International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249 – 8958, Volume-1 Issue-5, June 2012

Energy Efficient Conflict Free Query Scheduling For Wireless Sensor Networks

Pooja Kannadas, S. Joshua Daniel

Abstract -There is an increase in demand of high performance query services, with the emergence of high data rate applications. To meet this challenge we propose Dynamic Conflict-free Query Scheduling (DCQS), a novel scheduling technique for queries in wireless sensor networks. In contrast to earlier Time Division Multiple Access (TDMA) designed for query services in wireless sensor networks. DCQS has several unique features. First, it optimizes the query performance through conflict-free transmission scheduling based on the temporal properties of queries in wireless sensor networks. Second, it can adapt to workload changes without explicitly reconstructing the transmission schedule. Furthermore, DCQS also provides predictable performance in terms of the maximum achievable query rate. The nodes operate over the time-varying wireless channel whose quality significantly fluctuates over time due to fading and interference. Such time-varying nature of wireless channel imposes many constraints in designing an energy-efficient transmission scheme. In this work, we derive a tight bound on the maximum query rate achieved under DCQS. Such a bound is of practical importance since it can be used to prevent network overload. NS2 simulations demonstrate that energy efficient DCQS significantly outperforms 802.11 in terms of energy efficiency, over head, query latency, and throughput, thereby increasing the network life time.

Keywords: DCQS, TDMA, NS2

I. INTRODUCTION

Wireless Sensor Networks (WSN) is often used for real time applications [1], such as environment surveillance, medical care and traffic control. The emergence of high data rate sensor network applications has resulted in an increasing demand for high-performance query service. The key challenge was to provide a high throughput query service that can collect data from large networks and adapt to workload changes. To meet this challenge, we propose Dynamic Conflict-Free Query Scheduling (DCQS), an integrated frame work for transmission scheduling designed to meet the communication needs of high data rate applications. A data collection application may express its collection interests as queries over subset of nodes, which may involve data aggregation.

These queries have to be executed periodically to collect data at the base station. The use of routing trees in executing query instances introduces precedence constraints among packet transmission. DCQS assumes a common query model in which source nodes produce data reports periodically.

This model fits many applications that gather data from the environment at user-specified rates. Such applications generally rely on existing query services [2].

In addition to that we also propose an energy efficient transmission strategy for wireless sensor networks, which is normally operated in energy constrained environments. Here we use the Markov decision process, from which we obtain the optimum threshold for a successful transmission. A plan is a sequence of steps, each comprised of a set of conflict-free packet transmissions. DCQS executes a plan sequentially by performing the transmissions assigned to each step. Upon the completion of a plan execution, the data reports from all sources involved in the query would have been delivered to the base station. DCQS could accomplish this by executing instances one at a time as they are released according to their constructed plans. DCQS provides not only better performance than the traditional transmission scheduling techniques, but have some additional features. They include that, DCQS can adapt to workload changes without having to recompute its transmission schedule. This is accomplished by dynamically determining the transmissions to be executed in each slot. DCQS also have low runtime over head and limited memory requirements making it suitable for resource constrained devices. A key advantage of DCQS is that it has a known capacity bound in terms of the maximum query completion rate thereby preventing the over load.

A. Architecture of Wireless Sensor Networks:

A Wireless Sensor Network (WSN) consists of spatially distributed autonomous sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants. Early research on wireless sensor networks (WSNs) has focused on low data rate applications, such as habitat monitoring [3]. In contrast, recent years have seen the emergence of high data rate applications, such as real-time structural health monitoring [4] and preventive equipment maintenance [5].The development of wireless sensor networks was motivated by military applications such as battlefield surveillance. They are now used in many industrial and civilian application areas, including industrial process monitoring and control, machine health monitoring, environment and habitat monitoring, healthcare applications, home automation, and traffic control.

A wireless sensor network is a collection of nodes organized in a network. Each node consists of one or more microcontrollers, CPUs or DSP chips, a memory and a RF transceiver, a power source such as batteries and accommodates various sensors and actuators. In addition to one or more sensors, each node in a sensor network is typically equipped with a radio transceiver or other wireless communications device, a small microcontroller,

Manuscript Received on June 22, 2012

Pooja Kannadas, M.E. Student, Department of EEE, Hindusthan College of Engineering and Technology, Coimbatore- 641 032, India

S.Joshua Daniel, Assistant Professor, Department of EEE, Hindusthan College of Engineering and Technology, Coimbatore- 641 032, India

and an energy source, usually a battery. A sensor node might vary in size from that of a shoebox down to the size of a grain of dust. The cost of sensor nodes is similarly variable, ranging from hundreds of dollars to a few pennies, depending on the size of the sensor network and the complexity required of individual sensor nodes.

