

RESEARCH

Open Access

Fuzzy-based congestion control for wireless multimedia sensor networks

Cagatay Sonmez^{1*}, Ozlem Durmaz Incel², Sinan Isik^{1,3}, Mehmet Yunus Donmez^{1,4} and Cem Ersoy¹

Abstract

Congestion is a challenging problem for sensor networks because it causes the waste of communication and reduces energy efficiency. Compared to traditional wireless sensor networks, the probability of congestion occurrence in wireless multimedia sensor networks is higher due to the high volume of data arising from multimedia streaming. In this article, problems for multimedia transmission over wireless multimedia sensor networks are examined and sensor fuzzy-based image transmission (SUIT); a new progressive image transport protocol is proposed as a solution. SUIT provides fuzzy logic-based congestion estimation and an efficient congestion mitigation technique which decreases the image quality on-the-fly to an acceptable level. In case of congestion, SUIT drops some packets of the frames in a smart way and thus transmits frames to the sink with lower, but acceptable quality. In this way, SUIT improves the continuity of the video streaming. We evaluate the performance of SUIT by comparing it with two different competitors. The first one is an example transport protocol, namely Fuzzy Logic-Based Congestion Estimation. The second one is a buffer occupancy-based congestion control mechanism which is commonly used in previous studies. According to the simulation results, SUIT provides better energy consumption, frame delivery, frame loss and frame latency performance than its competitors.

Keywords: Wireless multimedia sensor networks; Transport layer; Image transmission; Fuzzy logic; Congestion control

1 Introduction

With the recent developments in sensor technology, typical sensor nodes are able to capture multimedia data *via* low-cost hardware such as CMOS cameras and microphones. Such low-cost devices which are capable of capturing multimedia information from the environment have made it possible to develop applications for wireless multimedia sensor networks (WMSNs) [1]. A WMSN is formed with a large number of distributed embedded devices which are equipped with camera modules. These devices are able to retrieve multimedia content from the environment and are able to extract video and audio streams, still images as well as the scalar sensor data from the multimedia content. WMSNs have generated much interest in recent years due to their potential to enable a large class of applications such as surveillance systems, traffic control systems, environment monitoring, control of manufacturing processes in industry.

Congestion is one of the major challenges for wireless networks. With respect to the characteristics of WMSNs, multimedia traffic produces bursty high-load traffic in the network [2]. Therefore, probability of congestion in WMSNs is higher than traditional Wireless Sensor Networks (WSNs) due to the large amount of video traffic. Addition to the waste of communication and energy that it causes, congestion negatively affects reliability due to the packet losses and degrades the overall performance of the network and quality-of-service (QoS) of the application. Congestion control is a difficult task for wireless networks because identifying the occurrence of congestion is not as simple as in wired networks. Generally, congestion control protocols for WSNs consider specific parameters such as local buffer occupancy, packet arrival rate, and packet service time. However, deciding on the level of congestion based on a specific parameter may give incorrect results [3]. Instead of using a specific congestion indicator, using multiple indicators, such as a fuzzy logic-based mechanism proposed in this article, can provide better congestion detection performance.

*Correspondence: cagatay.sonmez@boun.edu.tr

¹Computer Networks Research Laboratory, Netlab, Department of Computer Engineering, Bogazici University, Bebek 34342, Istanbul, Turkey
Full list of author information is available at the end of the article

Implementation of multimedia coding techniques is another major challenge for WMSNs. The encoder used in wireless sensors should have high compression capability, low complexity and high error resiliency. Predictive video coding techniques such as H.263 or MPEG (Moving Picture Experts Group) family provide quite good compression. However, these coding techniques are asymmetric such that the encoder is more complex than the decoder. They use a complex motion estimation algorithm in the encoding phase. Since sensor nodes have energy, memory, and computational-speed constraints, such encoding techniques are not appropriate for sensor devices. Therefore, a significant number of the WMSN applications are using JPEG (Joint Photographic Experts Group) images due to its lower complexity and adequate compression capabilities [4-6].

Motivated by these discussed facts, in this article we focus on the following question: 'How much efficiency can be achieved in detecting and mitigating congestion in WMSNs by using a fuzzy logic-based congestion detection method and image quality adaptation?' To answer this question, we propose sensor fuzzy-based image transport (SUIT) which is a new progressive image transport protocol that provides fuzzy logic-based congestion estimation and subsequent mitigation techniques. SUIT is designed for target monitoring applications where sensor nodes start transmitting video frames when they detect any target in their field-of-view (FoV). Since sensor nodes generate event-based video data, there is bursty traffic in the intended application. According to the mobility pattern of the targets in our application scenario, they are expected to leave the FoV after a short period. If we used a congestion detection scheme where congestion is announced by the sink node, the congestion notification sent by the sink node may be received after the target leaves the sensor node's FoV because the target is moving and there are many hops from source to sink which may cause high packet delay. Due to the characteristics of short-interval bursty traffic, SUIT does not provide a back-pressure mechanism to inform sensor nodes about the congestion. Main goal of SUIT is increasing the frame delivery performance by decreasing the quality (while maintaining an acceptable threshold), as frames are flowing over the intermediate sensors. In case of congestion, SUIT adapts the frame sending rate and instead of dropping the whole frame, it reduces the frame quality (as well as the frame size) by dropping subframes of progressively encoded JPEG without corrupting the image file. Thus, SUIT transmits images to the sink with lower, but acceptable quality. In this way, SUIT improves the continuity of the video streaming. To the best of our knowledge, SUIT is the first transport protocol proposed for WMSNs which is able to mitigate congestion by adjusting image quality while image data are being transmitted. Extensive

simulations are performed to evaluate the performance of the SUIT protocol. For congested network cases, SUIT provides better performance, in terms of frame delivery, frame loss, frame latency, than its competitors while providing acceptable image quality.

The contributions of this paper are the following:

- We propose a fuzzy logic-based congestion estimation approach to detect congestion efficiently.
- We introduce a new combination of three different congestion indicators to get a more accurate measurement of the congestion.
- We propose a novel congestion mitigation technique which can perform quality adaptation on-the-fly and provide a considerable frame delivery and latency performance gain.
- We present a protocol which utilizes the cross-layer functionalities and makes use of cross-layer information exchange among the application, transport, MAC and routing layers.

