ADVANCEMENT IN DIRECTED DIFFUSION TO REDUCE ENERGY CONSUMPTION IN WSN

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

MUKHTAR ALI ANSARI
[2K16/CSE/07]

Under the supervision of

MR. VINOD KUMAR
Associate Professor

Dept. of Computer Science & Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY

(FORMERLY DELHI COLLEGE OF ENGINNEERING)
BAWANA ROAD, DELHI-110042

JUNE 2018

DECLARATION

I, MUKHTAR ALI ANSARI, ROLL NO.-2K16/CSE/07 Student of M.TECH(COMPUTER SCIENCE AND ENGINEERING) hereby declare that the thesis work entitled "**ADVANCEMENT IN DIRECTED DIFFUSION TO REDUCE ENERGY CONSUMPTION IN WSN"** which is submitted by me to the department of Computer Science and Engineering, Delhi Technological University, in partial fulfillment of the requirement for the award of the degree of Master of technology, is a bonafide report of Major Project-II. This work has not previously formed the basis for the award of any degree, diploma, fellowship or other similar title or recognition.

Place: Delhi

Date:

MUKHTAR ALI ANSARI

M. Tech CSE

Roll No.- 2K16/CSE/07

CERTIFICATE

I hereby certify that the project Dissertation titled "**ADVANCEMENT IN DIRECTED DIFFUSION TO REDUCE ENERGY CONSUMPTION IN WSN**" which is submitted by MUKHTAR ALI ANSARI, Roll No.- 2K16/CSE/07 [Computer Science and Engineering], Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of technology, is a record of the project work carried out by the students under my supervision. To the best of my knowledge this work has not been submitted in part or full for any degree or diploma to this university or elsewhere.

Place: Delhi

Date:

SUPERVISOR

Vinod Kumar

Associate Professor

Department of Computer Science and Engineering

Delhi Technological University

ABSTRACT

Continuous advances in computation power, compactness of memory, energy efficiency in radio technology, had made wide-scale deployment of sensor networks inevitable in today's scenario. Nodes in WSN (Wireless Sensor Networks) autonomously collect information about ambience and dump them to sink nodes (central entity of a WSN). A crucial cost for autonomous behavior is energy. Since, WSN are data-centric, there is a need to analyze data to avoid redundancy, before data extraction. Directed Diffusion is a protocol proposed to gain the same goal. In this paper, we presented some additional paradigms using random numbers to convert the two-phase model in one phase, making backtrack to sink node easier and also chosen path on the basis on energy efficiency and survivability rather than latency.

ACKNOWLEDGEMENT

It is a distinct pleasure to express my deep sense of gratitude and indebtedness to my learned and enlightened supervisor **MR. VINOD KUMAR,** Associate Professor in department of computer science and engineering, Delhi technological university, for his valuable guidance, encouragement and patient review. Without his help and guidance, this dissertation would have rather impossible.

I am also thankful to all my friends who have continuously helped and motivated me during the course of this dissertation.

Last but not the least, I express regards to my parent, who have been a constant source of inspiration for me.

MUKHTAR ALI ANSARI

Roll No.-2K16/CSE/07

CONTENTS

LIST OF FIGURES

Figure No.	**Caption**	**Page No.**

LIST OF TABLES

Table No. **Caption** **Page No.**

LIST OF ABBREVIATIONS

WSN	Wireless Sensor networks
SPIN	Sensor Protocol for Information via Negotiation
DD	Directed Diffusion
MACA	Medium Access with Collision Avoidance
S-MAC	Sensor-MAC
T-MAC	Timer-MAC
D-MAC	Dynamic-MAC
M-SPIN	Modified SPIN

CHAPTER- 1

INTRODUCTION

1.1 INTRODUCTION

Wireless sensor network [1,2,3] consists of multiple nodes and their major function is to sense data (thermal, electromagnetic ,seismic, magnetic, biological, chemical ,optical, accelerometers, etc) and convey *useful* data to the sink nodes. This task can be categorized as: Sensing, Computation, Communication. Today, both sensing and computing are cheaper as compared to the energy consumption of communication. The work of Pottie and Kaiser [4] showed a fine demonstration of this that says, about 3000 instruction could be executed with same energy as to communicate 1 bit for a distance of 100m by radio. Also, day by day this gap is further increasing since computing is becoming more and more cheaper with time. So, we make a trade-off between Computation and Communication for the ultimate achievement of energy efficiency. For sensor networks, energy is the major factor as their discharging batteries cannot be replaced at frequently since WSN are usually deployed in hostile environment so the frequency of battery replacement should be in order of multiple years. For achieving this much of efficiency, trade-off should be bent towards computation and minimum possible amount of data communication has to be done.

WSN takes data centric approach instead of taking address centric i.e. there is no global addressing of nodes. There is an interface between WSN and outer world called Base station. In WSN, Data is retrieved through this base station. The working starts with when a query is inserted into sink node. Sink node is a node responsible for provide the data in response to that query. This was the abstract problem statement for extraction of useful data from WSN network. Two families of protocols were proposed for this problem:

- Family of SPIN(Sensor Protocols for Information via Negotiation) protocols
- Directed Diffusion

SPIN protocols basically aims at restricted flooding of data throughout the sensor network whereas Directed Diffusion, in an energy efficient way, gives a complete standard approach to extract data from WSN.

1.2 PERFORMANCE ISSUES AND RESEARCH CHALLENGES

For sensor networks, energy is one of the paramount factors since replacement of their discharged batteries cannot be done at frequent intervals because WSN are ususally deployed in hostile places where human involvement is troublesome. Thus the battery replacement frequency should be in order of multiple years. For achieving this extent of efficiency, trade-off should be bent towards computation and minimum possible amount of data communication is to be done.

At MAC [5, 6, 7] there occurs a collision if a node receives more than one packet at same time. Now, data re-transmission will be required which is certainly infeasible in WSN. Also, other problems such as *idle listening* (listening to idle channel), *overhearing/Noise* (i.e. receiving packets that are intended to reach other nodes), and *over emitting* (transmitting messages when receiver is not ready for it) are to be handled for a gentle data transmission. Medium Access protocols were not designed keeping energy efficiency as a paramount desire in mind, thus they are not directly feasible to sensor networks. So, there are a few MAC protocols named S-MAC, T-MAC etc which conjunct earlier protocols like CSMA/CA [8], CSMA/CD etc with newer paradigms. I've shown most of these protocols and also a comparative analysis of all existing MAC protocols for WSN. Most of these protocols aim at decreasing any communication redundancy within the network thus, saving energy in a way. Data dissemination aims for minimum possible data transmission through looking into the abstract of data (e.g. Metadata and Value-Attribute pairs) such that *redundant data* is not transmitted, whereby, MAC protocols keeps a check for proper data exchange over the medium such that *no redundant transmission* takes place. So, both these paradigm are equally important for energy efficiency in WSN.

BACKGROUND AND LITERATURE REVIEW

2.1 DATA DISSEMINATION

Data-centric approach in WSN implies that detected data (from a node) may be processed by the nod before transmission and can transmit only vital data disposing redundancy. It also infers that there is no need of node addressing for neighbor-neighbor communication in spite of *address-centric* paradigm. Sensor networks might be thought as distributed-database because every node is in charge formicro-sensing only. Data detected from various nodes are joined together to finally project the genuine data sensed by the sensors. A typical example can be use of sensor networks in *distributed camera calibration*where diverse pictures from various angle at same instance of time are consolidated to create massive graphic effect in film generation. This approach encourages data aggregation and safeguard on a node failure.

2.1.1 Problems:

Transmission of data without any prior computation on it, may lead to redundant transmission thus results in energy wastage. A few problems which might persist and should be handled are:

- *Implosion: In case of flooding used for data transmission*, multiple copies of single data may reach at a node thus involving an overhead of atleast two sendings and two receivings (and both sending and receiving are the two most costly tasks in terms of energy) as shown in Figure 1:

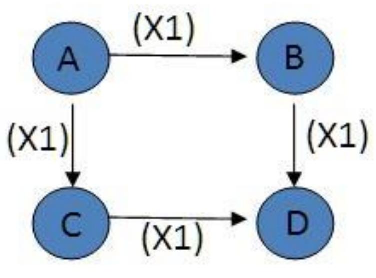

Figure 1: (X1) data received along the paths A-C-D and A-B-D. Two redundant sendings (from C and A), two redundant receivings (at C and D).

- *Overlapping Data:* In WSN, the distribution of nodes is very dense (close in proximity) over a given geographical area i.e. they may be very close to one another. So, therefore, any two nodes may sense some overlapping area of their domain of region hence they will sense redundant information i.e. multiple copies of same data reflecting information about same geographical area, as can be seen in Figure 2:

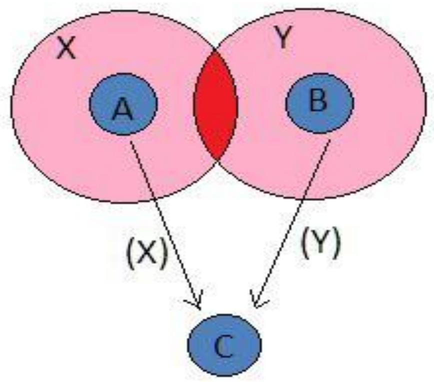

Figure 2: Node A and B having regions X and Y respectively. Sensed data of region X□ Y is received two times at node C.