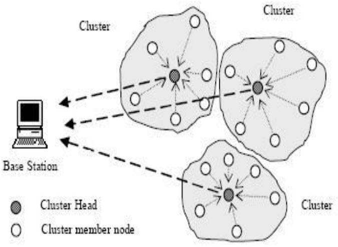


Fig. 1 Architecture of Wireless Sensor Networks

Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and bandwidth. A sensor network normally constitutes a wireless ad-hoc network, meaning that each sensor supports a multi-hop routing algorithm.

However sophisticated applications require preprocessing of the data to extract important information so that transmission bandwidth can be preserved by simply transmitting the essential information (e.g., alerting the operator of a critical event). Local processing capability is also important for applications in which the sensor supports bidirectional communication. In these cases, the users can query the sensor either for status or for a history of previous samples of data. The communication module consists of a short-range radio transceiver.

B. Energy Constraints

The design of each system component can be optimized to minimize energy consumption. Energy consumption occurs in three aspects: sensing, communication, and data processing. Algorithmic modifications can often result in significant energy savings. Usually the communication of data consumes much more energy than sensing and data processing. Therefore, highly localized and distributed solutions for different levels of communication protocols are required.

Energy is the scarcest resource of WSN nodes, and it determines the lifetime of WSNs. WSNs is meant to be deployed in large numbers in various environments, including remote and-hostile regions, with ad-hoc communications as key. Therefore the algorithms and protocols are designed with the following features.

- Lifetime maximization
- Robustness and fault tolerant
- Self-configuration

The algorithms implemented in this paper improve 70% with respect to the plain 802.11 while achieving a comparable throughput.

II. PROTOCOL DESIGN

A. Dynamic Conflict-Free Query Scheduling

Dynamic Conflict-free Query Scheduling (DCQS), an integrated framework for transmission scheduling designed to meet the communication needs of high data rate applications. A data collection application may express its collection interests as queries over subsets of nodes which may involve data aggregation [9]. DCQS separates the problem of conflict-free transmission in two parts. Initially we consider the problem of scheduling each query instance in isolation when all the network resources are dedicated to its execution. Next we consider the execution of queries submitted by the user. These queries can be executed one at a time as they are released according to their constructed plans. DCQS can support queries with in-network data aggregation, such as average and histogram, as well as more common forms of aggregation such as packet merging and data compression [6]. While DCQS can optimize the performance of queries with aggregation, it also supports queries that do not perform aggregation. The working of the DCQS can be explained in the sequence of four steps.

1. When a new query is submitted by the user, DCQS identifies a plan for its execution. Normally many queries can be executed using the same plan. If no plan is reused then the planner constructs a new plan for executing the query.
2. The base station performs rate control to ensure that the total query rate remain within the maximum query rate under DCQS. If not the query rate is decreased proportionally, not to exceed the maximum query rate.
3. The phase, period and the aggregation function of the query are disseminated to all nodes.
4. At runtime, the scheduler executes all query instances.

DCQS dynamically determines the transmissions to be executed in each slot and, as a result, it may adapt to workload changes more effectively than traditional TDMA protocols and 802.11[6] with fixed transmissions schedules. DCQS has low runtime overhead and limited memory requirements making it suitable for resource constrained devices conflict-free transmissions are assigned.

B .The Centralized Planner

When the query involves aggregation, the plan must respect the precedence constraints introduced by aggregation: a node is assigned to transmit in a later step than any of its children. Note that DCQS does not impose any constraint on the order in which a node's children transmit.

Each node is assigned in sufficient steps to meet its workload demand a node to wait for data from its children even for queries that do not involve aggregation because this results in transmission schedules that have long contiguous periods of activity/inactivity. The node transitions from a sleep state to the active state just-in-time to receive the data from its children and transitions back to sleep after it completes collecting data from its children and relaying it to its parent.

