Congestion Avoidance Routing in Wireless Sensor Networks

N. Thrimoorthy
Assistant Professor,
New Horizon College of Engineering
Bangalore, India.

T. Anuradha, PhD
Associate Professor,
Dept. of Computer Science,
Dravidian University, Kuppam, A.P.,
India

ABSTRACT

Wireless sensor nodes consist of group of self organized sensor nodes with limited resources in terms of processing power and battery energy. Wireless sensors are used increasingly in many industrial and consumer applications. Sensors detect events and send via multi hop routing to the sink node for processing the event. The routing path is established through proactive or reactive routing protocols. Congestion happens due to flow of packets exceeding the capacity of link & exceeding the capacity of nodes in terms of Queue Size. Due of congestion packet loss occurs and it affects the quality of application services built on top of sensor network. In this work, we address the congestion from point of reducing the probability of congestion rather than make it to occur and solve it. We propose a NOCO routing protocol which is built on top of AODV and follows reservation based mechanism to avoid congestion and also alleviate the congestion if it happens.

Keywords

Sensor Networks, Multi hop routing, Congestion, AODV, NOCO.

1. INTRODUCTION

In wireless sensor networks hundreds or thousands of sensor nodes transmit packet about a physical phenomenon to a single or more sinks using multi-hop routing. Congestion in wireless sensor networks happen due to simultaneous transmission, buffer overflow and time varying channel conditions. Congestion is more challenging to control in wireless networks than wired networks because of the shared channel.

Due to congestion, the energy consumption of sensor nodes increases. Congestion results in retransmission of packets and sensor nodes energy deplete at faster rate due to this retransmission. Energy drain slowly reduces the life time of sensor network and makes it unusable. Packet loss occurring due to congestion reduces the Quality of service of applications running on WSN platform. Network throughput is decreased even thought network capacity is not fully utilized. Loss of events detection and transmission to sink will result in serious disasters in case the sensor node is deployed in industrial safety applications like fire monitoring.

Previously many studies have been made on congestion control in wireless sensor networks. Studies proposed solutions in different layers like physical, network and transport layers and also cross layers to detect and reduce congestion. Previous works can be classified into following types

1. Rate Based
2. Buffer Based
3. Priority Based
4. Cluster Based
5. Multipath routing based

Rate based algorithms estimates the number of flows from upstream nodes and modulates the rate of packet flow.

Buffer based algorithms tune their transmission rate and time based on the buffer occupancy of nodes in the routing flow.

Priority based algorithms assigns different priorities to flow and make their forwarding decisions based on the priority.

Cluster based algorithms decentralize the congestion control to are of scope by clustering the network and manage congestion.

Multi path routing algorithms divides and forwards packets across multiple path to reduce the congestion.

In this current work, we propose a hybrid approach to eliminate congestion from happening at first go and even if occurs reducing the congestion.

2. RELATED WORK

In this section we survey the current congestion detection and control protocols.

CODA [1] is an energy conserving and efficient control technique that is designed to solve congestion in the upstream direction i.e., the sensor to sink direction. It involves of two main schemes: 1) Open loop hop by hop backpressure mechanism. 2) Closed loop multisource regulation. The detection method in CODA is the receiver based congestion detection. It considers a combination of both present and past loading conditions of the current buffer occupancy in the receiver. If the occupancy exceeds the threshold value, then congestion is inferred. The node detecting the congestion will notify its upstream neighbors to reduce the flow by backpressure mechanism. CODA detects congestion based on queue length and wireless channel loading. It uses AIMD rate adjustment technique and jointly used end-to-end and hop-by-hop controls for regulation.

Event to Sink Reliable transport is a unique transport solution that is designed to achieve reliable event detection with minimum energy expenditure and congestion resolution [2]. This technique overcomes on of the disadvantages of CODA. ESRT works based on two parameters: Event reliability and reporting frequency. Event reliability is defined as the number of data packets received at the decision interval at the sink. The end-to-end data delivery services are regulated by adjusting the sensor report frequency. If the reporting frequency is too low, the sink will not be able to collect enough information to detect the events. But on the other hand, if the reporting frequency is too high, it endangers the event transport reliability. ESRT adjusts the reporting frequency such that the observed event reliability is higher than the desired value to avoid congestion. The congestion detection in ESRT is by local buffer level of the sensors nodes. The sensor node adds a congestion notification bit on the packet's header when congestion occurs. When the sink receives this CN bit, it knows that congestion has happened in WSN.