The organization of the paper is as follows: In Section 2, literature review on congestion and transport protocols for wireless sensor networks is detailed. In Section 3, design and architecture of the SUIT protocol are introduced. In Section 4, the performance of the proposed SUIT protocol is evaluated with a real-life scenario. Section 5 concludes our study and provides possible directions for future research.

2 Related work

A transport protocol is responsible for the reliable delivery of data from source to destination. However, an efficient transport protocol should also handle network congestion because congestion deteriorates the QoS by causing buffer overflow that may lead to larger queuing delays and higher packet loss [7]. Moreover, it can lead to transmission collision at the MAC layer which requires retransmission of the colliding packets. Since wireless sensor nodes are energy-constrained devices, retransmission wastes the limited energy. Energy consumption is an important challenge for the sensor devices due to their limited and mostly irreplaceable battery. Therefore, improvement in energy efficiency while transmitting data over WSNs has been widely studied [8,9]. Misra et al. [2] discuss the performance of protocols performing congestion control while multimedia streaming over WMSN. Basically, congestion control protocols should detect the congestion successfully, notify the associated sensor nodes and then apply the subsequent mitigation techniques.

Detecting congestion is the most difficult step of congestion control since there is not an exact congestion indicator available for WMSNs. Protocols which address congestion control mostly use specific metrics for

congestion estimation. In WSNs, commonly used congestion indicators are local buffer occupancy, channel load, packet arrival rate, and packet service time. Event to Sink Reliable Transport (ESRT) [10] protocol and Fusion [11] detect congestion by monitoring local buffer occupancy, whereas Congestion Control and Fairness (CCF) [12] and Rate-Controlled Reliable Transport (RCRT) [13] protocols use packet service time to estimate congestion. Congestion Detection and Avoidance (CODA) [14] uses both local buffer occupancy and channel load for congestion detection. Similarly in [15], local queue lengths and channel conditions are used for predictive congestion control. Fuzzy-Based Congestion Estimation (FCE) [16] and SenTCP [17] calculate congestion level using the buffer occupancy and packet arrival rate. In [18], congestion is estimated by comparing the input traffic rate and the maximum allowable transmission rate. Interference-Minimized Multipath Routing (I2MR) [19] detects congestion by monitoring the size of data transmit buffer using exponential weighted moving averages. In [20], the occurrence of a congestion is decided when the queue occupancy is greater than a given threshold or when the collision rate is above a given threshold.

After detecting the congestion, neighbor sensor nodes should be informed about it. Two common methods are used for this issue. Congestion can be informed explicitly or implicitly. Protocols which perform notification process explicitly send a message to the relevant sensor nodes. Other protocols perform this step implicitly by inserting a congestion notification flag into the header of outgoing packets as a piggyback. Explicit notification brings an extra communication overhead to the network. Implicit notification requires fewer packet transmissions, but takes longer to effect. CODA and Fusion use explicit notification to inform neighbor nodes, whereas Siphon [21] and senTCP inform neighbor nodes implicitly. In [18], congestion is also notified implicitly, but the message includes the new rate of each child node instead of the congestion level. In I2MR, if an intermediate node detects congestion, it removes all pending packets from its own data transmit buffer and then sends a congestion notification packet, which is relayed reliably by all the upstream nodes along the path, to inform the source explicitly. In [20], congestion is notified explicitly. When the congestion has occurred, a congestion notification message, which contains the node id and path id, is sent back to the sources for each path id known by the node. On the other hand, SUIT does not use either explicit or implicit congestion notification because both methods are too slow to react for preventing congestion in event-based target monitoring applications where targets leave the FoV of sensors within a short interval.

For congestion mitigation, various methods are proposed. These methods are adapting data transmission

rate, redirecting data traffic to different sensors, and using back-pressure techniques to inform neighbor sensor nodes not to forward data. The proposed method in [18] is adjusting the source traffic rates at the upstream nodes to achieve low packet loss probability. Adapting the data transmission rate, which is performed by CODA, Fusion, CCF and I2MR is the most commonly used method for mitigating congestion. In [20], as the congestion notification message is arrived, the non-congested links are marked with the *inUse* flag. Then the traffic is repartitioned to the non-congested link to prevent congestion and provide fairness. Siphon proposes a virtual sink concept. Virtual sinks are specialized nodes and they are different from the physical sink. They have both primary low power mote radio and secondary long-range radio. For congestion mitigation, instead of changing the event rate, Siphon redirects the packet to the virtual sinks. On the other hand, the SPEED protocol [22] uses a back-pressure re-routing scheme to reduce the congestion in the network.

Fuzzy logic-based congestion control is another technique used in WSNs. For example, Zheng et al. propose a dynamic access category (AC) selection mechanism to provide dynamic protection of the video frames while transmitting the video over IEEE 802.11e networks [23]. Since the video data rate, coding structure and the network load may change very frequently, they use a fuzzy logic controller to adjust parameters of their dynamic frame assignment algorithm. In this controller, queue length of ACs and frame loss rates of frame priorities are used as an input. CONSEQ [24] is another work which avoids congestion via fuzzy logic-based dynamic rate adaptation. As the fuzzy control inputs, CONSEQ uses error and change-in-error of effective queue length which is computed based on the virtual queue lengths, observed packet drops, and the effect of probabilistic load balancing. Urathal et al. propose a Fuzzy-Based Congestion Control (FCC) algorithm which has three fuzzy parameters as congestion indicators [25]. The first parameter is the node degree which is basically calculated by using the location of the sensors and the number of hop count among them. The second and the third parameters are queue length and the data arrival rate, respectively. In case of high congestion level, FCC performs rate adaptation and also it uses a back-pressure technique to inform neighboring upstream nodes to send fewer number of packets. Zarei et al. propose a Fuzzy-Based Trust Estimation (FTE) as a congestion control scheme to find the corrupt or malicious sensors [26]. After finding these nodes, FTE isolates them to remove the traffic overhead arising from their packets. Each sensor monitors its neighbor node and finds their trustworthiness. FTE uses three fuzzy input variables which are forwarded packet ratio, delay and validity. Forwarded packet ratio is defined as

the number of packets forwarded by the suspected node divided by the number of packets forwarded by the other sensor nodes. For the packets forwarded by the suspected node, the delay is calculated by dividing the actual delay by the expected delay and the validity is calculated by dividing the number of valid packets to the number of all packets. According to the fuzzy result, the malicious nodes are detected and useless packets are dropped to increase the packet delivery rate.