- *Resource blindness:* A node, when in very frequent use, loses all its energy and finally becomes dead node. This might result in inappropriate data aggregation, and may complex the flow; so, the energy level should be taken in account.

2.1.2 EXISTING WORKS:

1. Sensor Protocol for Information via Negotiation (SPIN): SPIN [9] depends on *metadata* which used common data naming of Application Data Units under **ALF** (Application Level Framing) [10]. The protocols under SPIN suggested that transmission of metadata for complete message may not be costly and fulfill the need of energy efficiency. The nodes may transmit metadata before respective sensed data; which enables the recipient node to decide whether that data is already obtained by the node or not. And, therefore, complete message should be transmitted or not and saving energy. However, this scheme works iff metadata is

smaller than actual message, which usually is the case. There were two proposed protocols: SPIN1 and SPIN2.

(i) SPIN1: It is a 3-way handshake protocol for data scattering along the network. It has three stages, named as ADV-REQ-DATA, where ADV is Advertisement and REQ for Request. First a node, owning some data in hand to disseminate, sends an ADV message to all its neighbor nodes. On receiving ADV, the nodes checks if they already have that data, or whether already requested for it; if not, it sends a REQ message in response. Now, the sender node sends the data and completes the transmission. Although, this further raises the complexity during transmission but a remarkable amount of energy efficiency is achieved. This functioning is depicted by Figure 3:

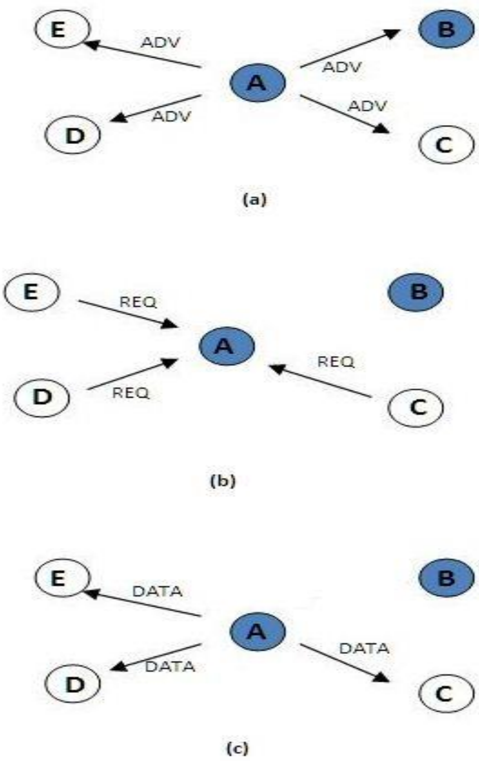

Figure 3: (a) A and B already contain the data. A is willing1to send data so it sends ADV1to its neighbors. (b) Since B contains1data already, except B every1neighbor replies with1REQ packet. (c) Finally, A sends1DATA already to its entire1neighbors from already

So, an outcome can be seen that therewas one less send and one less receive which is commendable for the increased computation, as, in WSN, we generally try to compensate energy in computation.

(ii) SPIN2: SPIN2 includes energy limitation on node during transmission i.e. when energy of a node is more than the pre-specified limit value for complete transmission process (ADV-REQ-DATA), then only, it may take part in any of the task. So, when energy level is low for a node it decreases part taking in data dissemination. This implies a node only initiate three-stage protocol if it assures that it can finish full protocol with everyone of its neighbors. However, it doesn't keep nodes from accepting ADV and REQ messages, but it checks sending ADV, REQ, DATA and furthermore checks receiving DATA.

(iii) M-SPIN: Modified-SPIN added ***Distance Discovery(DD)*** to original SPIN protocols. It was given by Zeenat et al [11]. DD determines hop distance of each node in the network, in terms of hop count. This disclosure is done in yet another stage i.e. distance discovery phase. In this phase, a packet called the startup packet, is broadcasted by sink node that consists of *node id*, *hop count and message type.* The hop count field is augmented by 1 at each receiving hop. If there occurs a case wherein node receives more than one startup packets, it checks all the packets and set the distance to minimum value. This process lasts until every single node has received startup packet at least once.

This phase can be shown as:

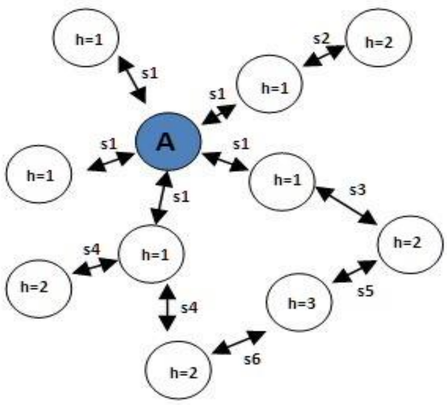

Figure 4: A is sink node and h is distance

This distance will then be utilized in negotiation phase of SPIN. The modification introduced is that the receiver node determines if it is closer to the sink node or with respect to the node that has sent the ADV. If hop distance currently received by the node is less than the hop distance it received as part of the ADV message, only then the receiving node sends REQ message to the sending node for current data. The energy utilization further increased with this idea.

(iv) Performance: Simulation of SPIN1 on extension of *ns* software package showed an energy saving with a factor of 3.5 over flooding with compromise of little convergence time. The observation regarding the redundancy percentage of various protocols is as follows:

-flooding sends 77% redundant data items,

-gossiping sends 96% redundant data items,

-SPIN1 sends only 53% redundant data items.

Though 53% is not much convincing for a factor of 3.5, but majority of these 53% redundant data items in SPIN are metadata not original message. A comparative study is shown below:

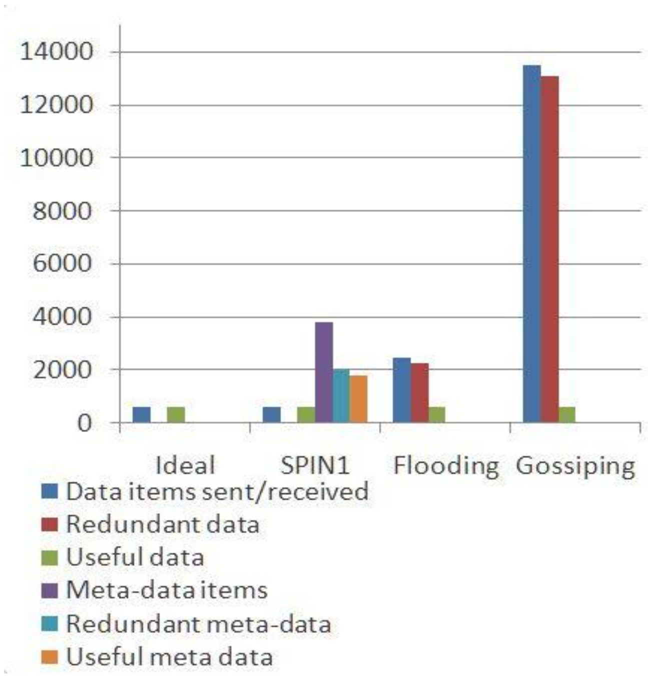

Figure 5: A comparison1of data items transmitted1in four1scenarios: ideal, SPIN1, classic flooding, and1gossiping [9]. Obviously, gossiping1is worst, whereas SPIN11is closest to ideal as metadata1packets are small in size.

SPIN2, which is aware of the problem of resource blindness, metric was to send maximum amount of data with given energy. SPIN2 distributed 73% in comparison of 68% of SPIN1, , 53% of flooding, 86% of ideal and 38% of gossiping. These figures shows the effectiveness of this idea.

Also, the number of ADV packets further reduced in M-SPIN which successively reduces corresponding REQ-DATA packets, by increasing the distance discovery overhead. But overall, it reflected lesser amount of data flow in the network, hence improving the energy efficiency.

2. Directed Diffusion: Directed diffusion [12] performs same task with the help of attribute-value pair. It is a good alternative for SPIN. This paradigm consists of various other elements, namely *interest, gradient, reinforcement.* Directed diffusion is also data-centric but unlike SPIN it doesn't use metadata, rather it uses value-attribute pair used to disseminate *interest* over the sensor network and collect the responses from the specific subset of the sensor network. Proceeding with the protocol, first we look up at every element, individually, of this paradigm and then combine them for understanding overall working of directed diffusion.

(i) Naming: As proposed in earlier works, directed diffusion was *application specific* i.e. for different applications it may be chosen differently. Likewise, we've taken an example to collect animal tracking information in a specific part of geographical area that falls under the domain of given wireless sensor network. Then, naming of task description can be done as follows:

Type	: animal //detect animal
Interval	: 100ms //data rate
Duration	: 10s //for next 10 sec
Rectangle	: [-100 -200 100 200]
Timestamp	: 06:23:26

In the above example we've taken a generalized value for coordinates of rectangle. Practically, GPS may be used for choosing the coordinates. Infact, not only a rectangle but any other shape can also be taken.

