International Journal of Automation and Computing 12(1), February 2015, 1-13
DOI: 10.1007/s11633-014-0861-y

A Comparison of Mamdani and Sugeno Fuzzy Based Packet Scheduler for MANET with a Realistic Wireless Propagation Model

Oche Alexander Egaji[1] Alison Griffiths[1] Mohammad S. Hasan[1] Hong-Nian Yu[2]

[1]Faculty of Computing, Engineering and Sciences, Staffordshire University, Stoke-on-trent, UK

[2]Faculty of Science and Technology, Bournemouth University, Fern Barrow Pool, Dorset, UK

Abstract: The mobile nature of the nodes in a wireless mobile ad-hoc network (MANET) and the error prone link connectivity between nodes pose many challenges. These include frequent route changes, high packet loss, etc. Such problems increase the end-to-end delay and decrease the throughput. This paper proposes two adaptive priority packet scheduling algorithms for MANET based on Mamdani and Sugeno fuzzy inference system. The fuzzy systems consist of three input variables: data rate, signal-to-noise ratio (SNR) and queue size. The fuzzy decision system has been optimised to improve its efficiency. Both fuzzy systems were verified using the Matlab fuzzy toolbox and the performance of both algorithms were evaluated using the riverbed modeler (formally known as OPNET modeler). The results were compared to an existing fuzzy scheduler under various network loads, for constant-bit-rate (CBR) and variable-bit-rate (VBR) traffic. The measuring metrics which form the basis for performance evaluation are end-to-end delay, throughput and packet delivery ratio. The proposed Mamdani and Sugeno scheduler perform better than the existing scheduler for CBR traffic. The end-to-end delay for Mamdani and Sugeno scheduler was reduced by an average of 52 % and 54 %, respectively. The performance of the throughput and packet delivery ratio for CBR traffic are very similar to the existing scheduler because of the characteristic of the traffic. The network was also at full capacity. The proposed schedulers also showed a better performance for VBR traffic. The end-to-end delay was reduced by an average of 38 % and 52 %, respectively. Both the throughput and packet delivery ratio (PDR) increased by an average of 53 % and 47 %, respectively. The Mamdani scheduler is more computationally complex than the Sugeno scheduler, even though they both showed similar network performance. Thus, the Sugeno scheduler is more suitable for real-time applications.

Keywords: Riverbed modeler, variable-bit-rate (VBR), constant-bit-rate (CBR), signal-to noise ratio (SNR), wireless mobile ad-hoc network (MANET).

1 Introduction

A wireless mobile ad-hoc network (MANET) comprises of randomly distributed mobile nodes that constitute a network without the need of a control centre or infrastructure. MANET has many useful applications, e.g., disaster relief, military operation, and recently reported civilian applications (this includes environmental monitoring, healthcare, etc.). The transfer of data between MANET nodes is peer-to-peer in nature. A pair of mobile nodes can communicate directly when they are within the radio range of each other. Hence, in order for a particular source to transmit data to a destination outside of its transmission range, the data from the source node must be relayed through one or multiple intermediate peer(s). This phenomenon is called multi-hop, which is a special characteristic of the MANET.

As a result of the dynamic nature of node movement, there are frequent disconnections between nodes which are connected either directly or indirectly[1].

As MANETs gain popularity, the need for them to support real-time and multimedia applications has increased. These applications have quality of service (QoS) requirements and some of the measuring metrics include throughput, end-to-end delay and packet delivery ratio (PDR)[2]. The QoS provision for a MANET can be provided over various layers in the open systems interconnection (OSI) protocol stack, starting from the physical layer to the application layer. For example, the physical layer is responsible for the quality of transmission. The link layer handles the variable bit error rate. The network layer is responsible for any change in the delay and bandwidth. The transport layer deals with the delay and packet loss due to transmission, whilst the application layer handles the regular disconnection and reconnection of the network link[3].

The random nature of node movement in a MANET causes frequent route changes. This can lead to high packet loss and high end-to-end delay. It can also decrease the throughput of the network. As the traffic load increases, the performance of the network decreases. A MANET is infrastructure-less, thus it is difficult for any single mobile node to have an accurate and up to date picture of the network topology. In addition to the band limited shared network and the error prone nature of the wireless channel, the infrastructure-less state of the network makes meeting

Regular Paper
Special Lssue on Recent Advance in Automation and Computing
Manuscript received January 2, 2014; accepted September 24, 2014
Recommended by Associate Editor Yi Cao

a specified QoS target more difficult to attain. All nodes in a MANET have the capacity to be a source, destination or just a relay. These various functionalities of a MANET node will create various queuing behavior, which are different from a traditional cellular or wired network. Hence by using a scheduling algorithm to determine what queue or packet needs to be served next, the overall network performance can be improved. The default scheduling scheme for packets in MANET is first in, first out (FIFO).

A great deal of research has been done to improve the QoS of MANET. Xiao et al.[4] focused on routing protocols to improve link stability, end-to-end delay and bandwidth optimisation. Chou and Ishii[5] proposed an efficient coding scheme for the dissemination of data between MANET nodes. Perkins et al.[6, 7] compared the performance of various routing protocols with regards to mobility, delay, packet loss and network congestion and Ramachandran et al.[8] discussed the link stability in MANET.

Manoj et al.[9] proposed a Mamdani fuzzy inference system with two input variables and a single output (priority index), to schedule packets in MANET. The two input variables are channel capacity and data rate. These were used to determine the Priority Index of packets to be scheduled. Gomathy and Shanmugavel[10] also presented some work on Mamdani fuzzy scheduling with MANET (based on buffer size and number of hops suffered by packet).

Based on [9], we explored a better way to improve the QoS of MANET. In the course of this paper, [9] will be referred to by the first name of the first author "Manoj".

This work is built on [11] to propose a Sugeno based fuzzy scheduler. This scheduler is less computationally complex than the Mamdani. The performance analysis of the scheduling algorithms was done for constant-bit-rate (CBR) and variable-bit-rate (VBR) traffic.

Packets are scheduled based on their priority index. The priority index for the individual packets is calculated by considering three input variables. These are data rate, queue size and signal-to-noise ratio (SNR). The fuzzy scheduler was developed in a riverbed modeler using Proto-C language. The Mamdani and Sugeno fuzzy schedulers have been optimised so that the algorithm runs quicker which is essential for real-time applications. The proposed schedulers improved the overall end-to-end delay, throughput and PDR of the network. This paper contains 6 sections. Section 2 introduces the various traffic profiles, focusing on CBR and VBR traffic. Section 3 defines QoS and some of its measuring metrics, it also explains scheduling schemes and some currently available schemes. Section 4 describes the fuzzy inference system (FIS) focusing on the Mamdani and Sugeno FIS. Section 5 presents the performance analysis, it also includes the results and discussion. Finally, Section 6 presents the conclusion.

2 Traffic profiles

Traffic flow can be classified into one of the following traffic profiles: CBR, VBR and Bursty-bit-rate. This paper focuses on CBR and VBR traffic because they model real-time applications for video, voice and control. These profiles are based on the inter-arrival times/distribution of the traffic[12].

2.1 Constant bit rate

The data rate for CBR traffic is shown in Fig. 1; it does not vary over time. The average data rate and the peak data rate are the same for CBR models. The maximum burst size is also constant, thus the QoS requirement for this type of traffic is constant and easily predicted so the network can allocate the bandwidth needed for a flow[13]. This type of traffic is delay sensitive as it consists of real-time traffic. The odd packet drop is allowable as long as the packets are delivered in a timely manner. An example of this type of traffic is voice, video, control or any type of on-demand service[12, 13].