FCE [16] is the closest work to ours and it also uses a fuzzy logic-based congestion control. Like SUIT, FCE calculates the congestion periodically via a fuzzy logic-based calculation and three different congestion levels described as *slight congestion*, *somewhat congested*, and *very heavy congestion*. However, FCE and SUIT differ in the parameters used for the fuzzy logic system (FLS) and the congestion mitigation technique. FCE uses buffer occupancy and packet arrival rate values as an input for the FLS. However, SUIT uses buffer occupancy of the parent (next-hop) sensor node, the ratio of incoming to outgoing packets and the number of contenders. As the congestion mitigation technique, both protocols adapt the data sending rate, but SUIT also performs a new technique which decreases the image quality by dropping the relevant packets while images are being transmitted. Additionally, FCE classifies packets into three categories depending on the type of service layer. According to the congestion level, it applies a packet drop policy to provide the best QoS. However, SUIT has only one service which contains only real time video packets and therefore does not classify packets.

Monitoring local buffer occupancy is a commonly used technique in congestion detection [10,11]. We implement a competitor transport protocol which utilizes similar congestion control mechanism with ESRT and we called it as Queue-Based Congestion Control (QCC) protocol. Similar to ESRT, QCC periodically checks the buffer occupancy to estimate congestion, and adapts the data sending rate in case of congestion. However, QCC does not use the back-pressure technique to inform sensor nodes about the congestion level. Since back-pressure technique is not necessary for video streaming application scenarios and provides poor performance, QCC discards the back-pressure feature of ESRT. QCC protocol is briefly explained in Section 4.

3 SUIT Protocol design principles

Sensor fuzzy-based image transport (SUIT) is an image transport protocol which provides a fuzzy logic-based congestion control mechanism. The main idea behind SUIT is transmitting the maximum number of frames to the sink by decreasing the frame quality to an acceptable level in case of congestion. With this method, it is possible to transmit frames with lower quality instead of dropping them. Thus, SUIT increases the average frame

delivery which is an important QoS metric for video streaming applications. In addition, a novel congestion mitigation technique which is presented in Section 3.6 makes it possible to adapt video quality on the intermediate sensors.

SUIT streams a sequence of JPEG images instead of conventional predictive video coding techniques. SUIT can work with both sequential JPEG (SJPEG) and progressive JPEG (PJPEG) formats. It also supports restart marker feature of JPEG coding. In SJPEG, each image component is encoded from left to right and top to bottom route. Therefore, while downloading a large image with slow connection, the image is displayed line by line. In order to see the whole image, the file should be downloaded completely. For SJPEG, a single bit error can lead to a complete loss of synchronism at the decoder which results in a completely unusable picture. To overcome this problem, JPEG has a restart marker feature. Restart marker is used in JPEG to divide data segments into independent blocks. It is mostly used on unreliable networks to provide resynchronization of the decoders. The advantage of the restart marker is its high packet loss resilience. If there is a bit error or data loss inside the marked block, only the related block becomes useless rather than the whole frame. Restart marker is used with both SJPEG and PJPEG images.

In PJPEG, image is encoded into a series of scans. Therefore, an image can be sent in layers rather than scan lines. When the first layer is received, a blurred but recognizable image can be seen. Whenever the subsequent layers arrive, the image quality gradually increase and more precise views can be achieved. The advantage of PJPEG is that an image can be viewed on-the-fly while it is being transmitted. The client can see a low quality version of the whole image very quickly with gradual improvement of quality. Another important advantage of PJPEG is its error resilience capability. Unlike SJPEG, PJPEG consists of several layers which compose a JPEG image frame. In case of a data loss or error in any layer, the whole frame is not corrupted. Only the related layer is corrupted but JPEG frame can still be reconstructed from the remaining layers. By using error resilience advantage of PJPEG, a novel congestion mitigation method is proposed. If the network is congested, the image quality can be decreased on the fly by dropping one layer or layers from the PJPEG. Only PJPEG makes it possible to decrease the image quality on intermediate sensors, because of its layered structure. The performance of different types of JPEG encoding techniques are evaluated with extensive number of simulations and progressive JPEG encoding comes out on top with respect to the frame sending rate, frame receiving rate, frame delivery and frame latency performance in case of congestion. Therefore, the progressive encoding is recommended for the SUIT protocol and the

performance of SUIIT is evaluated by using PJPEG images in Section 4.

3.1 Connectionless communication

Considering the example target monitoring application, SUIIT sends all the video data as a burst when the target is detected in the FoV. In case of congestion, frame rate is adapted on the source nodes and image frame quality is decreased on the intermediate nodes by the proposed congestion mitigation technique explained in Section 3.6. It provides connectionless communication and does not provide packet reliability. SUIIT also does not require confirmation message of a receiver about the sent packet. Receiver (sink) can detect the missing packets by monitoring the sequence number of the incoming packets but it does not inform the sender about the transmission result. Generally, reliable transport protocols send ACK/NACK-based feedback packet to the source. However, for streaming video data, this method is not appropriate because source node cannot wait for the response of the sink before sending packets or frames. For this reason, packet-based reliability cannot be applied for multimedia streaming. Instead of providing guaranteed delivery of all packets, connection between the sender and receiver can be monitored *via* control packets. For continuous video streaming, status of the connection should be known to detect errors during streaming and to provide better QoS. For example, video conferencing applications check the connection speed in order to adapt incoming and outgoing video resolution. However, SUIIT does not monitor the connection either because it is assumed that predominant captured video period is very short so that all of the data may be completely transmitted from the source before getting a control message from the sink.

3.2 Packet ordering

In WSN applications, packets may be received out of order due to multihop topology of the network [27]. Packets of the same sensor may have different paths because of the routing layer decision. Therefore, out of order delivery should be handled for WMSN applications. SUIIT has a mechanism for reordering packets.