Each node employs a local scheduler that schedules the transmissions of all instances. The state of the scheduler includes: the start time and period of all queries, the plan's length, and the minimum inter release time. Note that if all nodes have a consistent view of these parameters, they will construct independently the same schedule. The scheduler also knows the steps in which the host node transmits or receives. However, the scheduler does not need to know the specific steps in which any other nodes transmit or receive. The scheduler has two FIFO queues: a run and a release queue. The release queue contains all instances released but not being executed. The run queue contains the instances to be executed in the slot s. Although the run queue main y contains multiple instances, a node is involved in transmitting or receiving for at most one instance. Thus the scheduler is feasible to run on resource constrained devices, because it is very simple and efficient.

C. The Distributed Planner

A distributed planner uses only neighborhood information in constructing plans. Specifically, a node knows only its adjacent communication and interference edges. The minimum interrelease time of the global plan is the maximum of the minimum interrelease times of the local plans. This suggests that similar to the length of the plan, the global minimum interrelease time can be computed using in-network aggregation. In fact, the two may be computed concurrently. Once the aggregation process is complete, the root can compute the length, and minimum inter release time of the plan and then disseminate them to all nodes.

A node n constructs a plan in three stages: plan formulation, plan dissemination, and plan reversal. The formulation stage starts when a node n becomes the highest priority eligible node in its one-hop neighborhood. When this occurs, n broadcasts a Plan Request to gather information about transmissions which have already been assigned. To construct a conflict-free plan, n must know the steps in which its two-hop neighbors with higher priorities were assigned. Upon receiving the Plan Request from n, each one hop neighbor checks if there is a node in its own one-hop neighborhood that has a higher priority than n. If no such node exists, the receiver responds with a Plan Feedback packet containing its local plan. Otherwise, the node does not reply. After a time-out, node n will retransmit the Plan Request to get any missing Plan Feedback from its one-hop neighbors. Since all Plan Feedback are destined for n, to reduce the probability of packet collisions, nodes randomize their transmissions in a small window. Once n receives the Plan Feedback, it has sufficient information to assign its transmissions to its parent using the same method as the centralized planner. In the second stage, n disseminates its local plan to its one-hop neighbors via a Plan Send. Upon receiving a Plan Send, a node updates its plan and acknowledges its action via a Plan Commit. To ensure that DCQS constructs a conflict-free schedule, neighboring nodes must have consistent plans. We note that the distributed planner achieves this objective through retransmission when needed. If a Plan Feedback from a neighbor is lost, n assumes that a higher priority node has not yet been scheduled and retransmits the Plan Request until it has received Plan Feedback from each neighbor or reached the maximum number of retransmissions. Similarly, during the plan dissemination stage, node n retransmits the plan until all its neighbors acknowledge the correct reception of its Plan Send via a Plan Commit.

III. ALGORITHMS AND IMPLEMENTATION

As discussed early in this paper, to maximize the energy efficiency, the algorithm consists of two components: BDT and CBA. The nodes operate over the time-varying wireless channel whose quality significantly fluctuates over time due to fading and interference. Such time-varying nature of wireless channel imposes many constraints in designing an energy-efficient transmission scheme. For instance, a transmission attempt, when the wireless channel is temporarily bad, is highly likely to be failed and may lead to a waste of energy. To avoid this, the sender may wait until the channel becomes better. However, deferring the transmissions until the channel becomes better may decrease throughput, or equivalently cause a longer latency. This is a trade-off problem between energy efficiency and throughput. Thus an efficient transmission scheme for the WSNs must be able to adapt to variation of the wireless channel while maintaining a good balance between these two conflicting measures. In [26], a transmission scheme adopting multicast Ready-to-Send (RTS) and priority-based Clear-to-Send (CTS) was proposed to prioritize the terminal with a good channel in terms of channel access.

A. Binary Based Decision Algorithm

In this scheme the sensor node takes two actions, Transmit and Defer. As shown in the fig.2 the current channel condition is measured at the receiver side through the two frame RTS and data frame and is classified into two states, Good and Bad based on the received SNR. This information is notified back into the sender by sending the return frame, CTS or ACK frame. The SNR threshold used to classify the channel states is determined using the Markov Decision Process.