Fig. 1 Constant bit rate

2.2 Variable bit rate

The data flow for VBR traffic is shown in Fig. 2. It changes with time, and these changes are normally smooth, not sharp or sudden. The average data rate and the peak data rate are different for this flow. This traffic type is more difficult for the network to handle, because the network cannot readily predict the resources needed for a traffic flow. Examples of such types of traffic are compressed video and voice streams[12, 13].

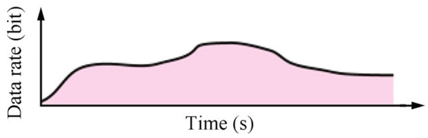


Fig. 2 Variable bit rate

3 Quality of service (QoS)

QoS is the network ability to provide better service for selected traffic. The purpose for having QoS is to provide guarantees on the ability of the network to provide a certain service quality. The network features used to measure the QoS are delay, throughput and PDR. These features are used as the measuring metric for performance analysis in this paper. Scheduling schemes can be used to improve the QoS of a network.

3.1 Scheduling scheme

A scheduling scheme is required to improve the QoS of MANET. This is an algorithm that determines the order in which a thread or data flow can access the available resources. Packets from various traffic flows arrive at a node, and the scheduler prioritises individual flows in the queue so they are served fairly in order to improve the QoS. Some of the conventional available scheduling algorithms are FIFO, priority queuing (PQ) and weighted fair queuing (WFQ)[14]. In FIFO, various packet flows are kept in the buffer until they are ready to be processed by the queue. Packets that arrived first at the queue are served first and any other packet that arrives afterwards will have to wait in the queue until all previous packets have been served. When the packet arrival rate is greater than queue processing rate, the queue will not be able to cope with the intensity of packet arrivals, thus congestion will occur. Hence packets will be discarded by the queue either because the queue buffer is already full or it has exceeded the waiting threshold in the queue. This conventional queuing scheme is not suitable for MANET because of the frequent changes of the network topology. Thus, an adaptive queuing scheme which adapts to the network topology change is needed.

4 Fuzzy inference system

FIS is a system that implements human experiences and preferences with membership functions and fuzzy rules. It can be used as a general methodology to incorporate knowledge, heuristics or theory into controllers and decision making[10]. A fuzzy model is made up of four blocks. These blocks consist of a fuzzifier, defuzzifier, inference engine and fuzzy knowledge base as shown in Fig. 3. The fuzzifier decides how to covert the crisp input into a fuzzy input to be used by the inference engine. This is achieved by mapping the crisp input to a set of input membership functions stored in the knowledge base. The inference engine applies reasoning to compute the fuzzy output using the "IF-THEN" type fuzzy rules which are stored in the knowledge base. It is used to convert the fuzzy inputs to fuzzy outputs. The defuzzifier converts the fuzzy outputs into a crisp value using an output membership function stored in the knowledge base.

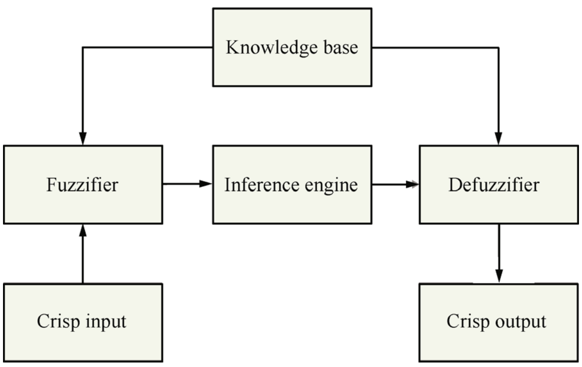


Fig. 3 Basic fuzzy system[9]

4.1 Fuzzy scheduler

The proposed fuzzy schedulers have three input variables and a single output variable which is the priority index of the packet. These input variables contribute to the QoS performance of a network. The three inputs for the fuzzy model are SNR, queue size and data rate as shown in Fig. 4. This is the queue size and data rate of the individual nodes that the packet is associated with as well as the SNR of the receiver. The inputs are fuzzified, implicated, aggregated and defuzzified to obtain a crisp value which is the output, i.e., priority index.

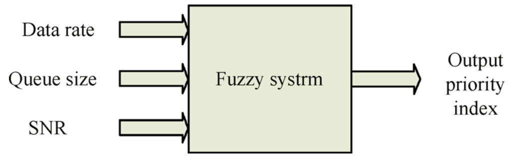


Fig. 4 The inputs and outputs of the proposed fuzzy schedulers

4.2 Membership function

There are a number of different membership functions. These include trapezoidal, triangular, piecewise linear, Gaussian and singleton. The most commonly used membership functions are trapezoidal, triangular and Gaussian shapes. The type of membership function used can be context dependent and is chosen arbitrarily by the user depending on their level of experience[15]. The triangular and trapezoidal membership functions (MFs) are considered in this paper for their simplicity and low computational complexity. The linguistic variables associated with the input variables are low (L), medium (M) and high (H). The input membership function for SNR, queue size and data rate are shown in Figs. 5−7, respectively. The x-axis represents the particular fuzzy input and was normalised for all input variables. The y-axis represents the certainty level and it varies between 0 and 1. There are two ways of mapping MFs, i.e., the number of MFs required for each input variable as well as the baseline. The first is knowledge elicited from experts in the field (manual mapping) and the second is knowledge extracted from trends in empirical data. The range of the fuzzy input on the x-axis was obtained through simple queuing formulas as well as trial and error to maximize the overall system performance. This was carried out by running multiple test simulation models. Reference [9] was also considered in determining the range of the fuzzy input.

