Routing in Wireless Sensor Networks

**Term paper for fulfilling the requirements of course*

CS625: Advanced Network

Instructor: Dr. Dheeraj Sanghi

Author: Nilesh Mishra (Dept. of Computer Science and Engineering IIT Kanpur, India) Email:nilmish@cse.iitk.ac.in

Abstract

Wireless sensor networks are catching up as the primary mode for monitoring and collecting data in physically challenging environments. They find applications in various fields varying from environment monitoring, military applications to monitoring patients in hospitals. The low cost, ad hoc deployment, distributed sensing, and adaptability of wireless sensors have given them a big advantage over other methods. The constraints due to their inherent features make routing in wireless sensor networks a big challenge. This paper covers a number of routing protocols being used in the current systems. The stress is on energy aware routing where the network optimizes between efficient routing and maximizing life of the network.

I Introduction

Advances in technology have led to a shift of devices from wired to wireless domain. Availability of better battery technology have prolonged the wireless devices service time. With recent advances in ASIC (Application Specific Integrated Circuits) design we are able to create more compact and efficient electronic circuits suitable for specific purposes. Wireless sensors are a fusion of techniques from ASIC designing, wireless protocol designing, MEMS (Micro Electro Mechanical Systems) based sensors, advanced battery technology and robust distributed computing algorithms. Recent improvements in them have lead to a rapid growth in this field.

Wireless sensors are small, inexpensive, low power devices which are deployed in large numbers over an area in an ad hoc fashion. Each individual node has a limited processing power, limited view of the environment and limited battery life. Presence of large number of such nodes which can co-ordinate amongst themselves give them huge advantage over centralized single sensor based techniques.

Wireless Sensor networks have severe resource constraints, asymmetric many to one data flow and unreliable network nodes. Thus their primary objective is for energy conservation and prolonging network lifetime, which overlooks at performance, bandwidth and QoS to optimize the primary goals. Several current protocols are based on energy efficient routing. SPIM [1] uses meta-data negotiations to eliminate redundant data transmission. LEACH [5] is a cluster based protocol to evenly distribute energy load among sensors in the network. Directed Diffusion [4] uses low rate flooding and subsequent reinforcement of better paths. Another approach is to probabilistically choose some sub-optimal paths at times to reduce drainage of nodes on optimal path [6]. PEAS [7] keeps only necessary nodes active and puts rest into sleep to conserve energy. In agent based energy aware routing an agent is created which on its route to sink does data aggregation etc. to reduce redundant flow of data [2].

In section II an overview of sensor networks is given which covers aspects such as the architecture of sensor nodes, their OS, MAC protocols, deployment and the energy models adopted by various researchers in simulations. Section III discusses different routing protocols in detail. Section IV has the conclusion, section V gives the references, and section VI has definitions of some important terms.

II Wireless Sensor Nodes and Networks

Wireless sensors have the potential to be deployed in a number of scenarios. Some prominent examples are networked sensor deployment in environment monitoring, surveillance, health care, manufacturing, transport, and inventory tracking. They can be used to monitor endangered species, enemy movement, chemical pollution/contamination levels, etc.

Wireless sensor networks exploit their large numbers to monitor the event more closely. In an environment with large number of obstacles deployment of single powerful sensor may not be able to address issues related with line of sight and high signal to noise ratio. A single node may leave a number of holes and shadows which can be effectively covered using a distributed deployment. Since they are used in large numbers they are able to record an event with greater redundancy i.e. if one sensor misses there are others which can track the event. Deployments in large numbers also give them robustness to point failures, which is very important in mission critical tasks, e.g. even when a large number of sensors are destroyed in a bombing raid the remaining sensors may keep monitoring enemy movement. As they are relatively cheaper to manufacture when produced in large numbers they can be used in larger quantities to cover a much larger area which is the union of the areas covered by individual sensors. Localization of events also will be much easier and precise by using more then one sensors. Also low cost means that they can be used in a manner where we need not recover the nodes once they expend their battery reserves.

With proper communication protocols the wireless sensor network may be able to prolong the network life much more then individual node lifetime by exploiting redundancy and presence of multiple paths. Also there is lot of redundancy in the data coming from these sensors. Data aggregation can be done at various levels in order to reduce this redundancy. Since communication is the major consumer of energy it is much better to process maximum information locally to get rid of unnecessary data or send just the summaries. Due to small processing power the sensor networks have to depend on distributed processing algorithms for large scale computing. Depending on the deployment model the network may consist of uniform nodes with one or more powerful base stations which act as sinks for the data collected. Alternatively the nodes can be non uniform with periodic deployment of more powerful nodes (in terms of energy and computational capabilities).

Sensor networks offer unique constraints in constructing hardware and protocols for them. The major challenges faced are:

- *Nature of deployment*: Sensor networks are to be deployed in an Ad-hoc fashion where there is little correlation in the nodes placement. It is left to the nodes to detect the network and its distribution. The hardware should be rugged to sustain the hardships of deployment in hostile conditions.

- *Self configuration*: The nodes are deployed to work in an unattended operation without human intervention. Thus, the system must be completely self configurable.
- *Energy constraints*: The nodes are untethered and are used in a use and throw fashion, thus they have to conserve the onboard energy source parsimoniously. Since communication dominates energy consumption the nodes need to minimize communication. Also as the small size limits the radio transmission limits the range of communication, it has to be done in small multi hop fashion.
- *Adaptability*: The network must be able to respond and take care of changes in network topology (addition/failure of nodes). Also for robustness environmental factors must also be taken care of.

