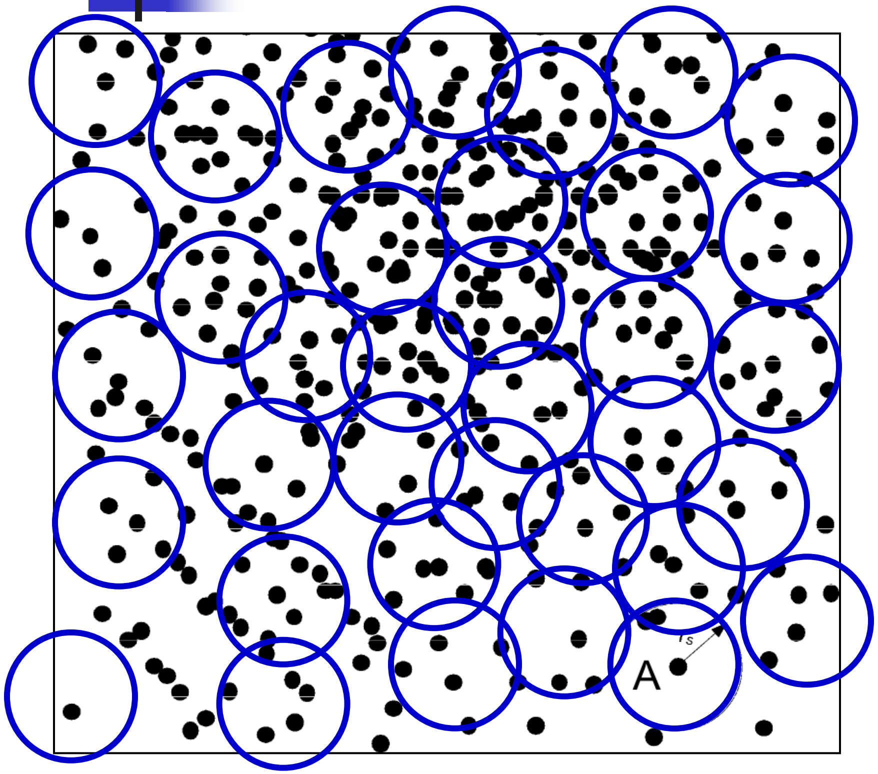
Randomly Deployed Sensor Networks

- **N sensors are randomly deployed uniformly in an area A**
- **Sensor density λ is given by:** $\lambda = \dfrac{N}{A}$
- **A point *x is* covered by a sensor s with the probability:**

$$e^{-\lambda * sensed\,area} = e^{-\lambda \pi r_s{}^2}$$

- **Fraction of area not covered:**
- **This value can approach 1 by either increasing *λ or r_s* , implying increasing either the number of sensor *N or the sensing radius r_s***
- **The sensing model is known as Boolean sensing model as any event occurring outside sensing radius *r_s is assumed to be zero***

Random Deployment and Coverage

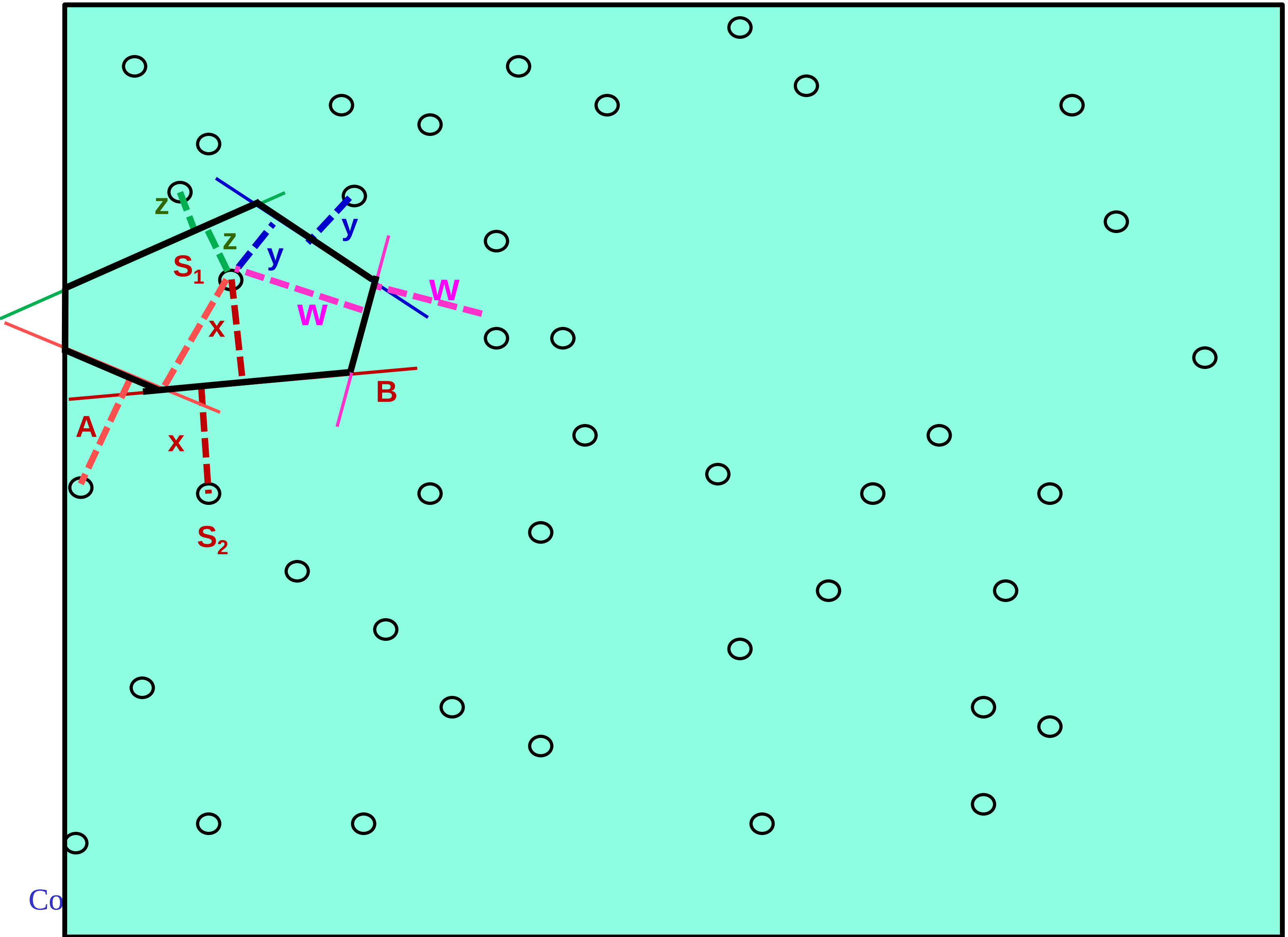


Random deployment of *N* sensors in a square area *A* = *L* × *L with sensing* range r_s

Fraction of coverage as a function of deployed sensors area A = 1000 × 1000, and sensing range r_s = *40 units*

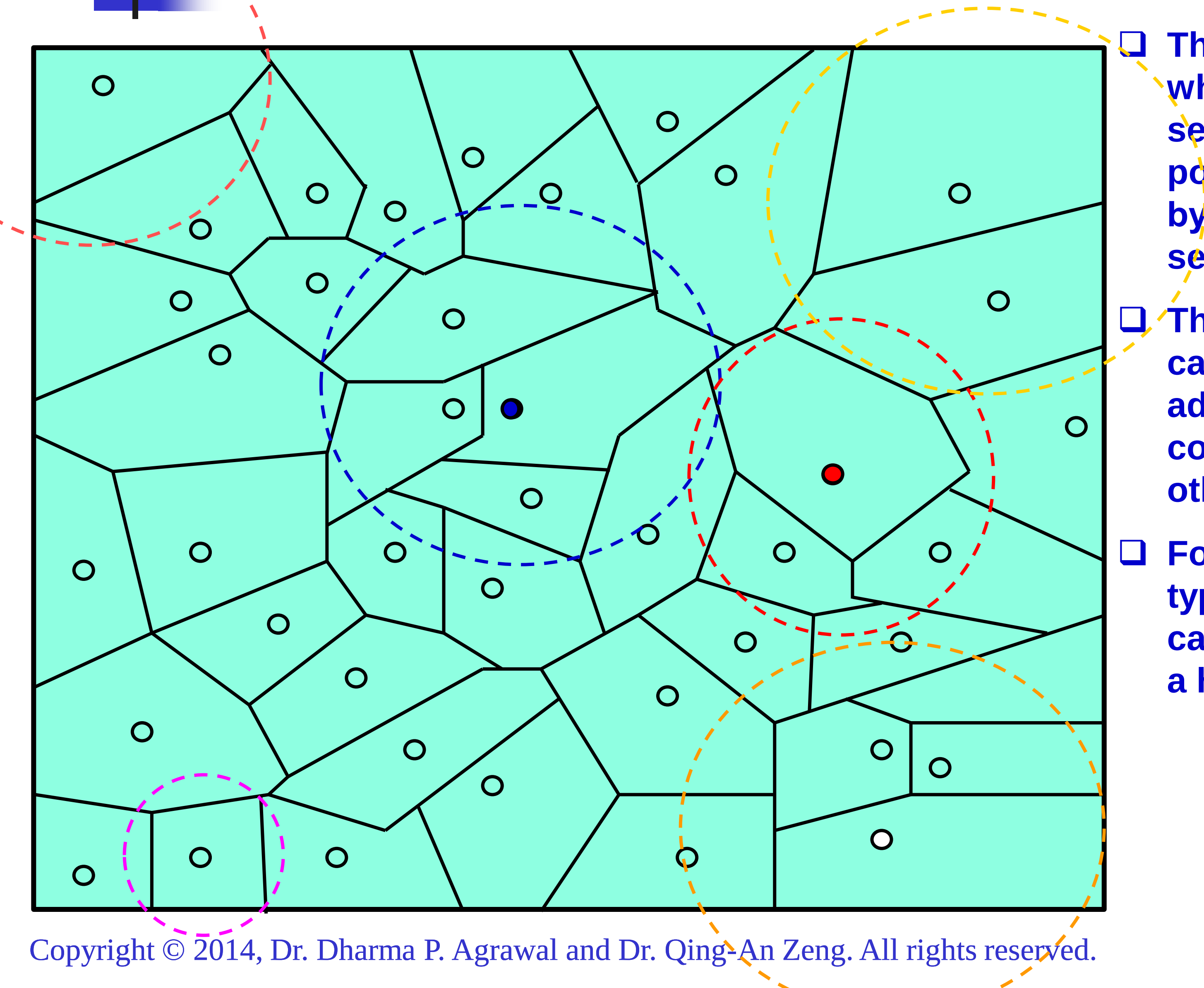
Number of sensors N	320	330	340	350	360	370	380	390	400	410	420
Coverage fraction	.79981	.80963	.81896	.82783	.83627	.8443	.85193	.85919	.86609	.87266	.8789

Voronoi Diagram for a WSN



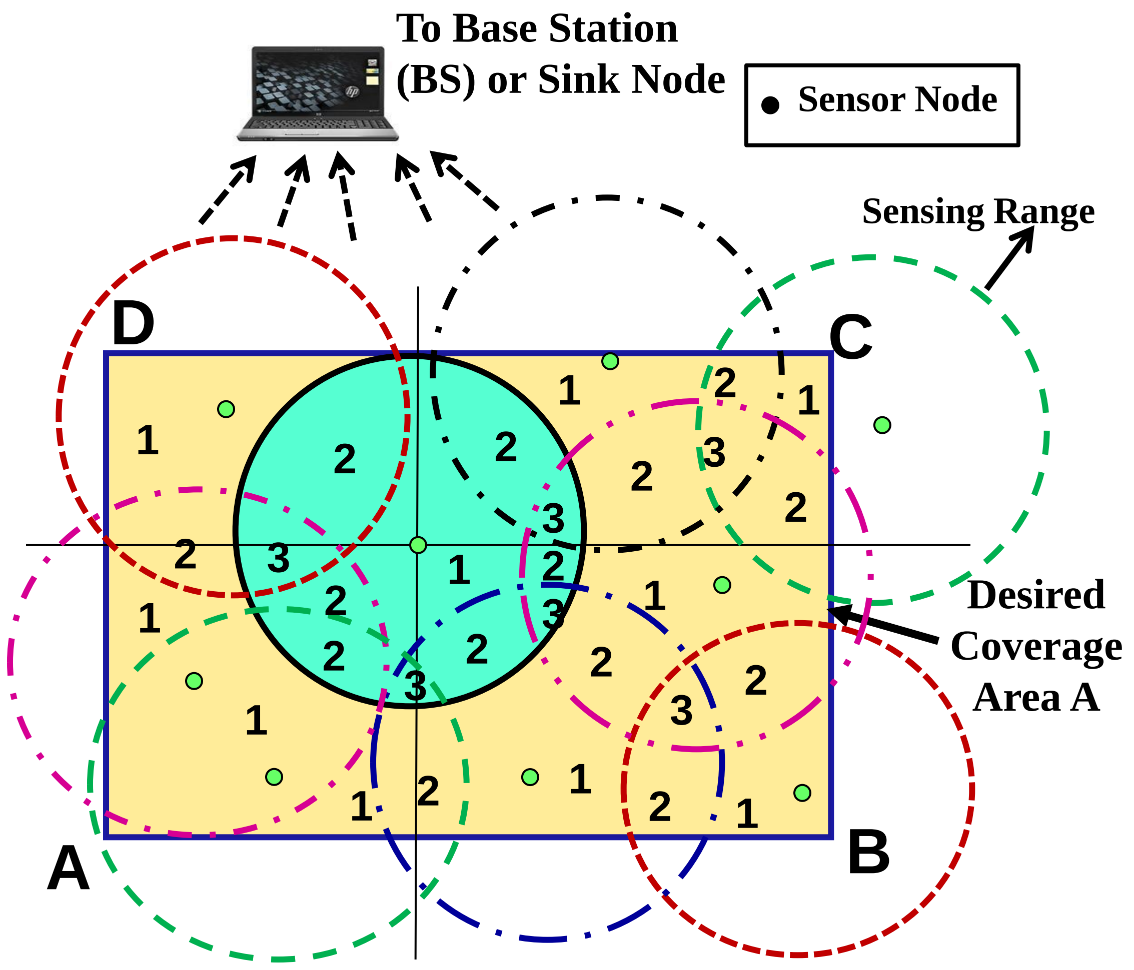
A line **AB** is drawn such that it is at equal distance ***x*** from two adjacent sensors S_1 and S_2