Congestion control for Multiclass Traffic (COMUT) is a framework that consists of a distributed and scalable congestion control mechanism. It is based on selforganisation of networks into clusters. Each cluster is equipped with a sensor that is autonomously monitors congestion within its scope [3]. These networks are designed to support multiclass of traffic in WSN's. Each cluster is governed by a sentinel. These sentinel roles are assigned to sensors to proactively monitor the system and collect the event rates that is used to infer the combined level of congestion. The local traffic is reported by the sensors to the sentinel en-route a local broadcast system. The sensor rates per cluster are regulated by exchanging only small amounts of control information via regulator packets between the sentinel sensors alongside the flow path.

Congestion control for Sink to Sensors (CONSISE) [4] is a technique that works downstream i.e., from the sink to sensor direction. Conventionally, congestion happens in the sensor-to-sink direction but, the reverse is also possible. The reasons are broadcast storm problem that refers to higher levels of collision that occurs on a series of local broadcast and reverse path traffic from sensors to sink. Congestion in the sensor-to-sink direction will not be rare if WSN is built over CSMA/CA type of MAC and flooding based routing protocol.

Priority based congestion Control protocol (PCCP)[5] is an upstream congestion control protocol in WSN which measures congestion degree as the ratio of packet inter arrival time to the packet service time. It is designed in a way that the data packets have a guaranteed weighted fairness so that sink can get different throughput from the sensor nodes but in a weighted way. PCCP is intended to improve energy-efficient and support traditional QoS in terms of latency, throughput and packet loss ratio. PCCP can be of three components: 1) Intelligent Congestion Detection (ICD). 2) Implicit Congestion Notification (ICN). 3) Priority-based rate adjustment.

In CCF[6] algorithm each node measures the average rate r at which packets can be sent from the node, divide the rate among the children nodes, adjust the rate if the queue is overloaded and propagate the rate downstream. It is designed to work with any MAC protocol in the data link layer and it exists in the transport layer. CCF uses packet service to deduce the availability of the service rate. It controls congestion in a hop-by-hop manner and each node uses exact rate adjustment based on its available service rate and child node number. It has two major disadvantages: The rate adjustment is based on packet service time which leads to low utilization as it has significant packet error rate. It cannot allocate the remaining effective capacity as it uses work conservation scheduling algorithm.

EB works in similar fashion to CODA. It uses congestion control in tree routing structure to all data sources to a sink. It uses the hop-by-hop backpressure mechanism. EB works in three steps: 1) each node calculates the average rate at which the data packets can be sent. 2) The node then divides the average data rate in to the number of children nodes to give the per-node data packet generation rate and adjusts the rate if the buffer is overflowing. 3) The node then compares the data rate of two children nodes with the parent nodes. The smaller rate among the two values is propagated such that data sources do not send packets beyond the minimum rate supported by the nodes along the path to the sink.

SenTCP [7] is a transport protocol that uses open loop hop-by-hop Congestion Control. It has two distinct features that it adopts while detection. It detects congestion using local Congestion degree and uses hop-by-hop for control [4]. The features include: 1) SenTCP conjointly uses average local packet service and average local packet inter-arrival time. These features determine the current local congestion degree in each intermediate sensor nodes. They effectively help to differentiate the reasons for packet loss and delay in wireless communication. 2) Each intermediate node issues a feedback signal backward and hop-by-hop control. This signal carries buffer occupancy ratio and local congestion degree. These parameters are used to adjust the sending rate of the neighboring nodes in the transport layer [1],[10]. SenTCP realizes higher throughput and good energy efficiency since it reduces packet dropping by hop-by-hop. The major disadvantage of SenTCP is that it guarantees no reliability.

Pump Slowly and Fetch Quickly (PSFQ) [8] control protocol aims at distributing data from sink-to-sensors i.e., it belongs to the downstream reliability guarantee. PSFQ is a mechanism that is proposed for reprogramming a group of sensors. PSFQ is based on slowly injecting packets into the network "pump operation" and performing aggressive hop-by-hop recovery in case of packet loss "fetch operation" and selective status reporting "reporting operation". The disadvantages of PSFQ include: 1) since it uses hop-by-hop recovery, it requires more buffer space. 2) The transmission of data packets is relatively slow in operation and hence there is large delay in the system. 3) PSFQ cannot detect a loss of single packets individually as it uses NACK signals for indication and the entire block is re transmitted upon request. 4) It cannot be used in the forward direction and does not address packet loss due to congestion.

