This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2007 proceedings.

Fuzzy Logic based Congestion Estimation for QoS in Wireless Sensor Network

Saad A. Munir *Yu Wen Bin*
State Key Laboratory of Networking & Switching,
Beijing University of Posts & Telecommunications,
Beijing, China
samunir@gmail.com, ywbwww@yahoo.com.cn

Ren Biao *Ma Jian*
Nokia Research Center
China
Ext-biao.ren@nokia.com, Jian.j.ma@nokia.com

Abstract: ***In this paper, we present a model for fuzzy logic based congestion estimation within a proposed QoS architecture. The architecture comprises of a QoS Management and Control module which is implemented both at node level and at the sink for a system level QoS administration. The QoS module implemented at sensor node forms a subset of the larger QoS Management and Control module implemented for the system wide information, so as not to encumber the resource constrained wireless sensor network. While much research has been conducted in wireless sensor network, little attention has been given to a holistic QoS approach for WSN. Energy efficiency has been the main QoS metric in research efforts. In this paper, we present a congestion estimation model for QoS in wireless sensor network, and implement it using fuzzy logic with fuzzy set variables. Simulations are conducted for our scheme which shows that with increased network dynamics and with increased packets generation rate, our implementation efficiently sorts out the traffic and minimizes the packet loss for prioritized event-driven traffic.***

I. Introduction

Wireless Sensor Network (WSN) owes its success mainly to the modern technological advancements in recent years that have enabled low-cost, low-power and minute sensor nodes in production. These tiny nodes work in the order of a few to tens of hundreds in areas ranging from sophisticated urban homes to extremely hostile, remote areas. The possible application scenarios, which are envisaged at the moment, include environmental monitoring, military surveillance digitally equipped homes, health monitoring, manufacturing process monitoring, conferences, vehicle tracking and detection (telematics), and monitoring inventory control.

In the absence of any central control authority and with its ad hoc nature, it is imperative that network should be self-configuring and self-healing in case of any challenge. While much work has been done in wireless sensor network, little research has focused on quality of service issues in wireless sensor networks. Energy efficiency, as a QoS metric, remains the major focus of research and other metrics are compromised to scavenge energy. This is in lieu of energy-constrained sensor nodes, working unattended and required to work reliably for a long period of time.

In this paper, a congestion estimation model for QoS in wireless sensor network is presented that is based both at node level and at the sink level for an overall system overview. In the node buffer, a queuing model is employed which classifies the traffic based on three application classes; source-event driven, source-continuous data, and sink query based. In our estimation for congestion at a particular node, we make use of fuzzy logic technique [1][2]. Fuzzy-logic based network monitoring is not a new concept and has been previously studied in various research papers [4][5]. However, to the best of our knowledge, no prior example exists of application of fuzzy logic theory in QoS control in wireless sensor networks. This paper presents a fuzzy-logic based mechanism to efficiently estimate congestion at node level so that a pro-active approach can be taken to mitigate congestion at node level and to maintain a desirable QoS.

The paper is organized as follows. Next section discusses some of the related work in implementing QoS in wireless sensor network. In section III, we will discuss our architecture for quality of service in sensor network covering both node level and system level. We will then highlight upon Fuzzy logic based system used in our buffer model in section IV. Section V will discuss about the simulation study and performance evaluation, followed by conclusion in section VI.

II. Related Work

Although the subject of wireless sensor network is an active research field for the past few years, QoS has largely been an unexplored area. Quality of service in a network is largely associated with energy efficiency in the network, and with network lifetime. However, with the growing applications of wireless sensor network, quality of service is now deemed necessary to be implemented. At the network routing level, a few protocols have been proposed so far. One of the earliest routing protocols with QoS impression is Sequential Assignment Routing (SAR) [6]. SAR creates trees originating from one-hop neighbor-hood of sink and takes into account two types of QoS metrics; energy resource and priority level of each packet. Multiple paths are created from a sink to source and a path may be selected based on the QoS metrics. However, SAR is believed to suffer from overhead of maintaining node state. Another QoS based routing protocol was presented by Akkaya [3], which classifies the traffic on the basis of real-time and non-real time application data. This protocol further makes use of a cumulative link-cost for each link and end-to-end delay and chooses the least-cost link, though the weightings allocated to different QoS parameters are not mentioned in detail. Our buffer queuing model is similar to the one presented by Akkaya, however, we classify traffic based on the application type, and do not take into account the real-time or non-real time traffic. At transport layers, notion of *reliability* exist in the PSFQ and ESRT protocols [10, 11]. Whereas PSFQ deals with data flow with strict delivery guarantees, ESRT is a solution to achieve reliable event detection with minimum energy expenditure and congestion resolution.