Since sensor's radio is small power, short range radio the communication is ideally multihop. As the distance increases the probability of a signal getting lost or corrupted in a multihop communication gets increased. The protocols must be designed in a manner to reduce global communication and emphasis must be on to do setup and control on basis of local interactions.

Architecture

The node's architecture can be divided into a number of subsystems. Since the major stress in sensor network is to optimize energy usage we can make all these subsystems energy aware. Figure 1 gives an overview of a node's framework.

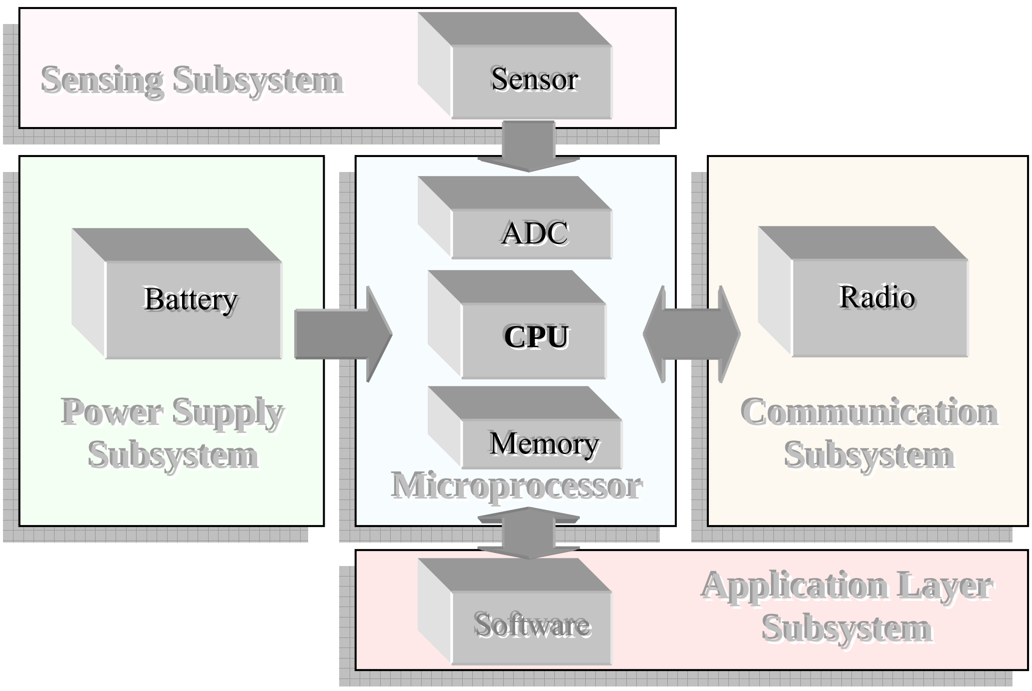

Figure 1: Architectural diagram of a wireless sensor node

- *Computing subsystem*: The computing subsystem has a microcontroller unit which in tandem with the software controls the sensor and communication protocols. The Microprocessor can have multiple levels of operation depending on power requirements. This power management requires frequent shuttling between various modes.

- *Power supply subsystem*: This subsystem supplies the energy to the node. To prolong the battery life we need to ensure that high current is not drawn from the battery for prolonged periods. A small current flow ensures longer battery life.
- *Communication subsystem*: It consists of a short range radio for communication. Radios normally have Transmit, Receive, Idle and Sleep modes of operation for power conservation. To conserve energy the radio must be turned off if not required as it is the major consumer of energy.
- *Sensing subsystem*: This module consists of sensors and actuators which link the sensor node with the outside world. To conserve energy, low powered devices must be used with provisions to turn off whatever not required.
- *Application subsystem*: This module is the application layer which has the software running the nodes hardware. It has the OS and routines handling the routing, communication over radio, etc. Application layer must also be energy aware as it controls the other subsystem's behavior, putting then in and bringing them out of, sleep to conserve energy.

The network architecture governs how information is taken care of at the network level. SINA (Sensor Information Networking Architecture) is a middleware architecture which facilitates adaptive organization of sensor information by allowing application to issue queries, collect information and monitor changes in the sensor nodes. This model has hierarchical clustering which clusters nodes based on power level and/or proximity parameters. The clustering is done recursively to obtain a hierarchy of clusters. This is done in order to decrease power requirements to prolong network life. The information flowing in from the nodes is accessed via attribute based naming, than direct addressing. In this instead of asking information from a particular node the network is enquired for nodes having specified information. It also defines a programming interface called SQTL (Sensor Query and Tasking Language) between sensor applications and SINA middleware.

The nodes in the network can increase their service life by adaptively going to sleep as and when their active participation is not required. PEAS (Probing Environment and Adaptive Sleeping) [7] is a protocol designed for robust operation of sensor nodes in hostile harsh environments by randomizing sleeping times. It overcomes the requirement of keeping a detailed table of the states of neighbors as required in other protocols which is difficult in dense deployment. The nodes are given exponentially distributed sleeping times initially. When a node wakes up it sends a PROBE packet in a certain probing range. If a node is active in the region it sends a REPLY message containing a parameter based on the probing frequency observed by it which is used to decide the next sleeping cycle of the just woken up node. Thus a node only resumes work when it fails to receive a REPLY message for a certain period.

