Wireless Sensor Networks: Issues, Challenges and Survey of Solutions

B.Sangeetha[1]
[1]Student
[1]Department of Computer Science and Engineering
[1]Saveetha School of Engineering

Abstract— Wireless ad-hoc sensor networks have recently emerged as a premier research topic. They have great long-term economic potential, ability to transform our lives and pose many new system-building challenges. Sensor networks pose a number of new conceptual and optimization problems such as location, deployment and tracking in that many applications rely on them for needed information.
The past works are scattered across all of the systems layers: from physical layer to data link layer to network and application layer. In this report, we present an overview of wireless sensor networks and issues involved in employing them. We make an attempt to provide a snapshot of solutions proposed in recently published literature for different issues like Medium Access Control, Data Dissemination, Security and Coverage determination.
Keywords: Sensor Networks, Wireless Sensor Networks, Distributed Sensor Networks, WINS, sensor nets.

I. INTRODUCTION

In the foreseeable future sensor networks have wide applicability from Observing scientific phenomenon to use in agricultural monitors and warehouse inventory management. In order to understand this scientific phenomenon, it is necessary for researchers to collect numerous measurements of a scientific event in a geographic region. While these measurements can be obtained at a distance (remote sensing), there is often no substitute for observations made firsthand within the region of interest (in-situ). One form of technology that can accomplish such in-situ science is the Wireless Sensor Network (WSN). In WSNs a number of probe devices are distributed throughout a geographic region to observe local scientific conditions. In addition to sensors, probes are equipped with computational resources for in-network data processing, as well as wireless transceivers for communication with neighboring probes. Recent advances in integrated circuitry, micro electromechanical systems (MEMS), communication and low-cost, low-power design have fomented the emergence of these wireless sensors.

Most current deployed sensor networks involve relatively small numbers of sensors, wired to a central processing unit where all of the signal processing is performed. In contrast, this survey focuses on distributed, wireless, sensor networks in which the signal processing is distributed along with the sensing.

A. Why distributed sensing?

When the precise location of a signal of interest is unknown in a monitored region, distributed sensing allows one to place the sensors closer to the phenomena being monitored than if only a single sensor was used. Line of sight, and more generally obstructions, cannot be addressed by deploying one sensor regardless of its sensitivity. Thus, distributed sensing provides robustness to environmental obstacles.

B. Why wireless?

When wired networking of distributed sensors can be easily achieved, it is often the more advantageous approach. Moreover, when nodes can be wired to renewable (relatively infinite) energy sources, this too greatly simplifies the system design and operation. However, in many envisioned applications, the environment being monitored does not have installed infrastructure for either communications or energy, and therefore untethered nodes must rely on local, finite, and relatively small energy sources, as well as wireless communication channels.

C. Why distributed processing?

Finally, although sensors are distributed to be close to the phenomena, one might still consider an architecture in which sensor outputs could be communicated back to a central processing unit. However, in the context of untethered nodes, the finite energy budget is a primary design constraint. Communications is a key energy consumer as the radio signal power in sensor networks drops off as r^4 due to ground reflections from short antenna heights. Therefore, one wants to process data as much as possible inside the network to reduce the number of bits transmitted, particularly over longer distances.

II. HOW ARE SENSOR NETWORKS DIFFERENT FROM OTHER KINDS OF NETWORKS?

Sensor Network is similar to a general purpose Mobile Ad-Hoc network (MANET) in many aspects, they are distributed, self-organized, multi-hopped and lack a fixed infrastructure.

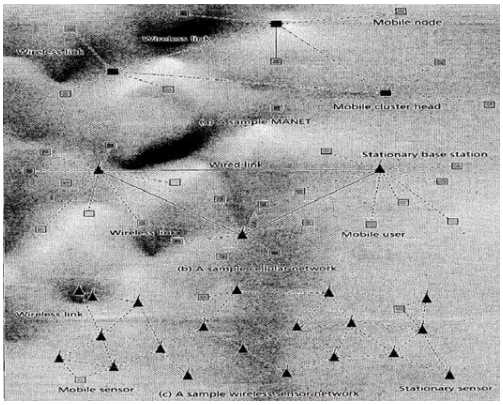

Fig. 1: sensor network

The main difference lies in the fact that the former basically has lower cost, lesser bandwidth, smaller processing power, higher redundancy and are more power-constrained. While MANET is a general structure with mobility as its main feature, the sensors have no or low mobility. Another special MANET is the Bluetooth technology [20] with cable replacement as its goal, also shares some features with sensornets. However, here the power constraint is not so strict, the processing power is much higher and the target applications are quite different. The main aim of any sensor

All rights reserved by www.ijsrd.com

net is to spatially densely and temporally continuously monitor and gather data, thus often forming many-to-one correlated traffic pattern from the sensors to the collection station. But the aim of Bluetooth, similar to General MANET, is to provide one-to-one independent connection.

III. MOTIVATING APPLICATIONS

A. Historical Example: Oceanography

Buoy networks are being used in oceanic environment evaluation to collect and monitor physical parameters, like sea temperature, wave direction, and current speed and so on. Many real-time, climactically buoy networks have been implemented around the world to successfully to capture such information.

B. Modern Applications

- In all fires, early warnings are critical in trying to prevent small harmless brush fires from becoming monstrous infernos. By deploying specialized wireless sensor nodes in strategically selected high-risk areas the detection time can be drastically reduced, increasing the likelihood of success in extinguishing efforts.
- Wireless sensor networks provide a viable alternative to several existing applications. Large buildings contain hundreds of environmental sensors that are wired to a central air conditioning and ventilation system. The significant wiring costs limit the complexity of current environmental controls and reconfigurability of these systems. Replacing the hard wired monitoring units with wireless sensor nodes can improve the quality and energy efficiency of the environmental system while allowing almost unlimited reconfiguration and customization .
- One field where these sensor-nets will be used is large scale environmental monitoring (air, water, soil chemistry). The goal is to enable scattering of hundreds of thousands of these nodes in areas that are difficult to access for study using conventional methods. The network could then monitor events, perform local computations on the data, and either, relay aggregated data, or configure local and global actuators.
- Biomedical sensor applications like Artificial Retina, Glucose Level Monitors, Cancer Detectors, General Health Monitors .
- Smart Kindergarten: The envisioned system would enhance the education process by providing a childhood learning environment that is individualized to each child, adapts to the context, coordinates activities of multiple children, and allows continuous unobtrusive evaluation of the learning process by the teacher .
- Habitat Monitoring: Long-term data-collection for systematic and ecological field studies .
- Commercial Applications: Agriculture Monitors, Warehouse Inventory, Product Maintenance, Smart spaces, Factory Instrumentation.