Much of the other QoS related work has focused on designing an optimal number of sensor nodes from which a sink would ask for data information. The approach is based on Gur Game theory, taking into consideration delays and addition and removal of sensor nodes. This approach has been studied in [7][8].

Fuzzy logic based management and control has been studied in the past for wireless networks and ad hoc networks. In [4], authors present a fuzzy congestion control approach for ad-hoc networks, in which a theoretical fuzzy logic based concept is used to control the congestion. Our methodology is considerably different from this approach, as we actually implement fuzzy logic with a fuzzy table derived from fuzzy sets and fuzzy variables; thus a more realistic implementation of fuzzy logic.

Nevertheless, to the best of our knowledge, no prior research work exists for the holistic QoS approach within the wireless sensor network, both from the node point of view and from system point of view.

III. QoS Model for WSN

The QoS model in our approach consists of two modules; QoS Management and QoS Control.

QoS Management is the mechanism for managing the QoS related issues in wireless sensor network. This management mechanism works on the basis of feedback system. We further propose that, this management system works on the principle of fuzzy logic approach. The fuzzy logic methodology helps us having a pro-active approach to solve QoS related issues. The pro-active approach helps achieve efficiency in more QoS constrained wireless sensor networks, with more dynamics involved. QoS Management system has two modules. One is the System Knowledge, in which knowledge of nodes' energy reservoir, buffer capacity, transmission range, and localization is stored. This information is periodically updated to reflect the true behavior of the system. The management mechanism helps us have a current knowledge of the network configuration and state. The second module consists of an interface for the feedback system which is then compared with the known values. The QoS Control module, upon making an analytical comparison, comes up with a decision and may take an action, such as adjust the duty cycle or transmission power of nodes, initiate a new path, and adjust the packet generation rate.

Now we will detail a description of our proposed QoS architecture for wireless sensor network, both at individual node level and at sink level (system view).

From a node point of view, our QoS architecture with different modules is depicted in figure 1. The node consists of processor, memory with buffer capacity, and transceiver at hardware level and QoS Management and Control along with Energy Model as software abstract. A description of different components will now be presented, highlighting the overall node architecture.

The most important QoS model in our approach, the Queuing model, located within the buffer is classified into three categories depending on the type of service layer. The three different service classifications are Sink Query based, Source Event Driven and Source Continuous data model. We argue that in most of the sensor network applications, event driven model has the highest priority as compared to query based and continuous data. This is quite reasonable in application scenarios such as motion detection, intrusion detection, smoke or fire indication, and similar event-monitoring applications. In these applications, it is of utmost importance that in case of any sensor-specific "event" the information should be relayed to the end user at its earliest with minimal packet loss.