The task descriptor actually tells the type of information needed i.e. the *interest* of the sink node. Thus, this task descriptor is known as *interest.* When a node finds itself eligible for sending the data it might send the data in the format shown below:

Type	: animal // type seen
Instance	: rat // instance of this type
Location	: [125; 180] // node location
Intensity	: 0.6 // signal amplitude measure
Confidence	: 0.85 // confidence in the match
Timestamp	: 1:20:40 // event generation time.

(ii) Interest Propagation*:* After naming-scheme selection, what comes next is the dissemination of interest over the sensor network. Interest is injected at the sink node which is spread in the network. This sink node is not the same as the base station of network. Any arbitrary node can be opted for injection of interest. As mentioned, interval represents the data rate and duration is the time in which data from the sink node will be purged out. Initially, the sink node broadcasts the interest packet to all the neighbors at regular time intervals with timestamp increasing monotonically, but with a smaller data rate. As obvious, initially sink node will only be concerned about finding out that if there exists a required node or not. So, a lower data rate or higher interval value is set, because high data rate will cause overheads and downgrade the efficiency.

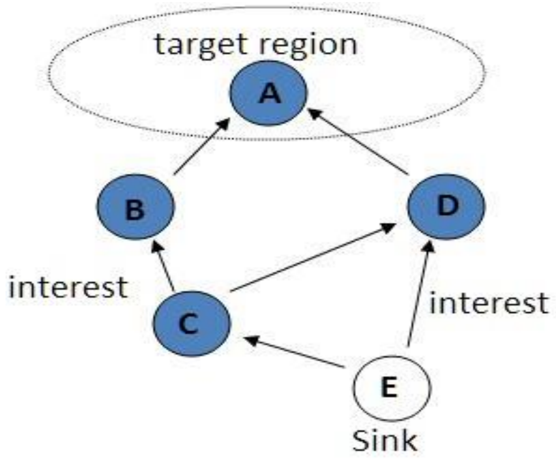

Figure 6: Interest propagation

Interest is broadcasted frequently to get greater dependability with confront of little overhead. Interest caches are kept in every node which includes an interest whenever it is unmatching as compared to others held by node already.

Two interests are said to be distinct iff attribute:

- Type is different
- Interval is different
- Rectangle is disjoint

Each section in interest field contains a **timestamp** and **gradient** field for each neighbor. Gradient field additionally contains **data rate** and **duration field.** Whenever a node receives an interest, it checks for its existence to include in cache and if it includes it fills the single gradient towards the neighbor and data rate as received. If, an interest entry is there but the gradient is not towards that neighbor, along with modification in other fields a gradient is included towards that neighbor i.e. similar type interest is received from two neighbors.

Subsequent to receiving an interest, node might choose to re-transmit the interest to a sub-set of its neighbor. A node may even ignore an interest if it has re-sent matching interest recently.

(iii) Gradient Formation: Gradient specifies both, the data rate as well as the direction. As we know, data rate is determined from interval attribute, located in the gradient section of interest cache.

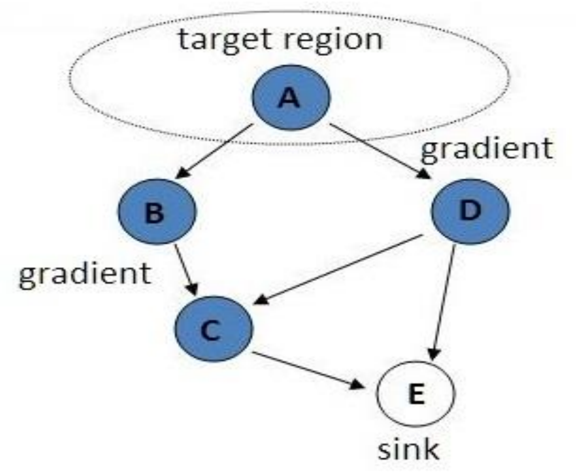

Figure 7: Gradient formation

Its use is depicted as: assume a node finds itself inside the rectangle, it initiates its sensors and begins gathering samples. A node that gets a target looks into its cache, if a match is found, it finds the entry which requires highest data rate and samples at the same rate. Then, it sends those data in the form as depicted previously. It is preferably unicasted to the neighbors instead of broadcasting. Nodes also keep up a *data cache* which have numerous usage like loop elimination and so forth. Other use is to synchronize i.e. a node checks rate of accepting by looking at current data and previous data from data cache using timestamps.

(iv) Reinforcement: Till now, disseminated interest was of higher interval (or lower data rate) majorly aimed at exploratory work but data rate should be increased for actual data transmission, which is attained by *reinforcing* the paths. When a node, known as source node, detects the useful target (as specified in the interest), data is sent with the same interval in multiple paths, but when the **sink** node receives this data, it reinforces a neighbor to get the data in actual rate as required.

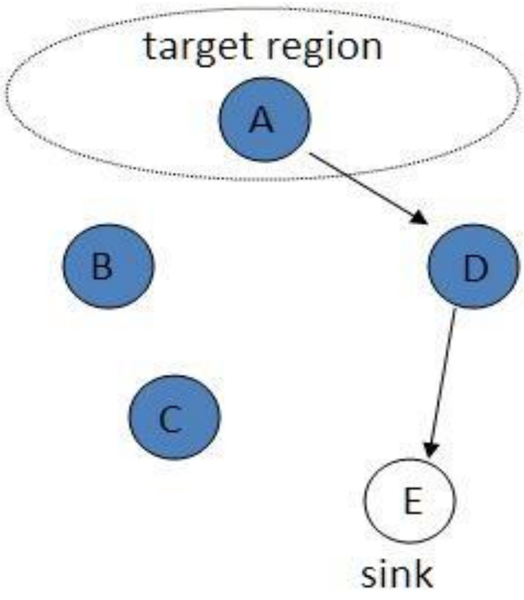

Figure 8: Reinforcement

It is still an open subject to decide which neighbour should be selected for reinforcing. One option, which is very intuitive obviously, is to choose the node from which it received a new data item first(as it is might deliver highest data rate). Sink node sends, to its neighbor, an interest packet with lower interval values (to denote higher data rate), for the purpose of reinforcing it. If this data rate is greater than any of the already existing gradient, it should subsequently reinforce one of its neighbors, and this way it propagates towards the source node.

(v) Performance: In directed diffusion, there is delay in delivery to evaluate its performance and classical flooding is referred considering energy efficiency. In flooding, all data will be broadcasted to each and every node in the sensor network without any further addition of the efficiency measures. Directed diffusion, to be accepted as a protocol, should perform better than any trivial paradigms and flooding is accepted be one of the easiest protocols that can be put in execution in almost any network.

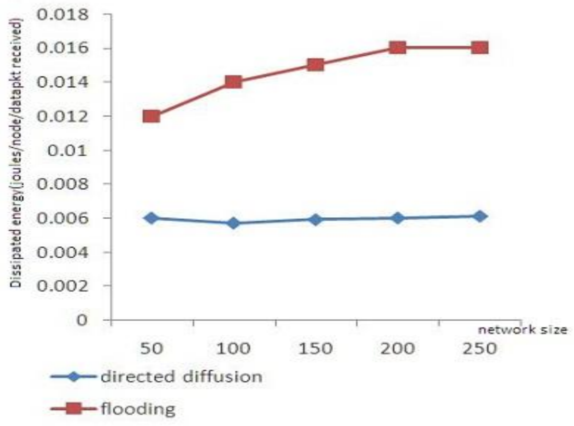

Figure 9: Average energy dissipated [12]

This graph depicts the results as expected and proves that directed diffusion is feasible in WSN on the critical factor mentioned by us i.e. Energy Efficiency. Although, only single factor would not confirm its complete admittance as one more factor is necessary & directed diffusion had to bypass it! This vital factor is the delay factor. Thankfully again, directed diffusion given positive results:

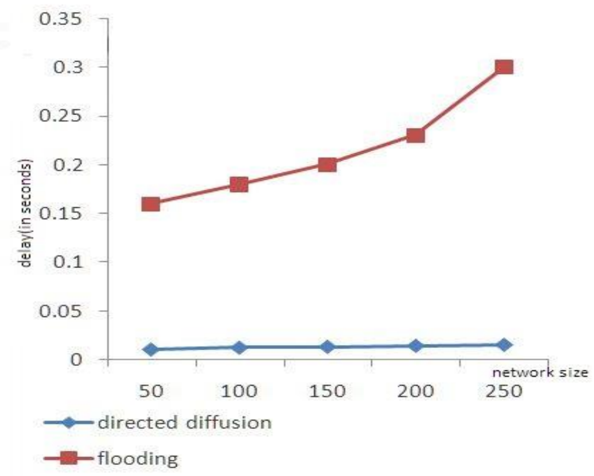

Figure 10: Average delay [12]

So, these remarkable results make directed diffusion very conversant in the dissemination of data over wireless sensor networks. Above this, its application specific behavior is a gem in the paradigm.

2.2. MEDIUM ACCESS PROTOCOLS

Till now, we were discussing about how to control the extent of data to be disseminated in the sensor network but this data transmission occurs through wireless mediums that are majorly responsible for the depletion of energy. Now, with an assumption that only useful data are transmitted, we'll check for redundant transmissions and try to minimize them.