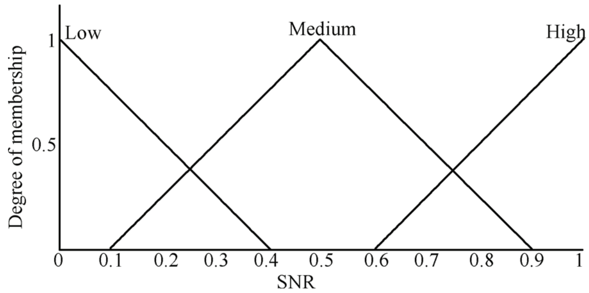


Fig. 5 Membership function for SNR

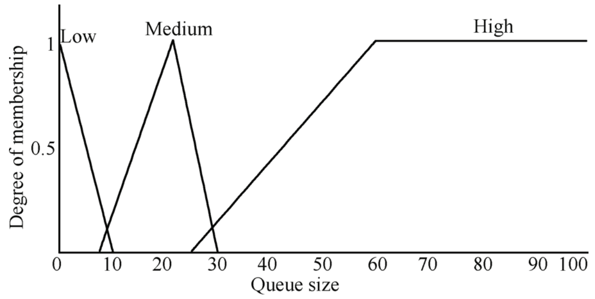


Fig. 6 Membership function for queue size

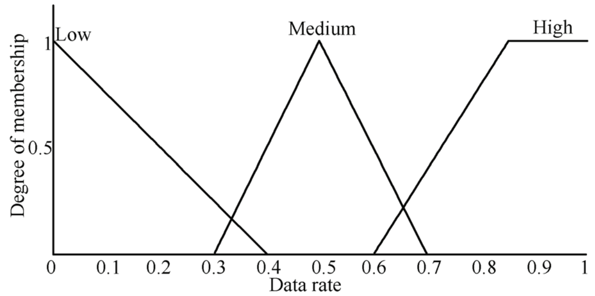


Fig. 7 Membership function for data rate

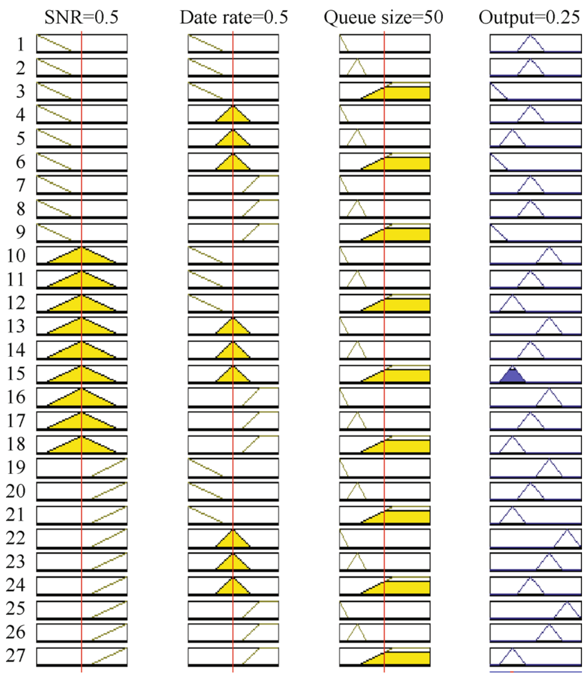


Fig. 8 Membership function and fuzzy rule base for the proposed schedulers

The rules are carefully designed based on the relationship between the input variables. The conditional rules for the fuzzy scheduler are shown in Fig. 8. The surface viewer which shows the relationship between the inputs and output is shown in Fig. 9. The fifteenth rule can be interpreted as if SNR is medium, data rate is medium and queue size is high, then the output is low. The other rules are formulated similarly. The output priority index ranges from 0 to 1, "0" meaning the highest priority in the queue and 1 means the least priority. Thus as the priority index increases from 0 to 1 the packet priority in the queue drops accordingly. There are three input variables with three associated linguistic variables which gives 3^3 combinations, prompting the 27 rules.

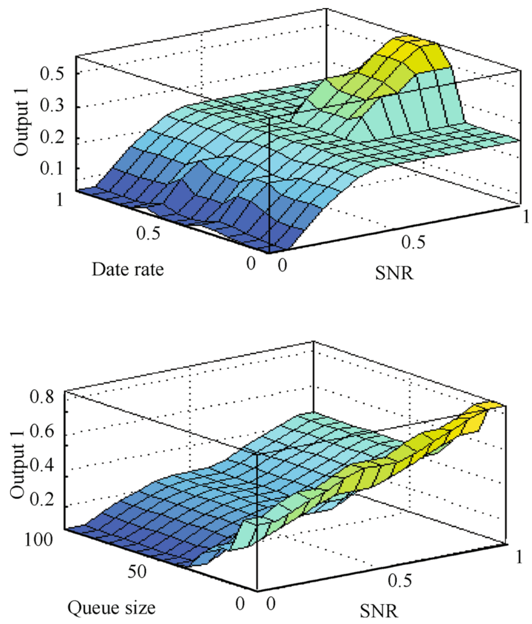


Fig. 9 Surface viewer for fuzzy scheduler

4.3 Types of FIS

There are two major types of fuzzy system, the Mamdani[16] and the Sugeno[17]. The Mamdani and Sugeno FIS for a given system have the same number of inputs and output membership functions. The rules are also the same, the only difference is the defuzzification of the fuzzy output. The next section highlights some of the variations between these two FIS systems.

4.3.1 Mamdani and Sugeno FIS differences

The most fundamental difference between the Mamdani and Sugeno FIS is how the crisp output is generated from the fuzzy inputs[18]. Some of the most popular Mamdani defuzzification techniques are usually a variation of the max criterion method. These include smallest of maxima (SOM), largest of maxima (LOM), and the mean of maxima (MOM). These methods select the smallest, largest and mean output value for inputs whose membership value reaches maximum. MOM is one of the most popular methods, it calculates the final output "Z" by averaging the set of output values that have the highest possibility degree "M" using the formula given in (1)[19].

$$Z = \sum_{j=1}^{l} \frac{x_j}{l}, \quad x_j \in M. \tag{1}$$

Two other commonly used defuzzification techniques are the center of gravity (COG)/centroid and center of area

(COA)/bisector method.

The COG/centroid method determines the crisp output by calculating the center of gravity of the possibility distribution of the output. For continuous values, the output "Z" is calculated using (2)[19].

$$Z = \frac{\int_{\mu} \mu(x) x \mathrm{d}x}{\int_{\mu} \mu(x) \mathrm{d}x}. \tag{2}$$

The COA is similar to the COG method. However, it calculates the position under the curve where the areas of both sides are equal. The COA can be calculated using (3)[19].

$$\int_{Z} \mu(x) \mathrm{d}x = \int_{Z} \mu(x) \mathrm{d}x. \tag{3}$$

Braae and Rutherford[20] presented a detailed analysis of various defuzzification techniques which include COG and MOM. They concluded that COG yields better results. For this reason, the COG/centroid defuzzification technique is used in this work.

The output membership function for the Mamdani scheduler is made of triangular membership functions, shown in Fig. 10. It consists of 5 linguistic variables, namely: very low (VL), low (L), medium (M), high (H) and very high (VH).

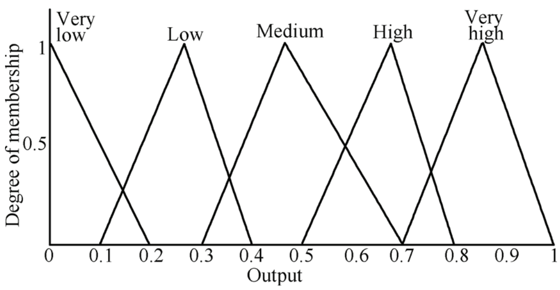


Fig. 10 Mamdani output membership function

Sugeno FIS uses the weighted average to compute the crisp output and thus the complex iteration process used by Mamdani is bypassed. The Sugeno FIS does not have an output membership function. The output for Sugeno FIS is shown in Fig. 11 and it is a constant value. It consists of five output points which are the same as the number of membership functions for the Mamdani output (very low (VL), low (L), medium (M), high (H) and very high (VH)). The Sugeno FIS is a less computationally complex algorithm than the Mamdani equivalent. The interpretability and the expressive power of the Mamdani FIS is lost in the Sugeno FIS because the consequent of the rules is not fuzzy[19]. It means that the output will be a constant rather than a fuzzy set when the rules are evaluated. Thus, the impact of this on the system performance will be evaluated.

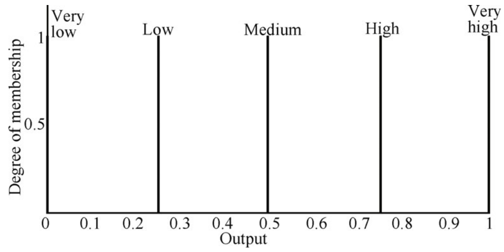


Fig. 11 Sugeno output membership function

5 Performance evaluation

The proposed fuzzy logic based packet scheduling algorithms are evaluated using a network simulation model called riverbed modeler and the measuring metrics are end-to-end delay, throughput and PDR. The results are presented in this section.

5.1 Simulation environment and methodology

Riverbed modeler is the leading simulation tool used in the academic circle for simulation of computer network and relevant technologies. It is used for modeling and analysing communication networks. It can model the performance of a network with a high degree of accuracy.

This simulation, models a network of 20 randomly distributed mobile nodes within a 500 m × 500 m area. The mobile nodes have wireless interfaces, which are configured to the IEEE 802.11n standard. A shadowing propagation model with path loss exponent (β) of 2.02 and a shadowing deviation (α) of 6.5 is used according to previously carried out outdoor experiment[21]. Each simulation is run for 600 s and multiple runs were carried out with varying seed values and the collected data was then averaged. The seed is used by the simulation random number generator, multiple seed value will provide multiple instances of the traffic generated.