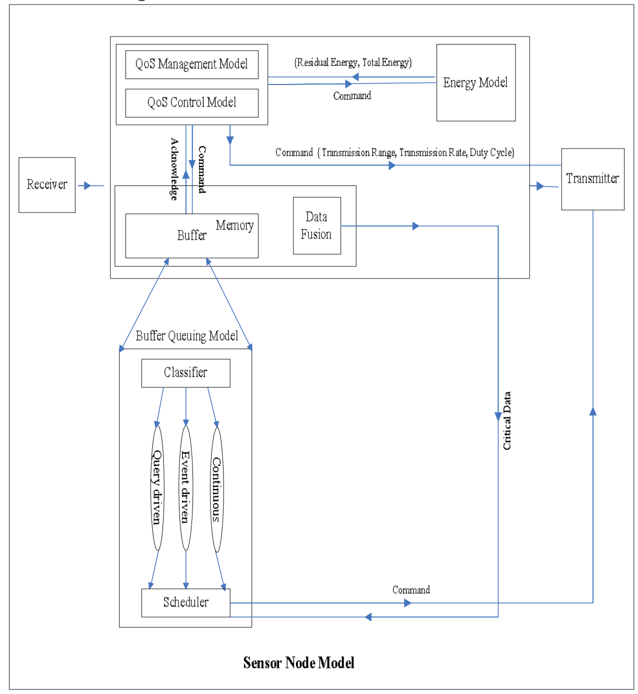

Figure 1. QoS Management and Control Model for Sensor Node

However, in case of temperature monitoring or similar "data" information applications, where there is no particular instance of an "event" but a continuous data stream, sink query based model assumes higher priority then normal continuous data. This classifier is inspired by the model proposed by Akkaya in [3], but instead of classifying data based on real-time and non-real time, we wish to classify the data based on service model, which we believe to be a realistic and more important parameter to consider. Further, real time applications are yet an arduous task for the already resources constrained wireless sensor network. The sample algorithm for our methodology is presented in figure 3 which, upon congestion, makes a packet drop decision based on packet service type. The classification in our methodology is followed by the "scheduler", which, based on the application type, schedules the data for ongoing transmission and buffer maintenance. In addition to this, with the help of known application types, "data fusion" at the sensor node may bypass the queuing model and a "critical" data may directly be sent to the scheduler, as shown in figure above.

Likewise, at the transceiver level, a relationship with the QoS management and control model exists which is basically a command based mechanism, a part of which includes duty cycle, transmission range and transmission rate. Transmission range of the node is increased or decreased depending on the network topology. The gains achieved are increased connectivity, energy saving, and mitigating congestion. Increasing transmission range helps us achieving a better connectivity in the network, with more number of neighbor nodes for each sensor node; thus having multiple route paths. Energy saving may seem to be contradictory and while it is true that energy expenditure increases with increased transmission range, we, however, come up with increased system-wide energy efficiency by having a fewer intermediate relay nodes.

The Energy Model, periodically or on-demand, reports about the residual energy and total energy of the sensor node. This information forms the basis of calculation and estimation of node energy at the QoS management and Control module for further efficient data and node management tasks.

The QoS Management and Control modules as well as Energy Model are the software abstracts present within our model. We wish to localize the QoS strategy at each node, so as not to put extra burden on computationally scarce sensor node. The QoS management and Control module implemented at each node is node-specific and carries out the QoS related tasks addressing the specific node.

At the system level, QoS Management and Control module is implemented in the sink, which is thought to be rich in energy and computational resources. In our approach, the QoS management and control module at sink consists of a system model, which essentially comprises of system energy table, system congestion table, system delay table, and system packet loss. These models, as depicted in figure 2 comprise of the related energy, congestion and delay information of the sensor nodes. The information is collectively analyzed in the "Controller" so as to have a system-view of the network. Scheduler forms the end-interface of QoS management and control module. QoS Management and Control module at sink will further act as an interface between wireless sensor network and other networks where the end user may be present, such as internet or cellular networks. We, however at the moment, only focus on the interface between the sink and the wireless sensor network. The sink QoS Management and Control module will further decide about the level of data fusion required at the sensor nodes, which is a trade-off between the energy efficiency gain and the level of accuracy obtained by different, multiple data measurements.

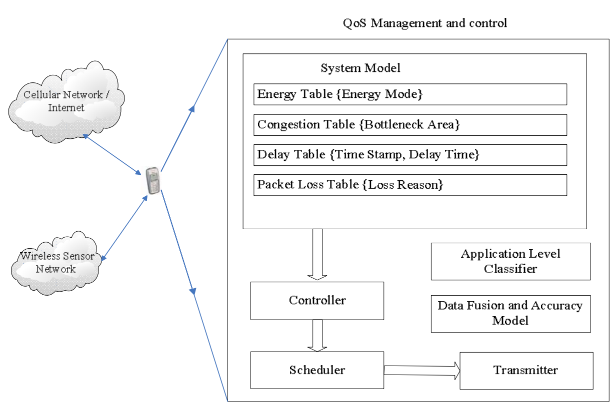

Figure 2: QoS Management & Control at Sink

Algorithm

If (value of $\mu_{p*s} = 1$) *// It is getting congested, as known from the fuzzy table*
{
switch (Packet Header)
{
case 'Continuous':
drop the packet;
break;