In order to display an image, SUIIT waits for all packets of a frame for sequentially encoded JPEG images. If the JPEG is progressively encoded, the application can display each received sub-frame gradually. Thus, a frame or sub-frame can only be displayed after the whole number of packets of the relevant frame or sub-frame are received. Every SUIIT packet has a source ID, frame number and a sequence number which is increased by one for each created packet. In the application layer, SUIIT tracks the sequence number of the incoming packets. When all of the packets of a frame (or sub-frame if image is

progressive) are received, the packets are ordered by the application and then displayed. In Figure 1, out-of-order packet handling mechanism of SUIIT is depicted.

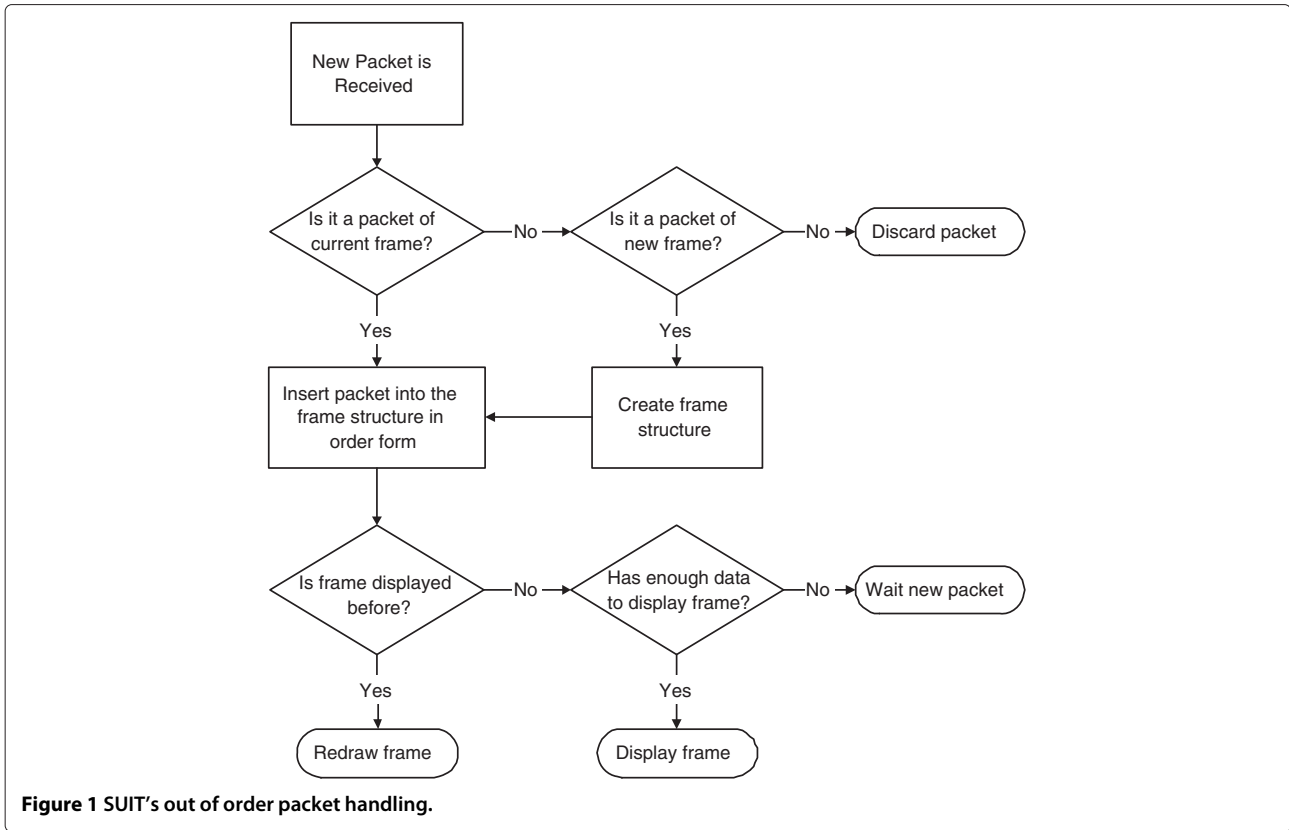
3.3 Packet prioritization

The buffer management in transport layer is an important issue in WMSNs because it has a considerable impact on the overall network efficiency and QoS which is dependent to the application requirements. For video surveillance applications, the primary requirement is delivering the video frames as fast as possible. SUIIT has only one type of packet (video fragment packet) so there is only one queue at the transport layer instead of having a couple of prioritized queues. SUIIT assigns a score for each packet for prioritizing them to improve QoS performance. When the transport layer buffer is full, the packets which have the lowest score are dropped. Score of the packet is calculated by using three parameters; current hop count of the packet, average delay of the packet, and frame index of the packet.

Current hop count value is embedded to the packet header. This value is assigned as zero when the packet is created at the source. At each hop, the receiver node increases the current hop value by one. The packet which visits more intermediate nodes until it reaches the sink has higher probability of drop-out than the packet which visits fewer sensors between the source and the sink. Therefore, SUIIT gives a higher priority to the packets whose current hop count is higher than the others. Depending on the maximum hop count that the topology allows, the possible hop-count value space is divided into four intervals using the points h_1 , h_2 , and h_3 , where $h_1 < h_2 < h_3$. The weight of current hop count (W_h) is calculated with the following formula:

$$W_h = \begin{cases} 1 & \text{if } \text{hop} \leq h_1 \\ 2 & \text{if } h_1 < \text{hop} \leq h_2 \\ 3 & \text{if } h_2 < \text{hop} \leq h_3 \\ 4 & \text{if } h_3 < \text{hop} \end{cases} \quad (1)$$