2.2.1 Problems:

In Adhoc networks, Medium Access Protocols are primarily focused on *fairness and the throughput*, but in WSN, we know that energy efficiency is primary requirement. The major important factors responsible are shown in *Figure 11.* So, the protocols introduced previously were not directly feasible with wireless sensor networks. The discussion of Adhoc networks's protocols are beyond the scope of this paper. The primary reasons for this infeasibility in WSN are:

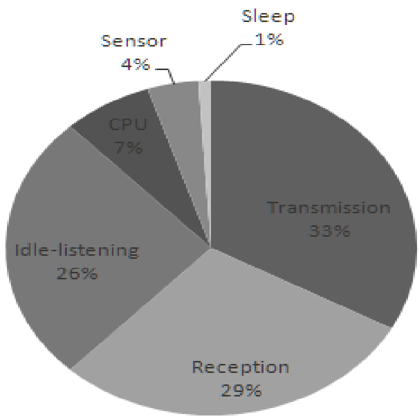

Figure 11: Energy consumption in WSN

1 Hidden and exposed terminals: Both of these cases leads to packet collision which causes plenty of energy loss. In the situation depicted in figure-11, node A & C are hidden from each other, so on being unaware of their presence, node C starts transmitting data to B and hence causing collision at B and both the received data will become void and there will be need of re-transmission. There may occur another reverse situation as depicted in the next topic.

These can be explained by following diagrams:

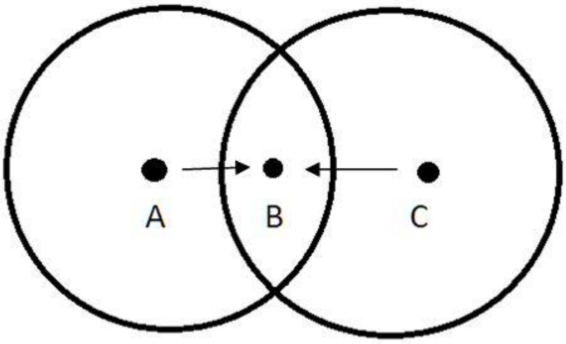

Figure 12: Hidden terminal

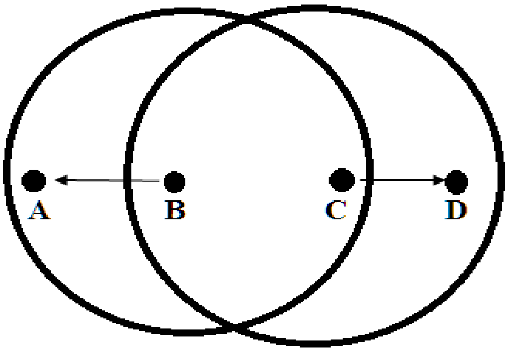

Figure 13: Exposed terminals

In this situation, B is sending data to A. Meanwhile, C also willing to transmit data to D but it senses that the medium is currently busy and concludes that collision will definitely occur but this is not the case in reality, actually the collision will not occur, this is called problem of exposed terminal.

2 Near and Far terminals: Due to the attenuation in signal strength along with distance (signal strength is inversely proportional to square of distance), another problem may arise named as Near and Far terminals. We take following example:

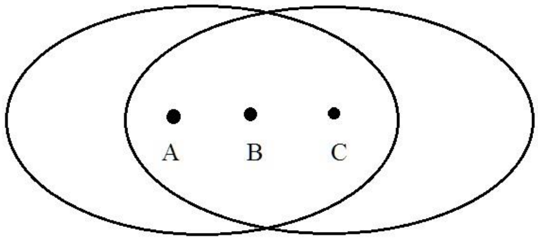

Figure 14: Far and Near terminals

In above situation, lets say B is sending data to any other node (except A and C), then since B is close to C, B's signal will dominate those from intended sender A.

3 Other problems:

- *Idle listening:*This problem occurs by turning on the radio transceiver even if there is no data transmission taking place.
- *Overhearing*: Sensor nodes receives the unintended packet and process it.
- *Overemitting:* There occurs a case when the receiver is not ready(sleep mode) but sensor nodes has started transmitting the packets.

2.2.2 EXISTING PROTOCOLS

We'll discuss the protocols which addresses these problems.

1. Multiple Access with Collision Avoidance (MACA): MACA[13] is a MAC layer protocol without any collision detection techique, usually deployed with wireless mediums. Firstly, a node that is willing to transmit data is supposed to send a RTS (request-to-send) message and get a CTS (clear-to-send) message in response by one of the receivers. Both RTS and CTS messages carry the names and length of both the nodes. Simultaneously any other node that has received a RTS will have to wait until these nodes complete the exchange of CTS message. And, as understood, any node hearing CTS message will have to wait for completion of ongoing data transmission. It is a solution to the problem of *hidden and exposed terminals* as depicted in following Figure:

.

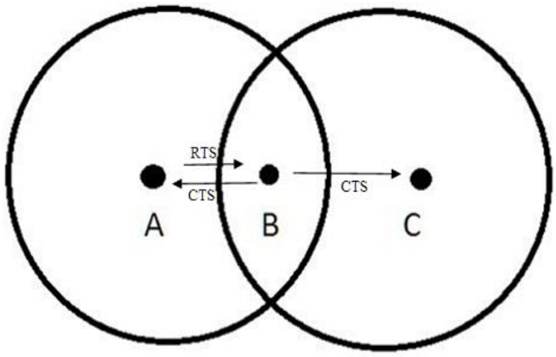

Figure 15: RTS/CTS exchange solving hidden terminal problem

Here, we're showing the solution of hidden terminal problem. Since A wants to send data to B, so as per the protocol, first it sends RTS message and get a CTS response from B, now as C already had got the CTS message from B, it is conversant that B is already involved in a transmission and will have to wait for it to get done. The problem of *far terminal is also solved in it.*

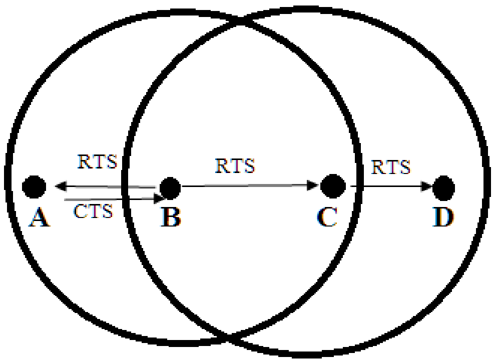

Figure 16: RTS/CTS exchange solving exposed

Here, let B wants to transmit data to A, so it broadcasts RTS request. when A receives that, it responds with CTS message, whereas, C has to wait for CTS, since C is out of proximity of A, it won't get any CTS message, thus it is free to transmit RTS message to D and need not wait for transmission of data between B and A and continue the transmission of data with it.

2. Sensor-MAC protocol (S-MAC): S-MAC protocol is one of the early age protocols designed for wireless sensor networks. As shown in *Figure 11,* 26% of energy is lost in idle listening. In S-MAC, we try to resolve idle listening by deploying a simple idea:

determine a schedule that tell the nodes when to set receivers into sleep mode and when to wake them up (activate the receivers). These are known as *sleep schedules* of the nodes. There is a SYNC message, locally distributed, to maintain the synchronization. For example, SYNC may contain node-id (local) and next-sleep time. The Communication of sender and receiver may be depicted as:

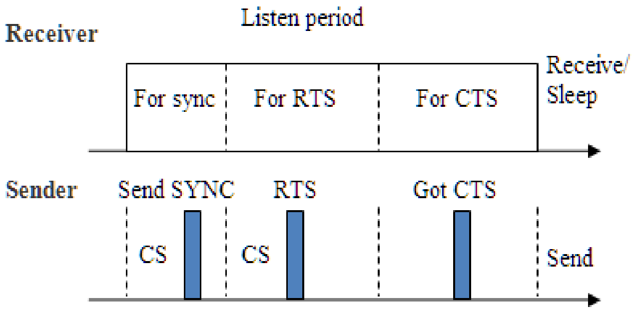

Figure 17: S-MAC scenario [5]

Here, first time slot is for SYNC, second and third for RTS/CTS exchange. Whenever a node is willing to send message, it waits for target node wake period and then start transmitting data in one go. This paradigm would complicate the flow if data to be sent is of longer length as retransmission would be exceptionally wasteful and cause immense unfairness. So, S-MAC additionally incorporates *message passing* system in which substantial data is divided into smaller parts and sent as individual data i.e. every piece follows after RTS-CTS exchange. Although, some unfairness and overhead still continues as now a data devours group of RTS-CTS exchange as opposed to one.

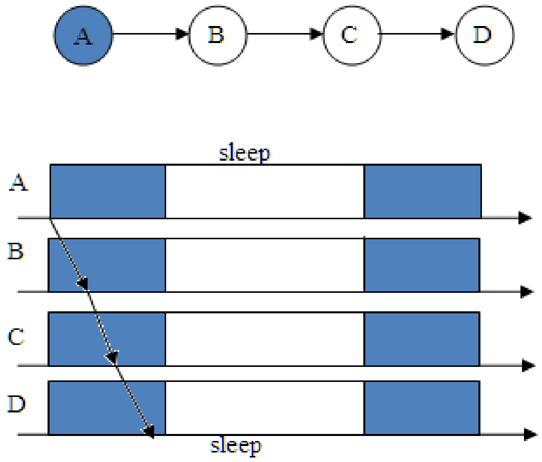

Figure 18: S-MAC [14]

Here, A is sending data to E through B and C. the sleep schedules used here are static, nodes synchronize their sleep schedules and keeps awake for a pre-specified timeframe, regardless of whether no data is being sent, degrading energy efficiency due to overhearing. S-MAC endeavors to cope up with all the four problems specified previously.