Table 1 shows the simulation parameters used. CBR and VBR traffic are generated and the performance of the scheduling algorithms is analysed. All mobile nodes served as a transmitter and receiver. The data payload is 1024 bytes[9]. The performance of the schedulers was evaluated under various load conditions (30, 40, 50 and 60 pkts/s). The random waypoint mobility model is used and the node speed ranges from 0 to 20 m/s with a pulse time of 4 s.

Table 1 Simulation parameters

Number of nodes	20
Area	500 m × 500 m
Simulation time	600 s
Mobility model	Random waypoint
Speed of the nodes	0–20 m/s
Propagation model	Shadowing model (β=2.02, α= 6.5)
Traffic type	CBR & VBR
Channel bandwidth	12–54 Mbps
Data payload	1024 bytes
MAC protocol	IEEE 802.11n (buffer size=16 MB)

5.2 Performance evaluation of fuzzy scheduler

The input variables were obtained from the network and the fuzzy rules are evaluated based on these inputs. Each evoked rule has a corresponding output membership function. This output membership function is then implicated, aggregated and the crisp value (priority index) is calculated from these aggregated curves by using a centroid defuzzification technique. The Proto-C language of riverbed modeler which implements the fuzzy system was verified using the fuzzy logic tool box in Matlab.

5.3 Performance evaluation using riverbed modeler

The output priority index of a packet is used to schedule the packet. By scheduling the packets this way, packets in highly congested queues are scheduled first. This differs from the standard priority scheduler because the packet priority index is based on individual packets rather than a traffic flow. If the queue of a node is full, it will cause an increase in the end-to-end delay and packet loss rate, thus newly arriving packets are discarded and packets in the queue that have exceeded the waiting threshold are also discarded. The cause of the degradation of network performance is not limited to the length of the queue. It also relates to the data rate and SNR. When the SNR of the receiving node is low, the network will suffer a higher packet loss because of the poor wireless communication link between nodes. The packet priority increases as the SNR decreases in order to reduce the packet loss rate and thus improve the end-to-end delay.

The final input is the data rate. At higher data rates, the end-to-end delay of a packet is low and the PDR is significantly higher. However, when the reverse is the case, there will be a higher packet loss rate and an increase in the end-to-end delay. Packets are given a higher priority when the data rate is low. Packets present in a crowded node will experience a high queue delay and higher packet loss rate. This algorithm monitors these aforementioned parameters and calculates an appropriate priority index in order to optimise the network and improve the QoS performance. When a packet reaches a node, its priority index based on the network properties of that node is calculated and attached to its header. Each node has three sub-queues in order to reduce the effect of sorting on the overall network performance. Arriving packets are en-queued in these sub-queues based on their priority index. The first sub-queue en-queues packets with priority index between 0 and 0.33. The second sub-queue en-queues packets with priority index greater than 0.33 but less than or equal to 0.66. The third sub-queue en-queues packets with priority index greater than 0.66 but less than or equal to 1. The net result is that packets are sorted in the various sub-queues based on their priority index (i.e., packets with the lowest priority index move to the head of the queue and are scheduled first).

5.4 Performance analysis of fuzzy schedulers

This work is an extension of [11], which proposes a Mamdani fuzzy based scheduler. In addition to the Mamdani, a Sugeno fuzzy based scheduler is also proposed. The Sugeno scheduler is faster than Manoj and [11] because it is less computationally complex and therefore more appropriate for real-time applications. The schedulers have varying degrees of complexity, hence the algorithms were run in Microsoft Visual Studio for 100 cycles. A timer is inserted at the beginning and end of each cycle to measure the duration, and the average time was calculated.

The average time measured is equal to the additional processing delay which both the algorithms will add to individual packet per hop. This is added as a constant value to the formulae that calculate packet processing delay in riverbed modeler.

The algorithms were optimised by measuring the number of times each rule is used by CBR and VBR traffic within the specified simulation parameters shown in Table 1. Rules used 200 times or less are eliminated.

For CBR traffic, the result shows that only 10 of the 27 rules were used according to Table 2.

Table 2 CBR rule used count

Rules	Count
Rule 1	57
Rule 2	0
Rule 3	0
Rule 4	202 551
Rule 5	24 989
Rule 6	186 714
Rule 7	0
Rule 8	0
Rule 9	0
Rule 10	0
Rule 11	0
Rule 12	0
Rule 13	271 724
Rule 14	31 477
Rule 15	310 278
Rule 16	0
Rule 17	0
Rule 18	0
Rule 19	0
Rule 20	0
Rule 21	0
Rule 22	1512
Rule 23	85
Rule 24	852
Rule 25	0
Rule 26	0
Rule 27	0
Total	1 030 239

This was because CBR traffic consists of constant data rate. The 17 unused rules were eliminated. Further optimisation is carried out with the 10 remaining rules. Two rules are found to have been used less than 200 times (rule 1 and 23) and were also eliminated, reducing the total number of rules for the scheduler to 8. The performance of the optimised scheduler with 8 rules is compared with the performance of the scheduler with 10 rules. This is done by classifying the test into four cases as shown in Table 3.

Table 3 Test case CBR

Cases	Total rules
Case 1	Contains 10 rules
Case 2	Contains 9 rules−only rule 23 is removed
Case 3	Contains 9 rules−only rule 1 is removed
Case 4	Contains 8 rules−rules 1 and 23 are removed

Table 4 Rule optimisation results CBR

Cases	Delay	Throughput	PDR
Case 1	33.60	111538.90	0.37
Case 2	33.60	111538.90	0.37
Case 3	33.60	111538.90	0.37
Case 4	33.60	111538.90	0.37

Table 5 VBR rule used Count

Rules	Count
Rule 1	102 324
Rule 2	14 632
Rule 3	145 251
Rule 4	81 886
Rule 5	9 626
Rule 6	89 361
Rule 7	80 126
Rule 8	8 076
Rule 9	71 367
Rule 10	130 786
Rule 11	17 515
Rule 12	192 349
Rule 13	100 258
Rule 14	11 380
Rule 15	117 355
Rule 16	96 985
Rule 17	9 612
Rule 18	93 387
Rule 19	2 881
Rule 20	186
Rule 21	1 032
Rule 22	1 929
Rule 23	117
Rule 24	655
Rule 25	2 149
Rule 26	121
Rule 27	689
Total	1 382 035

Table 4 shows the average delay, throughput and PDR for all test cases. According to Table 4, there is no performance degradation for all the test cases as all the results are the same. Hence, the final number of rules for the CBR traffic is optimised from 27 to 8 without any performance degradation.

A similar optimization technique used for CBR is applied to VBR. All 27 rules were used by the VBR model, 3 of those rules (rules 20, 23 and 26) were used less than 200 times in the course of the simulation as shown in Table 5.

The three aforementioned rules were eliminated. A series of simulations test were carried out to check the performance degradation resulting from eliminating any or all of these 3 rules. The simulation work is classified into seven cases as shown in Table 6.

Table 7 shows the results for the average delay, throughput and PDR for all test cases. The results showed no significant changes in the performance. Thus, the fuzzy rule for the VBR traffic is optimised from 27 to 24 by eliminating all the rules that were used less than 200 times.