Similar to the hop count, total delay of the packet is also embedded into the packet header. Before sending a packet, sensor nodes calculate the interval between the time it is received and the time it is ready to be sent. Then sensor nodes update the total delay value of the packet. The average delay can be calculated easily by dividing the total delay by the current hop count. The packet which has less average delay provides better performance for video streaming applications. Transmitting the packets with low delay instead of the packets with high delay decreases the overall latency. Therefore, SUIIT gives a higher priority to the packets whose average delay is less than the other packets. Depending on the maximum bandwidth of the sensor nodes, the possible average delay value space is divided into four intervals by using the points d_1 , d_2 , and



d_3 where $d_1 < d_2 < d_3$. The weight of the average delay (W_d) is calculated with the following formula:

$$W_d = \begin{cases} 3 & \text{if average delay} \leq d_1 \\ 2 & \text{if } d_1 < \text{average delay} \leq d_2 \\ 1 & \text{if } d_2 < \text{average delay} \leq d_3 \\ 0 & \text{if } d_3 < \text{average delay} \end{cases} \quad (2)$$

For surveillance applications, Durmus et al. show that the initial frames of an event are more valuable, so they assign higher priority to the initial frames [28]. Therefore, we take into consideration the use of frame index as another QoS metric. In order to distinguish packets based on their frame index, the source sensor embeds the frame index into the packet header. SUIT gives higher priority to the packets which is a fragment of the first frames of the detected target. While calculating the weight of frame index (W_f), a threshold F_{th} which can be changed depending on the application requirements is used. W_f is calculated with the following formula:

$$W_f = \begin{cases} 2 & \text{if frame index} < F_{th} \\ 1 & \text{if otherwise} \end{cases} \quad (3)$$

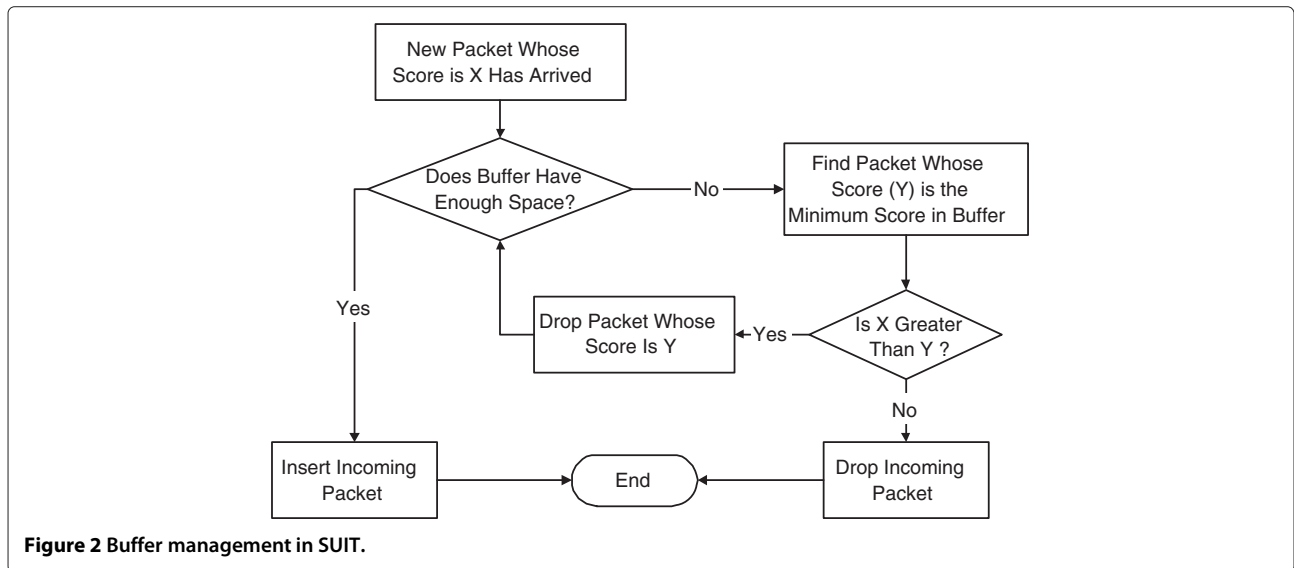
To assign a score to a packet, weights of each parameter are calculated and then multiplied with each other. The packet which has the minimum score is dropped when the queue is full. While calculating the weights, some

specific points and thresholds are used in the aforementioned formulations. These values can be varied according to the network topology and the network characteristics. The values used in the simulations in Section 4 are determined experimentally and the ones which provide the best performance gain are selected.

Compared to a non-weighting system, the advantage of a weighting system is that the system can favor a specific parameter value according to the system requirements. Our weighting system aims to distinguish the frames according to the traversed hop count, average delay and the index of the frame. It favors the frames traversing more hops to reduce energy waste, the frames with lower average delay to eliminate out-of-date frames to preserve energy and the first F frames from a sensor on detection of a target to preserve valuable information. Hence, by treating the frames differently, the weighting system provides the performance gain and energy conservation which are important metrics to be considered in MWSNs. The overall process is illustrated in the flowchart given in Figure 2.

3.4 Cross layer design

In the layered network design, operations are divided into hierarchical layers. Each layer is responsible for performing interrelated functions. These functions together provide a service that allows two systems to exchange



data. Layered network design has a lot of advantages. It reduces the complexity of the design, development and testing phases because each layer focuses on its own task and it does not have to consider other layers. Another advantage is that, the layered network allows different types of network hardware and software to communicate with each other.

Unlike the conventional network application, WMSN applications run on a specific device. They are not designed for different network hardware and software possibilities such as different network cards, operating systems and applications. So, separating layers strictly is not required for WMSN applications. Moreover, the layers may need to share some information with each other to provide better QoS. The main goal of the cross-layer approach is increasing the performance of wireless networks by using the interactions between various layers of the protocol stack. However, exchanging information over two or more layers does not mean that there is no layered architecture. Breaking all of the layers can destroy modularity and leads to spaghetti design which results in various negative consequences [29]. Therefore, the interaction of the layers is very important. There are several studies which propose cross-layer optimization techniques and explain why a cross-layer solution is needed for WMSNs citeSu2006,Ghalib2010.

SUIT uses a cross-layer approach to exchange information among layers. However, layering is not completely left, so that the advantages of the layered design are still preserved. SUIT has its own application layer and transport layer but it does not provide a MAC and routing layer. Any MAC and routing layer can be used under SUIT by extending them *via* a messaging module. SUIT can work with such protocols as long as they provide the necessary information which is mentioned later in this section.