Voronoi Diagram for a WSN

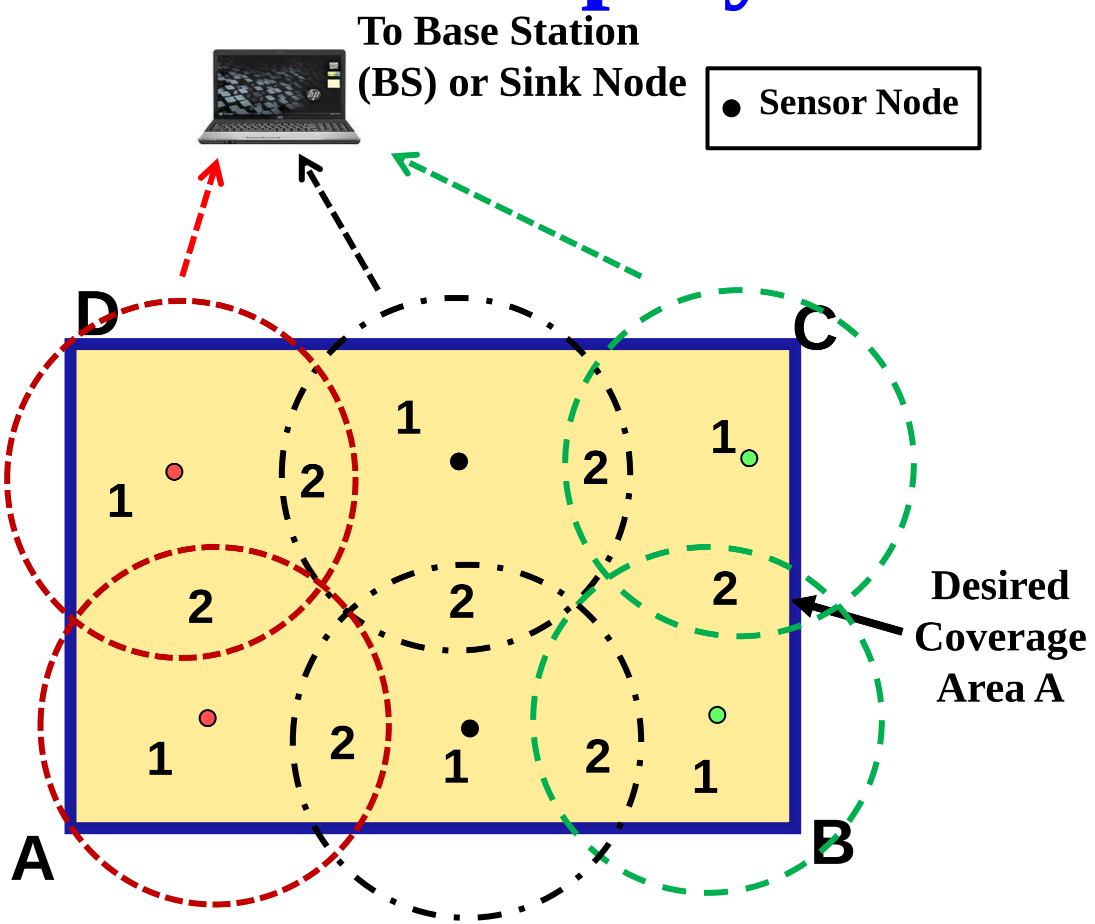


- The polygons indicate what kind of sensor to select so that each polygon can be covered by each sensor's sensing range
- The transmission range can be adjusted so that adjacent sensor can communicate with each other
- For simplicity, the same type of sensor nodes can be used, leading to a homogeneous system

Coverage of a given area *A* with multiple SNs



Covering an area Using Regular Deployment



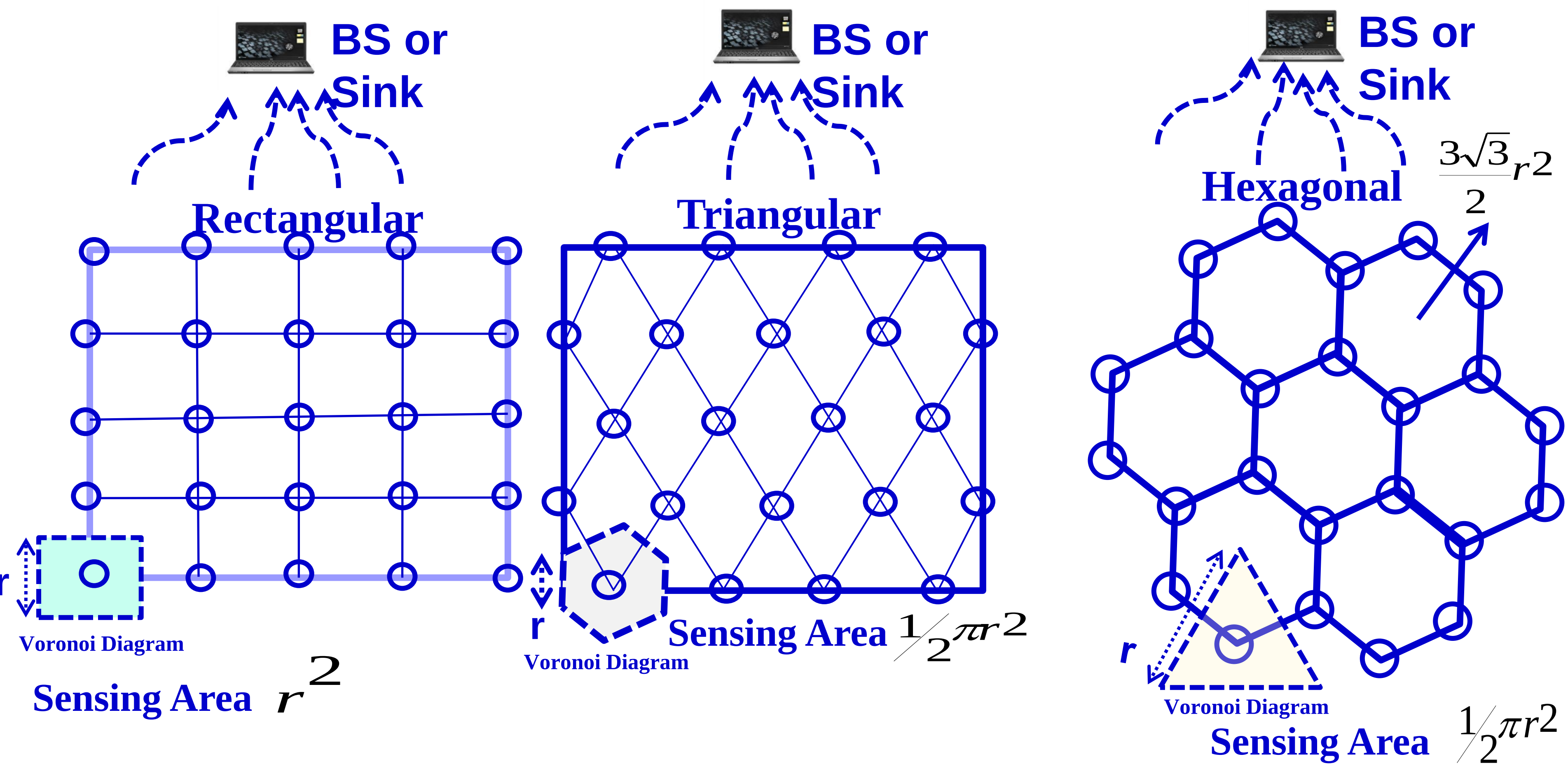
1-coverage

Coverage with only 6 SNs

Why use Regular Topology?

- Area is accessible
- Area can be covered with minimum number of sensor nodes
- Location of adjacent sensors known
- No need for self-organization or localization
- TDMA schedule is easy to utilize

Three regular Deployments and their sensing area based on Voronoi diagram



Required Number of Sensors for Full Coverage to cover an area A=100*100 units by three Lattices with varying Sensing Range

R_s in units	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
Square	20000	5000	2222	1250	800	556	408	313	247	200
Hexagon	30792	7698	3421	1925	1232	855	628	481	380	308
Triangle	15396	3849	1711	962	616	428	314	241	190	154

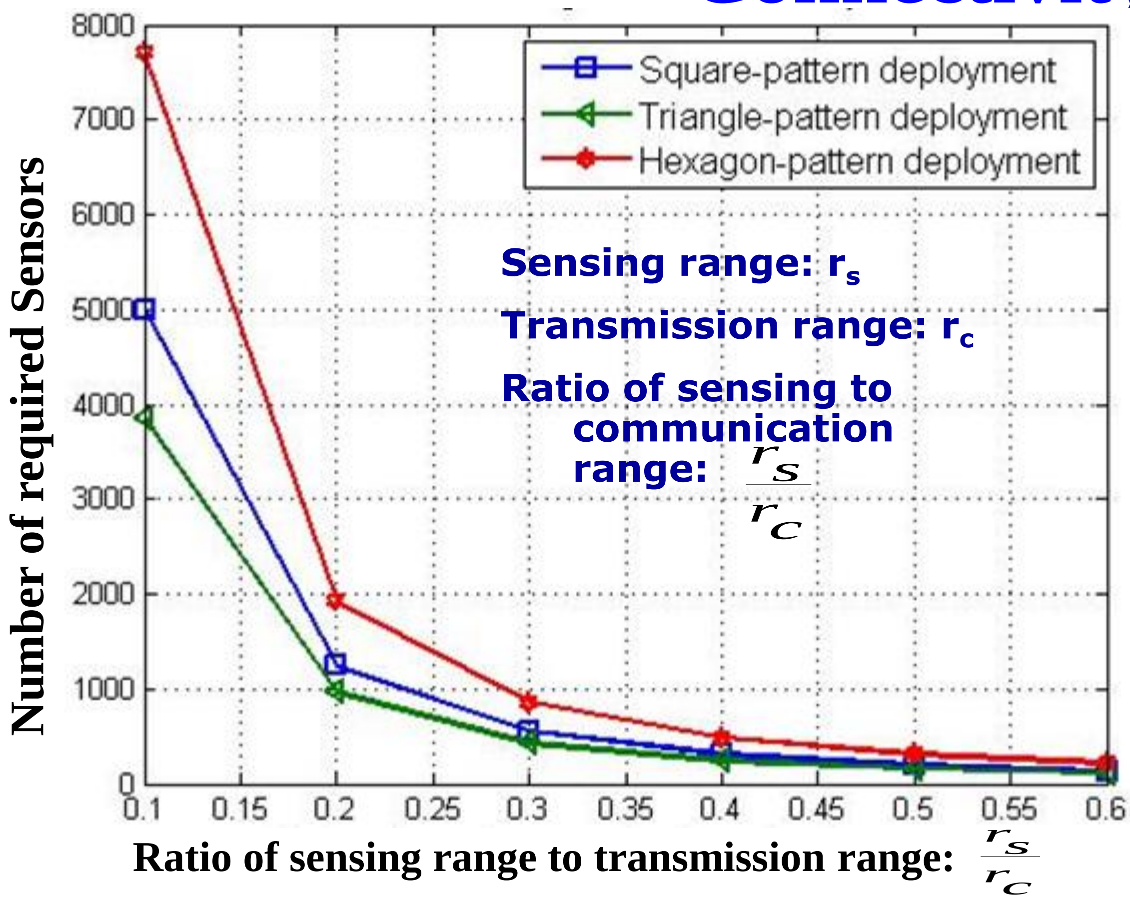
R_s in units	5	6	7	8	9	10	11	12	13	14	15
Square	200	139	102	78	62	50	41	35	30	26	22
Hexagon	308	214	157	120	95	77	64	53	46	39	34
Triangle	154	107	79	60	48	38	32	27	23	20	17

Required Number of Sensors for Full Connectivity to cover an area A=100*100 units by three Lattices with varying Communication Range

R_c	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
Square	40000	10000	4444	2500	1600	1111	816	625	494	400
Hexagon	30792	7698	3421	1925	1232	855	628	481	380	308
Triangle	46188	11547	5132	2887	1848	1283	943	722	570	462

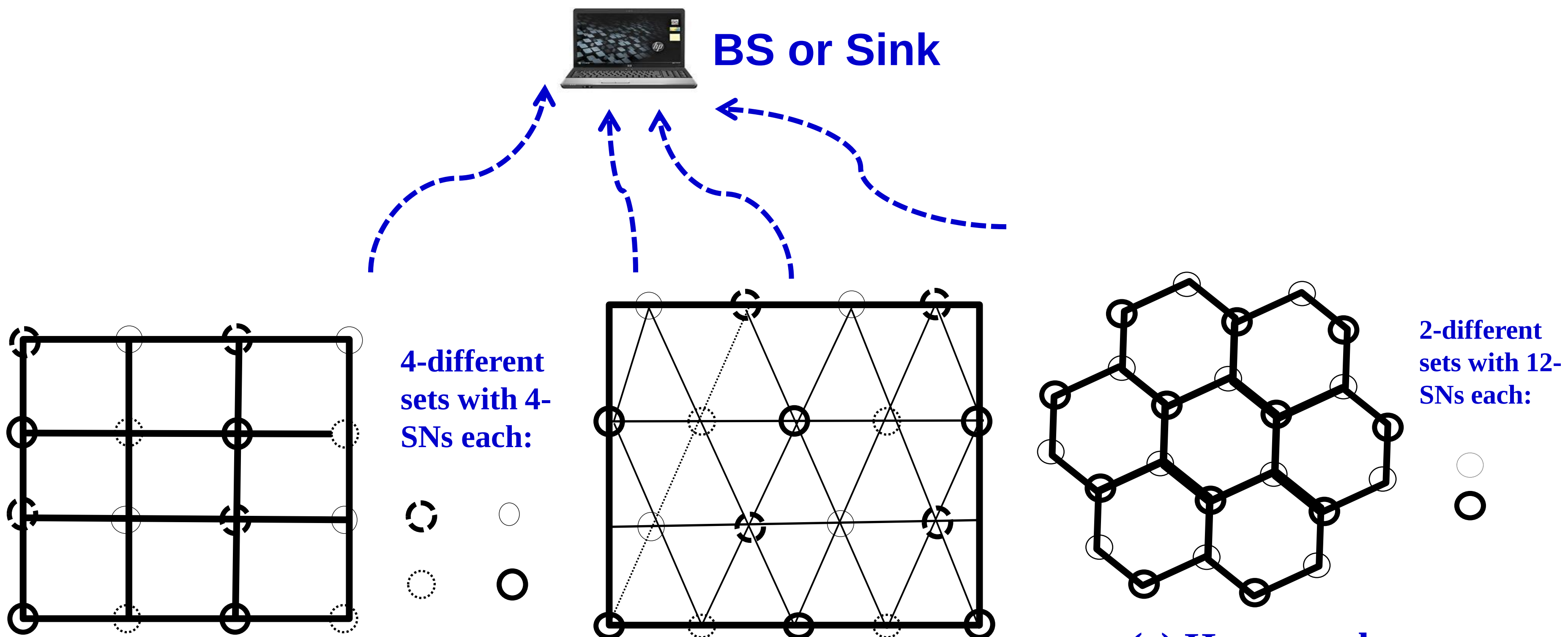
R_c	5	6	7	8	9	10	11	12	13	14	15
Square	400	278	204	156	123	100	83	69	59	51	44
Hexagon	308	214	157	120	95	77	64	53	46	39	34
Triangle	462	321	236	180	143	115	95	80	68	59	51

Required Sensors to cover 100x100 area as a Function of Ratio of Coverage and Connectivity