Table 6 Test case VBR

Cases	Total rules
Case 1	Contains 27 rules
Case 2	Contains 26 rules – only rule 20 is removed
Case 3	Contains 26 rules – only rule 23 is removed
Case 4	Contains 26 rules – only rule 26 is removed
Case 5	Contains 24 rules – rules 20, 23 and 26 are removed
Case 6	Contains 25 rules – rules 20 and 23 are removed
Case 7	Contains 25 rules – rules 20 and 26 are removed

Table 7 Rule optimisation results VBR

Cases	Delay	Throughput	PDR
Case 1	32.93515	82 462.04	0.276353
Case 2	33.10667	82 547.1	0.276452
Case 3	33.00937	82 479.36	0.276411
Case 4	32.93515	82 462.04	0.276353
Case 5	33.09063	82 553.96	0.276419
Case 6	32.93515	82 462.04	0.276353
Case 7	33.00937	82 479.36	0.276411

5.5 Performance analysis of CBR traffic

The service rate or capacity of the queue for the CBR model is 30 pkts/s for all traffic loads. The average end-to-end delay for CBR traffic for load 30 pkts/s and 40 pkts/s are shown in Figs. 12 and 13, respectively. The proposed schedulers (Mamdani and Sugeno) perform better than the existing (Manoj). However, as shown in Fig. 12, the Sugeno scheduler performs slightly better than the Mamdani scheduler. The performances of the Mamdani and Sugeno scheduler are very close according to Figs. 12 and 13.

The graph for the average end-to-end delay for 50 pkts/s and 60 pkts/s behaves similarly to that of 30 pkts/s and

40 pkts/s. Table 8 contains the values for the average end-to-end delay for all traffic loads. The difference between the Mamdani/Sugeno scheduler with the Manoj scheduler, as well as the percentage improvement for the Mamdani and Sugeno scheduler is also shown in Table 8.

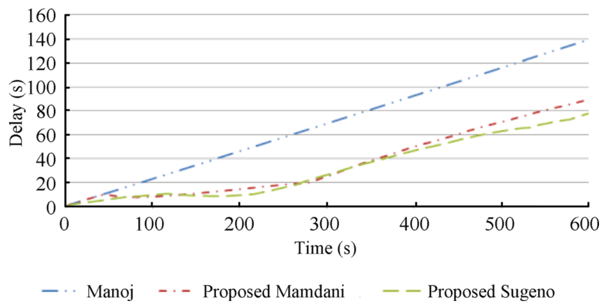


Fig. 12 End-to-end delay for 30 pkts/s CBR

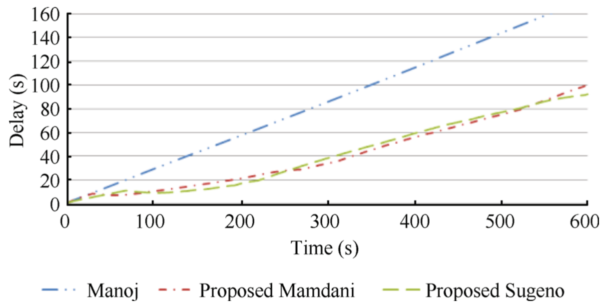


Fig. 13 End-to-end delay for 40 pkts/s CBR

Table 8 Average end-to-end delay CBR

Scheduler	Average end-to-end delay (s)			
	30 pkts/s	40 pkts/s	50 pkts/s	60 pkts/s
Manoj	69.43	86.11	99.75	109.00
Proposed Mamdani	35.92	41.11	46.54	49.96
Proposed Sugeno	32.03	41.36	42.04	49.98
%improvement Mamdani	48.26	52.26	53.35	54.16
%improvement Sugeno	53.87	51.97	57.86	54.15

According to Table 8, it can be noted that the proposed Mamdani and Sugeno scheduler performed better than the Manoj scheduler. For traffic load of 30 pkts/s and 50 pkts/s, the proposed Sugeno scheduler performed slightly better than Mamdani scheduler, whilst the proposed Mamdani scheduler performed slightly better at 40 pkts/s and 60 pkts/s.

The throughput and packet delivery ratio for all traffic loads behave similarly. Their performances are very close because of the nature of the traffic being sent. The queuing capacity is 30 pkts/s, thus the queue will forward packets at its maximum capacity for CBR traffic because all packets are of similar size. The throughput is approximately the same for all traffic loads as can be noticed from Figs. 14 and 15. According to Fig. 15, the throughput for the Sugeno scheduler is slightly lower than that of the Manoj and proposed Mamdani scheduler at the initial stage of the simulation. This occurred between simulation time 0 to 80 s for 40 pkts/s. The throughput becomes stable at 80 s simulation time.

The throughput for CBR traffic for all traffic loads are shown in Table 9, the throughputs for all the three schedulers are close. The negative sign shown in the table signifies Manoj scheduler performed better than the proposed ones, but it is by a small margin. At 60 pkts/s load, the proposed Mamdani scheduler slightly outperforms Manoj scheduler by 0.509 %.

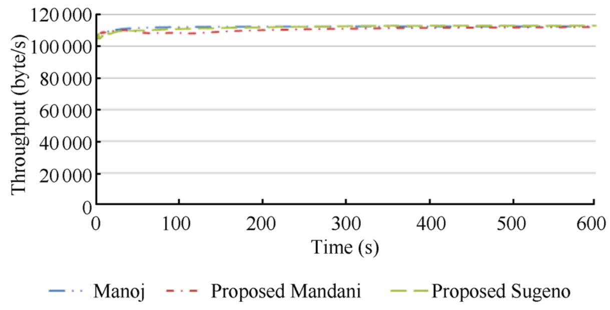


Fig. 14 Throughput for 30 pkts/s CBR

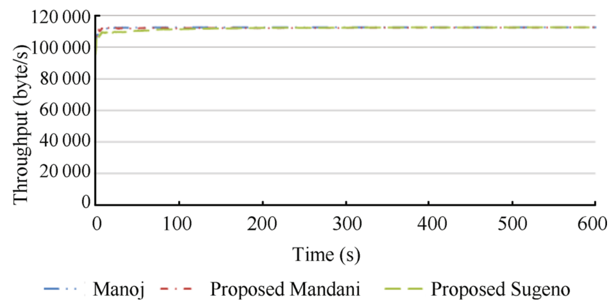


Fig. 15 Throughput for 40 pkts/s CBR

The service rate of the queue for all traffic loads is 30 pkts/s. CBR traffic has the same data rate. As a result, the queue will always forward packet at the maximum capacity for all traffic loads thus maintaining approximately the same throughput for all traffic loads.

Table 9 Throughput CBR

Scheduler	Throughput (bytes/s)			
	30 pkts/s	40 pkts/s	50 pkts/s	60 pkts/s
Manoj	112 011.8	112 368.2	112 297.2	110 832.9
Proposed Mamdani	110 225.4	112 095.8	112 097.5	111 397.3
Proposed Sugeno	111 505	111 638.8	111 306.8	110 612.3
%improvement Mamdani	−1.595	−0.242	−0.178	0.509
%improvement Sugeno	−0.452	−0.649	−0.882	−0.199

The PDR for 30 pkts/s and 40 pkts/s are shown in Figs. 16 and 17, respectively. The performance of the PDR for all schedulers is almost the same at each traffic load.

The PDR for all loads is summarised in Table 10. According to Table 10, the PDR decreases as the network load increases. This is a bottleneck effect. The Mamdani and Sugeno perform slightly better than Manoj.