C. NASA Applications:

NASA uses sensor networks primarily in In-situ data collection, Precision landing guidance, Vehicle health sensors, Trail markers and exploration of distant regions such as surface of Mars.

IV. ENGINEERING CHALLENGES

Most envisioned sensor network applications encounter the following challenges:

- Untethered for energy and communication requiring maximal focus on energy efficiency.
- Ad hoc deployment, requiring that the system identifies and copes with the resulting distribution and connectivity of nodes.
- Dynamic environmental conditions requiring the system to adapt over time to changing connectivity and system stimuli.
- Unattended operation requiring configuration and reconfiguration be automatic (self-configuration).

To address these technical challenges several strategies are going to be key building blocks/techniques for sensor networks:

Collaborative signal processing among nodes that have experienced a common stimulus will greatly enhance the efficiency (information per bit transmitted)

Exploiting redundancy has application when the cost of deploying the initial set of sensors is much less compared to the cost of replacing defective or failed nodes or renewing node resources. Thus redundancy can be exploited to extend system lifetime. Another application is when sensors cannot be positioned carefully; redundancy can be exploited to extend coverage by using a subset of the nodes, which are positioned favorably.

Adaptive fidelity signal processing can be exploited to strike a balance between energy, accuracy and rapidity of results. The timeliness and accuracy of the signal processing can be adapted keeping in mind the energy resources and latency requirements.

Hierarchical, tired architecture can greatly contribute to overall system lifetime and capability. Whenever possible, higher capacity system elements can be used to offload drain on small factor elements, while the latter can be exploited to obtain the desired physical proximity to stimuli. Moreover, even among elements with homogeneous capabilities, creating clusters and assigning special combining functions to cluster heads can contribute to overall system scalability and lifetime. However, to ensure robustness, such clustering/hierarchy must be self-configuring and reconfiguring in the face of environmental or network changes.

V. SENSOR NODE ARCHITECTURE

In sensor networks, the architecture of a node is highly dependent on the purpose of the deployment. But a generalized architecture can be shown as in Figure 1 . Each node consists of a sensor, processor, and radio for communication, battery, and memory. According to their operational, we can divide its functionality into two broad categories: Heavy-duty part, includes sensor and data converter and signal processing, has to operate at low power level, using in real-time system. The low-duty part with extra energy performs further processing and communication.

Figure 2 shows the general layered architecture of a sensor node. In this proposed architecture the network functionality is divided between main CPU and radio board. The main idea behind this architecture is to decrease the

All rights reserved by *www.ijsrd.com*

functionality on the sensor CPU by transferring some of the functionalities to the radio board. The radio boards process the information in the form of Micro Controller Units (MCU), which are used for the physical and MAC layer implementation. Thus part of the network functionality is transferred to the radio board as shown in Figure 3.

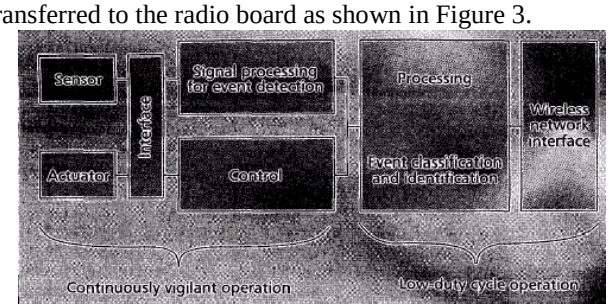

Fig. 2: The Architecture of a Sensor Node

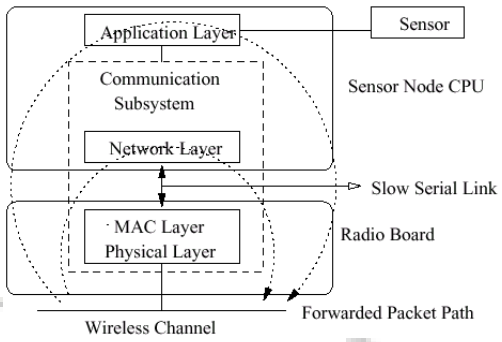

Fig. 3: Layered Architecture 1

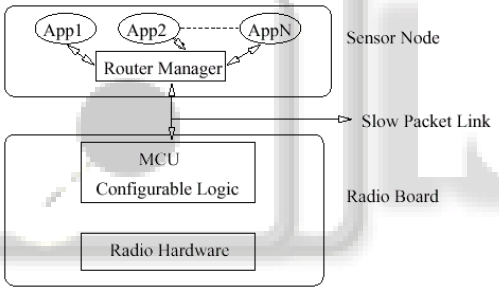

Fig. 4: Layered Architecture 2

Several institutions have begun large-scale projects to develop system and protocol architectures for wireless sensor networks. The projects include:

- AWIARS: Adaptive Wireless Arrays for Interactive, Reconnaissance, Surveillance, and Target Acquisition in Small Unit Operation (UCLA/Rockwell Science Center)
- WINS: Wireless Integrated Network Sensors (UCLA/ Rockwell Science Center)
- SCADDS: Scalable Coordination Architectures for Deeply Distributed Systems (USC/ISI) .
- Smart Dust: Autonomous Sensing and Communication in a Cubic Millimeter (U.C.Berkeley) (see Figure 4 for the Berkeley Mote).
- µ-AMPS: Micro-Adaptive Multi-domain Power-aware Sensors (MIT)

A. TinyOS

J.Hill. et.al propose a event driven operating system to reduce the burden of application development by providing convenient abstractions of physical devices and highly tuned implementations of common functions while providing efficient modularity and robustness. The TinyOS is designed to fill the role of the software platform to support and connect the tiny wireless sensor nodes where the current real time embedded operating systems are unsuitable. It fits in 178 bytes of memory and supports the concurrency intensive operations required by networked sensors (hence the choice of event based model) with minimal hardware requirements. TinyOS supports an application level-messaging model, a variant of Active Messages (AM) called Tiny Active Messages. This concept strives for overlapping of communication and computation and is well suited for the event based execution model of TinyOS. The low overhead associated with event-based notification is complementary to the limited resources of the networked sensors.