$\frac{r_s}{r_c}$

Lattices Topologies divided into different Sleep-Awake sets

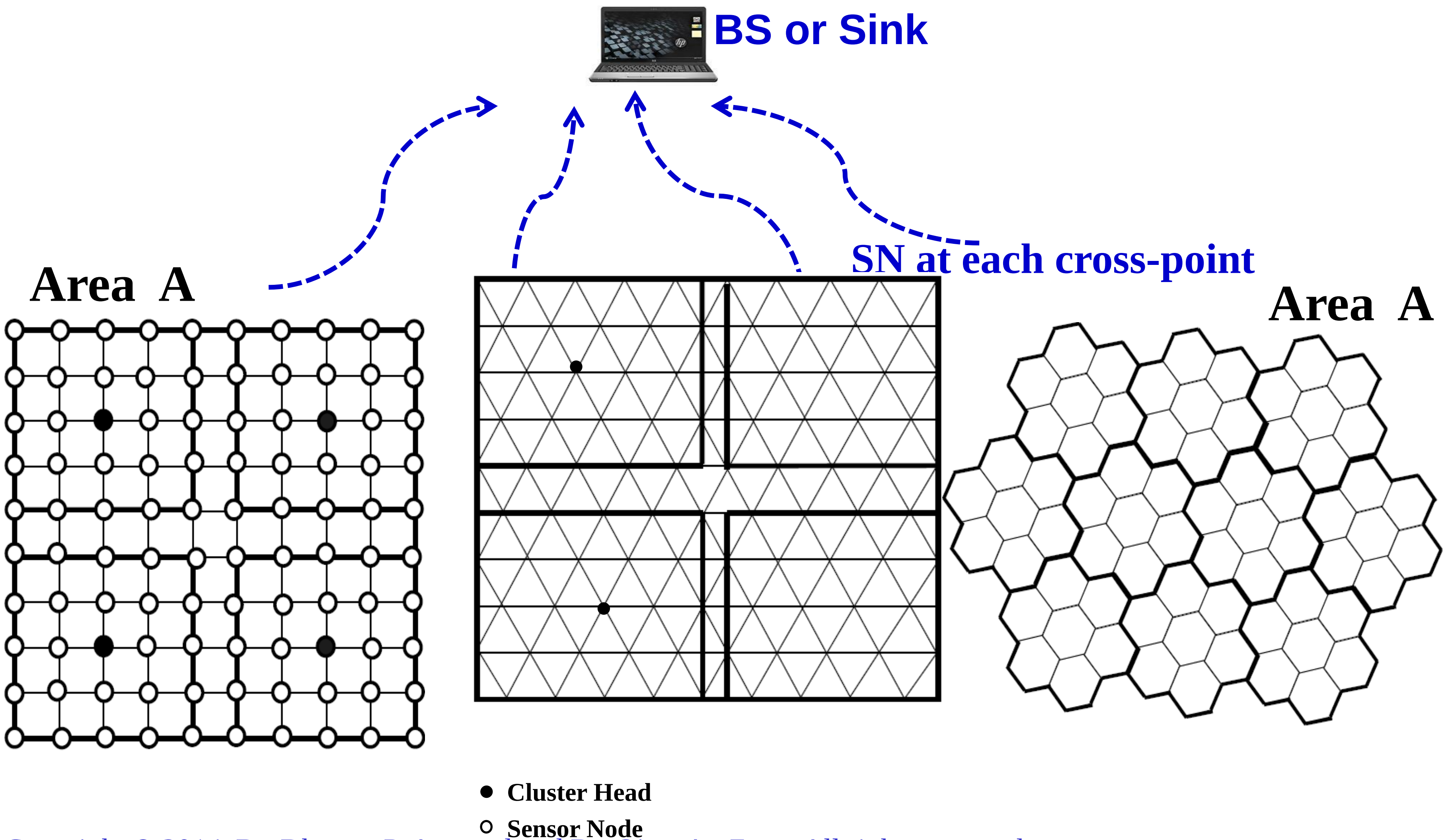


(a) 2-D Mesh

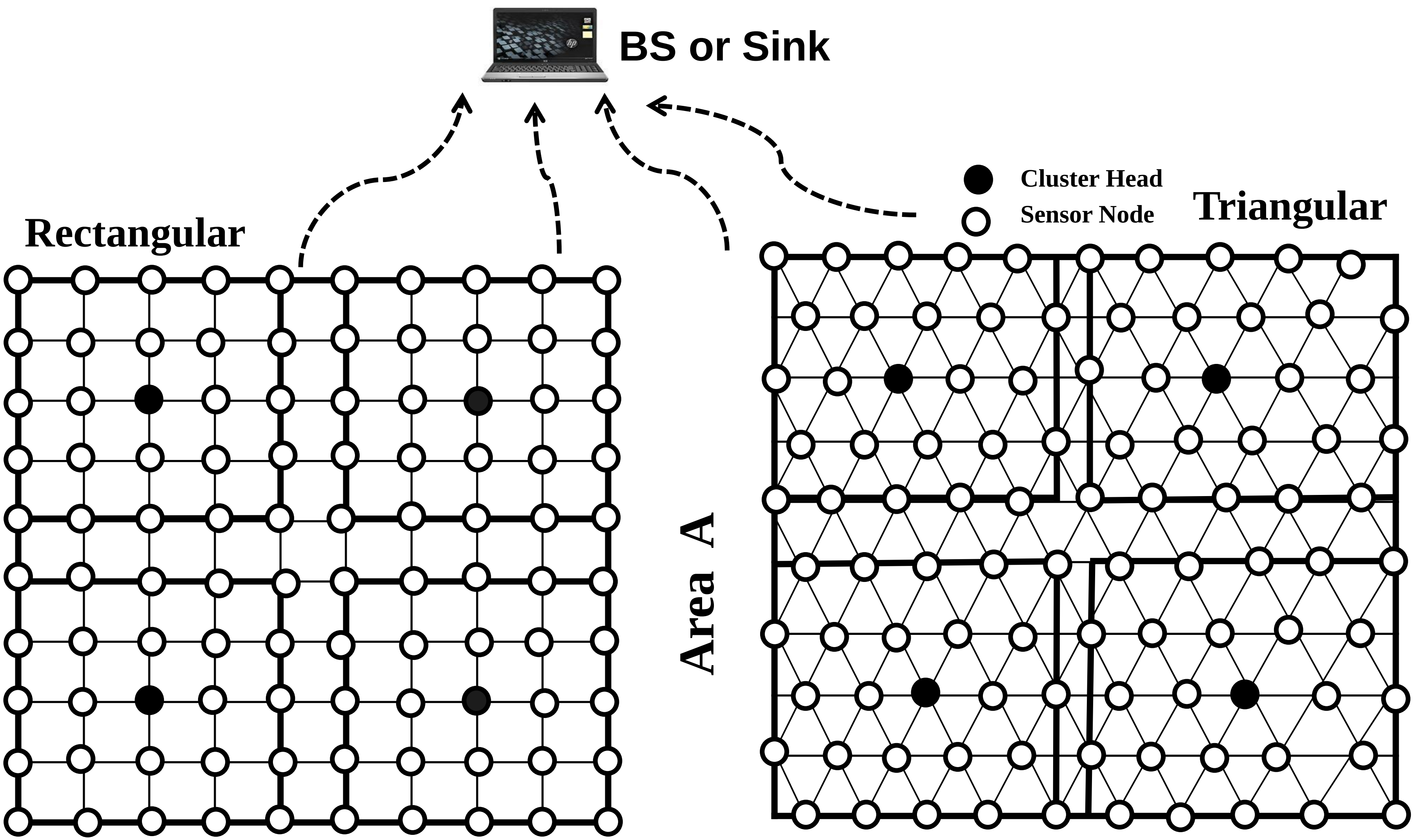
(b) Triangular

(c) Hexagonal

Larger Area A covered clusters of three regular topologies



Larger Area A covered with clusters of regular topologies



Covered Sensing Area

Placement	Distance between Adjacent Sensors	Area of the topology	Sensing Area to be covered by each SN based on Voronoi Diagram	Sensing area to be covered by N SNs
Rectangular	r	r^2	r^2	$N.r^2$
Triangular	r	$\frac{\sqrt{3}}{2}r2$		$N.\frac{3\sqrt{3}}{2}r2$
Hexagon	r	$\frac{3\sqrt{3}}{2}r2$	$\frac{\sqrt{3}}{2}r2$	$N.\frac{\sqrt{3}}{2}r2$

Full Coverage and Connectivity in Three Lattices

Required Number of Sensors for Full Coverage by three Lattices with different Sensing Ranges to cover an area A=100*100 units

Lattice Type	r_s= 5	r_s= 6	r_s= 7	r_s= 8	r_s= 9	r_s= 10	r_s= 11	r_s= 12	r_s= 13	r_s= 14	r_s= 15
Square	200	139	102	79	62	50	42	35	30	26	23
Hexagonal	924	642	472	361	286	231	191	161	137	118	103
Triangular	154	107	79	61	48	39	32	27	23	20	18

Required Number of Sensors for Full Connectivity by three Lattices with different transmission ranges for an area A=100*100 units

Lattice Type	r_c=5	r_c=6	r_c =7	r_c =8	r_c =9	r_c=10	r_c=11	r_c=12	r_c=13	r_c=14	r_c=15
Square	400	278	205	157	124	100	83	70	60	52	45
Hexagonal	924	642	472	361	286	231	191	161	137	118	103
Triangular	462	321	236	181	143	116	96	81	69	59	52

Full Coverage and Connectivity in Three Lattices

Required Number of Sensors for Full Coverage and Connectivity by Three Lattices for different ratios of sensing range to transmission range (Area A=100*100 units)

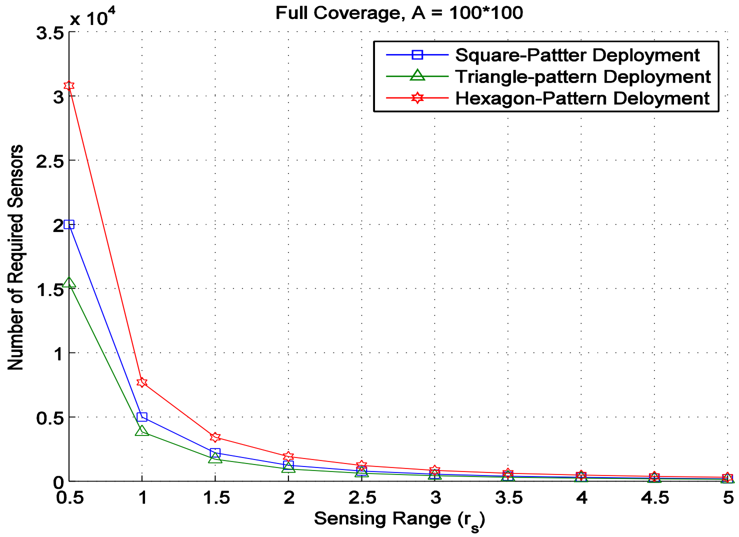
Lattice Type*	r_s= 5	r_s= 6	r_s= 7	r_s= 8	r_s= 9	r_s= 10	r_s= 11	r_s= 12	r_s= 13	r_s= 14	r_s= 15
Square	200	139	102	100	100	100	100	100	100	100	100
Hexagonal	308	214	158	121	95	77	77	77	77	77	77
Triangular	154	116	116	116	116	116	116	116	116	116	116

***Transmission range r_c is set as 10 meters, and sensing range is varied from 4 to 15 meters**

Full Coverage in Three Lattices when $R_s <= 5$

Required Number of Sensors for **Full Coverage** by three Lattices with different Sensing Ranges to cover an area A=100*100 units

(R_s <= 5)

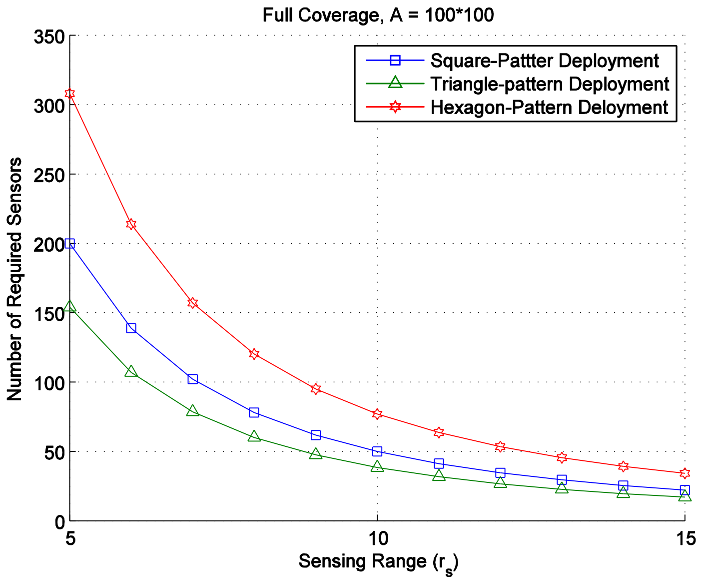


R_s	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
Square	20000	5000	2222	1250	800	556	408	313	247	200
Hexagon	30792	7698	3421	1925	1232	855	628	481	380	308
Triangle	15396	3849	1711	962	616	428	314	241	190	154

Full Coverage in Three Lattices When $5 <= R_s <= 15$

Required Number of Sensors for Full Coverage by three Lattices with different Sensing Ranges to cover an area A=100*100 units

($5 <= R_s <= 15$)

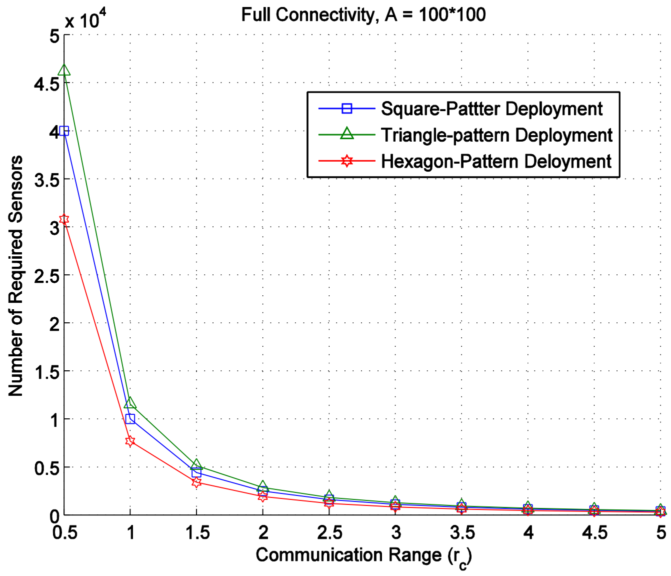


R_s	5	6	7	8	9	10	11	12	13	14	15
Square	200	139	102	78	62	50	41	35	30	26	22
Hexagon	308	214	157	120	95	77	64	53	46	39	34
Triangle	154	107	79	60	48	38	32	27	23	20	17

Full Connectivity in Three Lattices when R_c <=5

Required Number of Sensors for Full Connectivty by three Lattices with different Sensing Ranges to cover an area A=100*100 units

(R_c <= 5)

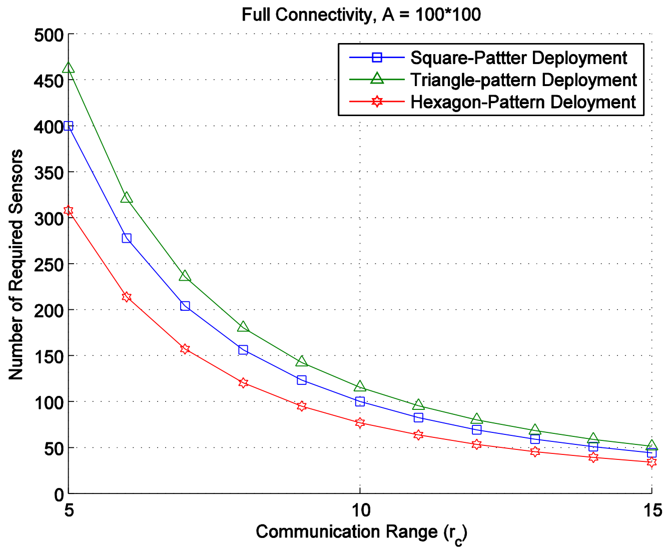


R_c	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
Square	40000	10000	4444	2500	1600	1111	816	625	494	400
Hexagon	30792	7698	3421	1925	1232	855	628	481	380	308
Triangle	46188	11547	5132	2887	1848	1283	943	722	570	462

Full Connectivity in Three Lattices when $5 <= R_c <= 15$

Required Number of Sensors for **Full Connectivty** by three Lattices with different Sensing Ranges to cover an area A=100*100 units

($5 <= R_c <= 15$)

