An Efficient method for the advancement of performance of wireless sensor network

Priyanka M. Lokhande

Abstract — In this paper, the main focus or emphasis is given on the upgradation of the performance of the wireless sensor network or it may be a general wireless network. The main issue is towards the monitoring of the delay issue & network lifetime issue. Wireless sensor network also suffer from the different problems or vulnerabilities. WSN suffer from attacks due to which there is an impact on the performance of the wireless sensor network. Our main aim in this paper is towards the advancement in the performance of the wireless network. Also towards the maximization of network lifetime & minimization of delay term. Wireless sensor networks consist of many sensing nodes that captures the changes in the environment enclose data in data packets and receives these packets to sink node present in the network. A WSN consists of a number of sensors spread across a geographical area. Each sensor has wireless communication capability and sufficient intelligence for signal processing and networking of the data. The network lifetime is usually defined as the time until the first node fails because of energy depletion. So sleep-wake scheduling is effective mechanism to increase network lifetime. Sleep-wake scheduling is efficient to increase network lifetime but it could result in substantial delays because a transmitting node needs to wait for its next-hop relay node to wake up. Anycast forwarding schemes are used to forward the data packet to next hop node which minimizes the expected packet-delivery delays from the sensor nodes to the sink node. CMAC avoid the synchronization while supporting the latency.

Index Terms — WSN, Security in network, Delay, lifetime, anycast, sleep wake scheduling, CMAC, routing.

1 INTRODUCTION

This paper highlights on the important issue of the network system. For any system which is general system or computer system or here in this paper there is a consideration of the wireless sensor network. The system performance is ultimately depend on its speed, accuracy, reliability. The network system performance gets affected due to some attacks or some vulnerabilities or some security issues or it may be security threat to the system, which is ultimately degrades the performance of the wireless network. Our main focus in this paper is given on two issues first is the maximization of network lifetime & second issue is the minimization of delay term. Also there is a some issue related to network design & routing. A wireless sensor network is an infrastructure less, self organizing communication network, in which resource constrained sensors individually sense the environment but collaboratively achieve complex information gathering and dissemination tasks. Wireless sensor networks (WSN) are used to remotely sense the environment. Researchers see WSNs as an “exciting emerging domain of deeply networked systems of low-power wireless nodes with a tiny amount of CPU and memory, and large federated networks for high-resolution sensing of the environment”. Sensors in a WSN have a variety of purposes, functions, and capabilities A sensor network is composed of a large number of sensor nodes that are densely deployed. To list just a few venues, sensor nodes may be deployed in an open space. There are four sources of synchronized scheduling have been proposed in [1,2]. The concept of sleep wake scheduling is used to increase the network lifetime of event driven sensor network. Asynchronous sleep wake scheduling is used where each node is used to wake independent to save energy but to this process additional delay is encountered at each node along path to sink node because each node has to wait for next node before the transmission of packets. Anycast packet forwarding is used to control over this. Sleep wake scheduling is the important method which is used to increase the network lifetime. In many WSN applications, the deployment of sensor nodes is performed in an ad hoc fashion without careful planning and engineering. Once deployed, the sensor nodes must be able to autonomously organize themselves into a wireless communication network. Sensor nodes are battery-powered and are expected to operate without attendance for a relatively long period of time. In most cases it is very difficult and even impossible to change or recharge batteries for the sensor nodes. The traditional routing protocols have several shortcomings when applied to WSNs [3]. The design of routing protocols for network is challenging.

1.1 Network Design objectives

Many sensor networks are application oriented and have different requirement towards the applications. Following design...
Objectives are taken into consideration when designing the sensor network.