B. Channel aware back off adjustment

The channel aware back off adjustment algorithm can be explained as follows. Each and every sensor node maintain a table called link state table that lists up the channel states of each link to its neighbors and to validate their information. For each link a pair of channel, validity information is maintained. Online measurements are taken in order to set the channel coherence time. This is based on a priority based scheme, if the channel is good is now, it remains good in the near future. Such probability is realized by assigning different sizes of contention window (CW) to the sensor nodes based on the channel quality. The initial contention window is set according to the channel state

$$CW=\begin{cases}\alpha(t)CW, upon\ good\ channel\\ CW, upon\ \ meduim\ channel\\ \beta(t)CW, upon\ bad\ channel\end{cases}$$

where α(t) and β(t) are multiplicative constants for prioritizing the nodes whether the using node's channel condition is good or bad, t is the timer value, initially set to T, the validity period which is decremented over the time. In this paper we assume that α (t)=1/2 and β(t)= 3/2, unless stated otherwise.

IV. SIMULATIONS AND RESULTS

CSMA-based MAC protocols, such as used in 802.11, are known to have poor performance in heavy traffic situations and less energy efficient. Several measurement studies have documented this in real world scenarios. CSMA protocols are also not easily amenable to rigorous mathematical modeling for throughput and capacity. The problem with 802.11[6] is expected to be worse in mesh networks, where high capacity backbone links are needed and multihop interference plays a significant role.

The performance comparison between the MDCQS, DCQS and the 802.11 has been performed and observed the graph. The comparison is being done on the basis of the parameters, namely bandwidth, energy efficiency, overhead, throughput, and query latency.

A. Bandwidth

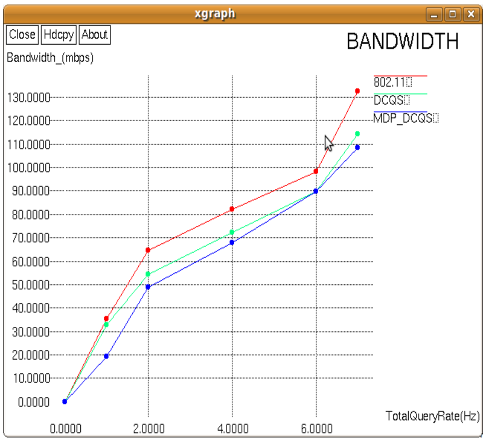
Bandwidth is defined as the data transfer rate in bytes. Figure 3.1 shows the graphical analysis of the bandwidth of 802.11, DCQS and MDCQS. The graphical analysis shows that bandwidth required for the MDCQS is small compared to the other protocols, thereby reducing the estimated time required for the transmission of queries in the wireless sensor networks.

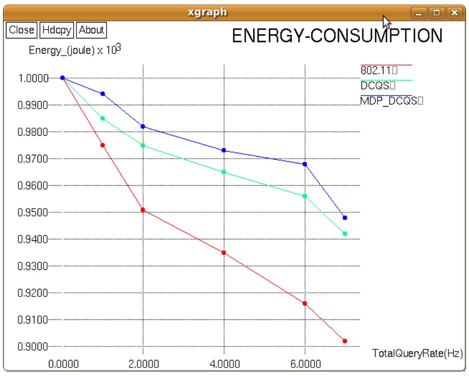
B. Energy Efficiency

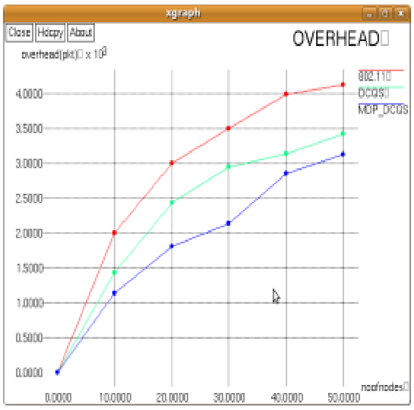
Energy efficiency expresses the lifetime of a sensor network. In the figure 3.2 comparison graph, the energy consumption is very less for MDCQS is very less when compared to the DCQS and MAC protocol.

C. Overhead

Overhead is expressed as a percentage of non application bytes divided by the total number of bytes in the message. Fig 3.2 shows the graphical comparison in terms of over head.