The application layer is responsible for generating and viewing images. Images can be both SJPEG and PJPEG with or without restart markers. The source sensor is capable of fragmenting the image and the sink is capable of defragmenting it. In conventional layered designs, the application layer does not request information from the lower layers but in SUIT, the application layer gets the congestion level from the transport layer. According to the congestion level, it adapts the video rate as explained in Section 3.6.

One of the required features of the MAC layer is sharing and collecting sensor-specific information in order to predict the status of the medium more accurately. In SUIT, the MAC layer can access the occupancy of the transport layer queue and piggybacks this value into each outgoing packet including RTS and CTS packets. The receiver sensor should read the packet header at the MAC layer and generate a table which stores the buffer occupancy of the neighbors. During the fuzzy logic-based congestion estimation, the transport layer uses the elements of this table. Moreover, the transport layer requests the ID of the next hop sensor node from the routing layer during the process of congestion estimation. As a result, the transport layer uses data from both MAC and routing layers.

In summary, the cross-layer design of SUIT creates interactions across layers by enabling flow of information on both sides. The main aim of this design is to achieve gains in the overall system performance without corrupting modularity. This does not leave the philosophy of layered designs but provides a messaging system among layers.

3.5 Congestion detection mechanism

Because of not having an exact indicator of the congestion, it is a difficult task to detect congestion when it

exactly occurs in WMSNs. The most commonly used indicator for WSNs is high buffer occupancy. However, monitoring a sensor node's buffer may not be sufficient since event-based multimedia data produces bursty high-load traffic in WSNs [3]. SUIT proposes three indicators to be used for congestion decision. These indicators are the following:

- Ratio of incoming to outgoing packets within a time window: The ratio is calculated based on a sliding window. Hence, it overcomes the momentary changes in traffic and leads to more stable results than the instantaneous buffer occupancy.
- Number of contenders: The contenders are the neighbor nodes which have packets in their queue waiting to be sent in the contention region at the MAC layer, not on the path. Number of contender is calculated by using RTS/CTS packets which are generated by the neighbor nodes. If there are too many contenders, collision probability is higher. Therefore, the number of contenders is an effective indicator for congestion.
- Buffer occupancy of the parent (next-hop) sensor node: It is an important metric since it provides information about the environment. If the buffer occupancy of next hop sensor node is high, the congestion probability will also be high.

SUIT uses a fuzzy logic system (FLS) to map these three congestion metrics into a single value. In this study, we prefer to use fuzzy logic system for congestion estimation because of the following advantages. A fuzzy control system does not require a precise model and a formulation for dynamic and complex systems where the relationship between system output and control inputs is very hard to be formulated explicitly and precisely with mathematical equations [32]. This feature of fuzzy logic control makes it possible to fully exploit the dynamics in the congestion detection in WMSNs. Moreover, as a formal methodology to emulate the intelligent decision-making process of a human expert, fuzzy logic control provides an effective and flexible way to arrive at a definite conclusion based on imprecise information. Therefore, it can easily deal with various uncertainties inside WMSNs.

The components of the proposed FLS are singleton fuzzifier, product inference engine and centroid defuzzifier. Since FLS uses three metrics, the fuzzy set consists of three fuzzy variables as

$$A = (\alpha, \beta, \gamma) \quad (4)$$

where α is the number of contenders, β is the buffer occupancy percentage of parent (next-hop) sensor node and γ is the fuzzy variable term for the traffic load. Traffic load

is the ratio of incoming packets to outgoing packets, and is therefore defined as

$$\gamma = \frac{\gamma_{in}}{\gamma_{out}} \quad (5)$$

Inputs and outputs of the FLS are non-numeric linguistic variables. Instead of using numerical values, FLS uses variables from natural language. Our FLS has three linguistic variables named *LOW* (L), *MEDIUM* (M), and *HIGH* (H) which determine the congestion level of the system.

Membership functions are defined to be used in fuzzification and defuzzification steps. In our case, we have three membership function set and each of the set has different function for L , M and H linguistic variables. Membership functions can be in different forms such as triangular, trapezoidal, piecewise linear, Gaussian, or singleton [33]. We are using triangular shapes due to the ease of computation considering the limited processing power available on the sensor devices. The functions can be computed using four basic operations and do not require complex operations such as integration and quadratic formulas. In Figure 3 membership functions for the fuzzy variables α , β and γ are depicted, where n_{max} represents the number of maximum contenders, b_{max} represents the maximum buffer occupancy and r_{max} represents the maximum traffic rate, up to the transmission in progress. Generally, the degree of membership value for each fuzzy variable is decided based on experimentation. In the simulations, the maximum and average values for these variables are determined experimentally. The critical points in membership functions can be calculated in terms of the upper bounds of the network indicators. In this work, n_{max} is assigned as 8, b_{max} is assigned as 100 and r_{max} is assigned as 2.

The first step of our FLS is the fuzzification step. The proposed FLS uses a singleton fuzzifier to map each crisp value to a fuzzy value using membership functions. Since FLS has three linguistic variables, the fuzzifier defines three fuzzy classes named L , M and H for each fuzzy variable:

$$F : (\alpha, \beta, \gamma) \rightarrow (\alpha^f, \beta^f, \gamma^f) \quad (6)$$

$$\chi^f = \left[\mu_{\chi}^L(\chi), \mu_{\chi}^M(\chi), \mu_{\chi}^H(\chi) \right] \quad (7)$$

where χ represents the variables α , β and γ . Fuzzy values are calculated *via* membership functions of the fuzzifier which are presented in Figure 3. For example, if the number of contenders, α , of the sensor node is 5, then the corresponding α^f is $[0, 0.5, 0.5]$ since $\mu_{\alpha}^L(5) = 0$, $\mu_{\alpha}^M(5) = 0.5$ and $\mu_{\alpha}^H(5) = 0.5$ as can be seen in Figure 3a. If the buffer occupancy of the next-hop sensor node, β , is 50, then the corresponding β^f is $[0, 1, 0]$ since $\mu_{\beta}^L(50) = 0$, $\mu_{\beta}^M(50) = 1$ and $\mu_{\beta}^H(50) = 0$ as can be seen in Figure 3b. If