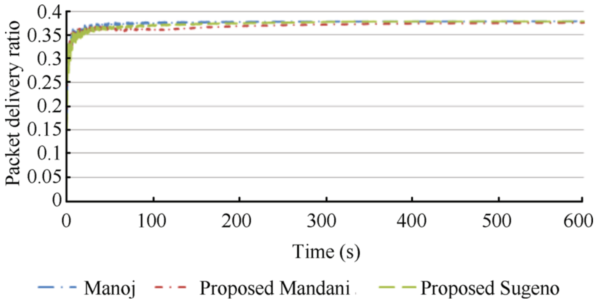


Fig. 16 Packet delivery ratio for 30 pkts/s CBR

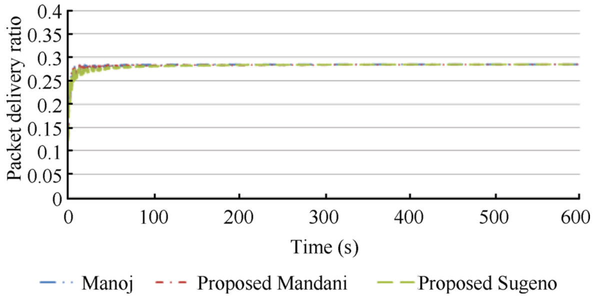


Fig. 17 Packet delivery ratio for 40 pkts/s CBR

Table 10 Packet delivery ratio CBR

Scheduler	Packet delivery ratio			
	30 pkts/s	40 pkts/s	50 pkts/s	60 pkts/s
Manoj	0.195	0.125	0.093	0.073
Proposed Mamdani	0.276	0.191	0.147	0.116
Proposed Sugeno	0.277	0.188	0.140	0.108
%improvement Mamdani	42.030	53.000	57.770	58.560
%improvement Sugeno	42.530	50.120	49.640	47.380

5.6 Performance analysis for VBR traffic

The service rate of the queue for VBR model is 30 pkts/s for all traffic loads. The performance analysis was done for the network under congested conditions. Hence, the reason for the high queuing delays. The average end-to-end delay for the traffic generation rate of 30 pkts/s is shown in Fig. 18.

The proposed (Mamdani and Sugeno) schedulers perform better than the existing scheduler (Manoj). The performance of the Mamdani scheduler in the first 0 – 30 s is slightly higher than Manoj, whilst Sugeno scheduler is close to Manoj between this simulation time. Thus, Manoj and Sugeno perform better than Mamdani between simulation time of 0 – 30 s. The performance of the Mamdani scheduler, improves significantly as compared to Manoj during simulation time of 30 – 600 s. The Sugeno scheduler also starts to improve significantly as compared to Manoj between simulation time 90 – 600 s. Sugeno also performs slightly better than Mamdani between simulation time 120 – 600 s.

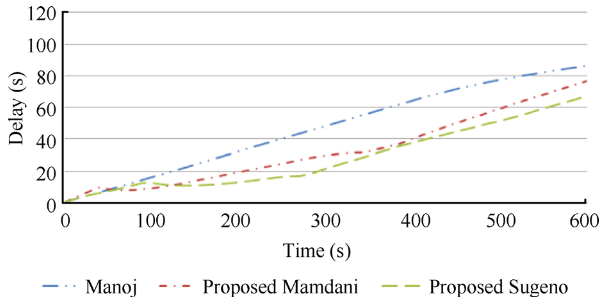


Fig. 18 End-to-end delay for 30 pkts/s VBR

The limited network resources cannot cope with the intensity of packets arrivals at the queue, thus congestion occurs. The average end-to-end delay increases linearly with time as shown in Fig. 18. The behavior of the delay graph for 40 pkts/s is shown in Fig. 19. It is similar to that of 30 pkts/s in Fig. 18. The end-to-end delay graph for 50 pkts/s and 60 pkts/s also shows a similar trait as in Figs. 18 and 19. The values for the average end-to-end delay for all traffic loads are shown in Table 11.

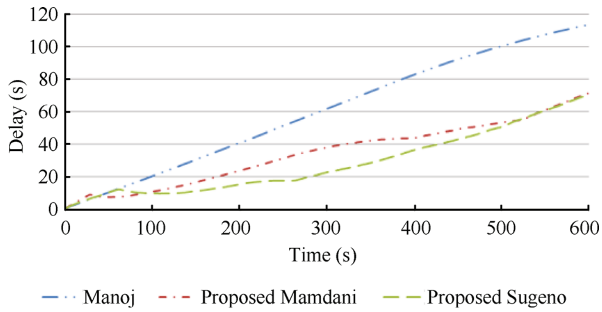


Fig. 19 End-to-end delay for 40 pkts/s VBR

Table 11 Average end-to-end delay VBR

Scheduler	Average end-to-end delay (s)			
	30 pkts/s	40 pkts/s	50 pkts/s	60 pkts/s
Manoj	47.04	60.50	69.14	76.41
Proposed Mamdani	32.94	34.29	40.03	49.78
Proposed Sugeno	28.25	28.20	28.23	32.96
%improvement Mamdani	29.98	43.33	42.10	34.85
%improvement Sugeno	39.94	53.39	59.16	56.87

According to Table 11, the proposed scheduler (Mamdani) performs 29.98 % better than Manoj for 30 pkts/s, 43.33 % better for 40 pkts/s, 42.10 % better for 50 pkts/s, and 34.85 % better for 60 pkts/s. The performance of the algorithm (Mamdani) improves as the network load increased from 30–40–50 pkts/s but drops slightly at 60 pkts/s. The proposed scheduler (Sugeno) also performs better than Manoj scheduler with an average percentage improvement of 39.94 %, 53.39 %, 59.16 % and 56.87 % for 30 pkts/s, 40 pkts/s, 50 pkts/s and 60 pkts/s, respectively. It also showed a better performance than the Mamdani scheduler for all traffic loads. As the traffic load increases from 30–40–50–60 pkts/s, the performance of the Sugeno scheduler as compared to Manoj and Mamdani scheduler improves. Thus, appropriately combining input features such as the SNR, data rate and queue size, the proposed schedulers scheduled packets better than Manoj. When a network is congested, the gradient of the end-to-end delay graph gets steeper as more packets arrive at the queue and the network tends towards congestion.

Thus, the gradient for the end-to-end delay graph shows the rate of increase of the network congestion. As the network load increases, the gradient also increases. Therefore, to avoid congestion or prevent a severe case of congestion, the gradient of the end-to-end delay graph needs to be prevented from increasing abruptly. The Manoj model becomes congested more quickly as shown in Figs. 18 and 19. The end-to-end delay graph for the proposed fuzzy schedulers (Mamdani and Sugeno) produced a lower gradient than Manoj, this was because incoming packets to the queue are given higher priority when the queue size is high, the SNR is low and the data rate is also low. These are the characteristics of the input variables when the network tends towards congestion.

The gradient for the end-to-end delay graph shown in Fig. 18 for the Mamdani scheduler is 20.02 % less than Manoj, whilst that of the Sugeno is 28.96 % less than Manoj for the same load. Thus, the network congestion is reduced by 20.02 % and 28.96 % for the Mamdani and Sugeno respectively. At an increased load of 40 pkts/s and 50 pkts/s, the gradient for end-to-end delay for the Mamdani scheduler is 45.61 % and 40.61 % lower than Manoj respectively. The performance slightly dropped to 33.92 % when the load was increased to 60 pkts/s.

The gradients for the Sugeno scheduler at an increased load of 40 pkts/s, 50 pkts/s and 60 pkts/s are 46.60 %, 52.47 % and 60.60 %. Thus, the gradient increases as the load increases, showing that the performance of the Sugeno scheduler improves as the load increases.

Fig. 20 shows an improvement in the throughput for the proposed schedulers (Mamdani and Sugeno) for 30 pkts/s and Fig. 21 shows that for 40 pkts/s.

The throughput for the proposed Mamdani and Sugeno scheduler is almost the same at 30 pkts/s as shown in Fig. 20. The performance of the throughput at 40 pkts/s is similar for the proposed Mamdani and Sugeno scheduler as shown in Fig. 21.

Table 12 shows the percentage improvement of the throughput for the proposed schedulers (Mamdani and Sugeno) as compared to Manoj scheduler for all loads.