- It tackles the problem of *collision* by executing RTS-CTS exchange
- It takes care of the problem of *overhearing* by turning the radio off when transmission does not include that specific node.
- It additionally takes care of the problem of *idle listening* by turning on-off the radios periodically.
- Over-emitting is also solved as nodes now wait for the neighbor nodes to awake and then send the data.

The solution for over-hearing can be filtered even more since radio is not turned off before scheduled sleep time i.e. static behavior. There were some other protocols too. Data packets broadcasting do not use RTS/CTS, which raises the probability of collision. Versatile listening brings about overhearing or idle listening if the packet is not bound to the listening node. Sleep and listen periods are predefined and steady, which reduces the efficiency of the algorithm if there is variable traffic load.

3T-MAC: After S-MAC, numerous protocols were proposed that addressed the deficiency in it. T-MAC is concerned about reducing idle energy further by deciding the duration of wake period dynamically. As opposed to S-MAC, where messages are transmitted in predetermined active period, T-MAC sends data as burst in the beginning of the frame. On the off chance that there is no "activation event" occurring, then after waiting for a threshold time duration, nodes goes into sleep mode again independent of the prior schedule. "Activation events" might be any event which ought to be responsible of active state of the radio of the node, for instance, receiving or transmission of data, any radio communication detected, any neighbor participating in data exchange, timer goes off for timer for sleep etc.

In Figure 20, the case is same as in S-MAC, since D don't identify any neighbor activity at the beginning of wake-up period, it goes into sleep mode after predetermined time T0 and after that return to sleep mode. Once more, subsequent to having the data, C waits for T0 and goes to sleep mode and proceeds with its data transfer in next wake-up period. Waiting threshold time should be determined with care, as extremely smaller value of T0 may increase the delay and bigger value may decrease the energy efficiency. It may also be chosen in an application specific way,

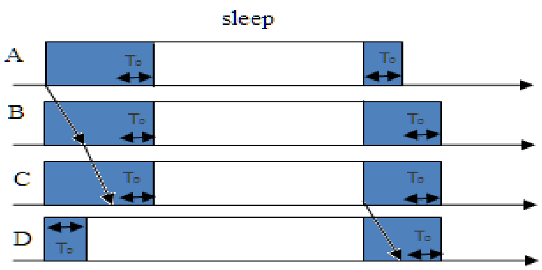

Figure 19: T-MAC sleep-schedule [15]

estimation based on average size of data packets and on location can be a smart thought! (As nodes close to the base station would be in more traffic). The betterment accomplished by T-MAC are because of the way that S-MAC may require its active period to be longer than necessary to accommodate traffic on the network with a given latency constraint. While the duty cycle can simply be tuned down, this won't represent burst of data that can frequently occur in sensor networks (e.g., following the detection of an occasion by numerous encompassing neighboring sensors).

4 D-MAC: As we probably are aware, in wireless sensor network, data is usually submitted to the base station at last, so convergecast is most normal communication design in WSN. Unidirectional ways towards the sink node can be diagramed into a tree structure. Using this reality, D-MAC was created which goes for same objective as for T-MAC i.e. energy efficiency by reducing *idle listening*. Further, it aims for low latency. In D-MAC, wakeup times for nodes are staggered to expel the delays. This can be depicted as:

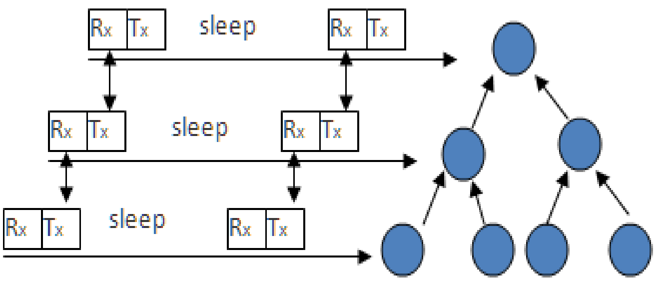

Figure 20: D-MAC tree structure [16]

From *Figure20*, during the receive period of a node, the greater part of its child nodes have transmit periods and battle for the medium. Low latency is accomplished by appointing subsequent slots to the nodes that are successive in the data transmission path. One other significant problem with D-MAC is its static topology, i.e. tree structure is pre-assumed and accordingly any adjustments in it, say, node failure and addition of a node isn't functionally supported. Another conceivable disadvantage can be collision at the time when nodes having same level, simultaneously transmit data. As no collision avoidance method is utilized here, it might cause significant loss sometimes. Likewise node in the tree initiates their receiving period after a delay offset of **t*d**, where t is receive/send period and d is the depth of the node in the tree. Conflict in send period is done by arbitrary back-off plan and without implementing RTS-CTS exchange.

5 Traffic-Adaptive Multiple Access protocol (TRAMA): TRAMA [17], quite similar to Node Activation Multiple Access (NAMA), is a TDMA-based protocol. While, MAC protocols tends to decrease idle-listeninng in order to increase energy efficiency, TRAMA does it by preventing the number of collisions because collision leads to retransmission. For each time slot In NAMA, a election is held locally in a distributive way in the range of two-hops. This consumes energy but will indeed reduce collision occurrences. In TRAMA, we have two time periods:

- Random access period, governed by NP (Neighbor Protocol)
- Scheduled access period, governed by SEP (Schedule Exchange Protocol).

In random access period, we maintain the information of two-hop topology and scheduled access transmission of schedule information and data takes place. Using NP and SEP, the neighbor information and the traffic schedule information is acquired, and the nodes determine their own radio state. Each node, with the help of a hashing function, estimates a priority for itself and of all the two-hop neighbors of the current slot. For that slot, if a node with highest priority has data to transmit, it wins that slot and sends its data. If one of its neighbors has the highest priority and the node finds that it is the intended receiver as per the information attained during SEP, it sets itself into receiving mode. Otherwise, it can go to sleep and save energy. The schedule packet uses *bit-map* to give information about the intended receiver. In bit-map, each of the local neighbors is represented by a bit position. The exact list is also held by those local neighbors, so they are conversant with the receiver as well. It also mentions if a node has won but still won't send any data, so that other interested senders can be re-evaluated to reuse that vacant time slot. Since two nodes within the two-hop neighborhood of a node may consider themselves slot winners if they are hidden from each other, nodes must keep track of an Alternate Winner, as well as the Absolute Winner for a given time slot, so that messages are not lost.

As aimed, TRAMA offers less collision probability and more sleep time. But in random access period, for exchange of schedule packets a node should either be in receive mode or send mode. So, irrespective of data, nodes have to lose their energy at least during random access period.

6 Sparse Topology and Energy Management (STEM): STEM [18] redefines a new idea for efficient transmission of data within the network. The new idea is not to switch on the radio till an activity has been noticed. Now since we require noticing the events even while *data channel* is reserved for data delivery, one more channel i.e. *wakeup channel* is responsible to wake the node up whenever traffic is generated. So, rather activating radios **proactively,** they are activated **reactively**. This is fully dynamic implementation for sleep schedules in the node as compared to any MAC protocol mentioned earlier in this paper.

Whenever any node generates data, it sends a signal over the *wakeup channel* to signal its downstream neighbors to activate their radios to receive the data and this paradigm continues in the network. There are two variants of STEM: STEM-T (Tone) and STEM-B (Beacon). IN STEM-T, a tone is send over a different radio channel continuously; and all other nodes should continuously listen to this channel which is little more energy consuming, but it offers minimum delay. It also may require extra hardware for additional radio. The preamble consists only of simple busy tone similar to traditional network protocols.

In STEM-B, node generating data sends beacons on a paging channel (special time slots in the main data channel). These beacons contain the MAC address of sender and intended receiver; on receiving these beacons node check whether they are intended receiver; and if they are, they send an acknowledgement to the sender and switch onto the data channel; on receiving an ACK message, sender stops sending beacons and start transmitting data.

This way, STEM-B can shorten the long preamble length but may cause delay as compared to STEM-T if used on the paging channel. Also, STEM-T uses simpler transceiver that can be lesser overhead as compared to communication over the main channel.

CHAPTER – 3

R-FEEDD and Improved Reliability

3.1 INTRODUCTION

Although, directed diffusion proved out to be humongous success in WSN on wide scale deployment but still there are a lot of scope in making this paradigm further efficient. R-FEEDD introduced two massive changes in trivial paradigms:

1. Used OPM (One Phase Model)
2. Chosen energy efficient path rather than latency

Utilizing OPM rather than TPM (Two Phase Model) means it shifted the selection of optimal path to the source node rather than sink node. This constrained R-FEEDD to work in scenario of bidirectional path only, rather than unidirectional paths as in trivial paradigms.