3. PROBLEM DEFINITION

Given a wireless sensor network covering a area of A and N sensor nodes distributed randomly in the network and M sinks located in different sides and there can be any number of uni-cast flows from any node to sinks via multi hop routing, the problem is to find routing path for uni-cast flows in such a way to reduce the congestion and if in any path congestion happens how to reduce the congestion using rate adjustment mechanisms.

4. NOCO SOLUTION

The congestion control mechanism in the NOCO solution consist of finding solutions to following sub problems

1. Selection of Sink based on current congestion
2. Finding the routing path
3. Rate allocation on the routing path to avoid congestion
4. Rate adaptation to reduce the congestion

When a node wants to send packet stream to sink, it will broadcast RouteReq. Every node which receives the RouteReq will check if the rate of data packet can be accommodated and will forward RouteReq only if it can accommodate. Each sink receiving the RouteReq will send RouteRes to the source node. When the RouteRes is received at intermediate node, they forward only if the rate of data packet can be accommodated. When the RouteRes is received at the source node, it will select the sink which is close to it.

When sending the data packet through the path selected, the data packet has a field to mark first packet, intermediate packet or last packet. When the first packet marker is set, all the intermediate nodes reserves the rate and it clears based on the two conditions

1. Data packet with last packet marker is set
2. Intermediate packets are not received in a timeout period.

Each sensor node has varied data rate based on the applications running on it. So sensor nodes better know the rate. In RouteReq, we add a parameter rate and the intermediate nodes broadcast RouteReq and RouteRes only if they are able to satisfy the data rate requested.

In each sensor node there is a queue for buffering the data packets from each node and the size of queue is allocated based on the rate requested. If the nodes sends packet exceeding their rate, then packets would be dropped only for that node and it will not affect all other nodes traffic through it.

Sometimes due to processing at node, the queue reserved for a flow may get full. For each queue, expected packet service time is calculated when allocated and when nodes packet service time is continuously higher than expected packet service time, than one of two decisions is taken

1. If the buffer memory for queue enlargement is available, then queue size is increased and the expected packet service is updated.
2. If the buffer memory for queue enlargement is not available, the data packet or ACK packet to the sender is marked with reduce rate field to reduce the data rate.

By this rate adaptation is done and the congestion is avoided.

Link congestion happens if there is may competing nodes for the shared channel. If the time slot scheduling based on MAC is known at higher layer, then the node can use it to calculate and verify if the expected rate in RouteReq can be granted and can abstain from forwarding the RouteReq. By this way NOCO solution is able to make reservation considering the link congestion.

The modifications required in Packet is given below

```
RouteReq

{

   Rate :  // requested rate

}

DataPacket

{

  Typemarker //   0 first packet , 1 inter packet , 2
//last packet

  Reducedrate

}
```

5. MATHEMATICAL MODEL

We model the expected rate calculated based on time slot scheduling and current rate usage in the node and queue size and rate adaptation mathematically in this section.

Let the number of current session through node be N and the rate reserved for each session Rs.

Let the processing speed of node expressed as number of packet processed per second is Tp.

Le the size of packet be Sp

Let the waiting time for slot to send packet is Tw.

Expected rate of data transfer at node Er is given as

$Er = (Tp-Tw/Tp) / (N*Rs)$.

The queue size Qr to meet the rate requested Rr is given as

$Qr = Qi + Rr* Sp$

Qi is the initial queue size allocated for a session.

Once the packet loss is observed at the queue, immediately rate must be reduced by a decrement value.

Let the number of packets lost for a session be Nloss over a period of time Tobs and the new reduced rate Rred is calculated as

$Rred = Rr – (Nloss/Tobs)*\alpha$.

Where α is a constant from 0 to 1 and it be tuned to achieve desired level of control on rate. For some application scenario rapid reduction in rate is not possible for such case α can be given low value.

If the enough queue size is available instead of reducing rate, the queue for session is increased as

$Qr = Qi + Rr*Sp + Nloss/Tobs* \alpha$

6. RESULTS

To test the performance of proposed solution we implemented the solution on NS2. The simulation was conducted with following parameters

Parameters	**Values**
Number of Nodes	100 to 200
Communication range	100m
Area of simulation	1000m*1000m
Packet Rate	10 to 40 packet per second
Simulation time	30 seconds
Interface Queue Length	50
MAC	802.11
No of sink	4

At each time interval of 5 sec, 10 nodes generated packet with rate of 10 to 40 as configured and maintained the traffic rate for 5 sec. The proposed solution is compared with priority based congestion control scheme mentioned in [5]. We measured following parameters

1. Throughput
2. Packet Success ratio
3. Packet Delay
4. No of congestion points

Throughput is measured as the number of packets received at sink and in our approach we sum up the throughput at all sink.