VI. LINK LAYER ISSUES

The two major services the link layer provides to higher layers are formation of link-layer topology (or infrastructure) and regulation of channel access among the nodes. Like in all shared-medium networks, medium access control (MAC) is important for successful operation of network. Current MAC design for wireless networks can be broadly divided into two categories: contention based and explicit organization in time/frequency/code domains. The various flavors of MACA, MACAW, are examples of the former. These contention based schemes are clearly not suitable for sensor networks due to their requirement for radio transceivers to monitor the channel at all times. This is particularly expensive operation for the low radio ranges of interest for sensor networks, where transmission and reception have almost have the same energy cost. One would like to turn off the radios when no information is to be sent or received. The other class of MAC protocols is based on reservation and scheduling. The task of assignment of channels (i.e., TDMA slots, FDMA frequency bands or CDMA spread spectrum codes) to links between radio neighbors avoiding the collisions is a hard problem and needs a hierarchical structure to make the channel assignment task more manageable. The problem in this approach is how to determine cluster memberships and cluster heads such that the entire network is covered. Moreover, when the number of nodes within a cluster changes, it is not easy for a TDMA protocol to dynamically change its frame length and time slot assignment. So its scalability is not as good as that of a contention-based protocol. For example, Bluetooth may have at most 8 nodes in a cluster.

Relevant issues in designing a good MAC protocol for the wireless sensor networks:

A. Energy-Efficiency:

This is the foremost important factor for any issue in the sensor nets.

B. Scalability:

A good MAC protocol easily accommodates changes in network size, density and topology. Some nodes may die over time and new nodes may join later; some nodes may move to different locations.

All rights reserved by *www.ijsrd.com*

C. Fairness:

In traditional wireless voice or data networks, each user desires equal opportunity and time to access the medium i.e., sending and receiving packets for their own applications. Per-hop MAC level fairness is this an important issue. However, in sensor networks, all nodes cooperate for a single task and normally there is only one application running at any time. In this case fairness is not important as long as application-level performance is not degraded.

D. Latency**:**

Latency can be important or unimportant depending on what application is running and the node state. During a period when there is no sensing event, there is normally very little data flowing in the network and most of the time nodes are in idle state. Sub-second latency is not important, and we can trade it off for energy savings by letting the node turnoff their radios to reduce the energy consumption due to idle listening.

Thus energy conservation and self-configuration are primary goals for sensor networks, while other attributes like fairness, latency, and throughput and bandwidth utilization are only of secondary concern

While the sensors are mostly stationary, mobile nodes are usually introduced in a sensor network to serve as gateway to the outside world. Thus MAC solutions can be classified into kinds. One category caters to the communication between the mobile nodes and the stationary sensors and the other category caters to communication between stationary nodes.

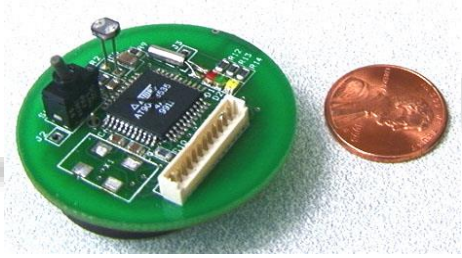

Fig. 5: Circuit Diagram

1) SMACS

Sohrabi and Pottie proposed this Self-organizing Medium Access Control for Sensor networks (SMACS). This distributed protocol enables nodes to discover their neighbors and build a network for communication without any master nodes. It builds a flat topology i.e., there are no clusters or cluster heads. Each node maintains a TDMA-like frame, called *super frame*, in which the node schedules different time slots to communicate with its known neighbors. The structure of this frame can change from time to time. The TDMA schedule consists of two separate regions. The first region is called the bootup period, when nodes randomly search on a fixed frequency band for new nodes to include in the network or to rebuild severed links. The other region is reserved for communication tasks with neighboring nodes. At each time slot a node talks to only one neighbor. To avoid interference between adjacent links, the protocol assigns different channels, i.e., FDMA or CDMA, to potentially interfering links. Although the super frame structure is similar to a TDMA frame, it doesn't prevent two interfering nodes from accessing the medium at the same time. The actual multiple accesses is achieved by FDMA or CDMA.

One interesting feature of Piconet Radio Protocol is that it also puts nodes into periodic sleep for energy conservation. The scheme that piconet uses to synchronize neighboring nodes is to let a node broadcast its node is before its starts listening. If a node wants to talk to a neighboring node, it must wait until it receives the neighbor's broadcast.

Woo and Culler [47] propose a contention based medium access control scheme with the goal to be energy efficient and fair bandwidth allocation to the infrastructure for all nodes in the multihop network. They conclude that limiting the length of listening, the introduction of random delay in addition to backoff, and phase shift at the application level are necessary for the traditional CSMA mechanism. They claim that the proposed *adaptive rate* control mechanism is effective in achieving their fairness goal while being energy efficient for both low and high duty cycle of network traffic. But in the context of sensor networks fairness is only of secondary importance and it can even be traded-off for further energy savings.

Zhong and Shah et.al. [50] propose a distributed access mechanism combining best of CSMA and spread spectrum techniques. It trades the bandwidth in broadband applications for higher power efficiency and throughput. This access protocol does not require a dedicated control channel, or synchronization, whether global or local. Additionally, it has very low delay and does not have the problem of coordinating broadcast and scheduled unicasts.

2) S-MAC (Sensor MAC)

The S-MAC [46] is designed with the primary goals of energy conservation, collision avoidance and self-configuration. It utilizes a combined scheduling and contention scheme. The Protocol tries to reduce energy consumption from all sources causing energy waste i.e., idle listening, collision, overhearing and control overhead. S-MAC uses three novel techniques to overcome these factors. To reduce energy consumption in listening to idle channel, nodes periodically sleep. Neighboring nodes form virtual clusters to auto-synchronize on sleep schedules. Similar to PAMAS, S-MAC also sets the radio to sleep during transmissions of other nodes. Unlike PAMAS, it only uses in-channel signaling. Finally, S-MAC applies message passing to reduce contention latency for sensor network applications that require store-and-forward processing as data moves through the network. The basic idea is to divide the long message into small fragments and transmit them in a burst. The result is that a node who has more data to send get more access time to the medium. This is unfair from a per-hop MAC level perspective. But this method can achieve energy savings by reducing control overhead and avoiding overhearing. The most important feature of wireless sensor networks, in-network data processing requires store-and-forward processing of mechanism. In this case, MAC protocols that promote fragment-level fairness actually increase message-level latency for the application. In contrast, message passing reduces message-level latency by trading off the fragment-level fairness.