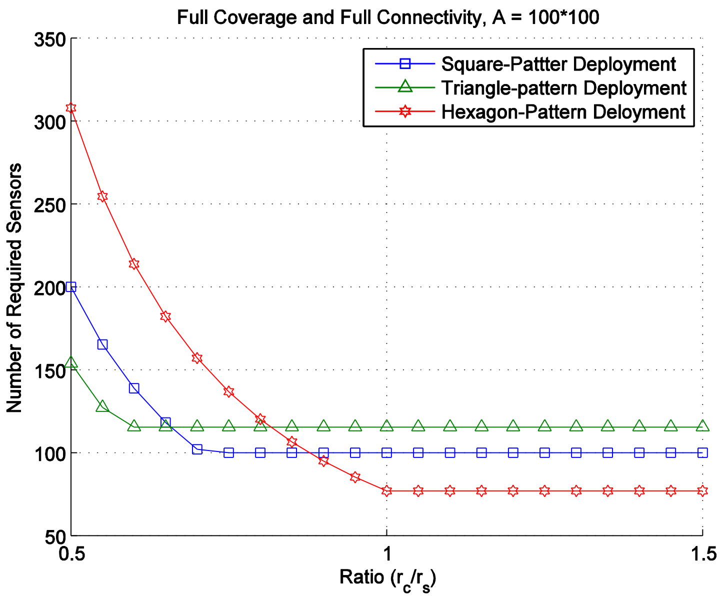


R_c	5	6	7	8	9	10	11	12	13	14	15
Square	400	278	204	156	123	100	83	69	59	51	44
Hexagon	308	214	157	120	95	77	64	53	46	39	34
Triangle	462	321	236	180	143	115	95	80	68	59	51

Full Coverage and Connectivity in Three Lattices

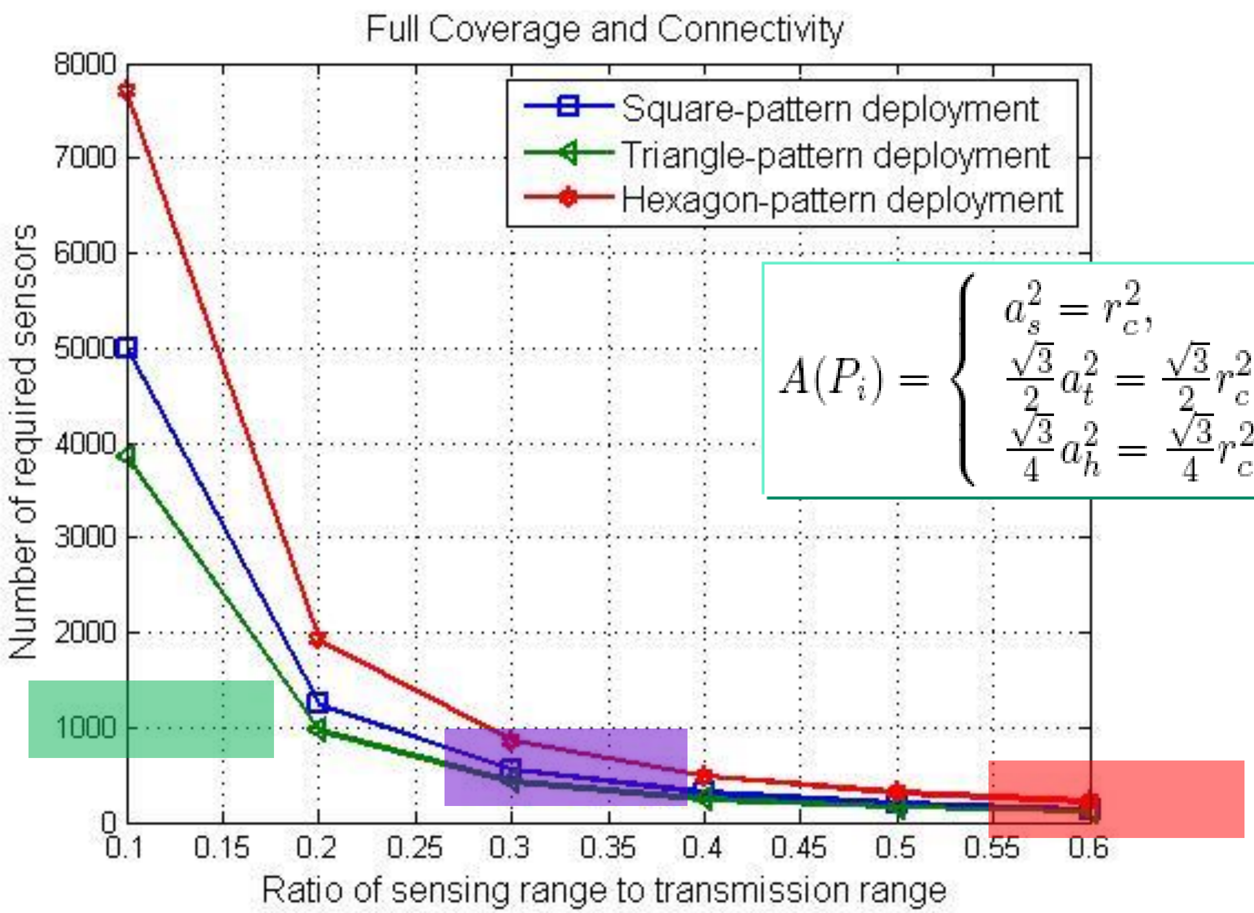
Required Number of Sensors for Full Coverage and Connectivity by Three Lattices for different ratios of sensing range to transmission range (Area A=100*100 units)

***Transmission range is set as 10 meters, and sensing range is varied from 5 to 15 meters**



Lattice Type	r_s= 5	r_s= 6	r_s= 7	r_s= 8	r_s= 9	r_s= 10	r_s= 11	r_s= 12	r_s= 13	r_s= 14	r_s= 15
Square	200	139	102	100	100	100	100	100	100	100	100
Hexagonal	308	214	158	121	95	77	77	77	77	77	77
Triangular	154	116	116	116	116	116	116	116	116	116	116

Required Sensors as a Function: Ratio of Coverage and Connectivity



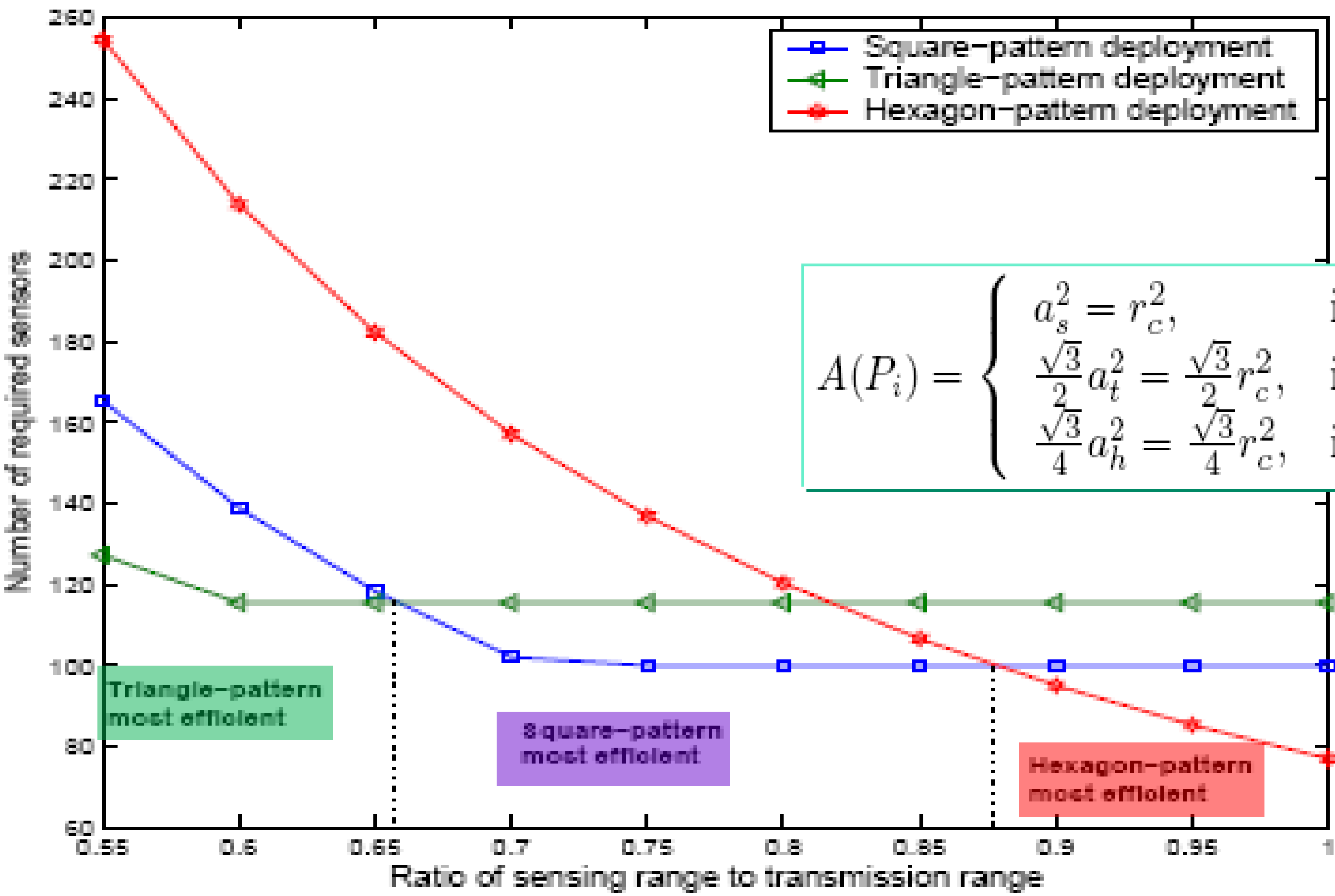
Sensing range: r_s

Transmission range: r_c

$$A(P_i) = \begin{cases} a_s^2 = r_c^2, & \text{i=s;} \\ \frac{\sqrt{3}}{2} a_t^2 = \frac{\sqrt{3}}{2} r_c^2, & \text{i=t;} \\ \frac{\sqrt{3}}{4} a_h^2 = \frac{\sqrt{3}}{4} r_c^2, & \text{i=h.} \end{cases}$$

Number of required sensors for connectivity and coverage (A=100*100)

Required Sensors as a Function: Ratio of Coverage and



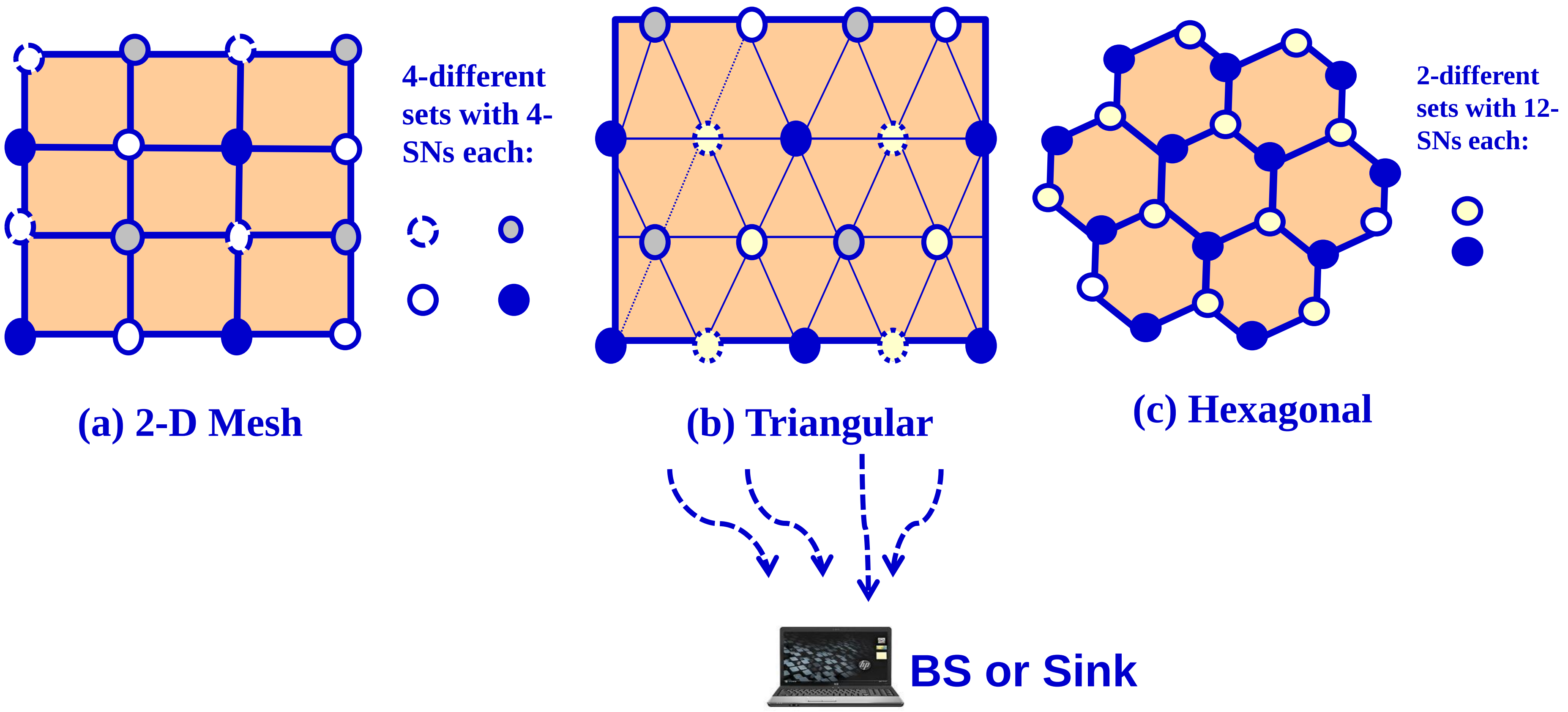
$$A(P_i) = \begin{cases} a_s^2 = r_c^2, & \text{i=s;} \\ \frac{\sqrt{3}}{2} a_t^2 = \frac{\sqrt{3}}{2} r_c^2, & \text{i=t;} \\ \frac{\sqrt{3}}{4} a_h^2 = \frac{\sqrt{3}}{4} r_c^2, & \text{i=h.} \end{cases}$$