The performance of the Mamdani scheduler improves as the network load increases. There were increases of 41.47 %, 54.16 %, 57.89 % and 58.81 % for the throughput of the proposed Mamdani scheduler for 30 pkts/s, 40 pkts/s, 50 pkts/s and 60 pkts/s, as compared to the Manoj. The percentage increases in throughput for the proposed Sugeno scheduler are 41.79 %, 50.17 %, 49.60 % and 48.23 % for 30 pkts/s, 40 pkts/s, 50 pkts/s and 60 pkts/s, respectively. The throughput of the proposed Mamdani scheduler is slightly higher than the throughput of the proposed Sugeno scheduler as shown in Fig. 21. This is different from the throughput performance of CBR traffic because of the variation in data rates of VBR traffic, thus the queue capacity might not be used at the maximum network load for the entire simulation duration. CBR traffic maximizes the use of the available resources more than VBR traffic. Hence, the scheduler increases the VBR throughput by maximizing the amount of traffic that can be forwarded from the queue in a second to make the network more efficient and also improves the network QoS performance.

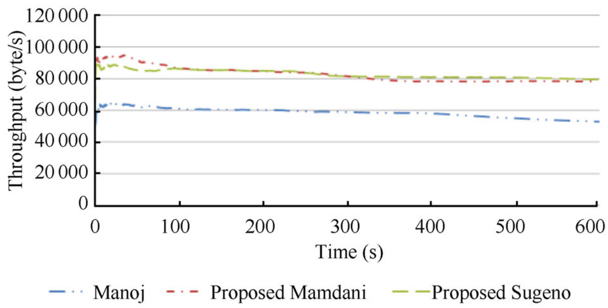


Fig. 20 Throughput for 30 pkts/s VBR

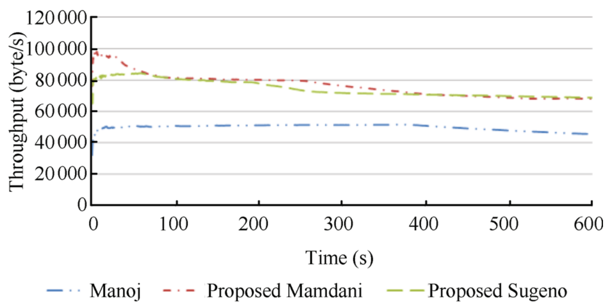


Fig. 21 Throughput for 40 pkts/s VBR

Table 12. Throughput VBR

Scheduler	Throughput (byte/s)			
	30 pkts/s	40 pkts/s	50 pkts/s	60 pkts/s
Manoj	58 289.81	49 540.70	46 055.62	44 113.80
Proposed Mamdani	82 462.04	76 370.53	72 717.57	70 058.59
Proposed Sugeno	82 651.59	74 395.56	68 899.18	65 389.32
%improvement Mamdani	41.47	54.16	57.89	58.81
%improvement Sugeno	41.79	50.17	49.6	48.23

Fig. 22 shows an increase in the PDR for the proposed schedulers (Mamdani and Sugeno) as compared to Manoj for 30 pkts/s. The PDR for 40 pkts/s is shown in Fig. 23. Table 13 shows that both proposed fuzzy schedulers perform better, thus resulting in a higher PDR than Manoj.

The percentage improvements of the PDR for the proposed Mamdani scheduler as compared to Manoj are 42.03 %, 53.00 %, 57.77 % and 58.56 % for 30 pkts/s, 40 pkts/s, 50 pkts/s and 60 pkts/s, whilst the percentage improvement for the proposed Sugeno scheduler are 42.53 %, 50.12 %, 49.64 % and 47.38 % for 30 pkts/s, 40 pkts/s, 50 pkts/s and 60 pkts/s, respectively. The proposed scheduler delivered more traffic per second than the Manoj. The PDR of the proposed Mamdani scheduler performs slightly better than the proposed Sugeno scheduler. The PDR improvements are similar to that of the throughput.

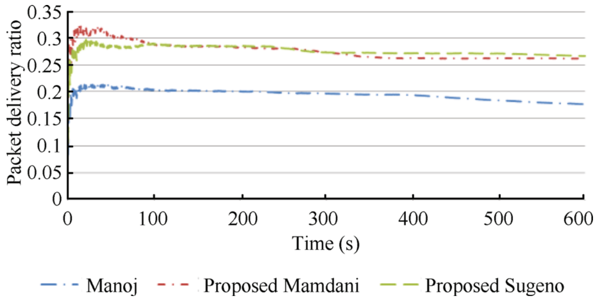


Fig. 22 Packet delivery ratio for 30 pkts/s VBR

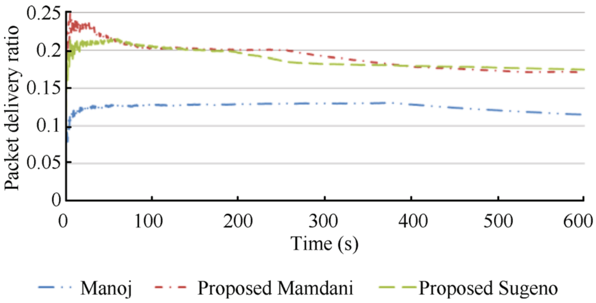


Fig. 23 Packet delivery ratio for 40 pkts/s VBR

Table 13 Packet delivery ratio VBR

Scheduler	Packet delivery ratio			
	30 pkts/s	40 pkts/s	50 pkts/s	60 pkts/s
Manoj	0.195	0.125	0.093	0.073
Proposed Mamdani	0.276	0.191	0.147	0.116
Proposed Sugeno	0.277	0.188	0.140	0.108
%improvement Mamdani	42.030	53.000	57.770	58.560
%improvement Sugeno	42.530	50.120	49.640	47.380

6 Conclusions

Two optimised fuzzy logic scheduling algorithms based on the Mamdani and Sugeno are proposed for the MANET. The performance of these schedulers was compared to an existing fuzzy scheduler. Both schedulers consider three inputs (data rate, queue size, and SNR) as opposed to the existing scheduler, which considered two inputs (data rate and channel capacity). The inputs to the fuzzy system were fuzzified, implicated, aggregated and defuzzified to obtain the crisp value. The crisp value ranges from 0 to 1 and it represents the packet priority index. Zero "0" is the highest priority and one "1" the least priority. Each node consisted of three sub-queues to reduce the effect of sorting on the network performance. Individual packets are inserted in each sub-queue and served based on their Priority Index. The membership functions and the fuzzy rules were carefully designed. The number of rules has been optimised without affecting the performance of the CBR and VBR traffic.

The performance of the proposed scheduling algorithms (Mamdani and Sugeno) was analysed for CBR and VBR traffic. The measuring metric for performance analysis are end-to-end delay, throughput and PDR.

The proposed schedulers perform better in terms of end-to-end delay for CBR traffic, whilst the throughput and PDR are all very similar. This is because of the nature of CBR traffic, which consists of constant data rate over the entire simulation duration. Thus, the maximum network resource is utilized for all simulation time.

The proposed schedulers perform better than Manoj in terms of end-to-end delay, throughput and PDR for VBR traffic. The proposed Sugeno scheduler performs better than the proposed Mamdani in terms of end-to-end delay, whilst the throughput and PDR for all traffic loads showed similar performance as of the proposed Mamdani scheduler.

Although the proposed Mamdani scheduling algorithm is more computationally complex than Manoj, it compensates for its complexity by optimally scheduling the network better than Manoj.

According to the simulation results, there is no significant difference between the performance of the Mamdani and Sugeno scheduler for VBR and CBR traffic, the Sugeno scheduler will be the better choice for real-time applications because of the simplicity of its design and it is less computationally complex.

References

[1] S. K. Goel, M. Singh, D. Y. Xu, B. C. Li. Efficient peer-to-peer data dissemination in mobile ad-hoc networks. In *Proceedings of the International Conference on Parallel Processing Workshops*, IEEE, Washington, DC, USA, pp. 152–158, 2002.