3) Eavesdrop and Register (EAR)

This mobile MAC protocol is designed to provide the required connectivity to mobile nodes as they interact with

All rights reserved by *www.ijsrd.com*

the static sensor network, while adhering to the constraints for the entire network. Since it is desirable to setup connection with as few message exchanges as possible, the mobile node assumes full responsibility of connection setup. The mobile node keeps a registry of all the sensors in its neighborhood and makes handoff decisions whenever the SNR drops below a threshold value. The EAR algorithm is designed to be transparent to the protocol followed by the stationary nodes. The first slot following the bootup period is reserved for mobile nodes thus giving them higher priority. The EAR algorithm uses the invitation message broadcast during the boot up period as a trigger. The mobile node simply eavesdrops on to these messages and forms a registry of all stationary nodes within hearing range.

VII. ROUTING

Perhaps the most directly relevant to sensor networks is the ongoing work on ad-hoc wireless networks. A central focus of the work in the ad-hoc wireless networking has been the design of proactive routing protocols like Ad Hoc On Demand Distance Vector (AODV), Destination Sequenced Distance Vector (DSDV), Temporally Ordered Routing Algorithm (TORA) and reactive routing protocols, and combination thereof. Proactive routing protocols continuously compute nodes to all nodes so that a route is already available when a packet needs to be sent to a node. Such continuous route computation is highly energy inefficient. Reactive routing protocols on the other hand start route discovery process only when a packet needs to be sent. However, these protocols may redundantly flood requests throughout the network. A combination of proactive and reactive routing protocols may overcome the above-mentioned disadvantages, but still wont be as energy efficient as schemes that can exploit the application specific knowledge to route data and queries. Another aspect of sensor networks that makes these multi-hop ah-hoc protocols not appropriate is the low mobility or lack of it. In these ad-hoc protocols QoS is important and has to route large amounts of multi-media traffic in the presence of high mobility.

A. Sensor Protocols for Information via Negotiation (SPIN)

Kaulik et.al. proposed this family of adaptive protocols to efficiently disseminate information among sensors in an energy-constrained sensor network. SPIN uses metadata negotiation and resource -adaptation to overcome several deficiencies of traditional dissemination approaches. Using metadata names, nodes negotiate with each other about the data they possess. This negotiation ensures that nodes only transmit data when necessary and never waste energy on useless transmissions. Because the nodes are resource aware, they are able to cut back on their activities whenever their resources are low to increase their longevity. SPIN uses three kinds of messages for communication:
ADV - When a node has data to send it advertises using this message containing metadata.
REQ - A node sends this message when it wishes to receive some actual data.
DATA - Data message containing the data with a metadata header.
The four specific SPIN protocols are:

1) SPIN-PP:
This point-to-point communication protocol assumes that two nodes can communicate with each other without interfering with other nodes communication and also that packets are never lost. A node, which has information to send, advertises this by sending an ADV to neighboring nodes. The nodes that are interested in the data express their interest by sending a REQ. The originator of ADV then sends the data to the nodes that sent a REQ.
2) SPIN-EC:
This protocol just adds energy heuristic to the previous protocol. A node participates in the process only if it can complete all the stages in the protocol without going below a energy threshold.
3) SPIN-BC**:**
This broadcast channel based protocol differs from the previous protocols in that nodes do not immediately send out REQ messages on hearing an ADV. Instead each node waits a random amount of time before sending out the REQ message. The other nodes whose timers have not yet expired cancel their timers on hearing the REQ, thus preventing redundant copies of REQ being sent.
4) SPIN-RL:
This protocol was designed for lossy broadcast channels by incorporating two adjustments. First each node keeps track of the advertisements it receives and re-requests data if a response from the requested node is not received within a specified time interval. Second nodes limit the frequency with which they will send data. Every node waits for predetermined time before servicing requests for same piece of data.

B. Directed Diffusion

D.Estrin et.al. propose a communication model where each sensor node names data that it generates with one or more attributes similar to the metadata concept of the SPIN suite. A sink may query for information by disseminating an interest. Syntactically, an interest is simply a range of values for one or more attributes. An example for such a scenario can be a sensor network in which each node can detect motion (and possibly some other information) within some vicinity. One or more sink nodes may query the sensor network for motion information from a particular section of the terrain (e.g., from the southeast quadrant). Moreover, because of the relatively short life span of nodes, as well as the large amount of data and system redundancy possible, it may be more useful to address data instead of individual nodes . Each sensor may name the data it generates using a single attribute motion, which has a geographic location (e.g. latitude/longitude or relative location with respect to some landmark) as its value. Motion data may be described using several attributes: (type: seismic, id =12, location = 75N/120E) . Interests may be of the form (type = seismic, location = 70-80N/100-140E). Each node disseminates interests based on the contents of the interest. In the above motion detection example, nodes send the interest towards the neighbor in the direction of south-east quadrant. Conceptually, the path of interest propagation sets up a reverse data path for data that matches the interest (query). In the above diffusion model this data propagation is said to have an associated gradient. The notion of gradient is useful for robustness when each intermediate node propagates

All rights reserved by *www.ijsrd.com*

interest towards multiple neighbors. The authors call that the strength of the interest is different towards different neighbors, resulting in source-to-sink paths with different gradients. In its simplest form gradient can be a scalar. Negative gradients are also possible, which inhibit the distribution of data along a particular path and positive gradients encourage the propagation of data along the path. The value of the gradient can have application specific semantics. In the aforementioned motion detection example, if a node has two outgoing paths, one with a gradient of 0.7 and other with a gradient of 0.3, then the node may send twice as much detail along the higher gradient path than along the lower.

The diffusion model allows intermediate nodes to cache or locally transform (e.g. aggregate) data. Caching and aggregation can increase the efficiency, robustness and scalability of coordination. Locally cached data can be accessed by other sinks with much lower energy consumption. The diffusion model's data naming and local data transmission features capture the data-centricity and application specificity that is possible in sensor networks.