Sensing range: r_s

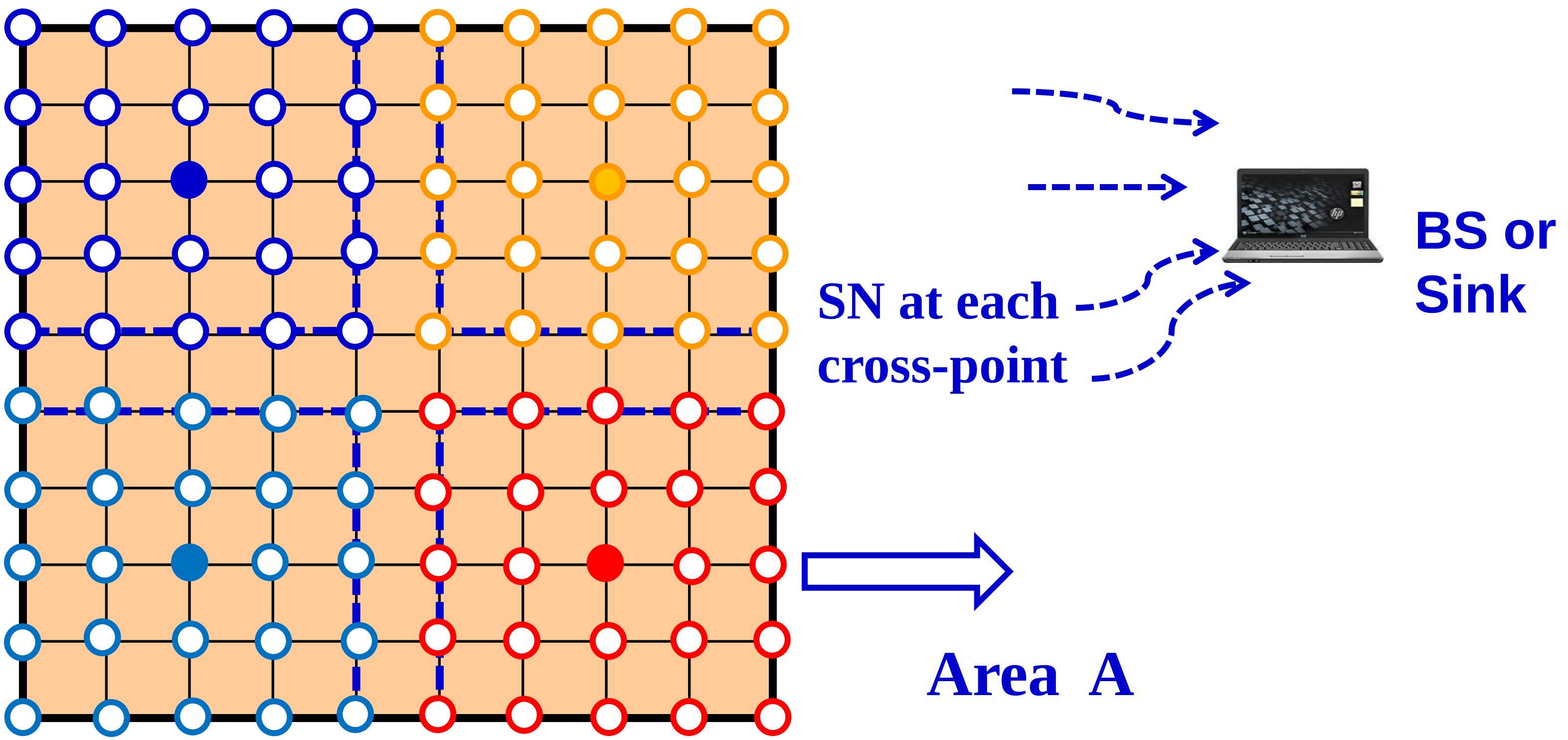
Transmission range: r_c

Number of required sensors for connectivity and coverage (A=100*100)

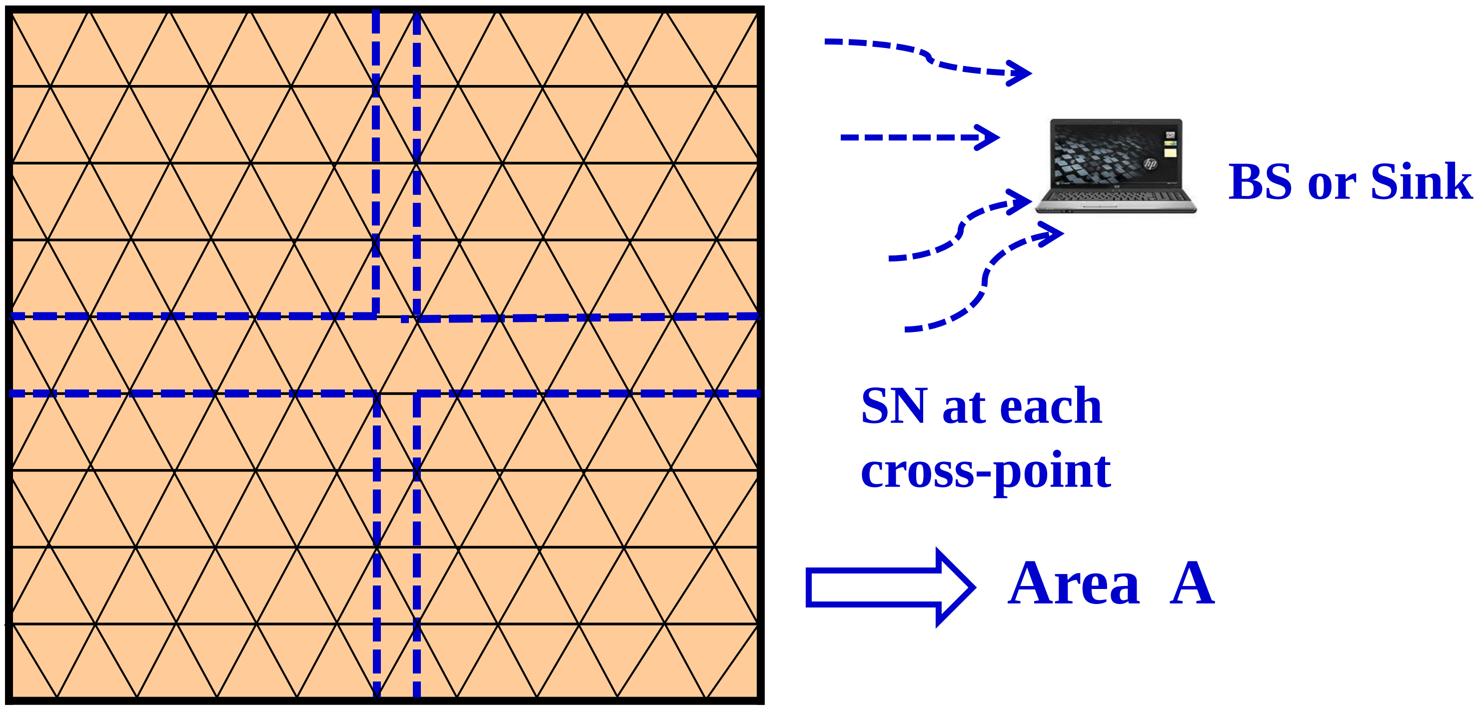
Three Lattices Topologies divided into different Sleep-Awake sets



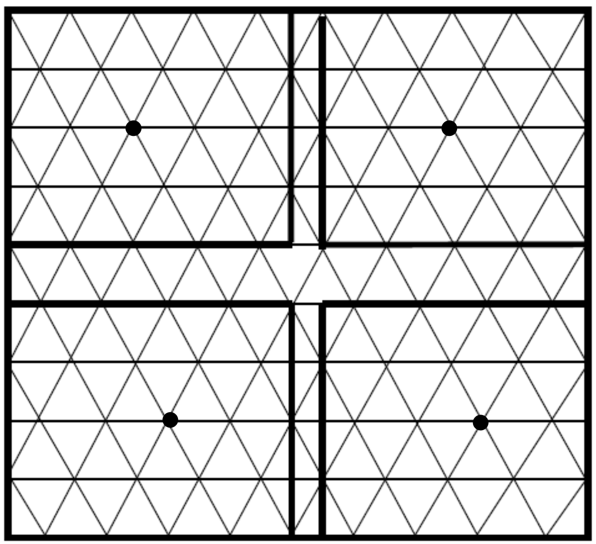
Larger Area A covered by clusters of Rectangular SNs

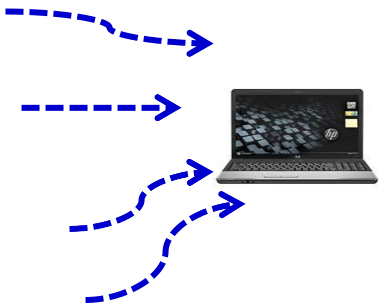


Larger Area A covered by clusters of Triangular SNs



Larger Area A covered by clusters of Triangular SNs





BS or Sink

SN at each cross-point

Area A

Larger Area A covered by clusters of Hexagonal SNs

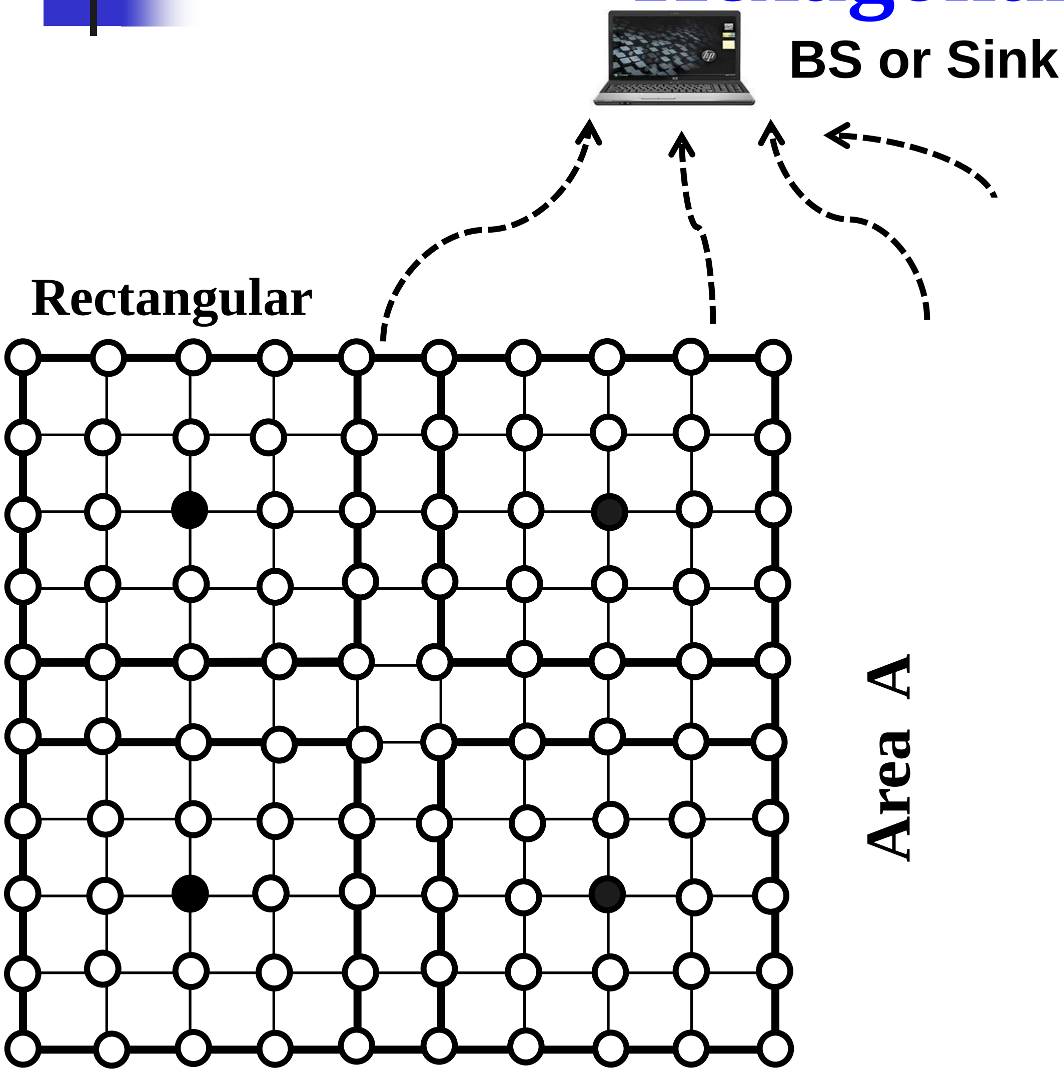
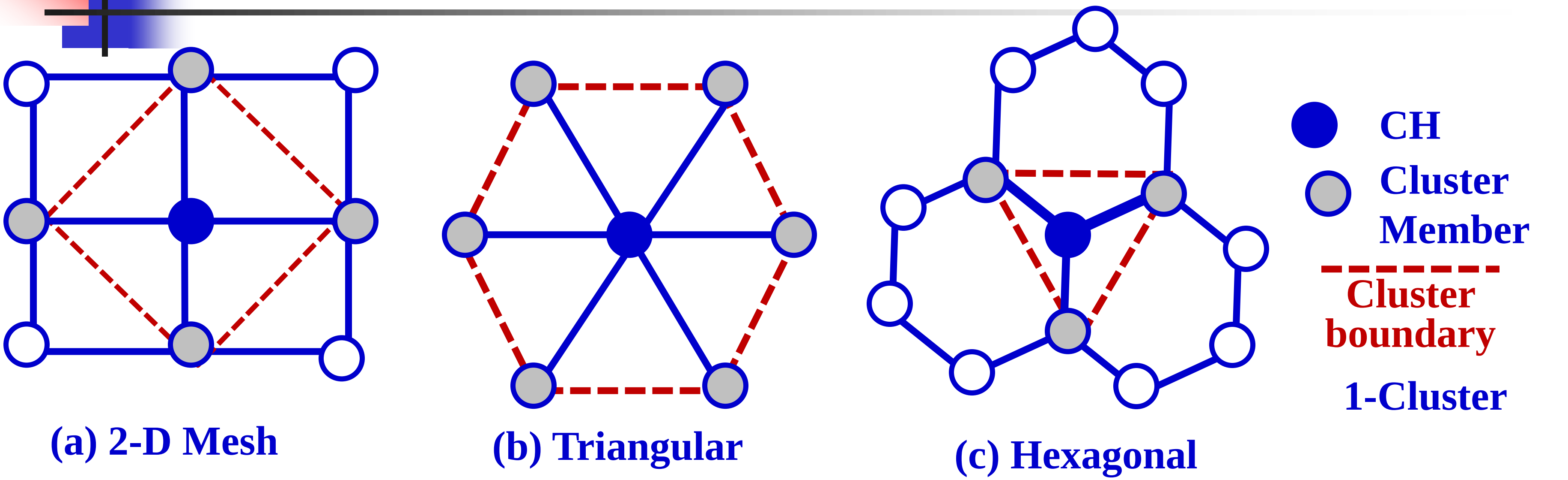
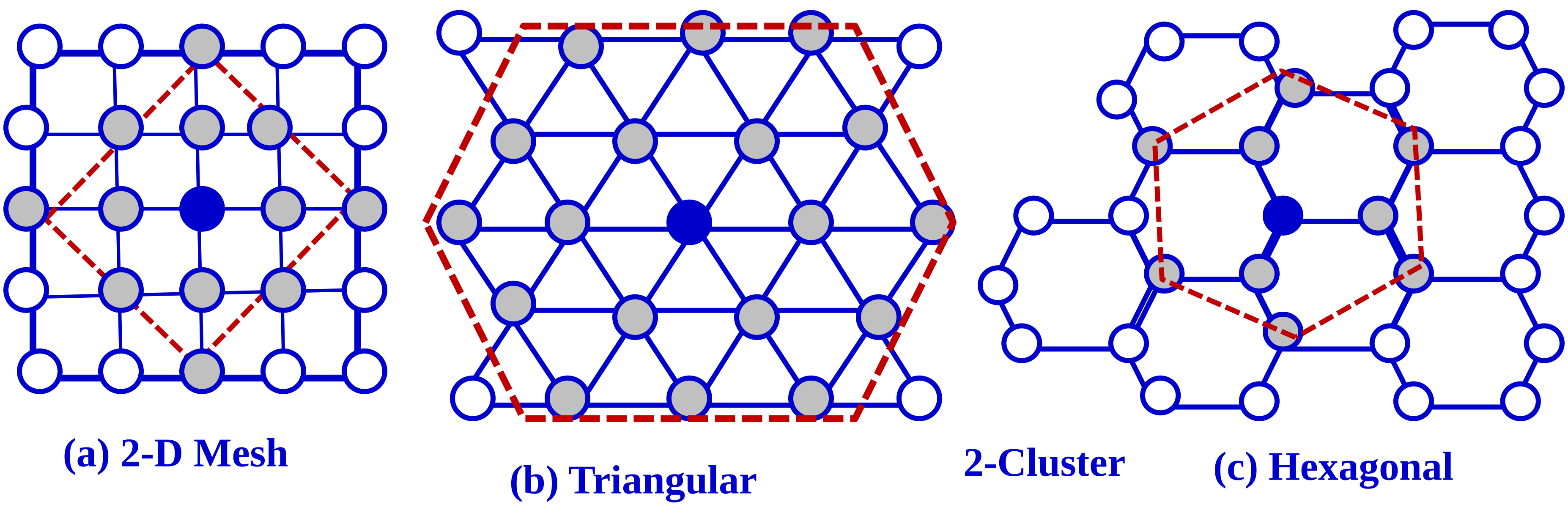