case 'Query':
{
if (there is a "Continuous" packet in buffer)
{
drop the "Continuous" packet and insert the incoming "Query" packet into the buffer.
}
else
drop the "Query" packet;
break;
}

case 'Event':
{
if (there is a "Continuous" packet in buffer)
{
drop the "Continuous" packet and insert the incoming "Event" packet into the buffer.
}
else
if (there is a "Query" packet in buffer)
{
drop the "Query" packet and insert the incoming "Event" packet into the buffer.
}
else
drop the "Event" packet;
break;
}
end if;

Figure 3: Algorithm for classifying traffic based on service-type

In designing a QoS aware sensor network with respect to delay metric, an efficient queuing model is of paramount importance. This is because; queuing delay is often the major portion of the overall delay involved in sending packets from the source to the destination. In the next section, an overview of the fuzzy logic based queuing model for congestion estimation is presented.

IV. Node Specific Fuzzy Logic Approach for Congestion Estimation

We have used fuzzy logic to enhance QoS mechanism in wireless sensor networks. Fuzzy logic promises to be used as an effective tool for efficient buffer management by using simple, robust techniques. The approach follows that in wireless sensor networks, the QoS metrics may be defined by a non-distinct boundary, that is in case of delay, we might not be concerned about the exact amount of delay we are having; instead, with the help of the current network state, a decision needs to be made as to whether it is getting congested or not.

Our problem definition is as follows,

"In buffer model, with the given packet arrival rate, given buffer size and the current transmission rate, we are interested in calculating and maintaining a fuzzy table for the conditions of buffer getting congested at the current time."

We will now describe our methodology for fuzzy logic approach to mitigate congestion in the network. In estimating congestion, the two most important variables are the net packet arrival rate and the current buffer occupancy. With fuzzy logic, we assign grade values to our two variables at hand, which are packet arrival rate and buffer capacity. Our fuzzy set therefore consists of two fuzzy variables

Fuzzy Set, **A** = $\{p, s\}$

Where p is the fuzzy variable term for the net packet arrival rate P_{net} in the buffer. P_{net} is the ratio of incoming packets to outgoing packets at a particular node, and is therefore defined as,

$$P_{net} = \frac{P_{in}}{P_{out}} \tag{1}$$

Similarly, s is the fuzzy variable for buffer size S, which corresponds to the percentage level of buffer fullness at a particular instance.

The fuzzy set variables in our example take grade values of 0, 0.5 and 1, corresponding to no congestion, medium congestion, and heavy congestion. The grade value of a particular fuzzy variable takes a value between 0 and 1 and the number of values taken determines the number of states we wish to observe in our readings. These fuzzy grade values are not absolute and we may introduce more grade values for such linguistic terms as "slight congestion, somewhat congested, very heavy congestion" and any such term. However, for simplicity and for simulation purpose, we take the grade values in three terms.

Thus, after assigning the grade values to our fuzzy set variable p, we have a set of equations

$$p = \begin{Bmatrix} 0, & for P_{net} < 0.5 \\ 0.5, & for 0.5 \le P_{net} < 1 \\ 1, & for P_{net} \ge 1 \end{Bmatrix} \tag{2}$$

Similarly, for buffer size fuzzy variable s, we come up with three fuzzy states corresponding to the percentage fullness of the buffer,

$$s = \begin{Bmatrix} 0, & for S = 0 \\ 0.5, & for S \le 0.6 \\ 1, & for 0.6 < S \le 1 \end{Bmatrix} \tag{3}$$

As stated earlier, the values for the fuzzy variables are not absolute and may change according to the application at hand, and user requirements.

Having obtained the fuzzy values for the two variables, we now come up with the fuzzy table for these two variables. This fuzzy table is obtained by the outer max product of the fuzzy set values.