D.Braginsky and D.Estrin in further analyze the method of routing queries to nodes that have observed a particular event. This follows the philosophy of retrieval of data keyed on the event, not on the underlying network addressing scheme. They define an *event* as an abstraction, identifying anything from a set of sensor readings, to the node's processing capabilities. Similarly a *query* is defined as a request for information. If the amount of returning data from a query is significant makes sense in discovering short paths from the sources to the sink. Methods such as Directed Diffusion need an initial flood of query for exploration. GEAR (Geography and Energy Aware Routing) relies on localization information of the nodes and provides savings over a complete network flood by limiting the flooding to a geographical region.

Flooding need not be restricted to queries. For applications where there are few events and many queries, it makes sense to flood the event, and set up gradients towards the query. However, unless the number of queries per event and the amount of data conveyed by each event is quite high, the setup cost for event flooding cannot be effectively amortized. GRAdient broadcast (GRAB) contributes in this direction of event centric routing state in the network. GRAB describes a way of building a cost field toward a particular node, and then reliably routing queries across a limited size mesh toward that node. It comes with the overhead of a network flood to set up the cost field, but queries are routed along an interleaved set of short paths, and can thus be delivered cheaply and reliably.

C. Rumor Routing

The rumor routing is intended to strike a balance between query flooding and event flooding. It is only useful if the number of queries compared to the number of events is between the two intersection points. An application aware of this ratio can use a combination of rumor routing and flooding to best utilize available power (see Figure 4).

The idea here is to create paths leading to each event; whereas event flooding creates a network-wide gradient field. In this way when a query is generated it can be sent on a random walk until it finds the event path; instead of flooding it across the network. As soon as the query discovers the event path, it can be routed directly to the event (see Figure 5). If the path can't be found, the application can retry and can as a last resort flood the query. This makes sense, since two straight lines (although neither the path nor the query is entirely straight) in a plane are very likely to intersect. The algorithm employs a set of long-lived agents that create paths (in the form of state in nodes) directed towards the events they encounter. Whenever an agent crosses a path to an event it has not yet seen, it adapts its behavior and creates path state that leads to the event.

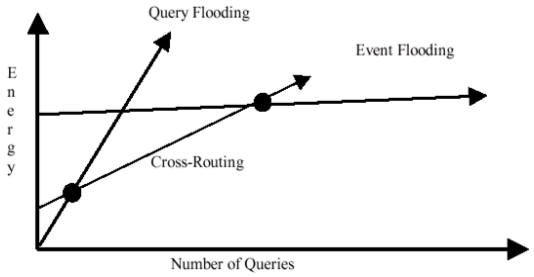

Fig. 6: query flooding and event flooding

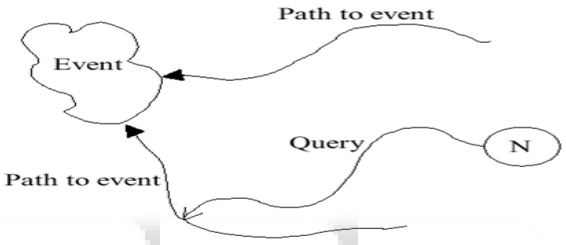

Fig. 7: Path to event

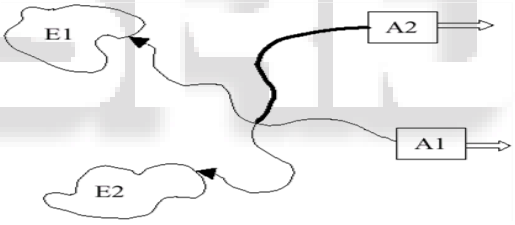

Fig. 8: Path with node

In Figure 6, agent A1 has been creating path state leading to event E1. Agent A2 has been creating path state leading to E2. When A2 crosses the path created by A1, it begins to create aggregate path state, leading to both E1 and E2. The agents can also optimize the paths in the network if they find shorter ones. When an agent finds a node whose route to an event is more costly than its own, it will update the node's routing table to the more efficient path (see Figure 7).

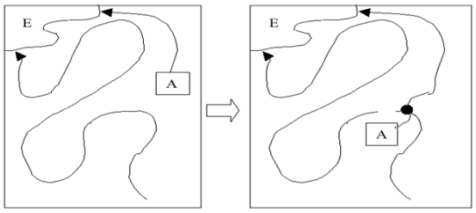

Fig. 9: update nodes

D. Adaptive Local Routing for Cooperative Signal Processing

It is clear that some layering of distributed signal processing and networking functions is necessary for energy efficiency. Since the communications dominate the energy cost when

All rights reserved by *www.ijsrd.com*

cooperative functions among nodes are needed, the question that arises is to what extent the signal processing hierarchy demands a corresponding networking hierarchy. K. Sohabri et.al. Presented algorithms for setting up sub-networks to perform cooperative signal processing the two types of cooperative signal processing are non-coherent and coherent. For non-coherent processing, raw data is - preprocessed at the node itself before forwarding it to a central node (CN) for further processing. For coherent processing, raw sensor data is forwarded after minimal processing to the central node. Because of fairly low data traffic load for non-coherent processing energy efficient algorithmic techniques assume importance. On the other hand, since coherent processing generates long data streams, energy efficiency must be achieved by path optimality.

1) Non-coherent Cooperative Function

Three phases are involved in this algorithm: Phase I involves target detection, data collection, and preprocessing. After preprocessing if a node finds that its information might be of interest, it declares its intention to participate the cooperative function (Phase II). In Phase III, a CN is elected for more sophisticated information processing by taking into account the energy reserves and computational capabilities of nodes. The CN election algorithm has two components: Single Winner Election (SWE) and Spanning Tree (ST) algorithm. The first component involves the necessary signaling in single candidate election. The second component computes a minimum-spanning tree rooted at the CN. An *Elect* message is broadcasted by every node willing to be a CN along with a set of parameters that serve as the election criteria. The nodes that receive these messages then compare the criteria with themselves and respond with a second set of messaged with the result and store the winner in their registry. The routing information is piggybacked along with the Elect message thus allowing the calculation of a minimum-spanning tree rooted at the winner simultaneously. Thus the winner's information diffuses through the network and the spanning tree gradually increases its coverage and finally covers the whole network.