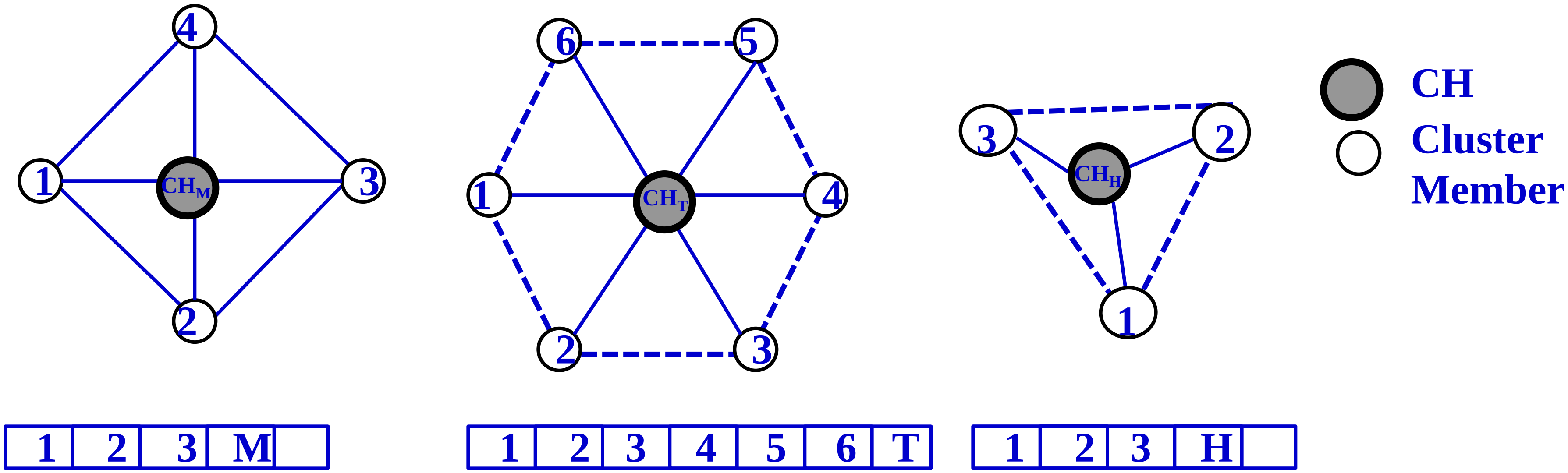
Figure 14.12: Larger Area A covered with clusters of regular topologies

k-Clustering of SNs for three topologies





TDMA Schedule for three Regular Topologies



(a) 2-D Mesh

(b) Triangular

(c) Hexagonal

Characteristics of Sensor Networks

- **Proactive Networks**
  - **The nodes in the network periodically switch on their sensors and transmitters, sense the environment and transmit the data of interest. This provides a snapshot of relevant parameters at regular intervals (periodic monitoring)**
- **Reactive Networks**
  - **In this scheme the nodes react immediately to sudden and drastic changes in the value of the sensed attribute (time-critical applications)**

Fundamentals of MAC Protocol for Proactive Wireless Sensor Networks

- **For Proactive Networks**
  - Report time
  - Attributes

- **Static Channel Allocation**
  - In this category of protocols, if there are N nodes, the bandwidth is divided into N equal portions either in frequency (FDMA), in time (TDMA), in code (CDMA), in space (SDMA: Space Division Multiple Access) or OFDM (Orthogonal Frequency Division Multiplexing)

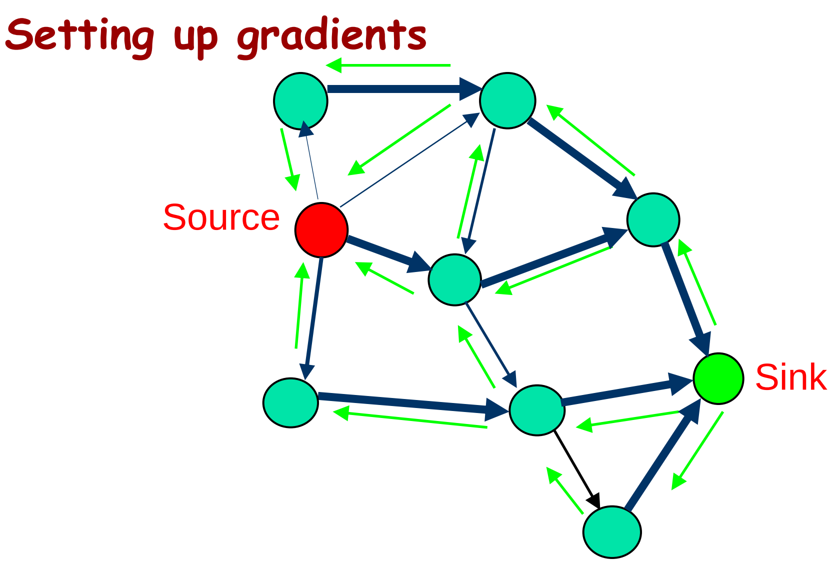
- **Dynamic Channel Allocation**
  - In this category of protocols, there is no fixed assignment of bandwidth. There are contention-based algorithms- carrier access multiple access (CSMA) and multiple access collision avoidance protocol

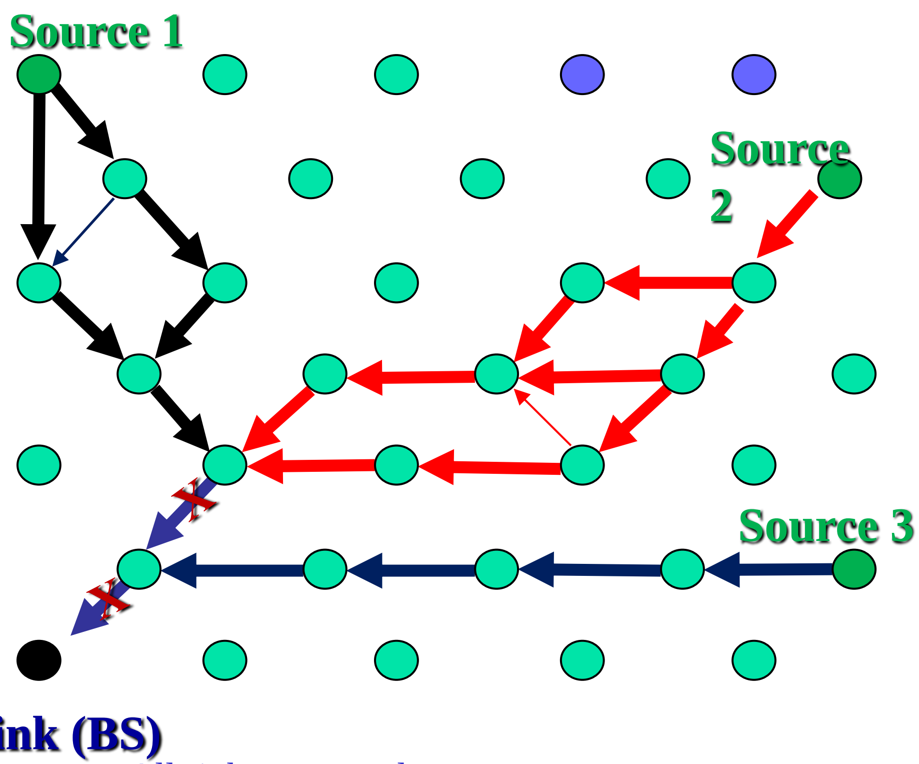
Flat Routing in Sensor Networks

- Routing in sensor networks is different from traditional wired or wireless networks
- Sensor networks are data centric, requesting information satisfying certain attributes and thus do not require routing of data between specific nodes
- Adjacent nodes may have almost similar data and might almost always satisfy the same attributes, rather than sending data separately from each node to the requesting node, it is desirable to aggregate similar data in a certain region before sending it (data fusion)
- The requirements of the network change with application, hence it is application specific

Directed Diffusion

- Directed Diffusion is a data aggregation and dissemination paradigm for sensor networks and gradients are used to send data towards requesting node
- If two paths have the same length, then one path is selected arbitrarily based on some criteria
- Data flowing from three sources are aggregated at *x*





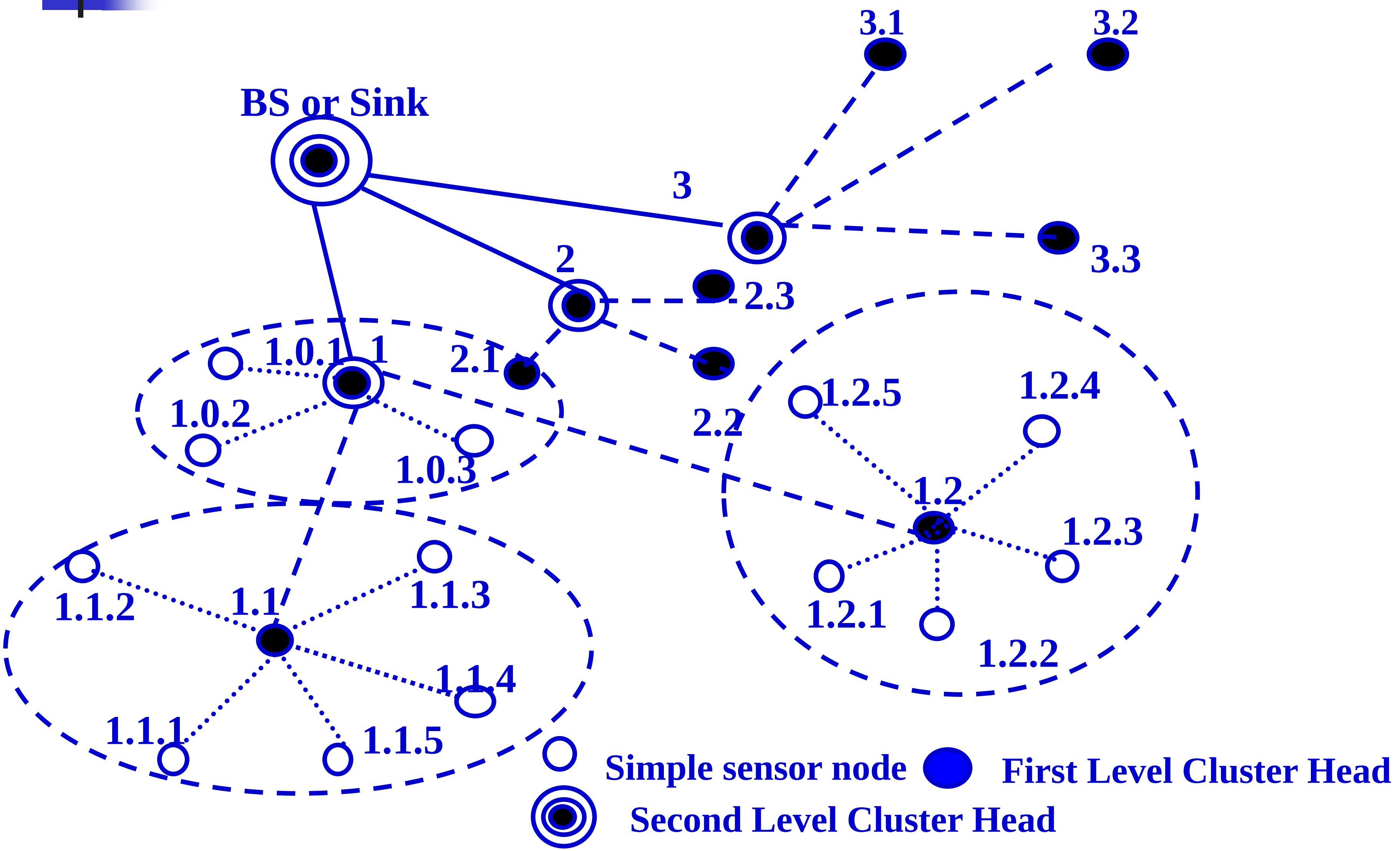
Routing in Sensor Networks – Flat Routing

- **Sensor Protocols for Information via Negotiation (SPIN)**
  - Disseminates the information at each node to every node in the network
- **Cougar**
  - Two approaches for processing sensor queries: warehousing and distributed
  - In warehousing, the data is extracted in a pre-defined manner and stored in a central database (BS)
  - Subsequently, query processing takes place on the BS
  - In distributed approach, only relevant data is extracted from the sensor network, whenever data is needed
  - COUGAR provides user representation and internal representation of queries
  - COUGAR has three tier architecture
  - -- Query proxy
  - -- Front-end component
  - -- Graphical user interface (GUI)

Hierarchical Routing in Sensor Networks

- Hierarchical clustering schemes are the most suitable for wireless sensor networks
- The network consists of a Base Station (BS), away from the nodes, through which the end user can access data from the sensor network
- BS can transmit with high power
- Sensors cannot reply directly to the BS due to their low power constraints, resulting in asymmetric communication.
- Nodes transmit only to their immediate cluster head (CH), thus saving energy
- Only CH needs to perform additional computation on data (aggregation)
- Adjacent cluster members have similar data

Hierarchical Routing (Cont'd)



Features of Reactive Sensor Networks

Case Study of a typical query:

- Report immediately if the temperature in the northeast quadrant goes below 5^0 C
- Retrieve the average temperature in the southwestern quadrant
- For the next two hours, report if the temperature goes beyond 100^0 C
- Which area has the temperature between 5^0 C and 100^0 C in the past two hours

- Complexity of a sensor node depends on expectations or functional requirements
- Mobile sensor network node remains the same
- Medium access using FDMA, could use CDMA model or fixed time slot in TDMA
- Sensed values are passed to other sensors and to a central controller for appropriate decision

Cluster Based Routing Protocol

Cluster Based Routing Protocol (CBRP)

- Here the cluster members just send the data to the cluster head (CH)
- The CH routes the data to the destination
- Not suitable for a highly mobile environment, as a lot of HELLO messages are sent to maintain the cluster

Scalable Coordination

- Clustering done using periodic advertisement
- Changes in networks conditions or sensor energy level results in reclustering

Low-Energy Adaptive Clustering Hierarchy (LEACH)

- LEACH is a family of protocols containing both distributed and centralized schemes and using proactive updates
- It utilizes randomized rotation of local cluster heads (CHs) to evenly distribute the energy load among sensors
- It makes use of a TDMA/CDMA MAC scheme to reduce inter and intra-cluster collisions
- LEACH is a good approximation of a proactive protocol
- The report time is equivalent to time frame
- Appropriate for constant monitoring applications

Reactive Network Protocol:TEEN

TEEN (Threshold-sensitive Energy Efficient sensor Network protocol)

- **It is targeted at reactive networks and is the first protocol developed for such networks**
- **In this scheme at every cluster change time, the CH broadcasts the following to its members:**
  - ***Hard Threshold (HT):* This is a threshold value for the sensed attribute**
  - ***Soft Threshold (ST):* This is a small change in the value of the sensed attribute which triggers the node to switch on its transmitter and transmit**