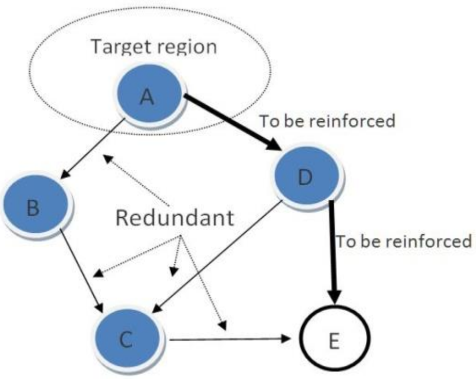

Figure 21: Redundant packets in Directed Diffusion

As depicted in Figure 21, if data is not required for longer period of time, paths depicted redundant will have no use as only reinforced path is utilized.

3.2 *Elements*

R-FEEDD introduced directed diffusion along with power of randomized numbers. One assumption that is being made in given WSN is symmetry of nodes which is generally the case. By symmetry we mean same radio transmission power of two nodes i.e. if node A can send data to node B then B can also send the data to A without inclusion of any additional hop. We first need to explain few elements that follow.

3.2.1 Random identifiers:

In WSN, there is no way for a node to identify any of the nodes globally but still it can identify itself and thus can identify the packets forwarded by it. To address itself every node generates a random number (Xi) and stamps it on every packet. Application of this will be discussed shortly in section 3.2.4. To accomplish identifying nodes within vicinity of neighbors following paradigm is implemented. Every node when receives a packet, it notes down (in the pseudo-routing table) from which neighbor it received the packet along with random number (Ri) received by it, then generate a new random number and forwards the packet and associate these numbers into its pseudo-routing table as described in section 3.2.2. This methodology is illustrated in Figure 2:

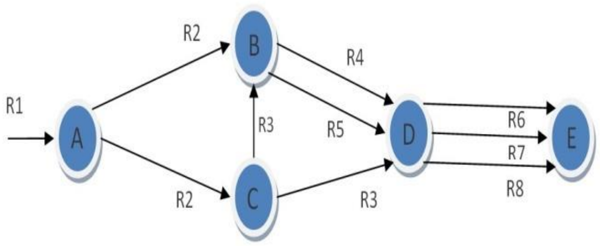

Figure 22: Random numbers associativity

Pseudo-routing table: Although in WSN, no global addressing is employed, but still the nodes can uniquely identify each of its neighbors, for example using Bluetooth address [7] etc. We maintained pseudo-routing table for each node in the network. This table contains 3 entries:

1. Sender node
2. Received random number
3. Forward random number

For Figure 22, pseudo-routing table will be formed as follows:

Table 1: Pseudo-Routing table for node A

Sender	Received random	Forward random
X	R1	R2

Table 2: Pseudo-Routing table for node B

Sender	Received random	Forward random
A	R2	R4
C	R3	R5

Table 3: Pseudo-Routing table for node D

Sender	Received random	Forward random
B	R4	R6
B	R5	R7
C	R3	R8

Now suppose for example, E wants to track the originator of a particular packet, it simply checks the random number of that packet, match it with received random number, find the associated neighbor with that random number and unicasts to that particular neighbor. This process can be iterated all the way to the originator. For example, suppose E checks the random number and found it to be R6, it notifies D, D looks into its table (depicted in Table 3) and finds R6 associated with R4 of B, thus it notifies B, then B checks its table (as in Table 2), find R4 associated with R2 of A, thus it notifies A, and A notifies the earlier node (using Table 1), similarly the originator could be notified. Now, as we utilized random numbers, we now analyze what these random numbers are and what may be shortcomings of them.

3.2.2 Random number in WSN:

Random numbers are generated by every node as discussed above. Now, a problem arises that random numbers generated may be accidentally same. But, fortunately, order of number of nodes in WSN is 1000 or even less. Ensuring large range of random number generation minimizes the probability of this accident, and even if they do are same, there is very micro probability that it was in the vicinity of close enough neighbors to create a problem. But still, clashes may occur, again fortunately, any protocol developed for WSN are self repairing as nodes frequently die. So, the failure introduced may be considered as some type of specialized node failure and the network will repair it with another path.

3.2.3 Path trace:

In interest propagation, initial step of directed diffusion, we tried that source node decides which path it should use for actual data transfer. And, data may reach to a node in multiple paths, so we cannot drop duplicate interests as it may come from a new and better path; neither can we forward each packet as it will stray towards flooding. To aid this, we use path trace; every node just appends its random id (Xi) in the interest.

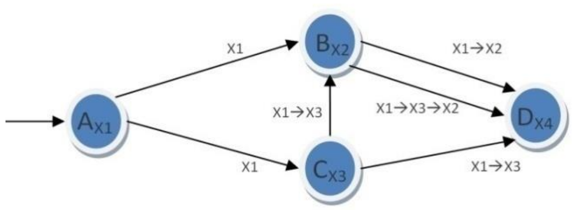

Figure 23: Path traces

In Figure 23, node D receives two packets from B; when path trace is not used, D cannot just receive all packets from B as it may result in flooding; neither can it reject all packets from B as it may lose a better path. When path trace is applied, D receives two packets one with path trace exactly as {X1,X2} and {X1,X3,X2}. Now, D realizes path traces are not same and that the packets are different, thus none are dropped. Also, this can be used to remove packet-looping, as when a node looks at random identifier and finds it to be same as its own random id, it drops the packet. Now, when random number identifier of two nodes becomes the same (as the unpredictable nature of random number); the path just dies but as discussed in section 3.2.3, it may recover with any other path.

3.3 R-FEEDD Funtioning

3.3.1 Interest propagation:

As in trivial directed diffusion, interest packet will contain attribute-value pair. But we inserted few more information into it. We added path trace and every node that comes in the path adds its residual energy in the path energy and increment hop count (utilized to calculate average energy of path by source node). Also, there are two more fields, Min and

Max energy field that are also updated by the node forwarding the interest. A node accepts an interest and forward it only if it do not have a same interest packet or if it do have the packet but not with same path trace. After accepting the interest, it adds its residual energy into the interest, appends its random-id (thus updates the path trace) and forwards it to its neighbors and store the updated path trace into its buffer. While an interest is received if node finds the same path trace as it has in its buffer, or even if it finds its own random id into the path trace, it refrains from forwarding it. This is done to remove packet looping. In this manner, finally interest is received by the source node.

3.3.2 Choosing optimal path:

Now, as we asserted that the interest carries energy information of nodes in the path, now we are equipped with enough data to choose an optimal path. Two factors that we considered for this choice are: (i) **Average energy** (ii) **Min-Max**.

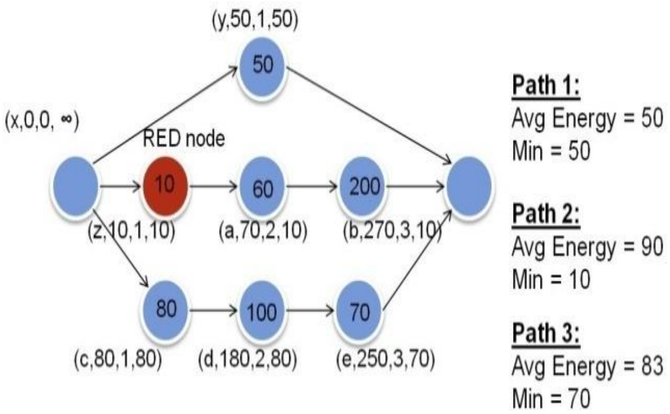

Figure 24: Optimal Path Selection

Source node calculates the average energy of path, for every path trace received by it, then it finds out the maximum of it, let it be M. All paths having energy greater than aM will be qualified for average energy round. Let us suppose an example of Figure 4 where tuples represents (interest identifier, total energy, hop count, MIN). Here for a = 0.7, path 1 will be disqualified. Here, a $(0 \leq a \leq 1)$ is application specific parameter and could be chosen in application specific manner; higher the value of **a** more the paths will be discarded due to lesser enery.Now, although path 2 has more average energy, still it is obvious that Path 3 is better as compared to path 2 because path 2 contains node that are far less reliable as compared to nodes in path 1; we call it RED node. To cope up with this problem we decided to add one more field in the header named MIN. It just keeps the record of minimum amount of energy possessed by a node in path. So, source node chooses the path having maximum

MIN field of the path out of the paths which qualified for average energy constraint. Here, path 3 (having MIN 70) will be chosen over path 1 (having MIN 10).

3.3.3 Reinforcement:

Now after selecting the optimal path from the alternatives, that path should be reinforced for actual data transfer. The source node not only recognizes the node from which it received the packet but also has the corresponding random number associated for the unique transfer. Now, a packet of reinforcement type can be send all the way to the sink node, these packets upon receipt by the nodes in between cognizes that it has been chosen to be a part of reinforced path, and thus it only unicasts this packet to next hop in the optimal path and also the data packet following this packet. This process has already been explained in Section 3.2.2. Let us take a case in Figure 2, that node E is source node and node A is sink node. Reinforcement in this scenario will proceed as follows. Let us suppose in advance that A,C,B,D,E is the optimal path that is to be reinforced. Now, E gets interests from D, each of which corresponds to each three different paths from A to E. This interest along with itself also contains path energy measures as discussed in section 3.2.2. E receives three interests in which, E chooses packet corresponding to random number R7, it checks its pseudo-routing table and discovers D to be previous hop, so E sends a special reinforcement packet on arrival of which D cognizes that it has been selected in reinforced path, so when it receives data packets followed by this reinforcement packet, it just unicasts it to its neighbor B only and similarly each node does the same. Hence, path A,C,B,D,E can be reinforced in such manner.