2) Coherent Cooperative Function

Since the energy cost of uploading long data stream to the CN is high, a Multiple Winner Election (MWE), which is a simple extension of SWE, is used to limit the number of sensor source node (SNs) providing the data. Instead of keeping record of one best candidate, each node will now keep up to n of them. Just as in SWE, for each winning SN candidate, a minimum energy path can be computed. After computing the total energy consumption to upload data from each SN to each node, with the use of SWE process, the node with minimum energy consumption is used to act as the CN. In general this formation process has longer delay, higher overload, and lower scalability than the non-coherent processing.

VIII. LOCATION MECHANISMS

The problem of localization, that is, determining where a given sensor is physically located in a network, is a challenging one, and yet extremely crucial for many of the envisioned applications of sensor nets. For example, localization opens up new ways of reducing power consumption in multi hop wireless networks. GEAR (Geography and Energy Aware Routing) uses location information to achieve power savings in routing. In context aware applications, localization enables intelligent selection of appropriate devices, and also supports useful coordination among devices. The desired granularity of localization is application dependent

Global Positioning System (GPS) solves the problem of localization in outdoor environments for PC-class nodes. However for large networks of small, cheap, low-power devices like sensor networks, practical considerations such as size, form factor, cost and power constraints preclude the use of GPS on all nodes.

Some of the design goals of localization in wireless sensor networks are

- RF-based: Normally, the sensors have some kind of short-range radio transceivers for communication. By leveraging this radio for localization the high cost and size requirements of GPS can be avoided.
- Receiver-based: For greater scalability, the responsibility for localization must lie with the receiving node that needs to be localized and not with the reference points.
- Ad Hoc: For easy deployment, the solution should not require preplanning or extensive infrastructure.
- Low Energy: Since the sensors have modest processing capabilities, the mechanisms should minimize computation and message costs to reduce power consumption.
- Adaptive Fidelity: The accuracy of the localization algorithms should be adaptive to the granularity of available reference points.

Localization methods typically rely on some form of communication between reference points with known positions and the receiver node that needs to be located. Various localization techniques can be classified into two broad categories based on the granularity of information inferred during the communication. Fine-grained localization systems (e.g., GPS) provide high precision location information, typically estimated ranges or angles relative to beacons (reference points) and compute location of the unknown node using trilateration (position estimation from distance to three points) or triangulation (position estimation from angles to three points). Coarse-grained localization systems estimate unknown node location from proximity to beacons or landmarks

Doherty et.al proposed a coarse-grained localization system based on RF-connectivity induced constraints. Known peer-to-peer communication in the network is modeled as a set of geometric constraints on the node position. As a physical example, if a particular RF system can transmit 20m and two nodes are in communication, their separation must be less than 20m. These constraints restrict the feasible set of unknown node positions. Formally, the network is a graph with n nodes at the vertices (each node having a Cartesian position) and with bi-directional communication constraints as the edges. Positions of the first m nodes are known (x_1,y1...,x_m,y_m) and the remaining n-m positions are unknown. The feasibility problem is then to find (x_{m+1},y_{m+1}... x_n, y_n) such that the proximity constraints are satisfied. There will be constraints among open nodes though their positions are unknown.

All rights reserved by *www.ijsrd.com*

Connections that are not reported are not detrimental to the performance of this algorithm. To calculate the feasible solutions to the position estimation problem convex optimization is used. This methodology requires centralized computation i.e., all nodes must communicate their communicate their connectivity information to a single computer to solve the optimization problem. This solution doesn't scale well for networks of the order of 1000s of nodes since the problem becomes too computationally intensive to be handles at one place. And also the communication cost increases with the number of sensors. A decentralized approach, where large network is divided into sub networks and position estimation can be carried out for each member of the network based on unknown centroid of the local region. Following these local estimations, the sub network centroids can be abstracted to nodes in the larger network and placed accordingly with another iteration of position estimation.

Bulusu etal propose a GPS-less localization methodology suitable for outdoor environments using RF-connectivity. Multiple nodes in the network with overlapping regions of coverage serve as reference points. They are situated at known positions (these nodes can be capable of running GPS) (X1, Y1) - (X_n,Y_n), that form a regular mesh and transmit periodic beacon signals (period = T) containing their respective positions. Sensors listen for a period t >> T to evaluate connectivity. If the percentage of messages received from a beacon in a time interval t exceeds a threshold Cmthresh, that beacon is considered connected. When the beacon placement is uniform, the centroid of the positions of all connected beacons is a feasible solution in the region of connectivity overlap. For non-uniform placement, a feasible solution can be found using the convex optimization techniques used in the previous method. This coarse grained, decentralized protocol doesn't require coordination among reference points or sensor nodes. It is therefore potentially scalable to very large networks.

Niculescu and Nath propose APS (Ad hoc Positioning System), a method to extend the capabilities of GPS to non-GPS enabled nodes in a hop-by-hop fashion in an ad hoc network. Positioning is based on a hybrid method combining distance vector like propagation and GPS triangulation to estimate location in presence of signal strength measurement errors. This mechanism applied same principle as GPS with the difference that the landmarks are contacted in a hop by hop fashion rather than directly. This method is similar to the distance vector routing, in the sense that at any time each node communicates with its immediate neighbors and in each message exchange it communicates its available estimates to landmarks acquired so far. APS is distributed, doesn't require special infrastructure or setup, provides global coordinates and requires recomputation only for moving nodes. Actual locations obtained by APS are on average less than one radio hop from true location.

IX. TIME SYNCHRONIZATION

Time synchronization is an important aspect of the distributed wireless sensor networks but often have unique constraints in the scope, lifetime and precision of the synchronization required as well as time and energy that can be expended to achieve it. Different applications like beam-forming array, data aggregation, recognition of duplicate detection of same event from different sensors, ordering of logged events have different synchronization requirements and also any single synchronization mechanism is not appropriate for all circumstances sensors should have multiple methods available to them so that they can dynamically trade precision for energy, or scope for convergence time. Existing time synchronization methods like NTP conserve use of bandwidth and try to keep the clock synchronization at all times but are not aware of the stringent energy constraints and the heterogeneity of the hardware that may be deployed in sensornets.