Operating System

The operating system for sensor networks are real time OS configurable for a given task to optimize resource usage. TinyOS is one such OS having component based architecture.

MAC Layer Protocols

Current link layer protocols can be broadly divided into two categories: contention based and organized methods. Since efficient energy utilization is the primary goal, contention based methods are not common in sensor networks as they require continuous listening on radio and wastage in collision. Organized methods are generally clustering based. The nodes elect a cluster head which gives the various nodes time slots for communication. The clusters have to be elected locally in a distributed fashion so as to be scalable and handle node failures. In the contention based methods also the protocol differs from the traditional RTS-CTS-DATA-ACK mechanism. We need to limit the number of control packets to conserve energy.

The data traffic may be low for a long time with short burst in traffic. The traffic is multihop and usually goes towards the base station. At each node there is a contention between traffic originating at that node and the data being routed through it. This is because each node doubles as a data generator and router. The probability of corruption and contention at every hop is higher for the nodes residing farther from base station. Since energy is invested in routing the packets, the more time a packet spends in the network the more cost gets associated in dropping the packet. Thus, higher priority must be given to routed traffic than originating traffic.

SMAC is a well known protocol which caters to the requirements for sensor networks. Since the major source of energy drains are collisions, overhearing neighboring traffic, control packets overhead and idle listening, it looks into methods to avoid them. SMAC uses three methods to achieve this:

- Going for periodic sleep when the channel is idle or when a neighboring node is transmitting.
- Formation of local virtual clusters to synchronize wake up and sleep times to reduce control packet flow.
- Message passing where we break a long message into fragments to avoid the overhead and delay associated with long messages getting lost.

SMAC also has mechanisms for efficient neighbor discovery and setting up a communication network without central control.

Localization

The sensor nodes are deployed in an ad hoc fashion, where generally there is no prior knowledge of the locations where the nodes will get placed. The problem of estimating the co-ordinates of a node is referred as localization. Localization can be divided in two classes, a) Coarse grained localization and b) Fine grained localization. Generally the coarse grained localization problem is solved using recursive trilateration /multilateration. The sensor nodes can be seen in a hierarchy where there is a group of nodes in the network which are aware of their positions (via GPS, etc.). These nodes send beacons throughout the network advertising their location. Other nodes listen to these beacons and calculate their locations using one or more of these beacons. For those nodes which do not have access to these beacon nodes, the nodes having located their positions earlier act as beacon nodes. This is known as iterative multilateralization. One drawback of this method is that the errors are cumulative thus increasing with each iteration. In fine grained localization methods we use timing analysis, signal strength, signal pattern matching and directionality of signal.

There are algorithms for proper placement of beacon nodes. The beacon nodes can be distributed in a uniform fashion or in a very dense beacon placement. Uniform deployment need not ensure visibility while the dense deployment may introduce overhearing. There are algorithms for optimal beacon/node placement which do it iteratively. These methods use redeployment, addition and turning off of selective beacons to achieve efficiencies.

Energy Models [5][2]

In most of the simulations and test bed experiments the sensor nodes and the sensor network is studied by constructing a formal model for them. Different assumptions regarding the radio characteristics for transmission and receiving modes can tilt the balance for a protocol. Consider the first order radio model where to transmit a *k* bit data packet from node N_i to N_j the energy expended are:

- Source node N_i

  $$E_{N_i} = E_{elec} * k + E_{amp} * k * d^2,$$

  here E_{elec} is the energy dissipation to transmit and receive a bit, and E_{amp} is the energy expended by transmitter in achieving an acceptable signal to noise ratio. Typical values for E_{elec} = 50nJ/bit, and E_{amp} = 100pJ/bit/m^2.
- Receiver node N_j

  $$E_{N_j} = k * E_{elec}$$

Thus the energy usage at an intermediate node for transmitting and receiving a packet is $E_{N_i} = 2 * E_{elec} * k + E_{amp} * k * d^2$. To reduce energy usage we need to not only have to transmit for smaller distances but as there is energy consumption for receiving data we need to minimize the number of transmitted packets as well. As can be seen from the above equations it is likely that a single hop transmission for a larger distance may be at times more efficient then a multi hop transmission of the same packet.

III Routing Protocols

Making the complete network energy efficient we need to make all subsystems energy aware as well, including the node's hardware and communication protocols. Routing of data is the major consumer of energy in the network; efficient routing protocols can reduce the energy consumption drastically. Routing protocols for sensor networks can be classified based on a number of criterions such as proactive vs. reactive routing, hierarchical vs. non-hierarchical routing, etc. In proactive routing we setup the data path in advance and maintain suitable routing tables while in reactive routing the routing tables and paths are created on the fly as and when needed. Routing can be either source or destination initiated. Below we discuss a few routing protocols. Directed Diffusion is a reactive protocol which uses attribute-value paired naming for naming data. Energy aware routing protocols use some metric of energy remaining in the nodes/network to decide the routes to be chosen. SPIN is negotiation based data dissemination protocol which uses metadata for identifying redundant packets. LEACH is a hierarchical clustering protocol which uses randomized cluster head rotation to maximize network life.