[2] F. Gianfelici. Measurement of quality of service (QoS) for peer-to-peer networks. In *Proceedings of the International Conference on Virtual Environments, Human-computer Interfaces and Measurement Systems*, IEEE, Giardini Naxos, Italy, pp. 133–138, 2005.

[3] R. Asokan. A review of Quality of Service (QoS) routing protocols for mobile Ad hoc networks. In *Proceedings of the International Conference on Wireless Communication and Sensor Computing*, IEEE, Chennai, India, pp. 1–6, 2010.

[4] H. N. Xiao, W. K. G. Seah, A. Lo, K. C. Chua. A flexible quality of service model for mobile ad-hoc networks. In

Proceedings of the 51th Vehicular Technology Conference, IEEE, Tokyo, Japan, vol. 1, pp. 445–449, 2000.

[5] C. Chow, H. Ishii. Video streaming over mobile ad hoc networks: Multipoint-to-point transmission with multiple description coding. In *Proceedings of IEEE Region 10 Conference*, IEEE, Hong Kong, China, pp. 1–4, 2006.

[6] C. E. Perkins, E. M. Royer, S. R. Das, M. K. Marina. Performance comparison of two on-demand routing protocols for ad hoc networks. *IEEE Personal Communications*, vol. 8, no. 1, pp. 16–28, 2001.

[7] M. T. Hyland, B. E. Mullins, R. O. Baldwin, M. A. Temple. Simulation-based performance evaluation of mobile ad hoc routing protocols in a swarm of unmanned aerial vehicles. In *Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops*, IEEE, Niagara Falls, Ontario, Canada, vol. 2, pp. 249–256, 2007.

[8] K. Ramachandran, I. Sheriff, E. Belding, K. Almeroth. Routing stability in static wireless mesh networks. In *Proceedings of the 8th International Conference on Passive and Active Network Measurement*, Lecture Notes in Computer Science, Springer-Verlag, Louvain-la-Neuve, Belgium, vol. 4427, pp. 73–83, 2007.

[9] K. Manoj, S. C. Sharma, L. Arya. Fuzzy based QoS analysis in wireless ad hoc network for DSR protocol. In *Proceedings of IEEE International Conference on Advance Computing*, IEEE, Patiala, India, pp. 1357–1361, 2009.

[10] C. Gomathy, S. Shanmugavel. An efficient fuzzy based priority scheduler for mobile ad hoc networks and performance analysis for various mobility models. In *Proceedings of the Wireless Communications and Networking Conference*, IEEE, Tamil Nadu, India, vol. 2, pp. 1087–1092, 2004.

[11] O. A. Egaji, A. Griffiths, M. S. Hasan, H. Yu. Fuzzy logic based packet scheduling algorithm for mobile ad-hoc network with a realistic propagation model. In *Proceedings of the 19th International Conference on Automation and Computing*, IEEE, London, UK, pp. 1–6, 2013.

[12] B. A. Forouzan. Data Communications and Networking, 4th ed., New York, USA: McGraw-Hill Higher Education, 2006.

[13] B. A. Forouzan, S. C. Fegan. *Data Communications and Networking*, New York, USA: McGraw-Hill Higher Education, 2007.

[14] J. M. Kim, I. K. Park, C. H. Kim. A study on the performance enhancements of video streaming service based on MPLS network. In *Proceedings of the International Symposium on Intelligent Signal Processing and Communication Systems*, IEEE, Seoul, South Korea, pp. 601–603, 2004.

[15] J. M. Mendel. Fuzzy logic systems for engineering: A tutorial. *Proceedings of the IEEE*, vol. 83, no. 3, pp. 345–377, 1995.

[16] E. H. Mamdani, S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-machine Studies*, vol. 7, no. 1, pp. 1–13, 1975.

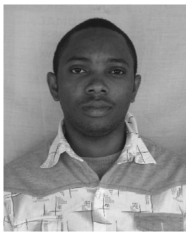
[17] T. Takagi, M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 15, no. 1, pp. 116–132, 1985.

[18] A. Hamam, N. D. Georganas. A comparison of Mamdani and Sugeno fuzzy inference systems for evaluating the quality of experience of hapto-audio-visual applications. In *Proceedings of IEEE International Workshop on Haptic Audio Visual Environments and Games*, IEEE, Ottawa, Ontario, Canada, pp. 87–92, 2008.

[19] N. Mogharreban, L. F. DiLalla. Comparison of defuzzification techniques for analysis of Non-interval data. In *Proceedings of the Annual Meeting of the North American Fuzzy Information Processing Society*, IEEE, Montreal, Que, Canada, pp. 257–260, 2006.

[20] M. Braae, D. A. Rutherford. Fuzzy relations in a control setting. *Kybernetes*, vol. 7, no. 3, pp. 185–188, 1978.

[21] O. A. Egaji, A. Griffiths, M. S. Hasan, H. Yu. Development of a realistic simulation model for IEEE 802.11n based on Experimental Results. In *Proceedings of the 7th International Conference on Software, Knowledge, Information Management and Applications*, Chiang Mai, Thailand, 2013.



Oche Alexander Egaji received the B. Sc. degree in electrical/electronic engineering from Eastern Mediterranean University, Turkey, the M. Sc. degree in communication engineering from the University of Manchester, UK. He is currently a Ph. D. candidate on optimisation of real-time wireless network control systems over mobile ad-hoc network at Staffordshire University, UK .

His research interests include traffic modelling, generation of a novel real-time scheduling algorithm to minimise delay, jitters, packet loss as-well as to increase the system robustness.

E-mail: o.a.egaji@staffs.ac.uk (Corresponding author)
ORCID iD: 0000-0001-6661-7415

Alison Griffiths received both the M. Eng. and first B. Eng. (Hons) degrees from Staffordshire University, UK in 1999 and 1998, respectively. From 1999 to 2003, she was a research associate on Engineering and Physical Sciences Research Council (EPSRC) funded project whilst being a Ph. D. candidate on the convergence of mobile computing and telecommunications at Staffordshire University, UK. Since 2003, she has been a senior lecturer in telecommunications at Staffordshire University, UK.

Her research interests include mobile agents, cellular and IP packet switched networks, communication of different types of media (voice, video conferencing, etc.), and control of mobile wire-less ad-hoc networks.

E-mail: a.l.griffiths@staffs.ac.uk

Mohammad S. Hasan received the B. Sc. and M. Sc. degrees in computer science. He obtained his second M. Sc. degree in computer and network engineering from Sheffield Hallam University, UK, and the Ph. D. degree in networked control systems over mobile ad-hoc network (MANET) at Staffordshire University, UK. Currently, he is a full time lecturer and a member of the Mobile Fusion (MF) Applied Research Centre (ARC) at Staffordshire University, UK.

His research interests include computer networks, networked control systems, remotely controllable mobile robot systems, real time systems, and wireless sensor networks.

E-mail: m.s.hasan@staffs.ac.uk

Hong-Nian Yu is currently a professor in computer science at Bournemouth University, UK. He has extensive research experience in modelling, control of robots, mechatronics devices, neural networks, mobile computing, modelling, scheduling, planning and simulations of large discrete event dynamic systems, radio frequency identification (RFID) with applications to manufacturing systems, supply chains, transportation networks, and computer networks. He has published over 200 research papers and held several grants from the Engineering and Physical Sciences Research Council (EPSRC), the Royal Society, and other funding bodies. He is a member of the EPSRC peer review college and serves on various conferences and academic societies.

His research interests include mobile computing, modelling, scheduling, planning, and simulations of large discrete event dynamic systems with applications to manufacturing systems, supply chains, transportation networks, computer networks and RFID applications and neural networks.

E-mail: yuh@bournemouth.ac.uk