J.Elson and D.Estrin in [8] propose a post-facto synchronization where clocks are normally unsynchronized. When a stimulus arrives, each node records the time of the stimulus with respect to its own local clock. Immediately afterwards, a third party node (a beacon) broadcasts a synchronization pulse. Nodes that receive this pulse use it as an instantaneous time reference and normalize their stimulus timestamp with respect to that reference. This method is limited by the transmit range of the beacon and creates only an "instant" of synchronized time. This method is inappropriate for an application that needs to communicate a timestamps over long distances or times. However it provides enough service for beam-forming applications, localization systems and other situations in which one need to compare the relative arrival times of a signal at a set of spatially local detectors. The receiver clock skew (clocks don't run at exactly same rates) variable delays in receivers (all receivers don't detect the signal at the same event) and propagation delay of the synchronization pulse affect the precision achievable by this method. NTP can be used to discipline the frequency of each node's oscillator.

Many network synchronization methods including the one described use a design where a server periodically sends a message containing its clock value to a client. J.Elson et.al propose a scheme Reference-Broadcast Synchronization or RBS [9] that synchronizes a set of receivers with one another" as opposed to the traditional model of synchronization of sender with receiver [2,3]. In this scheme nodes periodically send beacon messages to their neighbors using the network's physical-layer broadcast. Recipients use the message's arrival time as a point of reference for comparing their clocks. The message contains no explicit timestamp, nor is it important exactly when it is sent. The most significant limitation of RBS is that it can't be used in networks that employ point-point links since it requires a network with a physical broadcast channel. However it is applicable to a wide range of applications in both wired and wireless networks where a broadcast domain exists and higher precision synchronization is required that the 500?sec-2000?sec bound that NTP can typically provide in a LAN. To come over this limitation the paper also proposes a multihop scheme where it dynamically constructs a "time route" through a series of nodes which allows locally coordinated timescales to be federated into a global timescale, across broadcast domains with little loss in accuracy.

X. LIFETIME ESTIMATION

[24] explore the fundamental question concerning the limits of energy efficiency of sensor networks - What is the upper

All rights reserved by *www.ijsrd.com*

bound on the lifetime of a sensor network that collects data from a specified region using a certain number of energy-constrained nodes? The answer to this question allows one to calibrate the performance of collaborative strategies and protocols being proposed regularly. The exposure of lifetime's dependence on factors like source behavior, source region, basestation location, number of sensors, available initial energy, radio path characteristics allows one to see what factors have most impact on lifetime and consequently where engineering effort is best expended.

The authors measure the bounded network lifetime as the cumulative active time to the first loss of coverage. They state the Lifetime Bound Problem (LBP) as below: Given the region of observation(R), the source radius of observability (d_S), the node energy parameters ($?_1$,$?_2$ and n), the number of nodes deployed (N), the initial energy in each sensor (E), what is the upper bound on the active life time (L) of any network established using these nodes which gather data from a source residing in R with spatial location behavior $l_{source}(x,y)$. In the model, they have assumed that the sensor nodes are static, while the target moves around according to some location distribution. This is a reasonable assumption in most of the applications.

First the energy consumed for point-point communication is established as follows. Given a transmitter A and a receiver B separated by D meters, intermediate nodes between A and B are introduced as relay nodes to prevent any nodes from spending too much energy. They introduce the Minimum Energy Relay scheme, which transmits data between any two nodes such that overall energy dissipation is minimized. Suppose K-1 relays are introduced with distance between any two consecutive nodes di, i= 1...K. Then the total energy is

[equa 1 from SN survey]

After minmizing the function Plink (D), the following result is derived

[equa 2]

Where d_{char} = [equa 3]

The main observation from above bound is that for a given D, there are a certain optimal number of intervening nodes acting as relays that must be used. Using more or less than this optimal number leads to energy inefficiencies. Notice that this analysis is best-case analysis considering that the worst-case analysis is meaningless in this situation, since the lower bound of lifetime can be arbitrarily bad for any network. The work presented in this paper will enable a deeper understanding of the fundamental limits of energy efficiency of wireless sensor networks.

XI. MONITORING WIRELESS SENSOR NETWORKS

For extending the lifetime of a sensor network, the sensor itself must be made, as energy efficient as possible and the collaborative strategy, which coordinates sensors, must be energy efficient. However, in scenarios where battery replacement is infeasible, the network lifetime can't be extended beyond a certain time, which depends on the initial capacity of the batteries in the sensors. [37,38] deal with the situation where the replacement of batteries is feasible. The problem of fundamental importance in this scenario is identifying faulty (crashed) nodes in the network. The diagnostic information (i.e., the status - operational/crashed - of each node) gathered by operational sensors can be used by an external operator to maintain network functionality by replacing the depleted batteries. Since the traditional distributed diagnosis protocols are designed for multiprocessor computers or wired networks they are infeasible or extremely energy consuming. For this reason, the authors developed a distributed silent fault diagnosis protocol called WSNDiag explicitly designed for wireless sensor networks. The protocol takes advantage of the shared nature of communications and aims at minimizing the total number of bits exchanged for the purpose of diagnosis, thus reducing the energy consumption entailed by the protocol execution. The protocol first constructs a spanning tree of the graph representing the network topology, and then exchanges diagnostic information only along the edges of the tree. This allows a significant reduction in the number of messages to be sent for the purpose of diagnosis.

Zhao et.al. in [39] deal with the same problem but take a different approach. While they agree for the need to have continuously updated information about network resources and application activities in a wireless sensor network after its deployment in an unpredictable environment, they argue that due to the constraints of low user-to-sensor ration, limited energy and bandwidth resources, it is inefficient to extract state of each individual sensor. They propose sensor network scans as indicator of network health. The proposed mechanism for collection residual energy scan (eScan) applied localized algorithms in sensor networks for energy-efficient in-network aggregation of local scans. Rather than collect all local scans centrally, this technique builds a composite scan by combining local scans piece-wise. At each step of aggregation these partial scans are auto-scaled by varying their resolutions. They also propose to apply incremental updates to scans i.e., when the state of a sensor changes rather than continuously re-sending its entire scan, it sends a partial update to a scan only when its local state has changed significantly. Furthermore, update traverses up the aggregation hierarchy if it impacts some aspect of the overall representation. An aggregate scan may loose detailed information such as the residual energy level at each node, but the compactness of such an abstracted representation can reduce the communication and processing cost significantly. Through simulations they show that the trade-off between this reduced fidelity and increased energy savings is acceptable. This mechanism, besides helping the user to decide where new sensors be deployed to avoid energy depletion, can help verifying the behavior of energy aware routing protocols and guide in incremental deployment of sensors.