Directed Diffusion [4]

Directed diffusion is a data centric, data dissemination protocol which names data items by attribute-value pairs. A node requests data by sending interests for named data and data matching this interest is drawn towards the node. It is a destination initiated reactive routing protocol where the possible paths are set up during interest dissemination and later one of the paths is reinforced to have a single optimal path for data propagation. There are no separate route discovery and route maintenance protocols. Interests are periodically flooded for route maintenance. There are no separate routers and each node does its own interpretation, transmission and consumption of data. Propagation of data and its aggregation depend only on localized interactions. Given below are the key features regarding Directed Diffusion.

- *Naming:* Each task is named by a list of attribute value pairs. The naming scheme can be simple attribute-value pair, with attribute having an associated range of values, or hierarchical or intentional names.
- *Interest propagation:* Interests are injected into the network at any arbitrary node (sink). An initial exploratory interest message is broadcasted by the sink periodically. This message has a low data rate for response, marked in it. These interests are soft states that are periodically refreshed by the sink. The refresh rate is chosen so as to trade off overheads for increased robustness to lost interests. Each node has an interest cache which records distinct interest entries. Based on the definition of distinct interests we can have interest aggregation at these nodes. Each entry has several gradient fields up to one per neighbor. Upon receiving an interest an entry is created in the cache based on whether it is a distinct entry or not. Each entry has a single gradient towards the neighbor from which the interest was received. When a gradient expires it is removed from the cache. On receiving an interest a node may decide to resend the interest to some subset of its neighbors. To its neighbors it appears to originate from the sending node, although it might come from a distant sink. This is an example of a local interaction. This is the manner in which the interest is propagated to rest of the network. A node may suppress a received interest if it recently resent a matching interest (after checking from cache). In case of having no information regarding the nodes which will be able to satisfy the interest match the message is rebroadcast, which is equivalent to flooding. Additionally we can use geographical routing or using the cache entry for finding the appropriate neighbor to whom we could send the message.

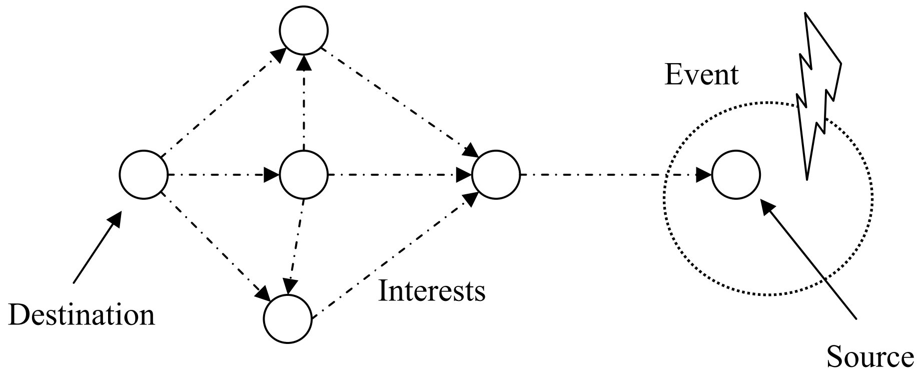

Figure 2: Interest propagation in a directed diffusion network

- *Gradient establishment:* Once a node receives an interest from its neighbor it sets a gradient towards it. Initially every pair of neighboring nodes establishes a gradient towards each other. This is because at a given time a node cannot disambiguate between two copies of the same interest received from two neighbors. This helps in fast recovery from failed paths and reinforcement of empirically better paths. The gradient specifies both the direction and value of data flow (rate of receiving and sending data for this particular neighboring node) in the network.

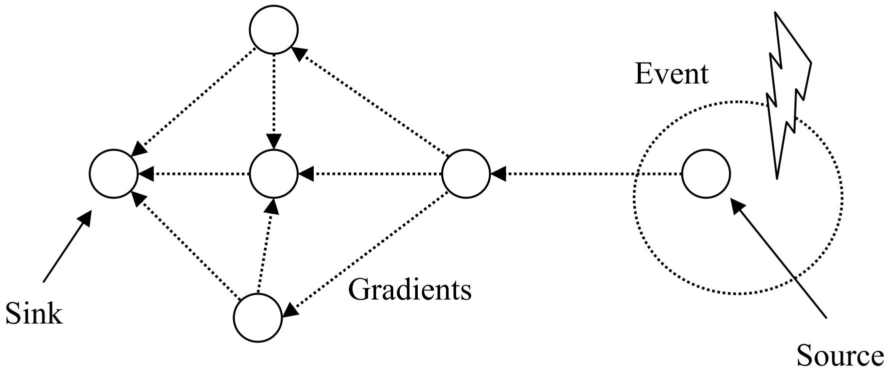

Figure 3: Gradient establishment in a directed diffusion network

- *Data propagation:* A sensor node matching the interest values processes the data and also starts reading the sensor values. These values after processing are sent on the network as an event. The event broadcast/unicast rate is the highest amongst the matching interests. The data message received by a node from neighbor is matched against the interest cache entry. If no match exists it is dropped else it is matched against a data cache entry. In this case we may drop the data message if it already exists and the message input rate is more then that mentioned in the interest value. This also helps in setting the data rate using selective reinforcement of paths.
- *Reinforcement for path establishment:* The sink initially sends low-rate event notification. These are called *exploratory events* and the subsequent gradient setup *exploratory gradient*. Once a source detects a matching target, it sends exploratory events, possibly along multiple paths, towards the sink. After the sink starts receiving these exploratory events, it reinforces one particular neighbor in order to draw down real data (events at higher data rate). The gradient setup for receiving high-quality data is called *data gradient*.
    - Path establishment using positive reinforcements: This is achieved by data driven local rules. The sink reinforces the link from the neighbor which sends a novel event, by increasing the data rate on that link. The neighboring node after receiving the interest requesting for increase in data rate chooses the neighbor from which it received this event and accordingly sends it an interest message asking to increase the data sending rate. Thus this message travels from source to sink along the path with minimum delay and increases the data transmission rates on this path. This method is very reactive to choosing the minimum delay path and hence we need more sophisticated rules for changing the path to consider other parameters as well.