3.4 IMPROVING RELIABILITY

3.4.1 Choosing optimal path

With an aim of improving the reliability of optimal path selection, we'll be handling a case which was skipped earlier by the R-FEEDD paradigm. Till the "interest propagation" element of R-FEEDD, everything remains the same i.e. there is no any possible change required. Although a fine observation was made in the process of choosing optimal path. There may occur a case when we get more than one optimal paths i.e. out of all those paths which are qualified for average energy round, there can be multiple paths with same value of MIN field (which is actually the maximum value of MIN field) as well as the same value of AVG Energy calculated. This is depicted in the figure- 25:

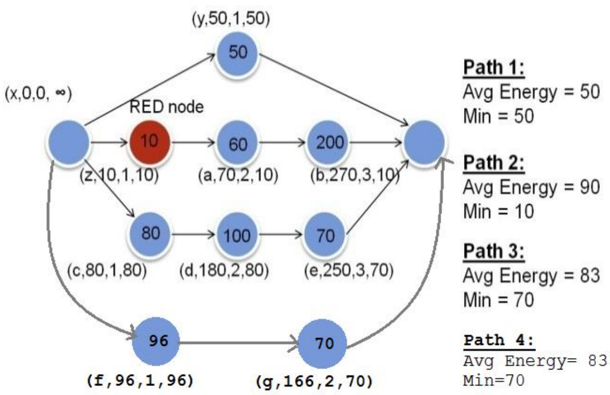

Figure 25: Optimal path selection with improved reliability

Now we end-up with a dilemma that which path is to be selected as the optimal path. As there are multiple paths qualifying the selection criteria, we must introduce an additional factor to tackle this scenario. It's by taking the Hop-Count in consideration, we can successfully escape the dilemma and get our optimal path finalized. As the paramount factor of wireless sensor network is energy efficiency, we'll select the path that causes least amount of energy consumption. So, the steps to select the optimal path are as follows:

1. Calculate the average energy path for each path trace and let the maximum value among them is M. So, all the paths having energy greater than aM will be qualified for average energy round.
2. Out of these qualified paths, one with the maximum value of MIN field will be selected as optimal path.
3. If there occurs a case that there are more than one paths having the maximum value of MIN field, then out of them the path which has the minimum hop-count will now be selected as the optimal path because it will cause minimum number of nodes to get discharge.

Path Reinforcement still remains same as it is proposed in R-FEEDD paradigm.

3.5 Analysis

In R-FEEDD, increase in energy efficiency is using local identifiers only and with the help of few data structures, most important being pseudo-routing table which makes a record about from which neighbor interest was received and forwarded (similar to gradient). Entries are frequently added and removed from these tables. And to gain further energy efficiency, we added energy regarding information into the interest packets that helped in optimal path selection. Table 4 depicts the difference in the proposed and trivial paradigm.

Table 4: DD vs R-FEED

Factors	**Directed Diffusion**	**R - FEEDD**
. Model	Two Phase Pull	One Phase Push
. Path Selection	Latency	Energy Efficiency
. Gradients	Data buffer	Pseudo-Routing Table
. Selecting node	Sink node	Source node
. Local Identifiers	Bluetooth address etc	Random Numbers
. Path Trace	No	Yes
. Control Packet size	Lesser(no energy info)	Higher
. Links	Unidirectional	Only Bidirectional
. Consideration of RED node	No	Yes
. Initial Path Formation delay	Considerable	Minimum

Other than initial path delay, all factors have been explained before. We assert that initial path delay for path selection will be lesser as our paradigm do not need exploratory replies from the source as in trivial paradigm. We applied two major changes in directed diffusion paradigm:

1. Reduced two phase path formation to one phase.
2. Chosen optimal path on the basis of energy efficiency rather than latency.

Consequence of first improvement is quite obvious and reduced the number of control packets drastically by totally removing one phase of control packet flow. However, we were bound to use another data structure Pseudo-Routing table in place of gradients. We also restricted our paradigm to bidirectional links which is generally the case for wireless sensor networks. The next improvement, certainly, need some proof before accepted to be a positive modification. For this purpose we simulated our paradigm on networks with different topology: grid topology as well as random topology. Fortunately, we obtained positive results. To make this simulation unbiased we took random status of the nodes like residual energy, links, position of source and sink node etc. We scaled the residual energy in range [10-1000]. In the simulation we did not transferred any actual data within the network because the changes we introduced were valid only until the path reinforcement after which both paradigms will behave same. The metrics that we choose for comparison are:

1. Optimal path energy per node
2. Energy of node with minimum energy in selected path

Since, we took random topology, to be certain that we did not have results randomly only once, we run the protocol for certain number of nodes several time and present the cumulative effect of results. Intuitively, higher the energy per node in the path, better is the paradigm and same goes for higher minimum energy of a node in selected path.

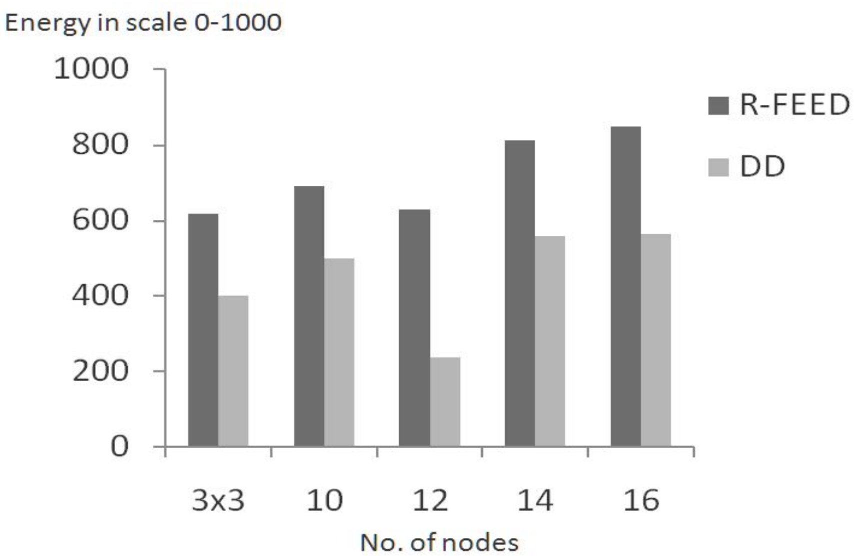

Figure 26: Comparison of energy per node of path

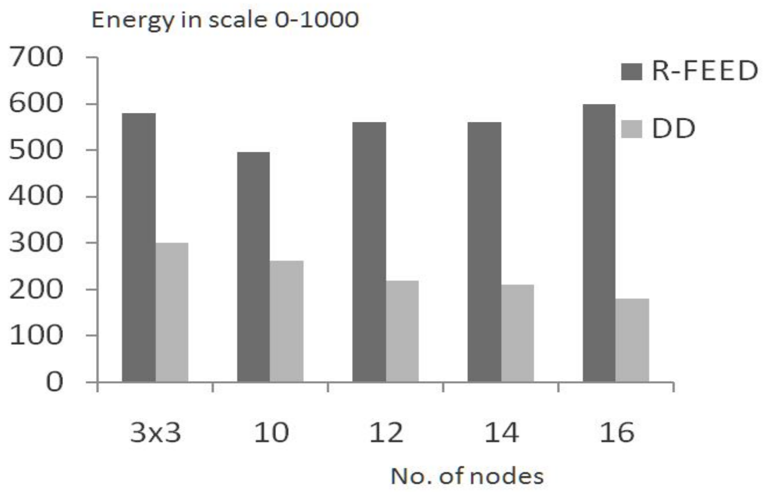

Figure 27: Comparison of Minimum energy of node in selected path

As we can see in above results that R-FEED performs well in choosing an optimal path from a given sensor network with minimized overhead. Most of the times we ran our simulation we noted approximately 20% increase on energy per node of selected path and 30% increase in minimum energy of a RED node in selected path; both of which are benign. Although, these provide energy efficiency in **actual data transfer**; choosing the one phase push model provides energy efficiency in **exploratory work** by reducing the number of control packets transferred.

CHAPTER – 4

CONCLUSIONS AND SCOPE FOR FUTURE RESEARCH

I've mentioned the preliminary protocols that were designed to meet with the energy constraints in WSN. These are discussed in the portion of data dissemination and MAC protocols. SPIN protocol introduced the concept of *negotiation*, hence it is considered as a revolutionary protocol. Afterward, this concept of negotiation was utilized in directed diffusion accompanying the idea of *interest.* Both of these protocols resulted in a massive improvement in performance as compared to traditional flooding and gossiping protocols. Then moving towards MAC protocols that were specially designed for wireless sensor network i.e. keeping energy efficiency as paramount factor. Although many works are proposed in wireless sensor networks, but it still lacks standardization. An efficient protocol that can be globally accepted is still a topic for researchers.