- Small node size: Since sensor nodes are usually deployed in a harsh or hostile environment in large numbers, reducing node size can facilitate node deployment. It will also reduce the power consumption and cost of sensor nodes.
- Low node cost: Since sensor nodes are usually deployed in a harsh or hostile environment in large numbers and cannot be reused, reducing cost of sensor nodes is important and will result into the cost reduction of whole network.
- Low power consumption: Since sensor nodes are powered by battery and it is very difficult or even impossible to charge or recharge their batteries, it is crucial to reduce the power consumption of sensor nodes so that the lifetime of the sensor nodes, as well as the whole network is prolonged.
- Scalability: Since the number sensor nodes in sensor networks are in the order of tens, hundreds, or thousands, network protocols designed for sensor networks should be scalable to different network sizes.
- Reliability: Network protocols designed for sensor networks must provide error control and correction mechanisms to ensure reliable data delivery over noisy, error-prone, and time-varying wireless channels.
- Self-configurability: In sensor networks, once deployed, sensor nodes should be able to autonomously organize themselves into a communication network and reconfigure their connectivity in the event of topology changes and node failures.
- Adaptability: In sensor networks, a node may fail, join, or move, which would result in changes in node density and network topology. Thus, network protocols designed for sensor networks should be adaptive to such density and topology changes.
- Channel utilization: Since sensor networks have limited bandwidth resources, communication protocols designed for sensor networks should efficiently make use of the bandwidth to improve channel utilization.
- Fault tolerance: Sensor nodes are prone to failures due to harsh deployment environments and unattended operations. Thus, sensor nodes should be fault tolerant and have the abilities of self-testing, self-calibrating, self-repairing, and self-recovering.
- Security: A sensor network should introduce effective security mechanisms to prevent the data information in the network or a sensor node from unauthorized access or malicious attacks.
- Quality-of-service support: In sensor networks, different applications may have different quality-of-service (QoS) requirements in terms of delivery latency and packet loss. Thus, network protocol design should consider the QoS requirements of specific applications.

1.2 Structure of Wireless sensor node

The sensor networks are mainly of two types - event-driven and time-driven (or continuous dissemination networks). In the event driven networks the sensor nodes sense the data and transmit it only if the data is considered critical enough to be communicated. In the time driven networks sensors sense the data and transmit it to the central controller periodically. The periodicity of relaying data packets is application dependent. The coverage area of a sensor node (or the approachability of a node) depends on its transmission range.

A sensor node basically consists of four basic components; a sensing unit, a processing unit, a communication unit and a power unit. Digital converters (ADC). Sensors observe the physical phenomenon and generate analog signals based on the observed phenomenon. They are the actual interface to the physical world. ADCs convert the analog signals into digital signals which are then fed to the processing unit. Processing unit usually consists of a microcontroller or microprocessor with memory which provides intelligent control to the sensor node (example Intel's Strong ARM microprocessor and Atmel's AVR microprocessor).

Communication unit consists of a short range radio for performing data transmission and reception over a radio channel. The power unit consists of a battery for supplying power to drive all components in the system. All these units should be built into a small module with low power consumption and low production cost. Each sensor performs two main responsibilities, namely

(i) sensing activities, and
(ii) Routing the sensed data to the sink. The main responsibility of sink is to collect information from various sensor nodes and process it for further actions.

Routing protocols for other wireless networks like mobile ad-hoc networks or cellular networks cannot be directly applied to WSNs due to the existing design challenges in WSNs like energy consumption, node deployment, QoS, data aggregation and node mobility. Energy dissipation at sensor node is a major concern, as in many applications sensors have to be deployed in inaccessible environments. Sensing alone is not an energy consuming activity, but networking and programming certainly are.

Prolonging battery life in sensor nodes and, by extension, the overall network lifetime is therefore a foremost task in the de-

* Priyanka M.Lohande received her Bachelor degree in engineering in Information Technology from Sant Gadge Baba Amravati University, Amravati in 2010. She also received Master degree in Computer science & Engg from Sant Gadge Baba Amravati University, Amravati. In 2013, she is currently working as an assistant professor in Comp sci & Engg Dept at Jawaharlal Darda Institute of Engg & Tech, Yavatmal (MS) India. Her area of interest is in network security. E-mail:priyanka20ht@gmail.com.
sign of practical WSNs. Another requirement of WSNs for applications such as tracking of intruders, detection of fire etc. is that the delay to transmit data from sensor node to the sink should be as low as possible. Network lifetime is defined as the time elapsed until the first node (or the last node) in the network depletes its energy (dies). Basic sensor node components consist of processing unit, power unit, communication unit & sensing unit.