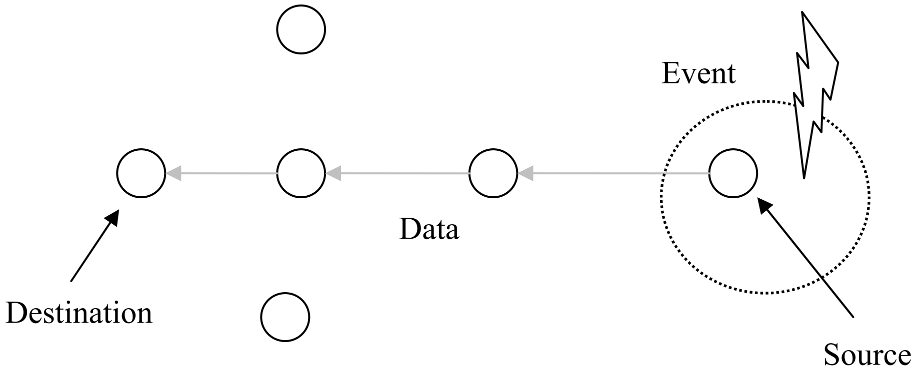

Figure 4: Data propagation in a directed diffusion network

- Handling multiple sinks and multiple sources: The algorithm is robust enough to handle multiple sources and multiple sinks generally. Typical cases can be handled with a few modifications.
- Local repair for a failed path: In case a node finds failure of source node or degradation in the rate of transmission from source node (here for each node the neighboring node transmitting the message appears as source) it can select a different source based on it receiving novel messages from other nodes or sending exploratory messages. Subsequently the node negatively reinforces the degraded source and chooses an alternative path. More complex rules can be used to ensure that the number of nodes affected by negative reinforcement is limited so that we do not waste resources rebuilding an optimal path.

- *Path truncation and loop removal:* In certain cases we may need to truncate some paths. This is done using negative reinforcement.
  - Truncation: In certain instances we may need to change the link from one neighboring node to another (e.g. in case of path degradation). In this case we use negative reinforcement to reduce the data transmission rate. One method is to use soft states where a path requires constant periodical reinforcements to keep a transmission rate. After a time period if a node does not receives reinforcement it reduces data rate to exploratory level. In the second method the negative reinforcement is the interest with lower data rate. If a node receives this it reduces its gradient towards the sink and if all its gradients are of exploratory type it negatively reinforces those neighbors sending it data.
  - Loop removal: primarily looping is prevented using the data cache. Negative reinforcements can also be used for removing local loops using the fact that loops do not provide the data first (i.e. their data is not novel). In certain cases we need to go for conservative rules to prevent truncation of paths which may occasionally contain duplicate messages.

The authors compare directed diffusion with omniscient multicasting on various parameters both analytically and in simulation. Result shows directed diffusion performing better then omniscient multicasting and order of magnitudes better then flooding.

Energy Aware Routing [2][6]

There are a number of ways in which the optimal paths can be selected from the multiple paths obtained during interest propagation in directed diffusion. Shah et. al. [] have proposed a mechanism where each node maintains a set of optimal and good paths leading from the source to the sink. This is done during interest propagation and route maintenance where each node incrementally calculates the cost of transmission to the sink. At the time of data propagation each node on receiving the data packet randomly chooses one of the neighbors leading to the sink based on probability which is inversely proportional to the cost of reaching sink via the node. This mechanism need not always choose the optimal path but it distributes the energy usage throughout the network. This is done in order to prevent fast drainage of nodes lying on the optimal path and thus prolonging network service lifetime.

In the proposed protocol the nodes are addressed by class based addressing where the interest packet specifies three things:

- Geographical location of the node from which the data is to be retrieved.
- Node type, which is the type of the node such as sensor, actuator, controller etc.
- Node subtype, where further classification of the node is done such as temperature sensor or chemical sensor, etc.

There are three phases in the protocol namely setup, data propagation and route maintenance. During the setup phase the sink or the destination node initiates the connection by flooding the network with interest packets in direction of the source. The intermediate nodes forward the request to only those nodes which are closer to the source. On receiving the request the node calculates the energy metric for the neighbor who sent the request is calculated and added to the cost of the path. Thus if N_i sends a request to N_j then the cost of the path is calculated as $C_{N_j,N_i} = Cost(N_i) + Metric(N_j, N_i)$ here $Cost(N_i)$ is the cost of the path for N_i.