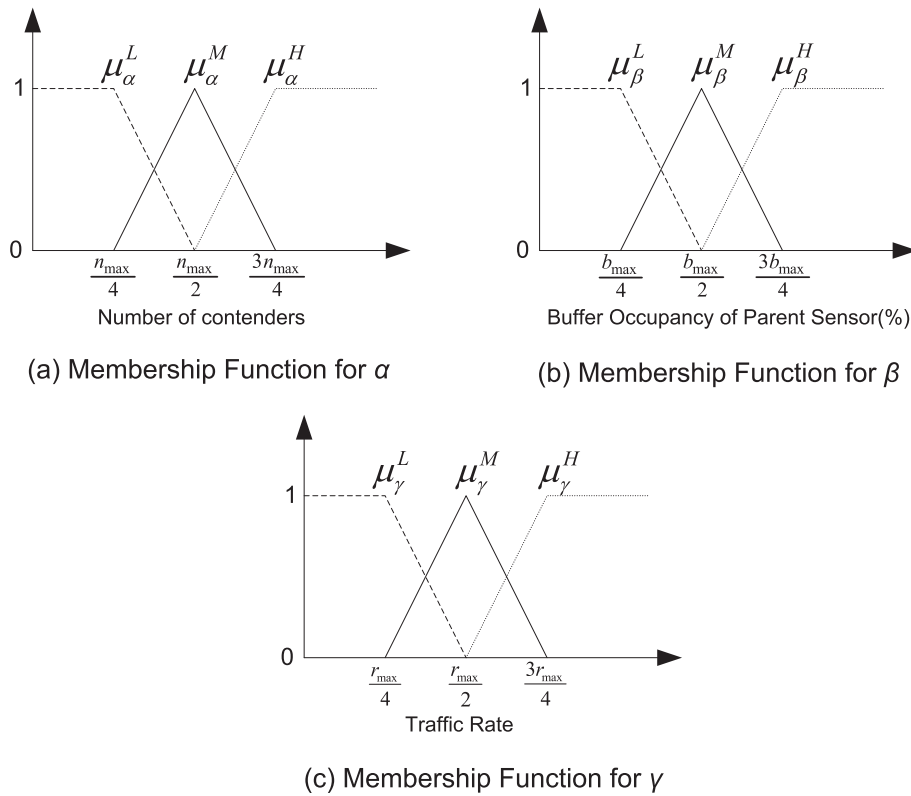


Figure 3 Membership functions for the fuzzy variables (a) α , (b) β and (c) γ .

the traffic rate, γ , of the sensor node is 1.25, then the corresponding γ^f is $[0, 0.5, 0.5]$ since $\mu_\gamma^L(1.25) = 0$, $\mu_\gamma^M(1.25) = 0.5$ and $\mu_\gamma^H(1.25) = 0.5$ as can be seen in Figure 3c.

The second and the most important step of our FLS is the inference step. Inference is used to combine fuzzy rules from the fuzzy rule base which is constructed to control the output variable. A fuzzy rule is a simple IF-THEN rule with a condition and a conclusion. Since there are three fuzzy variables and three fuzzy classes for each fuzzy variable, the rule base contains $3^3 = 27$ different rules. The output of each rule is calculated by the rule evaluation method (REM) and results are mapped to a consequence class c . The structure of r in our rule base is defined as:

$$r : \text{if } \alpha_r \text{ is } f_\alpha^r \text{ and } \beta_r \text{ is } f_\beta^r \text{ and } \gamma_r \text{ is } f_\gamma^r \text{ then output is } f_c^r \quad (8)$$

where each of f_α^r , f_β^r and f_γ^r is one of the fuzzy classes L, M or H .

In the proposed REM, all of the variables have the same effect for congestion decision. Therefore, all of the fuzzy variables have the same importance. The weights of the fuzzy classes for α , β and γ are set as 1 for L , 2 for M and 3 for H . The output of a rule θ_r is calculated by summing up the weight of f_α^r , f_β^r and f_γ^r . Table 1 shows the rule base of our fuzzy inference system (FIS), where the

θ column corresponds to the output values of the rules which are calculated using the REM and the c column corresponds to the consequence fuzzy classes determined by the unique values of θ . As shown in Table 1, θ value can be a number between 3 and 9, hence there are 7 consequence fuzzy classes.

We use the product-inference rule in our FIS, since it allows a mathematical simplification in the defuzzification process and is commonly used in practice. Hence,

$$\mu_r = \mu_\alpha^{f_\alpha^r}(\alpha) \cdot \mu_\beta^{f_\beta^r}(\beta) \cdot \mu_\gamma^{f_\gamma^r}(\gamma) \cdot \mu_c^{f_c^r}(\theta_r) \quad (9)$$

where $\mu_c^{f_c^r}$ is the membership functions of the consequence fuzzy classes defined in Figure 4 and $\mu_\alpha^{f_\alpha^r}$, $\mu_\beta^{f_\beta^r}$ and $\mu_\gamma^{f_\gamma^r}$ are the membership functions in the rule r corresponding to the fuzzy classes of the fuzzy variables for α , β and γ , respectively.

The last step of the proposed FLS is the defuzzification step. Defuzzification is a process of transforming fuzzy output inferred from the fuzzy inference system (FIS) to a crisp value. We use a centroid defuzzifier in the defuzzification step. The centroid defuzzifier returns the center of the inferred fuzzy outputs. The centroid calculation is the most popular defuzzification method which returns the center of the area under the curve. FCC [25]

Table 1 Fuzzy rules in the knowledge base

Index	α	β	γ	c	θ
1	L	L	L	1	3
2	L	L	M	2	4
3	L	L	H	3	5
4	L	M	L	2	4
5	L	M	M	3	5
6	L	M	H	4	6
7	L	H	L	3	5
8	L	H	M	4	6
9	L	H	H	5	7
10	M	L	L	2	4
11	M	L	M	3	5
12	M	L	H	4	6
13	M	M	L	3	5
14	M	M	M	4	6
15	M	M	H	5	7
16	M	H	L	4	6
17	M	H	M	5	7
18	M	H	H	6	8
19	H	L	L	3	5
20	H	L	M	4	6
21	H	L	H	5	7
22	H	M	L	4	6
23	H	M	M	5	7
24	H	M	H	6	8
25	H	H	L	5	7
26	H	H	M	6	8
27	H	H	H	7	9

also calculates center of gravity of the output membership function in the defuzzification phase of their fuzzy-based congestion control algorithm. However, calculating the center of the gravity of a trapezoid is a complex process and not appropriate for the sensor devices due to their limited processing capability and battery. In order to decrease the complexity of the defuzzification step, we determine the membership function of each consequence