Packet success ratio is the ratio of number of packet received successfully at sink to the number of packets sent.

Packet delay is the end to end delay for packet traversal from source to sink node.

A node is congestion point if its interface queue occupancy is more than 90%.

Throughput is calculated by increasing the packet rate from 10 to 40 insteps of 5 and results are plotted below. From the results we see that NOCO solution achieves better throughput than the priority congestion control.

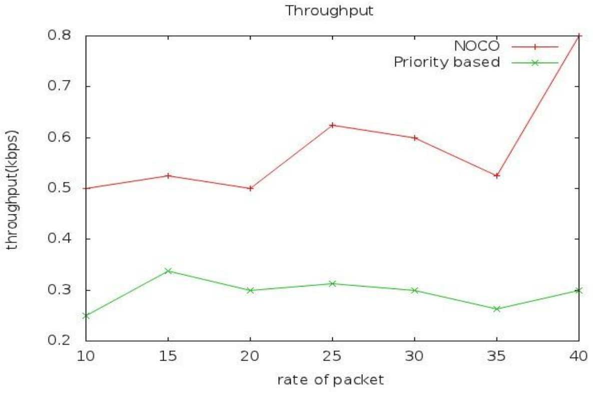

Figure 1: Throughput

Packet success ratio is calculated by increasing the packet rate from 10 to 40 insteps of 5 and results are plotted below. From the results we see that NOCO solution achieves better success ratio , the reason being the number of lost packets reduced due to rate reservation.

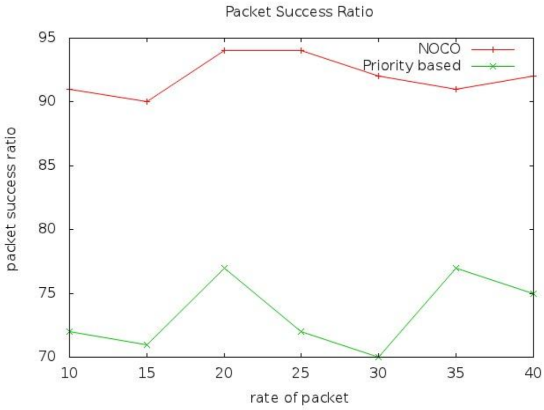

Figure 2: Packet Success ratio

Packet delay is calculated by increasing the packet rate from 10 to 40 in steps of 5 and the results are plotted below. From the results we see that the NOCO solution has comparatively higher delay the reason being some longest path for packets are possible in our approach.

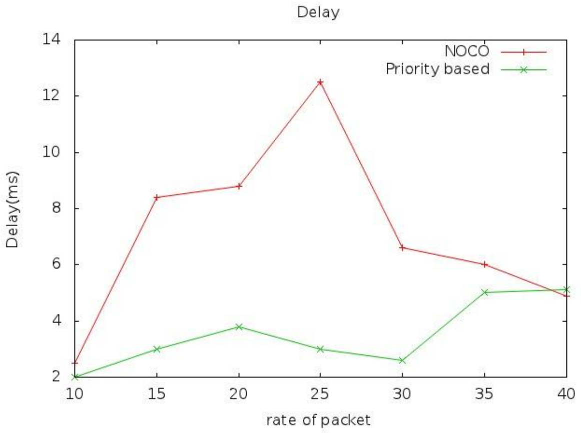

Figure 3: Packet Delay

We measured the number of congestion points by varying the packet rate from 10 to 40 in steps of 5 and the results show that number of congestion points is very less in our NOCO solution.