Metric(N_j,N_i) is calculated as $C_{ij} = e_{ij}^{\alpha} R_i^{\beta}$,where e_{ij} is the cost of transmission and reception on the link N_i to N_j and R_i is the remaining energy at node i. α and β are the parameters of the experiment and are obtained from simulations or empirically. Paths having very high costs are dropped, thus only neighbors having low cost are added to the forwarding table FT_j of N_j i.e. $FT_j = \{i \mid C_{N_j,N_i} \le \alpha(\min_k C_{N_j,N_k})\}$. Node N_j assigns a probability with each of nodes which is used while forwarding packets. The probability value is inversely proportional to the cost of the node calculated above. Thus $P_{N_j,N_i} = \frac{1/C_{N_j,N_i}}{\sum_{k \in FT_j} 1/C_{N_j,N_k}}$ where N_k are all neighboring nodes in the forwarding table of node N_j. Thus using this probability we can estimate the cost of routing through this node i.e. $Cost(N_j) = \sum_{i \in FT_j} P_{N_j,N_i} * C_{N_j,N_i}$. During the data propagation process a node after receiving a data packet looks into its cache for possible paths and chooses one randomly with probability P_{N_j,N_i}. Thus the data packet need not go through the optimal path but has a choice of paths which are good enough to have low and comparable costs to give sub-optimal paths. There is occasional flooding by the destination to maintain routes.

In comparison to directed diffusion the authors claim that this method gives an advantage over others in terms of prolonged network life (by 44%) and reduced energy consumption of network as whole (by 22%).

In a slight modification to this method [] authors claim to improve upon the results. Their major claim is that energy aware routing method does not adapt towards energy reserves remaining in the network. They provide five parameters based on which a sensor network protocol can be compared. These parameters are:

- Average remaining energy
- Standard deviation of remaining energy
- Minimum remaining energy
- Average energy used by a data packet
- Average packet delay.

SPIN (Sensor Protocol for Information via Negotiation) [1]

It is a negotiation based data dissemination protocol where unlike the other cases we want to send data to all nodes in the network and not just one sink node. Classical flooding of networks has the following drawbacks:

- *Implosion*: Since a node always sends a data packet to all its neighbor without considering whether it has already sent the packet earlier. The network wastes resources by transmitting multiple packets of same data item. This is because

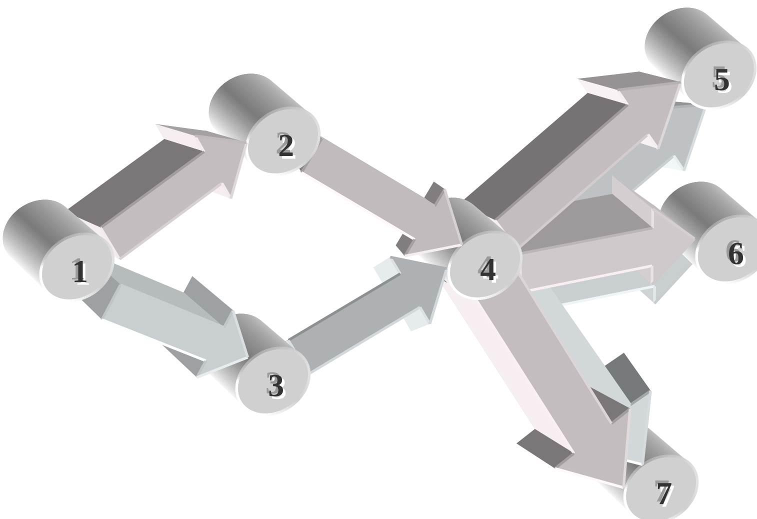

Figure 5: Implosion at node 4

  we lack mechanisms to uniquely identify a data item.

- *Overlap*: Sensor networks may have geographically overlapping regions where more then one sensors monitor events. Thus we may have situations when a common node receives multiple copies of a piece of data.

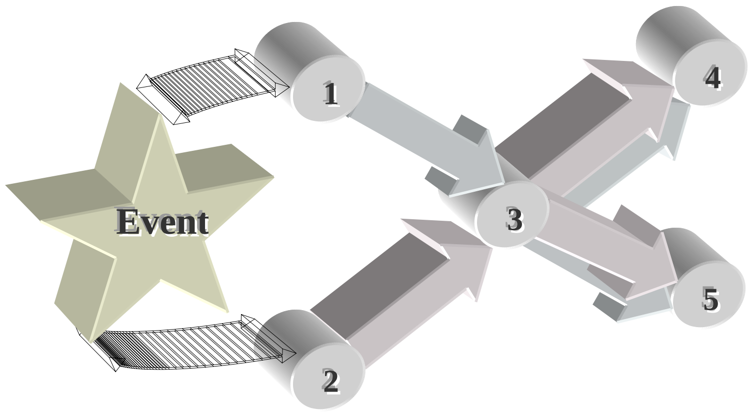

Figure 6: Overlap at nodes 1 and 2

- *Resource Blindness*: Nodes are not resource aware and do not adapt their behavior with change in energy level.

SPIN overcomes these limitations by implementing negotiation based data transfer and resource-adaptation. Nodes negotiate before transmission to transfer only unique data. This overcomes implosion. Also to do negotiations data descriptors called metadata descriptors are used. These metadata descriptors uniquely identify a data item and thus can be used to control overlap as well. Inclusion of a resource manager which is polled before transmitting introduces resource awareness in the nodes.