XII. SECURITY PROTOCOLS FOR SENSOR NETWORKS

As sensor networks edge closer towards widespread deployment, security issues become a central concern. All the work that was presented till now has focused on making sensor networks feasible and useful, and has not concentrated on security [23]. Despite the severe challenges of limited processing power, storage bandwidth and energy, security is important for these devices. These sensors measure environmental parameters and control air-conditioning and lighting systems. Serious privacy questions arise, if third parties can read or tamper with sensor data. In the future these wireless sensor networks will be used for emergency and life-critical systems and there these

All rights reserved by *www.ijsrd.com*

questions of security becomes foremost. The limited energy supplies create tensions for security: on one hand, security needs to limit its consumption of processing power, on the other hand, limited supply limits key life time (battery replacement reinitializes devices and zero out the keys). The aforementioned constraints make the current secure algorithms impractical. For example, the working memory of a sensor node is even insufficient to hold the variables required by asymmetric cryptographic algorithms like RSA It is found that purely symmetric cryptographic primitives (where both parties share a common key) are more suitable for the resource constrained sensor networks.

The security properties required by sensor networks can be classified as below:

A. Data Confidentiality:

A sensor network should not leak sensor reading to the neighboring networks. The standard solution is to encrypt the data with a secret key.

B. Data Authentication:

An adversary can inject messages, so the receiver needs to make sure that the data used in decision-making process originates from correct source, In two-party communication case, data authentication can be achieved through a purely symmetric mechanism. But the sensors need an authenticated broadcast mechanism and hence we need to construct an asymmetric mechanism from symmetric primitives.

C. Data Integrity:

This is necessary to ensure the receiver that the received data is not altered in transit.

Data Freshness: Given that all sensor networks stream some form of time varying measurements, it is not enough to guarantee confidentiality and authentication; we must make sure that each message is fresh. Data freshness implies that the data is recent and that no adversary replayed old messages. The possible two types of freshness are : weak freshness, which provides partial message order but carries no delay information and strong freshness, which provides a total order on a request-response pair and allows for delay estimation. Weak freshness is enough for sensor measurements, while strong freshness is useful for time synchronization.

Perrig et.al. present SPINS [23], a suite of security building blocks optimized for resource-constrained environments and wireless communication. The main achievement of the authors is that they show that it is feasible incorporate security mechanisms on minimalistic hardware. Their trust setup each node is given a master key, which is shared with the base station, and hence nodes implicitly trust the base station. Initially they setup secure channels between nodes and base stations to bootstrap other secure channels. SPINS has two secure building blocks: SNEP (Sensor Network Encryption Protocol) and ?TESLA (micro version of Timed Efficient Streaming Loss-tolerant Authentication). SNEP uses a shared counter between sender and receiver and thus avoid transmitting the counter value in contrast to other cryptographic algorithms. With the use of this counter SNEP achieves Data confidentiality, two-party data authentication, integrity, semantic security and weak freshness. SNEP achieves low communication overhead since it only adds 8 bytes per message and by keeping the counter state at each endpoint. A particularly hard and important mechanism for sensor networks is to provide efficient broadcast authentication. But this requires an asymmetric mechanism; otherwise any compromised receiver could forge messages from the sender. ?TESLA overcomes the computation, communication and storage overhead of asymmetric mechanisms through a delayed disclosure of symmetric keys. ?TESLA requires that the base station and nodes are loosely time synchronized and each node needs to know an upper bound on the maximum synchronization error. To send an authenticated packet, the base station simply computes a MAC (message authentication code) on the packet with a key that is secret at that point in time. When the node gets the packet, it can verify that the corresponding MAC key was not yet disclosed by the base station. Since the receiving node is assured that the MAC key is only known by the base station it is assured that no adversary could have altered the packet and hence buffers the packet. At the time of key disclosure, the base station broadcasts the verification key to all receivers. When the node receives the disclosed key, it verifies the authenticity of the key and uses it to authenticate the stored packet. Hence the key disclosure is independent of the packet broadcast and is tied to time intervals. The authors show that most of the overhead of adding security to the sensor networks comes from the transmission of extra data than computational costs.

Although SPINS addresses many security issues, it doesn't deal with information leakage due to covert channels. The suite merely ensures that a single compromised sensor doesn't reveal the keys of all the sensors. It is still an open problem on how to design efficient protocols that scale down to sensor networks, which are robust to compromised sensors. Finally, the problem of DoS attack on a wireless network by jamming the radio channel with a strong signal has to be dealt with.

XIII. CONCLUSIONS

In conclusion, wireless sensor networks present fascinating challenges for the application of distributed signal processing and distributed control. These systems will challenge us to apply appropriate techniques and metrics in light of the new technology opportunities (cheap processing and sensing nodes) and limitations (energy constraints).

We need a systematic analysis (similar to the SPEC benchmarks) of the architectural alternatives in the network sensor regime. Any proposed algorithm has to be experimented with larger number of nodes to further explore the scalability. Much of the current work is evaluated using ad-hoc simulations. Though current simulators are helpful in this regard, we need a common framework simulator, which can be used by everyone and hence one can make comparisons from the results. Furthermore the different solutions proposed have to be deployed in a real test bed and a detailed comparative analysis has to be made. Although hands-on experience with real embedded systems is essential for algorithm development in solving real problems, dealing with the real uncertainties, using real capabilities, it is difficult to isolate causes for specific behaviors and explore the space of possible interactions in this mode. An emulator with reasonable detail may prove

All rights reserved by *www.ijsrd.com*

helpful in this regard. Novel debugging and visualization technologies designed specifically for the new challenges of sensor networks will be very helpful in testing and maintenance of new algorithms and applications. We also see the need to come to a consensus on some characteristics of wireless sensor networks and the underlying assumptions that can be made while working on any solution.

REFERENCES

[1] Deborah Estrin, Ramesh Govindan, John Heidemann and Satish Kumar " Next Century Challenges: Scalable Coordination in Sensor Networks "In MobiCOM , August 1999.
[2] T.K.Srikanth and Sam Toueg. Optimal clock synchronization. J-ACM,34(3):626-645, July 1987.
[3] David L. Mills. Internet Time Synchronization: The Network Time Protocol. In Zhonghua Yang and T.Anthony Marsland,editors, Global State and Time in Distributed Systems. IEEE Computer Society Press, 1994.

All rights reserved by *www.ijsrd.com*