1.3 CMAC: MAC Layer Protocol Using Convergent Packet Forwarding for Wireless Sensor Networks

CMAC avoids synchronization overhead while supporting low latency. By using zero communication when there is no traffic, CMAC allows operation at very low duty cycles. When carrying traffic, CMAC first uses anycast to wake up forwarding nodes, and then converges from route-suboptimal anycast with unsynchronized duty cycling to route-optimal unicast with synchronized scheduling [10]. Duty cycling the radio is important to achieve long lifetime in wireless sensor network, but it usually causes performance degradation in throughput and latency which are critical metrics for various applications such as event tracking and surveillance. These conflicting objectives motivate our design of a new MAC layer protocol called Convergent MAC (CMAC). While transmitting packets, CMAC first uses aggressive RTS to anycast packets to potential forwarders which wake up first and detect the traffic using double channel check. Once the sender is able to transmit packets to a node with acceptable routing metric, CMAC converges from anycast forwarding to unicast to avoid the overhead of anycast. It include three steps aggressive RTS, Anycast based forwarding, conversion from anycast to unicast.

In fig 2 example, node C might be the earliest to wake up, followed by B and then by node A. Since A is already in the best region, the transmitter starts to unicast to A regardless if there is any other sleeping node in R1 with greater progress. However, it is possible that there is no node in region R1 in Fig. 2. Hence if the transmitter cannot find a better next hop than the current one after a duty cycle length, it switches to uncasting. In this way, the packet forwarding converges from anycast to unicast for each link. After some time without successful data packet reception, CMAC will timeout and nodes will again start following unsynchronized idle duty cycles.

1.4 Drouting Protocol

GeRaF was proposed by Zorzi and Rao [16], is the deterministic routing (Drouting) protocol which uses geographic routing where a sensor acting as relay is not known a priori by a sender. There is no guarantee that a sender will always be able to forward the message toward its ultimate destination, that is, the sink. This is the reason that GeRaF is said to be best-effort forwarding. GeRaF assumes that all sensors are aware of their physical locations, as well as that of the sink.

2 BACKGROUND

Extensive research has been done in the wireless network area. Various sleep-wake scheduling protocol have been proposed in literature.

In synchronized sleep–wake scheduling protocols [1-2] sensor nodes periodically or aperiodically exchange synchronization information with neighboring nodes. However, such synchronization procedures could incur additional communication overhead and consume a considerable energy.

On-demand sleep–wake scheduling protocols [6]-[7] is one scheduling where nodes turn off most of their circuitry and always turn on a secondary low-powered receiver to listen to “wake-up” calls from neighboring nodes when to relay the packet. But this on-demand sleep–wake scheduling can significantly increases sensor nodes cost due to the additional receiver.

In this, we are interested in asynchronous sleep–wake scheduling protocols in which each node wakes up independently.
of neighboring nodes for energy saving. However, this independence of the wake-up processes causes additional delays at each node along the path to the sink because each node needs to wait for its next-hop node to wake up before it can transmission of the packet. This delay could be unacceptable for delay-sensitive applications which require the event reporting delay to be very small. So for minimizing this event reporting delay, anycast packet forwarding technique is used.

In traditional packet-forwarding schemes, every node has one designated next-hop relaying node in the neighborhood, and it has to wait for the next-hop node to wake up when forward a packet has to be done. In contrast, under anycast packet-forwarding schemes, each node has multiple next-hop relaying nodes in a candidate set (we call it as forwarding set) and forwards the packet to the first node that wakes up in the forwarding set. It is easy to see that, compared to the basic scheme; anycast clearly reduces the expected one-hop delay. But anycast does not necessarily lead to minimum end to end delay because packet has to relay through time consuming path. Therefore to reduce this expected end to end delay, one challenge is how each node chooses its anycast forwarding policy like forwarding set this is to be solved and another important concept here is that implementing anycast in isolation does not give good performance it has to be jointly controlled with parameters of sleep scheduling like wake up rate. Anycast addresses these challenges.