SPIN uses three types of messages for negotiations. When a node receives a new data item it creates an ADV packet and sends it to all its neighbors. The ADV packet contains only the metadata descriptor of the data. On receiving the ADV packet a node checks its cache. If the metadata descriptor is present in the cache, the node does not reply to ADV. Otherwise it sends a REPLY message, containing the metadata descriptor it received in ADV message of the data item it wants. On receiving the REPLY message the source node dispatches a DATA message which has the data as the payload and a metadata descriptor header which is used to construct the ADV packets at the receiving node. It also has a resource manager which can poll all the resources of the node. The resource manager stops the node from active participation if it does not have sufficient energy to complete the full process. Thus if it does not have sufficient energy to receive DATA the node does not replies to ADV.

Authors have compared the protocol with three standard protocols namely

- *Classical flooding*: On receiving data the node forwards it to all its neighboring nodes without looking whether it has already passed a copy or not.
- *Gossiping*: Instead of indiscriminately forwarding as in classical flooding here we choose one of the neighbors randomly and forward the data to it. Although data dissemination is slow, the energy drain is slow.
- *Ideal case*: Here we send data to all nodes on the shortest path from the source. For this purpose we can use IP level multicasting, etc.

The results show that SPIN gives much better performance then classical flooding and gossiping. Also the performance is close to the ideal case. Also energy dissipation is smaller in comparison to classical flooding and gossiping. For a simulation with fixed amount of energy the SPIN protocol was able to disseminate 73%, when ideal method did 85%, flooding did 53% and gossiping dissipated only 38% of data.

LEACH (Low-Energy Adaptive Clustering Hierarchy) [5]

LEACH is a clustering based protocol in which the nodes randomly elect cluster heads amongst them and form clusters around them. Cluster heads collect data from other nodes in the cluster and after online processing and data aggregation sends it to base station. The three key features of LEACH are:

- Localized co-ordination and control for cluster setup and control.
- Randomized cluster head rotation.
- Local compression to reduce global data communication.

It is possible at times that direct communication of nodes with the base station can be more efficient than multihop communication. Consider the scenario when there are *n* nodes in the network and the left most node wants to send data to base station (Figure 7).

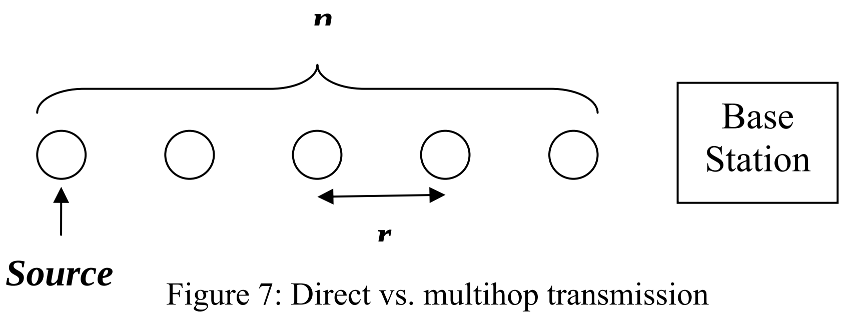

Figure 7: Direct vs. multihop transmission

There are *n-1* intermediate nodes so there will be *n* transmissions and *n* receives in the multihop case. Thus we have:

- Energy used for direct transmission from source to base station

$$E_{direct} = E_{elec} * k + E_{amp} * k * (nr)^2$$

- Energy usage for multihop transmission

$$E_{m-hop} = n * (E_{elec} * k + E_{amp} * k * (r)^2) + n * E_{elec} * k \,,$$

i.e. $E_{m-hop} = n * (2E_{elec} * k + E_{amp} * k * (r)^2)$.

Thus solving the inequality $E_{direct} < E_{m-hop}$, we have

$$\frac{E_{elec}}{E_{amp}} > \frac{r^2 n}{2}.$$

This employs that whether to go for multihop or direct communication depends on both topology and radio parameters of the sensor network. It can be easily seen that when the distance from base station is small and the transmission and receiving energies are of the same order then direct transmission is more efficient then multihop transmission. Based on simulation result it is observed that nodes closer to the base station die early in the multihop transmission, while nodes at the periphery die earlier in case of direct transmission. This asymmetrical node distribution is not preferable.

Clustering reduces energy usage within the cluster but drains the energy reserve for the cluster head. In case of networks where there are fixed cluster heads, the cluster heads need to be more powerful then other common nodes as it has to do maximum long distance communication.

The LEACH protocol overcomes these problems by adaptively choosing the cluster head in a randomized fashion where the cluster head is rotated amongst a set of nodes based on its energy reserve and its history as cluster head. This dynamic and random process reduces the chance of a single node acting as the cluster head as well as distributes the energy uniformly. LEACH's operation is divided into rounds. Each round can be further divided into the following steps:

- *Advertisement phase*: Where each node chooses to become a cluster head with a certain probability. Each node that elects itself as cluster head advertises and all the other nodes choose one of the clusters depending on signal quality.
- *Cluster setup phase*: In this phase each node informs the one cluster head about its decision to join. A node can receive multiple advertisement messages but it chooses one depending on criterions such as proximity, signal to noise ratio, etc.
- *Schedule creation*: The cluster head gives a TDMA schedule for transmission to each of its member for data transmission. The nodes can thus sleep/close radio when they are not transmitting.
- *Data transmission*: Each node transmits data in its allotted schedule and once the cluster head receives all the data it performs aggregation and compression of data before transmitting it to the sink. Some online processing can also be done such as summery creation etc. but depends on the tradeoff for computation vs. communication. To minimize overheads, time for d >> a+b+c.