TEEN (Cont'd)

- The nodes sense their environment continuously
- The first time a parameter from the attribute set reaches its hard threshold value, the node switches on its transmitter and sends the sensed data
- The sensed value is stored in an internal variable, called Sensed Value (SV)
- The nodes will transmit data in the current cluster period only when the following conditions are true:
  - The current value of the sensed attribute is greater than the hard threshold
  - The current value of the sensed attribute differs from SV by an amount equal to or greater than the soft threshold

TEEN

Important features:

- **Suited for time critical sensing applications**
- **Message transmission consumes more energy than data sensing. So the energy consumption in this scheme is less than the proactive networks.**
- **The soft threshold can be varied**
- **At every cluster change time, the parameters are broadcast afresh and so, the user can change them as required**
- **The main drawback is that if the thresholds are not reached, then the nodes will never communicate**

Adaptive Periodic Threshold-sensitive Energy Efficient sensor Network protocol (APTEEN)

Functioning:

The cluster heads broadcasts the following parameters:

Attributes (A): **This is a set of physical parameters which the user is interested in obtaining data about**

***Thresholds*:** **This parameter consists of a Hard Threshold (HT) and a Soft Threshold (ST)**

Schedule: **This is a TDMA schedule, assigning a slot to each node**

Count Time (CT): **It is the maximum time period between two successive reports sent by a node**

APTEEN (Cont’d)

- The node senses the environment continuously
- Only those nodes which sense a data value at or beyond the hard threshold transmit
- Once a node senses a value beyond HT, it next transmits data only when the value of that attribute changes by an amount equal to or greater than the ST
- If a node does not send data for a time period equal to the count time, it is forced to sense and retransmit the data
- A TDMA schedule is used and each node in the cluster is assigned a transmission slot

APTEEN (Cont'd)

Main features of the scheme:

- **It combines both proactive and reactive policies**
- **It offers a lot of flexibility by allowing the user to set the count-time interval (CT) and the threshold values for the attributes**
- **Energy consumption can be controlled by changing the count time as well as the threshold values**
- **The main drawback of the scheme is the additional complexity required to implement the threshold functions and the count time**

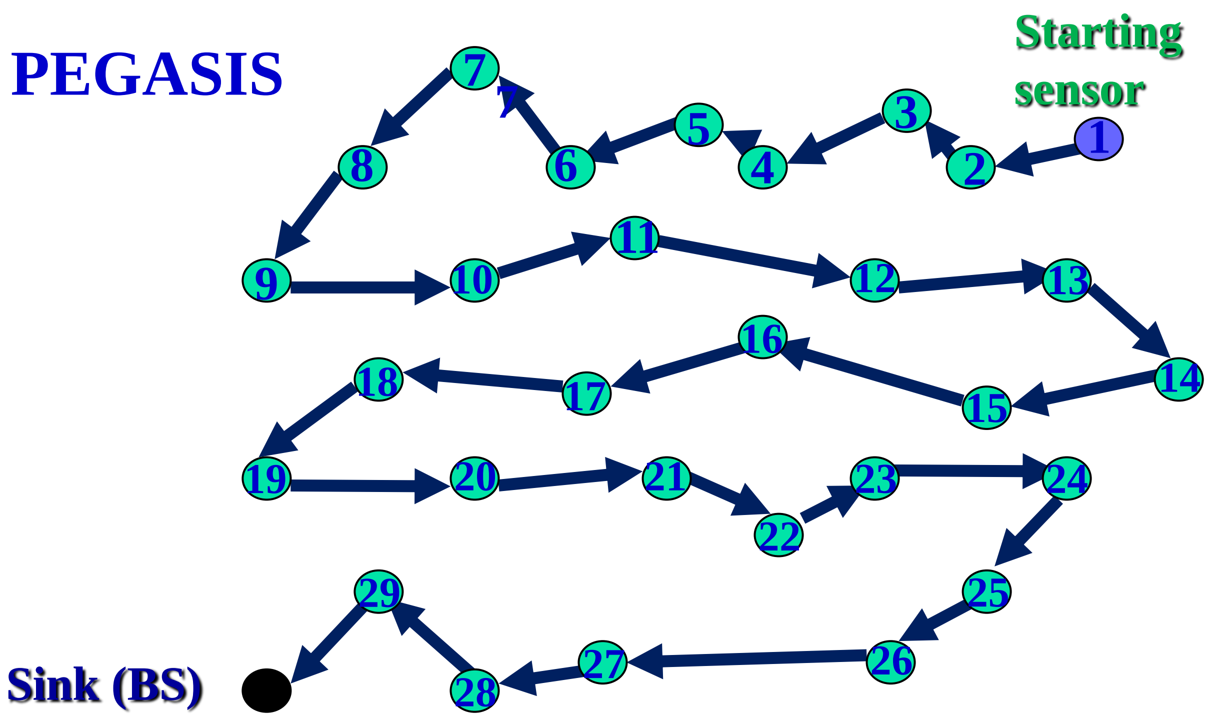
Hierarchical Vs Flat topologies for Sensor Networks

Hierarchical (Cluster-based scheme)	Flat
Reservation-based scheduling	Contention-based scheduling
Collisions avoided	Collision overhead present
Reduced duty cycle due to periodic sleeping	Variable duty cycle by controlling sleep time of nodes
Data aggregation by cluster head	Node on multi-hop path aggregates incoming data from neighbors
Simple but non-optimal routing	Routing is complex but optimal
Requires global and local synchronization	Links formed on the fly, without synchronization
Overhead of cluster formation throughout the network for random topologies	Routes formed only in regions that have data for transmission
Lower latency as multi-hop network formed by cluster-heads is always available	Latency in waking up intermediate nodes and setting up the multi-hop path
Energy dissipation is uniform	Energy dissipation depends on traffic patterns
Energy dissipation can not be controlled	Energy dissipation adapts to traffic pattern
Fair channel allocation	Fairness not guaranteed

Collaborative Information Processing

- Collaborative processing is another challenging area in sensor networks
- Power-Efficient Gathering in Sensor Information Systems (PEGASIS)
- All sensors are chained together so that each sensor receives only one message and transmits one till the message reaches the destination or the BS

PEGASIS



Node number	# Transmit	# Receive
1	1	
2 to 29	1	1
BS		1

PEGASIS applied to regular WSN

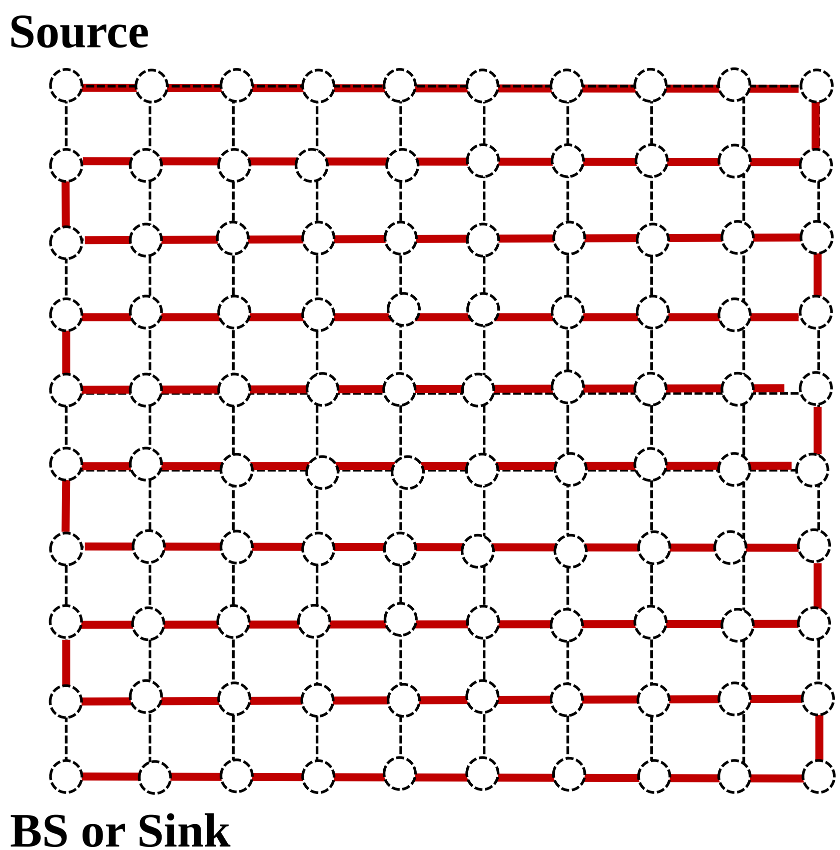


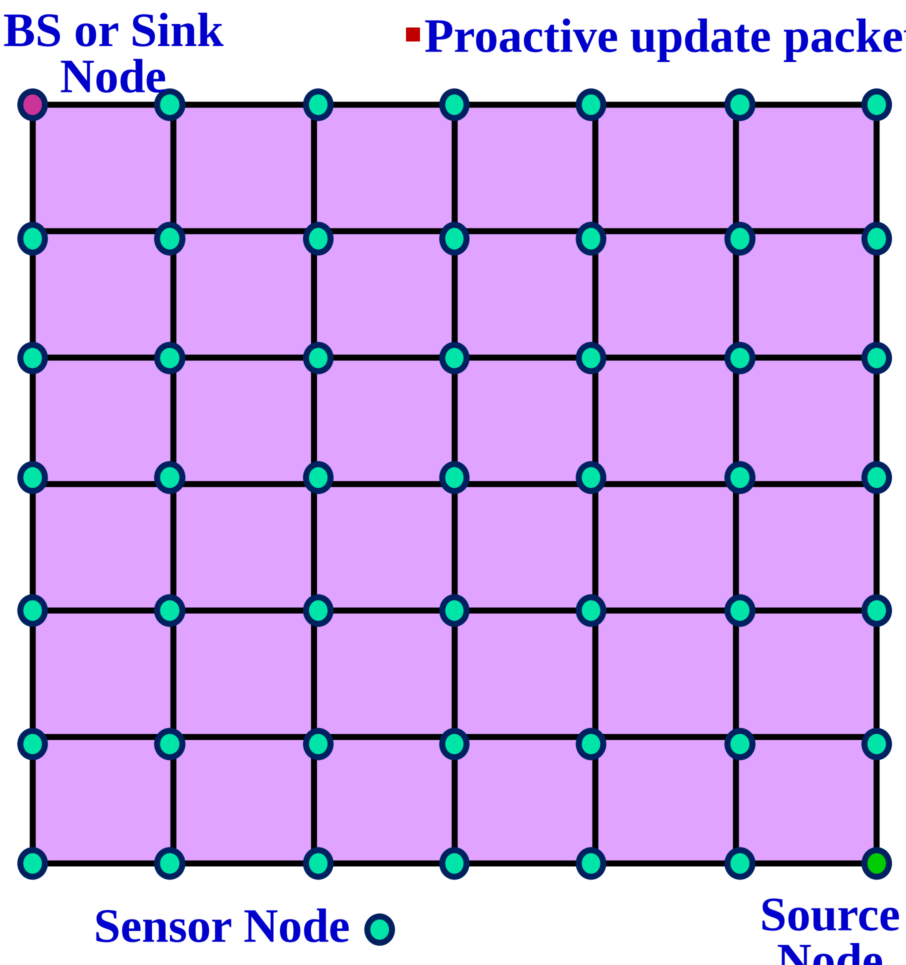
Figure 14.17 PEGASIS algorithm for a rectangular network.

Multi-path Routing

- Some queries are useful only when they are delivered within a given time frame
- Service differentiation is used to split the traffic into different classes based on QoS desired by each class
- Multihop routing must be able to adapt to the variation in the route length and its signal quality while providing the desired QoS
- The idea is to distribute routing of data packets between a given source and a sink on as many nodes as possible
- Multiple-path scheme preferable when either the density of active sources is high and their location is random
- Need a distributed and scalable scheduling algorithm that splits the traffic among multiple-paths in proportion to their residual energy

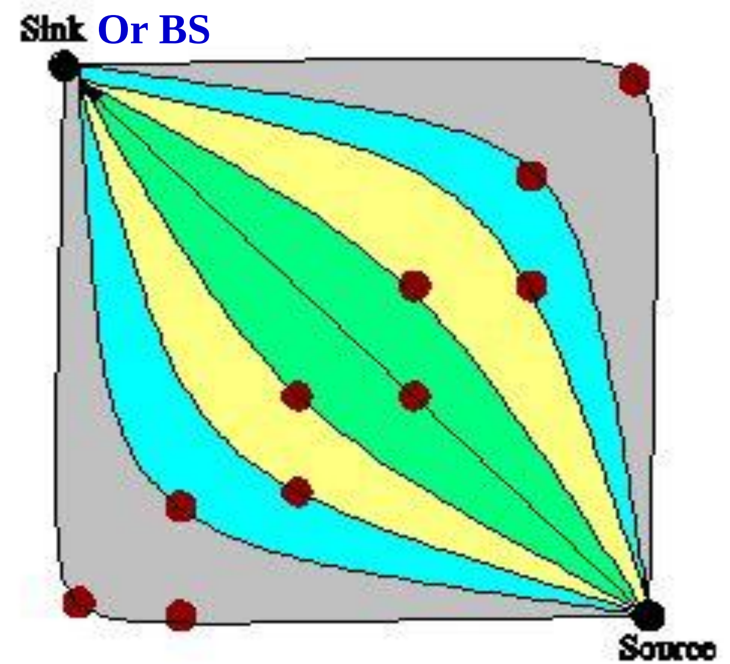
Multipath Routing

- Packets use a path based on criticality of data
- Urgent reactive response follow the shortest path
- Proactive update packets could be sent along a longer path



(a) 2-D Grid Architecture of Sensors

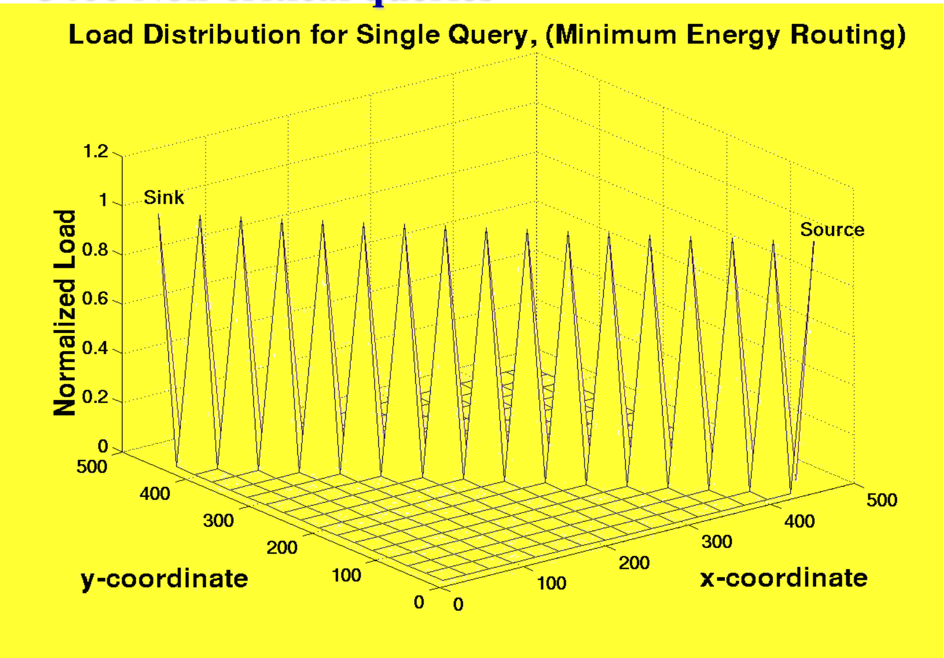
Sensors in 17 X 17 grid in 500m X 500m area
600 critical queries
5400 Non-critical queries