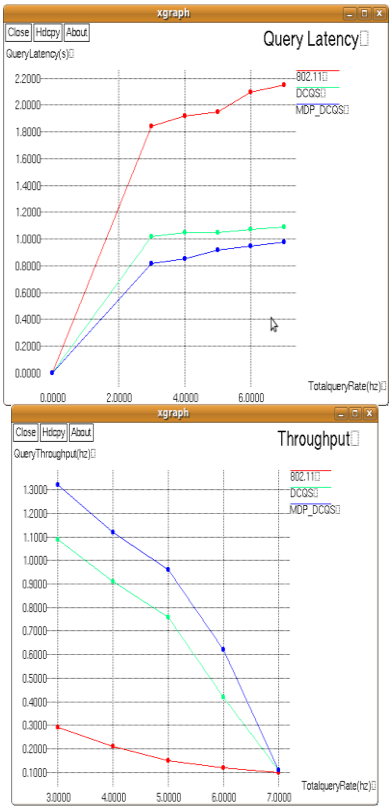


Fig.3 The comparison graphs of DCQS and 802.11, (3.1) Bandwidth (3.2) Energy Consumption, (3.3) Overhead,(3.4) Throughput (3.5) Query latency.

D. Throughput

Throughput is defined as the amount of data transferred from one place to another or processed in a specified amount of data. Data transfer rates for the disk drives and the networks are measured in terms of throughput. In other words it can be defined as the average data rate of successful data. The comparison of the DCQS and 802.11is shown in the Fig 3.4

E. Query Latency

The term latency refers to any of the several kinds of delays typically incurred in processing of network data. In a network, latency is the delay of how much it takes for a packet of data to get from one designated point to another. From that Fig 3.5, we can infer that the DCQS's performance is better than the 802.11 by 65%.

V. CONCLUSION

DCQS, a novel transmission scheduling technique specifically designed for query services in wireless sensor networks. The planner reduces query latency by constructing transmission plans based on the precedence constraints in in-network aggregation. The scheduler improves throughput by overlapping the transmissions of multiple query instances concurrently while enforcing a conflict-free schedule. In contrast, we derive a tight bound on the maximum query rate achieved under DCQS. Such a bound is of practical importance since it can be used to prevent network overload. Exchanging traffic statistics frequently may introduce no negligible communication overhead, which means DCQS can efficiently adapt to changes in workloads by exploiting explicit query information provided by the query service.

Furthermore, it features a local scheduling algorithm that can accommodate changes in query rates and additions/deletions of queries without explicitly reconstructing the schedule. In addition an energy-efficient transmission strategy for WSNs that operates in a strict energy-constrained environment was proposed. The proposed algorithm significantly improves energy efficiency without additional complexity. Our transmission algorithm consists of two components: an opportunistic transmission and a channel-aware backoff adjustment. The MDP formulation was used to obtain the optimum threshold of channel quality for the opportunistic transmission. By intelligently combining these ingredients our transmission algorithm outperforms the existing approaches in terms of energy efficiency, thereby prolonging the network lifetime further.

REFERENCES

1. T.F. Abdelzaher, S. Prabh, and R. Kiran, “On Real-Time Capacity Limits of Multihop Wireless Sensor Networks,” Proc.25th IEEE Int’l Real-Time Systems Symp. (RTSS). (2004),
2. S.R. Madden, M.J. Franklin, J.M. Hellerstein, and W. Hong, “Tiny DB: An Acquisitional Query Processing System for Sensor Networks,” ACM Trans. Database Systems, vol. 30, no. 1, pp. 122-173, (2005).
3. A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, “Wireless Sensor Networks for Habitat Monitoring,” Proc. First ACM Int’l Workshop Wireless Sensor Networks and Applications (WSNA), (2002).
4. R. Maheshwari, J. Cao, and S.R. Das, “Physical Interference Modeling for Transmission Scheduling on Commodity WiFi Hardware,” Proc. IEEE INFOCOM, (2009).
5. L. Krishnamurthy, R. Adler, P. Buonadonna, J. Chhabra, M. Flanigan, N. Kushalnagar, L. Nachman, and M. Yarvis, “Design and Deployment of Industrial Sensor Networks: Experiences from a Semiconductor Plant and the North Sea,” Proc. Third Int’l Conf. Embedded Networked Sensor Systems (SenSys), (2005).
6. R. Maheshwari, J. Cao, and S.R. Das, “Physical Interference Modeling for Transmission Scheduling on Commodity WiFi Hardware,” Proc. IEEE INFOCOM, (2009).
7. V. Rajendran, K. Obraczka, and J.J. Garcia-Luna-Aceves, “Energy-Efficient Collision-Free Medium Access Control for Wireless Sensor Networks,” Proc. First Int’l Conf. Embedded Networked Sensor Systems (SenSys), (2003)