The max-min composition of any two relations $P(U,V)$ and $Q(V,W)$, is defined by the membership function

$$\mu_{P*Q}(x,z) = \{(x,z), \max_y [\mu_P(x,y), \mu_Q(y,z)]\} \tag{4}$$

Therefore, applying equation (4) to our fuzzy variables at hand, the fuzzy table is shown in table 1,

p \ s	0	0.5	1
0	**0**	**0.5**	**1**
0.5	**0.5**	**0.5**	**1**
1	**1**	**1**	**1**

Table 1: Fuzzy Logic Table for buffer size and incoming/outgoing packet ratio

The above table is therefore the fuzzy table for the estimation of congestion considering the current packet arrival rate and buffer size. As can be shown, the outer-max product gives us a strict passive measure of the congestion state and will result in a stringent measurement for congestion estimation.

V. Performance Evaluation

Simulations were performed in NS-2 [9] to verify our approach. The algorithm presented in figure 3 was implemented in NS-2 and simulations were conducted to test its validation. Extensive simulations were conducted with varying packet generation rate, thus varying the network traffic within the network. Three service applications corresponding to source-event driven, continuous, and query-based traffic were generated and the appropriate buffer model was created which, upon receiving or generating the data packets, will schedule the traffic based on service classification type. Congestion was induced into the network by increasing the packet generation rate and with high network dynamics. The simulation scenario consisted of a total of 50 nodes in a random way-point model with the standard *DSR* and *802.11* routing and MAC level specifications, respectively. The simulations were run for 1000 seconds. We have made use of UDP agent to add a new field in initial common header (appli_packet_type) for our three service application categories corresponding to event, continuous, and query to differentiate their priority. The event-driven packets are given highest priority and continuous and query-based traffic packets may be dropped if there is a congestion state within the network. The simulation results are presented herewith and a comparison is shown between the congestion and packets dropped with and without implementation of our proposed fuzzy logic based congestion estimation.

Figure 4 shows the packets dropped for varying packets generation time interval. It is to be noted that for low packet generation time interval, i.e. for high packet generation rate, the number of event-driven packets dropped are almost 44-56 % less as compared to packets dropped for query-driven and continuous traffic, respectively. The graph shows that the number of event-driven packets dropped is considerably less than both continuous and query-based traffic. Please note that these figures account only for the packet drop in buffer. As can be seen, for less packets intensive applications, the number of event-driven packets drop reduce to as low as 5%.

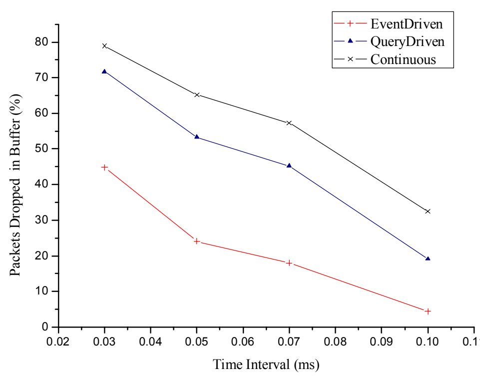

Figure 4: Packets Dropped for varying Packet Generation Rate

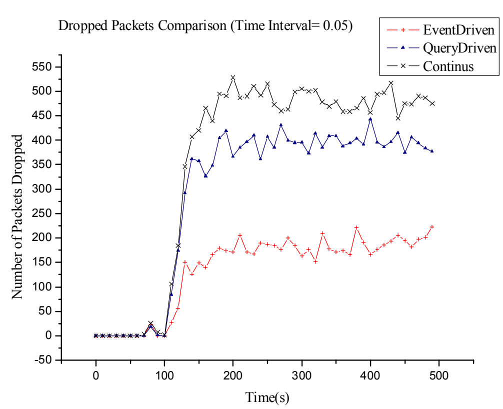

Figure 5: Number of Packets Dropped as a Function of Time for Three Traffic Classifications