This technique shows how to use the solution to the delay-minimization problem to construct an optimal solution to the lifetime-maximization problem for a specific definition of network lifetime. However, anycast does not necessarily lead to the minimum expected end-to-end delay because a packet can still be relayed through a time-consuming routing path. Therefore, the first challenge for minimizing the expected end-to-end delay is to determine how each node should choose its anycast forwarding policy (e.g., the forwarding set) carefully. Study of [8] proposes heuristic anycast protocols that exploit the geographical distance to the sink node. The work in [9] considers MAC-layer anycast protocols that work with the separate routing protocols in the network layer. However, these solutions are heuristic in nature and do not directly minimize the expected end-to-end delay. The algorithms in [10] use the hop-count information (i.e., the number of hops for each node to reach the sink) to minimize some state-dependent cost (delay) metric along the possible routing paths. However, these algorithms do not directly apply to asynchronous sleep–wake scheduling, where each node does not know the wake-up schedule of neighboring nodes when it has a packet to forward.

The second challenge stems from the fact that good performance cannot be obtained by studying the anycast forwarding policy in isolation. Rather, it should be jointly controlled with the parameters of sleep–wake scheduling (e.g., the wake-up rate of each node). It will directly impact both network lifetime and the packet-delivery delay. Hence, to optimally tradeoff network lifetime and delay, both the wake-up rates and the anycast packet-forwarding policy should be jointly controlled. However, related work is studied in the literature [11]–[12] but it is not jointly controlled.

3 ANALYSIS OF PROBLEM

In the section of problem definition, Anycast packet forwarding should be implemented along with sleep scheduling here in order to get good performance. Two main challenges that has to be addressed in enhancing the performance of event driven wireless sensor network are:

A] Delay-minimization problem: with the wakeup rates of the sensor nodes, optimally choosing the anycast forwarding policy to reduce the expected end to end delay from all sensor nodes to the sink.

B] Lifetime maximization problem: with constraint on the expected end-to-end delay, how to maximize the network lifetime by jointly controlling the wake-up rates and the anycast packet-forwarding policy.

Sleep-wake scheduling with anycast intend to solve these two main problems. The lifetime of event driven sensor network consists of two phases’ configuration phase and operation phase. In configuration phase, node optimizes control parameters of anycast forwarding policy and their wakeup rate. After configuration phase, operation phase begins in which each node alternates between two subphases sleeping subphase and event reporting subphase.

This Technique assume that the sensor network employs asynchronous sleep–wake scheduling to improve energy efficiency, and nodes choose the next-hop node and forward the packet to the chosen node using the following basic sleep–wake scheduling protocol.

This basic protocol generalizes typical asynchronous sleep–wake scheduling protocols in to account for anycast. In this basic protocol, we assume that there is a single source that sends out event-reporting packets to the sink. This is the most likely operating mode because when nodes wake up asynchronously and with low duty-cycles, the chance of multiple sources generating event-reporting packets simultaneously is small.

Apart from these problems there is an again different problem in WSN it may be problem related to the network design, routing problems, again some problems related to the security of the wireless sensor network.

4 SIMULATION RESULT & DISCUSSION

a] Anycast and sleep wake scenario with 30 nodes
Node 1 observes changes in environment and generates data packet. It has 6 nodes in its forwarding set, so node 1 sends beacon and ID signal to its neighboring nodes. At that time, any node in the forwarding set wakes up, i.e., node 3 wakes up. Then node 1 sends a packet to it. The same policy will be followed by node 3, and finally, the packet will reach the sink node 15. Remaining packets will follow the same path followed by the first packet.

Now after some time, node 29 becomes the source node and will send the packet to the sink 15. This is the second source node. The performance of Anycast and sleep wake is compared with two previous methods: CMAC and Deterministic Routing algorithm. The simulation result of CMAC is as below.