(b) Alternate paths from source to sink

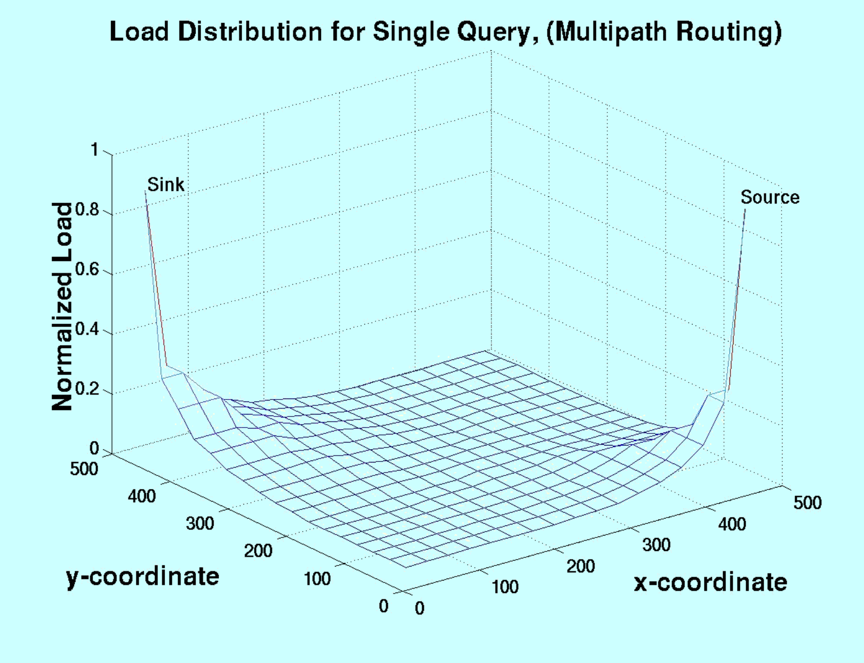
Energy Equalization with Multipath Routing

Sensors in 17 X 17 grid in 500m X 500m area
600 critical queries
5400 Non-critical queries



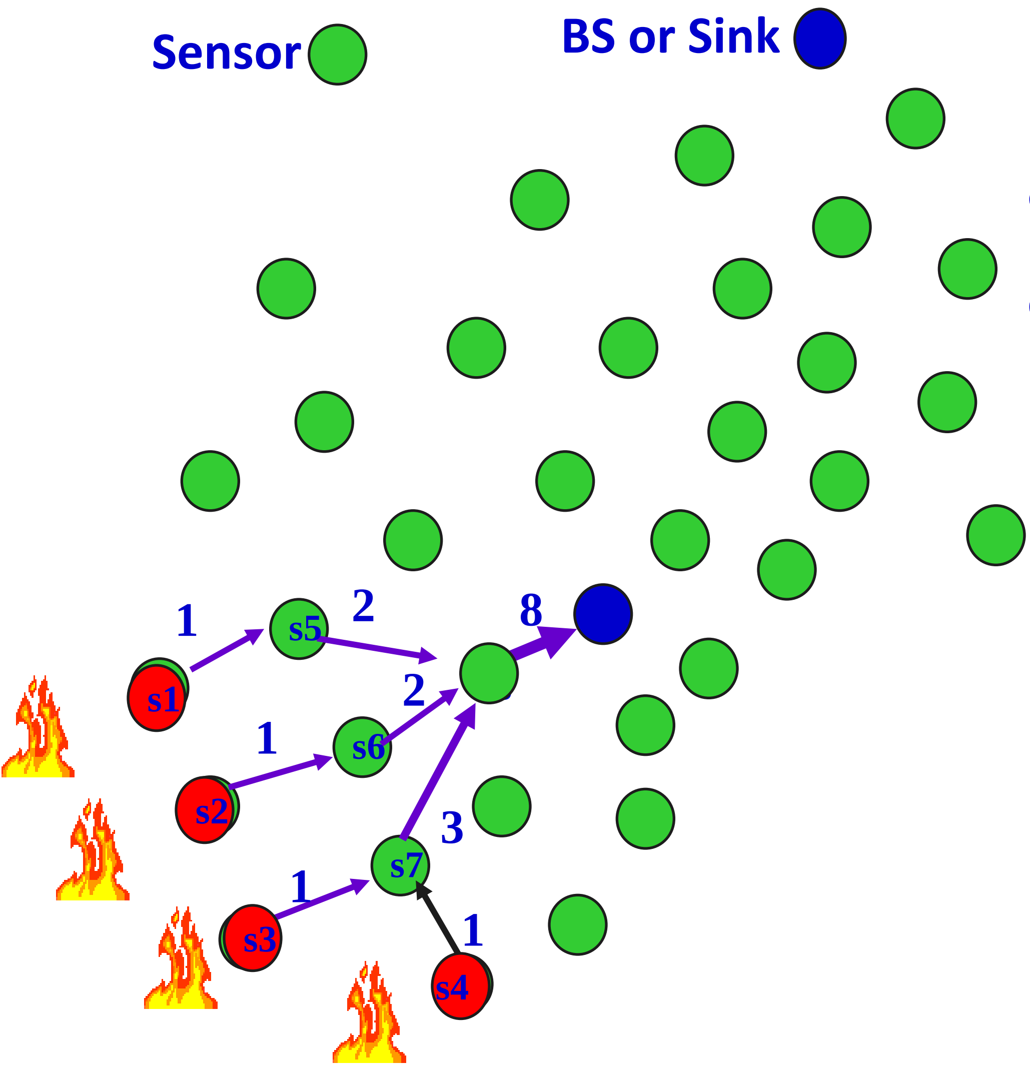
(B) Energy Consumption using single path routing

Sensors in 17 X 17 grid in 500m X 500m area
600 critical queries
5400 Non-critical queries



(b) Energy Consumption using multi-path routing

Energy Hole Problem



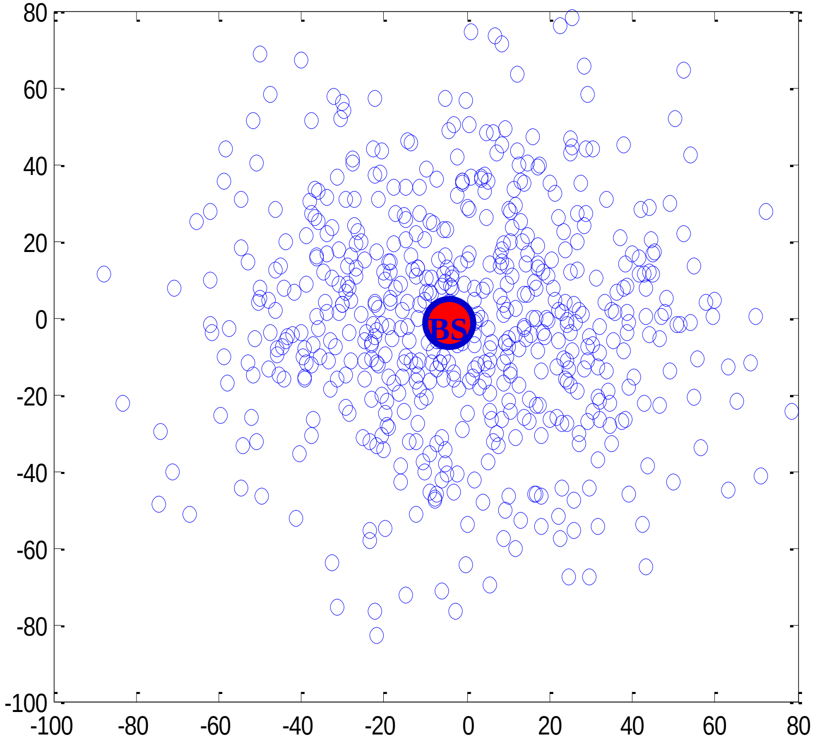
- Energy in a transceiver is consumed in:
  - Transmit
  - Receive
- Sensors have different amount of energy consumption
- Sensors close to the base station run out of energy at a much faster rate
- This is known as “energy hole” problem

Data Packets to be Received/Transmitted by Sensors

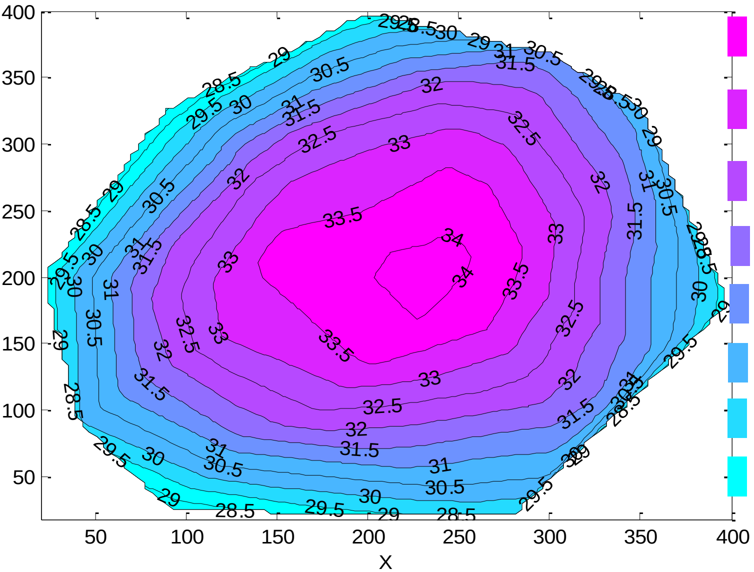
Sensor Number	Data Packets to be Received	Data Packets to be Transmitted
s1	-	1
s2	-	1
s3	-	1
s4	-	1
s5	1	2
s6	1	2
s7	2	3
s8	7	8

Gaussian Distribution of Sensors

- **Deploy more sensors near the base station and**
- **One such approach of Gaussian distribution of sensors**



Data Aggregation



) Contour of original rooftop data

at the University of Washington

- To form a tree in the area of interest and do aggregation at each tree node
- Aggregation Polynomial used:

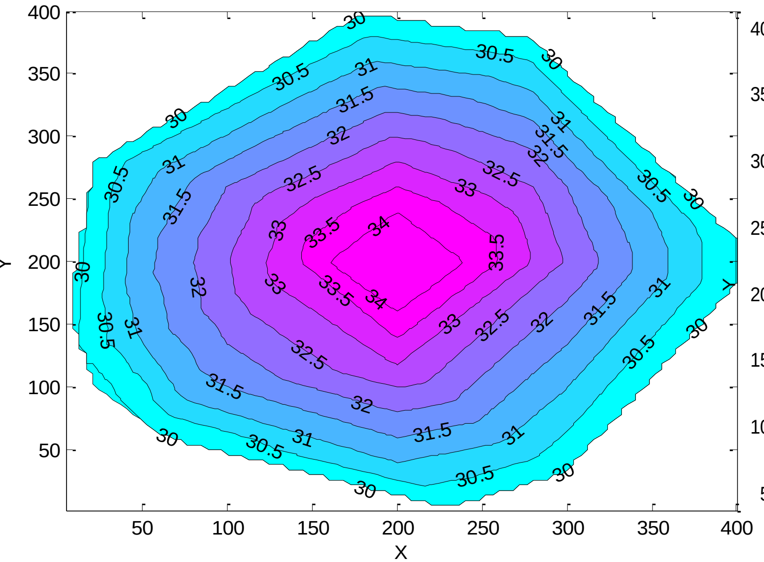
$p(x, y) = \beta_0 + \beta_1 y + \beta_2 y^2 + \beta_3 x + \beta_4 x y + \beta_5 x y^2 + \beta_6 x^2 + \beta_7 x^2 y + \beta_8 x^2 y^2$,

where x and y are the coordinates of the two dimensions

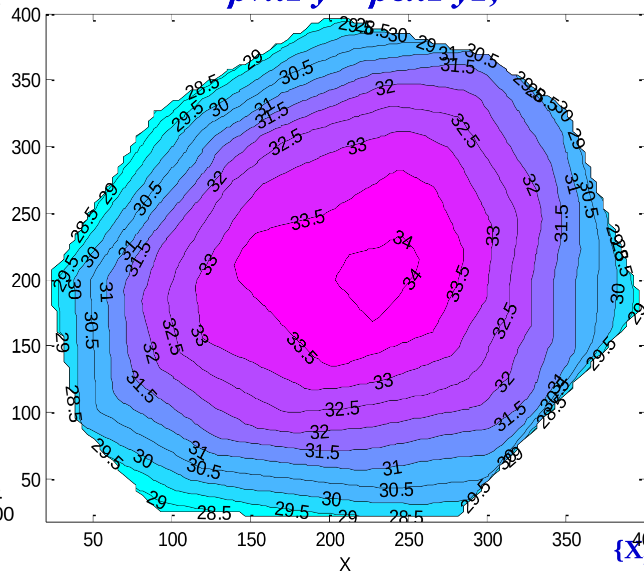
- Using received sensor data, the tree node computes these *β-coefficients* and pass that to the higher level tree node
- This process repeated till root node to give a final relation

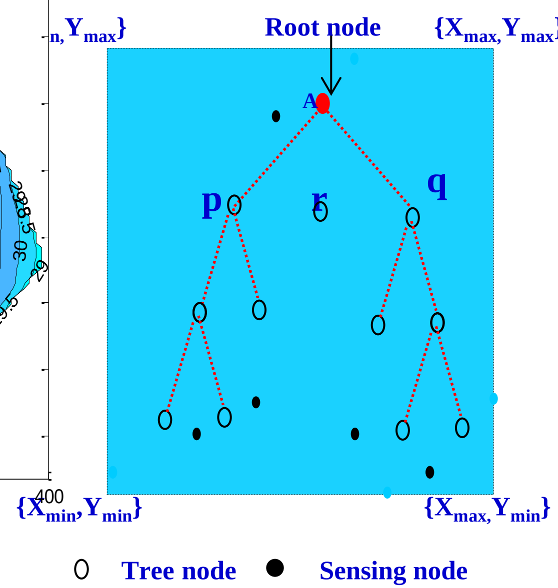
- **Volume of data depends on how frequently data is desirable**
- **There is a need to compress and combine data- "aggregation"**
- **Aggregated data is practically lossy due to compaction technique**
- **One novel approach is to utilize regression polynomial for spatially-distributed data**

Aggregation using Regression Polynomial



$p(x, y) = \beta0 + \beta1\ y + \beta2\ y2 + \beta3x + \beta4x\ y + \beta5x\ y2 + \beta6x2 + \beta7x2\ y + \beta8x2\ y2,$





(b) Data obtained by using Regression Polynomial

- $p(x, y) = 26.1429 + 0.0427163y - 0.000167934y^2 + 0.014x + 0.000249x\ y - 0.00000009231x\ y^2 - 0.0000181258x^2 - 0.00000860054x^2\ y + 0.00000000116143x^2\ y^2$
- Maximum error for a tree of depth 4 is limited to 5.64% , while most of the error is limited between 0-1.68
- Easier to find maxima/minima of *p by differentiating* the equation with respect to *x* and *y* and equating to zero

Operating System Design

- TinyOS architecture developed at Berkeley is an ultra-low power sensor platform
- Includes hardware and software, that enables low-cost deployment of sensor networks
- Combines advances in low-power RF technology with micro-electro mechanical systems (MEMS) transducer technology
- Most popular and freely available open source code is written nesC, a dialect of C programming language
- MagnetOS being developed at the Cornell University, is a single system image (SSI) operating system