As specified in [9], SPIN protocol need to be used with fusion within the MAC layer, to take the benefits of MAC-layer broadcast. Also SPIN was not really proficient in terms of resource adaptability, as SPIN-2 takes decisions only on the basis of communication cost and ignores any consideration to synthesizing cost. In directed diffusion, there are different scopes at different elements. In data propagation, different route selection on the basis of QoS. In interest propagation, initially flooding was used, which can easily be replaced with other paradigm like directional propagation using cache data. In reinforcement, concepts of negative reinforcement and its mechanisms are needed to be applied.

In case of MAC layer protocols, the issue still is that no any protocol has been marked as a standard, but the protocols are in execution because of presence of application dependency. Wireless network communication also shows that taking only data link layer in consideration may lead to inappropriate conclusion of system performance. So, other layers should also be considered in making any decision within a network. Infact, in sensor network layered approach should be avoided, as layer interfacing leads to more overhead on the nodes which results in more energy consumption. Rather in future, protocols should be developed that work in *integrated-layer environment*.

In this report we started with a brief introduction of wireless sensor networks and its applications. Thereafter we explored Directed Diffusion which is one of the most widely accepted paradigms in wireless sensor networks. We explored the R-FEEDD protocol and shown the technique for increasing its domain, hence the title- “improved R-FEEDD”. We were able to verify the enhancement to be positive in terms of energy efficiency and the performance. Although, R_FEEDD gives positive results, still there is a need to evaluate the effect of choosing random numbers in a network of large number of nodes.

REFERENCES

[1] Johannes Gehrke, Ling Liu, Sensor Network Application, IEEE Computer Society 1089-7801/06 © 2006 IEEE

[2] Ismail Ahmedy, Md. Asri Ngadi, Syaril Nizam Omar, Junaid Chaudhary, A review in WSN, ISSN 1992-2248 © 2011 Academic Journals

[3] Sanjay Kumar Padhi, Prasant Kumar Pattnaik, B. Putthal, Review of Routing Protocols in Sensor and Adhoc Networks, ISSN: 2076-3328, International Journal of Reviews in Computing, © 2009-2010 IJRIC. All rights reserved

[4] G. J. Pottie and W. J. Kaiser Embedding the internet: wireless integrated network sensors. Communications of the ACM, 43(5):51–58, May 2000.

[5] Luwei Jing, Feng Liu, Yuling Li, Energy saving routing algorithm based on SPIN protocol in WSN, ISBN 978-1-61284-879-2, © 2011 IEEE.

[6] Ilker Demirkol, Cem Ersoy, and Fatih Alagöz,MAC Protocols for Wireless Sensor Networks, IEEE Communications Magazine • April 2006, 0163-6804/06 © 2006 IEEE.

[7] Abdelmalik Bachir, Mischa Dohler, Senior Member, IEEE, Thomas Watteyne, Member, IEEE, and Kin K. Leung, Fellow, IEEE, MAC Essentials for Wireless Sensor Networks, IEEE Communications Surveys & Tutorials, VOL. 12, NO. 2, Second Quarter 2010, 1553-877X/10 ©2010 IEEE.

[8] Mark A. Perillo and Wendi B. Heinzelman, Wireless Sensor Network Protocols, Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), 2004.

[9] Y. C. Tay, K. Jamieson, and H. Balakrishnan, "Collision-Minimizing CSMA and Its Applications to Wireless Sensor Networks," IEEE J Sel. Areas Commun., vol. 22, no. 6, pp. 1048–1057, 2004.

[10] J.Kulik, W.R. Heinzelman, H. Balakrishnan: Negotiation–Based Protocols for Disseminating Information in Wireless Sensor Networks. In: Wireless Networks, Vol 8, pp. 169-185, 2002.

[11] Michael Fuchs, Christophe Diot, Thierry Turletti and Markus Hofmann, A NamingApproach for ALF Design.

[12] Zeenat Rehena, Sarbani Roy, Nandini Mukherjee, A Modified SPIN for Wireless SensorNetworks, 978-1-4244-8953-4/11©2011 IEEE.

[13] C. Intanagonwiwat, R. Govindan, and Estrin D. Directed diffusion: A scalable and robustcommunication paradigm for sensor networks. In Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks (MobiCom), 2000.

[14] P. Karn, “MACA - A New Channel Access Method for Packet Radio,” in 9th ARRL Computer Networking Conference, London, Ontario,Canada, 1999.

[15] Y. Wei, J. Heidemann, and D. Estrin. An energy efficient mac protocol for wireless sensornetworks. In Proceedings of the Twenty-First International Annual Joint Conference of the IEEEComputer and Communications Societies (INFOCOM), 2002.

[16] T. van Dam and K. Langendoen. An adaptive energy-efficient mac protocol for wirelesssensor networks. In Proceedings of the First ACM Conference on Embedded Networked SensorSystems (SenSys), 2003.

[17] G. Lu, B. Krishnamachari, and C. S. Raghavendra, “An Adaptive Energy-Efficient andLow-Latency MAC for Data Gathering in Wireless Sensor Networks,” Proc. 18th Int’l. Paralleland Distrib. Processing Symp, Apr. 2004, p. 224

[18] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna Aceves, “Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks,” Proc. ACM SenSys ‘03, Los Angeles,CA, Nov. 2003, pp. 181–92.

[19] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, “Optimizing Sensor Networks inthe Energy-Latency-Density Design Space,” IEEE Trans. Mobile Comput., vol. 1, no. 1, pp.70–80, January-March 2002.

[20] Lajevardi, A., Haghighat, A.T., Eghbali, A.N. Extending directed diffusion routingalgorithm to support sink mobility in wireless sensor networks, Communications (MICC), 2009IEEE 9th Malaysia International Conference ISBN: 978-1-4244-5531-7 © 2009 IEEE.

[21] John Heidemann Fabio Silva Chalermek Intanagonwiwat, Ramesh Govindan DeborahEstrin, Deepak Ganesan, Building Efficient Wireless Sensor Networks with Low-Level Naming,SOSP 2001, Lake Louise, Banff, Canada Copyright 2001 ACM.

[22] Introduction to Wireless Sensor Networks, Information Processing and routing in WSN ©World Scientific publishing Co. Pte. Ltd

[23] Ilker Demirkol, Cem Ersoy, and Fatih Alagöz, MAC Protocols for Wireless SensorNetworks: A Survey.

[24] Giuseppe Calí, Ileana Ghizd vescu, Anders Grauballe, Mikkel Gade Jensen, Fabio Pozzo,MAC Protocol for Wireless Sensor Networks, Spring 2007 Institute for electronic systemsCommunication systems.

[25] Pavlos Papageorgiou, Literature Survey on Wireless Sensor Networks.

[26] Jinglun Shi, Zhilong Shan, and Xuxun Liu, A Two-Level Routing Scheme for WirelessSensor Network, International Journal of Distributed Sensor Networks Volume 2012 (2012),Article ID 601389.

[27] Yun Li, Hao Jia, Zhan-jun Liu, Zhi Ren, Guo-jun Li, A MAC Protocol Based on S-MAC forPower Asymmetric WSN Network, ISBN: 978-1-4244-3746-7 © 2008 IEEE.

[28] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin, John Heidemann, Member,IEEE, and Fabio Silva, Member, IEEE, Directed Diffusion for Wireless Sensor Networking,1063-6692/03 © 2003 IEEE.

[29] Jamal N. Al-Karaki, Ahmed E. Kamal, Routing Techniques in WSN: A Survey, IEEEWireless Communications • December 2004, 1536-1284/04 © 2004 IEEE.

[30] K. Akkaya, M. Younis: A Survey on Routing Protocols for Wireless Sensor Networks. In:Elsevier Ad Hoc Network Journal, Vol. 3, pp. 325-349, 2005.

[31] I.F. Akyildiz, M.C. Vuran, O. Akan, W Su: Wireless Sensor Networks: A Survey Revisited.In: Computer Networks Journal (Elsevier Science), Vol. 45, no. 3, 2004.

[32] J.N. Al-Karaki, A.E. Kamal: Routing Techniques in Wireless Sensor Networks: A Survey.In: IEEE Wireless Communications, Vol. 11, no. 6, pp. 6-28, 2004.

[33] F. Zabin, S. Misra, I. Woungang, H.F. Rashvand, N.W. Ma, M. A. Ali: REEP: data-centric,energy-efficient and reliable routing protocol for wireless sensor networks. In: IET Commun.Vol. 2, No. 8, pp. 995-1008, 2008

[34] C.F. Garcia-Hernandez, P.H. Ibargüengoytia-González, J. Garcia-Hernandez, J.A. Pérez-Díaz: Wireless Sensor Networks and Applications: a Survey. In: IJCSNS, International Journalof Computer Science and Network Security, Vol.7 no.3, 2007.

[35] P.Levis and N.Lee: TOSSIM: A Simulator for TinyOS Networks, Included with the TinyOS1.1.0 software. 2003.

[36] S. S. Kulkarni, “TDMA Services for Sensor Networks,”Proc. 24th Int’l. Conf. Distrib.Comp. Sys. Wksps. , Mar.2004, pp. 604–09.

[37] Srivastava, N., Seth, S., Kumar, M. and Kumar, R., 2013. R-FEEDD: A Randomized Further Energy Efficient Directed Diffusion for Wireless Sensor Networks.