As shown, the source node will first send the packet to node 10 because it wakes up first. Then, when its best geographical routing node 3 wakes up, it switches from anycast to unicast and gives all packets to node 3. This process continues till packets reach to sink node 15. As packet traverse from more number of nodes, energy as well as delay required will be more in CMAC routing protocol. Also, its forwarding set concept does not consider sleep wake rate in consideration, so its performance decreases.
In drouting algorithm, the fixed routing path is already decided depending on protocol's performance metric e.g. geographical distance.

So when node 1 generates packet it has to wait for next hop node 3 to wake up then at node 3 it has to wait for node 24 to wake up and so on so there is considerable amount of delay at each hop which greatly increases end to end delay.

This is end to end delay for first packet during each set of transmission. Graph shows total simulation time on x-axis while delay on y-axis. When first node in the forwarding set wakes up then source will send packet to it so packet has to still relayed through time consuming path. This delay required will be different for each different set of data packets. Simulation result shows that it requires maximum delay i.e. 300 ns than other two protocols.

As shown in above graph it is clear that energy distribution in network is not uniform means there is unequal distribution of energy. Some nodes have too less energy while some having enough energy. Energy distribution of nodes is in the range of
11.0 to 15.3 joule for complete network node. As this includes both the concept of anycast and unicast, data packet traverse from more number of nodes so more energy is consumed during each transmission.

In drouting protocol, there is fixed deterministic path from source to sink so each packet has to wait for its next hop node to wake up before transmitting the packet. Energy consumption is greater than proposed solution because deterministic path is determined by considering geographical distance as routing metric so current energy of nodes will not be taken into consideration. Energy distribution is in the range of 12.4 to 15.2 joule which is greater than proposed solution so more energy is consumed for packet transmission.

In drouting protocol, as packet has to always wait for its next hop node the one hop delay increases as a result expected end to end delay is also much more than proposed anycast protocol. Delay required for packet transmission is highest i.e. 310 ns for drouting protocol.

This is combined energy distribution graph for proposed anycast with drouting and CMAC protocols. CMAC has unequal distribution of node energy in network its peaks are more fluctuating. Proposed sleep wake and anycast scheduling technique requires lowest energy of node than other two protocols so energy remains conserved in network and thus lifetime of wireless sensor node increases.
This is combined delay distribution graph for proposed anycast with drouting and CMAC protocols. CMAC requires 255 ns proposed sleep wake and anycast scheduling technique requires lowest delay to sink node i.e. 205 ns while drouting protocol requires highest delay to transmit packet to sink node 310 ns. Thus from simulation results, we proved the performance of sleep wake and anycast scheduling technique. It increases the lifetime of energy constrained wireless sensor network and minimizes the delay to transmit packet from source to sink node which is very important for delay sensitive applications.

5 CONCLUSION

In this paper, an anycast packet-forwarding Scheme is introduced to reduce the event-reporting delay and to prolong the lifetime of wireless sensor networks employing asynchronous sleep–wake scheduling. Event driven wireless sensor network with low data rate and delay sensitive applications requirements are focused. Specifically, study of two optimization problems is done. First, when the wake-up rates of the sensor nodes are given, an efficient and distributed algorithm is developed to minimize the expected event-reporting delay from all sensor nodes to the sink. Second, using a specific definition of the network lifetime, study of lifetime-maximization problem is done to optimally control the sleep–wake scheduling policy and the anycast policy in order to maximize the network lifetime subject to an upper limit on the expected end-to-end delay. Numerical results suggest that the proposed solution can substantially outperform prior heuristic solutions in the literature under practical scenarios where there are obstructions in the coverage area of the wireless sensor network. For future work, plan to generalize this solution to take into account non-Poisson wake-up processes and other lifetime definitions. Also consideration of more than one sink node can be done in future. Our main future work will stress on the advancement of the performance of the WSN & secure the network from different problems & make reliable network.

REFERENCES