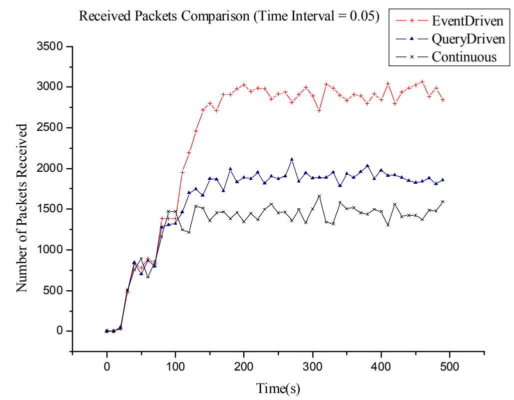

Figure 6: Number of Packets Received as a function of Time for Three Traffic Classifications

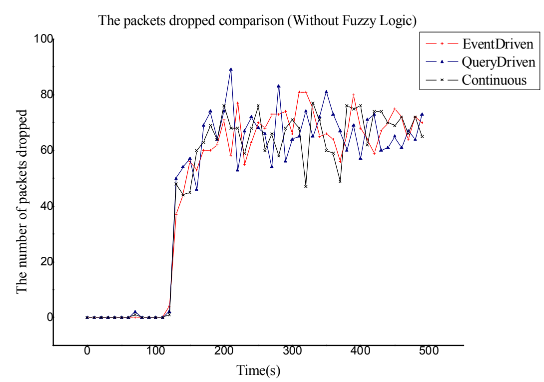

Figure 7: Number of Packets Dropped without Fuzzy Logic based Congestion Estimation

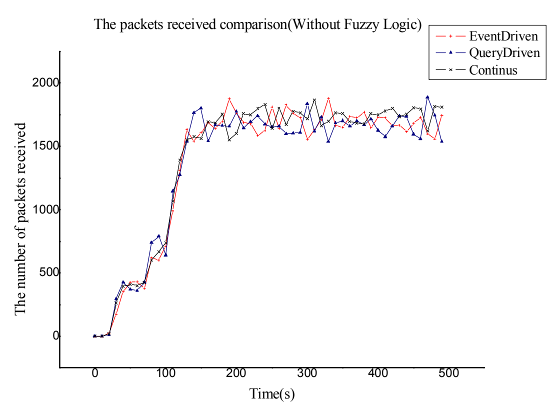

Figure 8: Number of Packets Received without Fuzzy Logic based Congestion Estimation

Figures 5 and 6 present the results achieved for number of packets dropped as a function of time. The results verify that as the dynamics increase in the form of network traffic, the number of packets dropped for event-driven is far less than the packets dropped for query-driven and continuous traffic.

This is to note that the number of packets generated in our simulation for the three kinds of service classifications was kept random with an equal probability for the generation of each kind of service types, so as to make our comparison results look more realistic. Further, the packets dropped in buffer are due to the implementation of our algorithm and contribute to the node as well as system efficiency.

As a comparison to our proposed fuzzy logic based congestion estimation, figures 7 and 8 show the result of induced network congestion without making use of fuzzy logic based congestion estimation. Figure 7 depict the number of packets dropped in case of congestion whereas figure 8 shows the number of packets received in case of congestion. As indicated in the figures, there is no differentiation between a possibly critical event-based sensor data and a continuous sensing traffic; thereby the number of packets dropped is equally likely for any of the three traffic classifications. The percentage packets dropped for a single simulation for a given packet generation rate is shown in figure 9. The numbers of event-driven packets which are dropped are only 5%, as compared to the 33% drop in case of continuous traffic. Similarly, figure 10 displays the percentage packets received for three traffic classifications. This figure is displayed so as to give the reader an estimation of the overall network state affected by our algorithm implementation. More than 55% packets generated are received as compared to around 32% packets received for continuous traffic; due to the fuzzy-logic based traffic classifier.