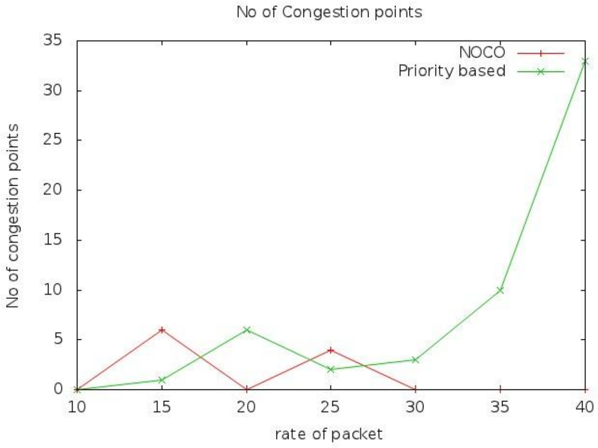

Figure 4: No. of congestion points

7. CONCLUSION

In this work, we have explained the NOCO congestion control mechanism. Through simulation we have proved that our proposed protocol has low congestion when compared to other congestion control algorithms. Due to congestion reduction packet loss is reduced and network throughput is increased. Also out algorithm uses the network capacity fairly and increases the life time of the sensor network.

8. REFERENCES

[1] Chien- Yin Wan, Shane B. Eiseaman, Andrew T. Champbell, “ CODA: Congestion Detection and Avoidance”.

[2] Ozgur B. Akan, Ian F. Akyildiz, “Event-to-Sink Reliable Transport in wireless Sensor Networks”.IEEE/ACM transactions on Networking. Vol. 13, No.5, October 2005

[3] KyriakosKarenos, VanaKalogeraki, Srikanth V. Krishnamurthy. “Cluster-based Congestion control for sensor networks”.

[4] RamujaVedantham, RaghupathySivakumar, Sueng – Jong park, “Sink-to-Sensors Congestion Control Strategy”.

[5] Mohammad HosseinYaghmaee, Donald Adjeroh. “ A

New priority Based Congestion Control Protocol for Wireless Multimedia Sensor Networks". IEEE 2008.

[6] Swastil Brahma, MainakChatterjee, Kevin Kwiat," Congestion Control and Fairness in Wireless Sensor Networks".

[7] C. Wang, K.Shoraby, B.Li, "SenTCP: A hop-byhop Congestion Control protocol for Wireless Sensor Networks" in IEEE INFOCOM 2005, USA, March 2005.

[8] C. Y. Wan, A.T. Campbell, "PSFQ: A reliable transport protocol for Wireless Sensor Networks" in proceedings of ACM WSNA'02, September 28, 2002, USA.

[9] K.ChenandK.Nahrstedt. EXACT:AnExplicitRate-basedFlowControlFrameworkinMANET(extendedversion). Technical Report UIUCDCS-R-2002-2286/UILU-ENG-2002-1730, Department of Computer Science, University of Illinois at Urbana-Champaign, July 2002.

[10] K. Chen and K. Nahrstedt. Limitations of Equation-Based Congestion Control in Mobile Ad Hoc Networks. In ICDCSW '04: Proceedings of the 24th International Conference on Distributed Computing Systems Workshops W7: EC, pages 756–761, 2004. DOI: 10.1109/ICDCSW.2004.1284118.

[11] K. Chen, K. Nahrstedt, and N. Vaidya. The Utility of Explicit Rate-based Flow Control in Mobile Ad Hoc Networks. In WCNC '04: Proceedings of the IEEE Wireless Communications and Networking Conference, volume 3, pages 1921–1926, Mar. 2004.

[12] K. Chen, Y. Xue, and K. Nahrstedt. On Setting TCP's Congestion Window Limit in Mobile Ad Hoc Networks. In ICC '03: Proceedings of the IEEE International Conference on Communications, Anchorage, Alaska, May 2003. DOI: 10.1109/ICC.2003.1204525.

[13] C. de M. Cordeiro, S. R. Das, and D. P. Agrawal. COPAS: Dynamic Contention-Balancing to Enhance the Performance of TCP over Multi-hop Wireless Networks. In Proceedings of the 10th International Conference on Computer Communication and Networks (IC3N), pages 382–387, Miami, FL, USA, Oct. 2002.

[14] R. de Oliveira and T. Braun. A Delay-based Approach Using Fuzzy Logic to Improve TCP Error Detection in Ad Hoc Networks. In WCNC '04: Proceedings of the IEEE Wireless Communications and Networking Conference, volume 3, pages 1666–1671, Mar. 2004.

[15] R. de Oliveira and T. Braun. A Dynamic Adaptive Acknowledgment Strategy for TCP over Multihop Wireless Networks. In INFOCOM '05: Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies, volume 3, pages 1863–1874, Mar. 2005. DOI: 10.1109/INFCOM.2005.1498465.