At the end of the round this process starts once again.

When the clusters are being created each node decides whether to become a cluster head based on the number of suggested cluster heads and the number of times it has become cluster head in the past. Each node chooses a random number and if the number is less then the threshold it becomes the cluster head. The threshold value T_{thresh} is given by:

$$T_{thresh} = \begin{cases} \dfrac{P}{1 - P * (r * \bmod \frac{1}{P})} & if \quad n \in G \\ 0 & otherwise \end{cases}$$

Where P = suggested number of cluster heads, current round and G is the set of nodes which were not cluster heads for 1/P rounds.

Collision between nodes in a cluster is avoided using TDMA. Contention between cluster heads for communication with base station is avoided using CDMA coding scheme. The structure of the clustering mechanism has provisions for having a hierarchical clustering where there are different levels of clustering, i.e. cluster heads themselves forming further clusters. Simulation based comparison with direct transmission, multihop communication and fixed cluster head protocols show that LEACH is able to prolong network life considerably. Also the nodes die in a much more random and distributed fashion than other three.

IV Conclusion

Wireless sensor networks are reshaping how we collect data. Their inherent benefits let them score over other methods. With mass production and reduced manufacturing costs they will get adopted for newer and more innovative applications. The crucial point in the construction of wireless sensor protocols and hardware is to keep in mind the resource crunch faced by these devices. Energy conservation and parsimonious energy expenditure must be built in every aspect of design. Data and packet routing is an important field in this respect as it is related to radio communication which hogs up major energy usage in these nodes. In this paper we discussed a number of protocols to deal with routing in sensor networks. Directed diffusion is one of the pioneering works in this field having a number of derivatives currently used in implementations. LEACH uses node clustering to reduce amount of global traffic and implement area level aggregation and compression. Rotating and randomizing the cluster heads distributes the energy level gradually over the network and nodes die in much more random fashion giving longer network life. SPIN introduces a novel negotiation based data dissemination protocol which employs metadata to uniquely identify data items to prevent sending multiple copies of the same data. It also introduces energy awareness into the system which help in prolonging life of the network. Energy aware routing introduces the concept of using sub-optimal paths at times to reduce and distribute energy consumption in routing thus increasing lifetime of the network.

V Reference

1. *Adaptive Protocol for Information Dissemination in Wireless Sensor Networks*, Heinzelman, W. R., Kulic, J., Balakrishnan, H., Fifth ACM/IEEE MOBICOM Conference, Seattle, WA, 1999.
2. *Agent-Based, Energy Efficient Routing in Sensor Networks*, Gan, L., Liu, J., Jin, X., pages 472-479, Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 1, 2004.
3. *A Survey on Sensor Networks*, Rentala, P., Musunnuri, R., Gandham, S., Saxena, U. University of Texas at Dallas.
4. *Directed Diffusion for Wireless Sensor Networking*, Intanagonwiwat, C., Ramesh Govindan, R., Estrin, D., Heidemann, J., and Silva, F., Pages 2-16, IEEE/ACM Transactions on networking, vol. 11, no. 1, February 2003.
5. *Energy-Efficient Communication Protocol for Wireless Microsensor Networks*, Heinzelman, W.R., Chandrakasan, A., and Balakrishnan, H. Proceedings of the Hawaii International Conference on System Sciences, January 4-7, 2000, Maui, Hawaii. (IEEE).
6. *Energy aware routing for low energy ad hoc sensor networks*, Shah, R. C., and Rabaey, J. M.,. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '02), March 2002.

7. *PEAS: A robust Energy conserving Protocol for Long-lived Sensor Networks*, Ye, F., Zhong, G., Lu, S., and Zhang, L. 10th IEEE International Conference on Network Protocols (ICNP02). Paris, France, November 12-15, 2002.
8. *Sensor Networks: An Overview*, Bharathidasan, A., and Ponduru, V. A. S. UC Davis.

VI Terms

- *Wireless Sensor Networks:* Collection of low cost, small range self configuring sensors which are deployed in an ad hoc manner to monitor events.
- *Proactive Routing:* Class of routing algorithms which maintain routing tables and path information beforehand.
- *Reactive Routing:* Algorithms where the routing information is generated and maintained only when required.
- *Implosion:* Multiple packet transmission of same data item due to lack of ability to identify/differentiate data packets uniquely during flooding of network.
- *Gossiping:* Random selection of a neighbor and sending of data packet to it during data dissemination.
- *Trilateration/multilateration:* Localization techniques used for locating events or finding co-ordinates of nodes in a network.
- *Coarse grained localization:* Localization technique where the nodes obtain their locations by listening to packets from beacons and obtaining co-ordinates based on proximity to beacon nodes.
- *Fined grained localization:* Localization done using timing, signal strength, directionality and signal pattern matching techniques.
- *Interest:* Data packets carrying the request from the destination flooded through the network to identify nodes carrying the relevant information.
- *Gradients:* Path set up by the interest packets containing link and data rate information.
- *Reinforcement:* Selective increment (decrement) of data rate along paths for positive (negative) reinforcement thus leading to single optimal paths rather then multiple paths.
- *Metadata descriptors:* Data descriptors uniquely identifying a data item in the network to minimize transmission of multiple copies of data packet during controlled flooding.