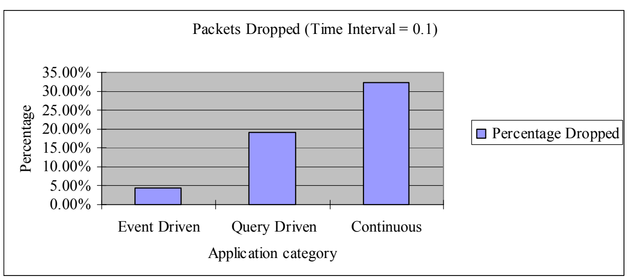

Figure 9: Percentage Packets Dropped for Three Traffic Classifications

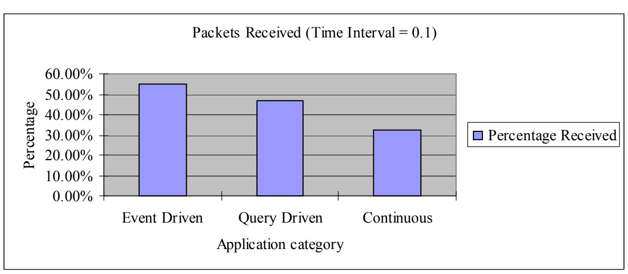

Figure 10: Percentage Packets Received for Three Traffic Classifications

VI. Conclusion

In this paper, we have described a fuzzy logic approach for efficiently estimating the congestion and then reacting to mitigate the congestion. A fuzzy table is maintained which takes the values of buffer state and incoming to outgoing packets ratio as input and gives us an output in the form of fuzzy variable as to whether a decision needs to be taken or not. Extensive simulations were conducted in NS-2 to verify our approach. We have presented our simulation results here which show that our mechanism works efficiently by reducing the number of important event-driven packets from being dropped as compared to normal continuous traffic and query driven traffic. Simulation results show that our implementation particularly works well with increased network traffic, that is, with increased packet generation rate. For resource scarce sensor nodes, which have insufficient memory and buffer capacity, our algorithm works especially well, ensuring the efficient delivery of prioritized event-driven packets. We have also presented a QoS model in which the fuzzy congestion estimation methodology resides. Future work in this regard relates to implementation of a fuzzy logic based QoS architecture for system wide congestion estimation in WSN and to consider the affect of increasing the congestion related parameters.

Acknowledgements

Our acknowledgements go to Dr. Xie Dong Liang and Tian Le of Broadband and Network Research Center, BUPT and to anonymous reviewers for their expert opinions and useful feedback.

References

[1] L.A. Zadeh, Fuzzy Sets, Information and Control, 1965

[2] L.A. Zadeh, Outline of A New Approach to the Analysis of Complex Systems and Decision Processes, 1973

[3] K. Akkaya, M. Younis, "An Energy-aware QoS Routing Protocol for Wireless Sensor Networks", 23rd International Conference on Distributed Computing Systems Workshops (ICDCSW'03) p. 710

[4] L. Khoukhi, S. Cherkaoui, "FuzzyCCG: A Fuzzy Logic QoS Approach for Congestion Control in Wireless Ad Hoc Networks", *Q2SWinet '05, October 13, 2005, Montreal, Quebec*

[5] X. Zhang, S. Cheng, M. Feng, W. Ding, "Fuzzy Logic QoS Dynamic Source Routing for Mobile Ad Hoc Networks", *The Fourth International Conference on Computer and Information Technology (CIT'04) pp. 652-657*

[6] K. Sohrabi, J. Gao, V. Ailawadhi, G. Pottie, "Protocols for self-organization of a wireless sensor network," *IEEE Personal Communications, Vol 7, No 5, pp. 16-27, October 2000*

[7] R. Iyer, L. Kleinrock, "QoS Control for Sensor Networks", *In Proc. of 2003 ICC, May 2003.*

[8] J. Kay, J. Frolik, "Quality of Service Analysis and Control for Wireless Sensor Networks", *Mobile Ad Hoc and Sensor Systems, MASS 2004, Oct 2004*

[9] Network Simulator: www.isi.edu/nsnam/ns

[10] C. Y. Wan, A. T. Campbell, L. Krishnamurthy, "Pump-slowly, Fetch-quickly (PSFQ): A Reliable Transport Protocol for Sensor Networks", *IEEE Journal on Selected Areas in Communications, Vol. 23, No. 4, April 2005*

[11] Y. Sankarasubramaniam, O. B. Akan, I. F. Akyildiz, "ESRT: Event-to-Sink Reliable Transport in Wireless Sensor Networks", *In Proc. of MobiHoc '03, June 1-3, 2003, Annapolis, MD*