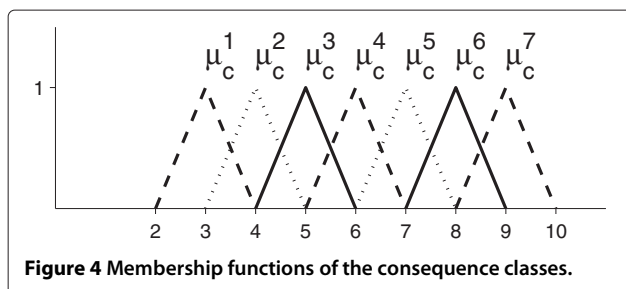


Figure 4 Membership functions of the consequence classes.

fuzzy class as an isosceles triangle whose abscissa value of its apex is equal to the θ value of the corresponding consequence class. In Figure 4, membership functions of the consequence classes are depicted.

Output of the centroid defuzzifier can be calculated easily by using Equation 10.

$$\omega = \frac{\sum_{r=1}^{27} \theta_r \mu_r}{\sum_{r=1}^{27} \mu_r} \quad (10)$$

where θ_r is the output of the rule r and μ_r is the output product-inference engine. Defuzzification is the final step in FLS. After applying the centroid defuzzifier, the output of the FLS, ω , becomes a crisp value between 3 and 9.

3.6 Congestion mitigation mechanism

SUIT classifies ω into three classes as *Not Congested* (NC), *Slightly Congested* (SC) and *Fairly Congested* (FC) by using two threshold values. Congestion mitigation is not applied for the NC level because in this case, sensor nodes have enough capacity to handle the network traffic. SUIT congestion mitigation algorithm is activated if the network is SC or FC. Our mitigation algorithm has two major procedures to decrease the congestion. These procedures are rate adaptation and quality adaptation.

Rate adaptation takes place in the source sensor which has a target in its FoV and generates image frames. The source sensor generates video in the application layer based on the congestion level which is calculated in the transport layer. Since SUIT is a cross-layer protocol, the application layer can request the required data from underlying layers. Application layer can define three different video rates according to three congestion level classes defined in SUIT. Depending on the sensors' data rate and encoded image size, the choice of rates should be configured by the application. A generic function for the rate adaptation is given in Equation 11.

$$r = \begin{cases} r_1 & \text{if } L = \text{NC} \\ r_2 & \text{if } L = \text{SC} \\ r_3 & \text{if } L = \text{FC} \end{cases} \quad (11)$$

where L is the congestion level and r is the video rate of application. r_1 , r_2 and r_3 are the video rates defined by the application and their values should be as $r_1 > r_2 > r_3$.

Quality adaptation takes place on the intermediate nodes on the fly. To the best of our knowledge, the quality adaptation on the fly is not applied for WMSNs before. We use PJPEG's structural advantage in this procedure. As explained in Section 3, PJPEG comprises a number of scans (subframes). If some packets of a sub-frame are dropped or corrupted, only the corresponding sub-frame becomes useless instead of the whole image. In our quality adaptation technique, we intentionally drop the packets

of some subframes of PJPEG. In order to distinguish subframes of the image, the SUIT packet has two types of information in its packet headers. These data are the number of total subframes that the related frame has, and the sub-frame index that the packet belongs to. Decreasing the quality is the responsibility of the transport layer. Thus, the application layer always generates full-quality images. In our quality adaptation technique, all PJPEG images are processed starting from the lowest score to highest score (scoring was explained in Section 3.3). In this process, subframes of the PJPEG images are randomly selected and dropped. According to the application requirements, fewer subframes are dropped for the SC case while more subframes are dropped for the FC case. However, the quality cannot be decreased more than a threshold defined for the application. This quality adaptation process is ended when adequate space in the buffer becomes available. If there is still not enough space in the buffer after all images are processed, images are dropped according to the packet prioritization policy.

In summary, in order to mitigate congestion, SUIT performs rate adaptation at the source node and quality adaptation on the intermediate nodes. The rate adaptation is commonly used by WSNs applications, but quality adaptation is a novel technique which can be applied for PJPEG images while they are being transmitted. The overall congestion mitigation process is illustrated in the flowchart given in Figure 5.

4 SUIT Performance evaluation

The performance of SUIT's congestion control is evaluated with extensive simulations and it is compared with two different congestion detection and mitigation techniques. These competitors use QCC and FCE [16] based congestion detection and rate adaptation as the congestion mitigation technique. The reason for selecting FCE as a competitor of SUIT protocol is to compare SUIT's fuzzy logic-based congestion estimation with another fuzzy logic-based approach. QCC is selected as the second competitor for comparing our protocol with a non-fuzzy logic-based approach but also a queue based congestion estimation technique, which is commonly used to estimate congestion in WSNs. To estimate congestion, QCC uses only buffer occupancy and FCE uses both packet arrival rate and buffer occupancy, whereas SUIT uses buffer occupancy, traffic rate and also number of contenders around a sensor node. The intention of selecting these competitors is to evaluate the performance gain achieved when using these 3 parameters instead of using 1 or 2 parameters while deciding on congestion.

The implementation of QCC is inspired by the protocols which monitor the local buffer occupancy of the sensor nodes in order to estimate congestion such as (ESRT) [10] and Fusion [11]. To determine the congestion level

of the network, instead of using single threshold, QCC uses two thresholds between 0% and 100% to adapt the result of QCC congestion to SUIT's rate adaptation. If the buffer occupancy rate is lower than the first threshold, the congestion level of the network is considered as *Not Congested*. If the buffer occupancy rate is between two thresholds, the congestion level of the network is considered as *Slightly Congested*. If the buffer occupancy rate is higher than the second threshold, the congestion level of the network is considered as *Fairly Congested*.

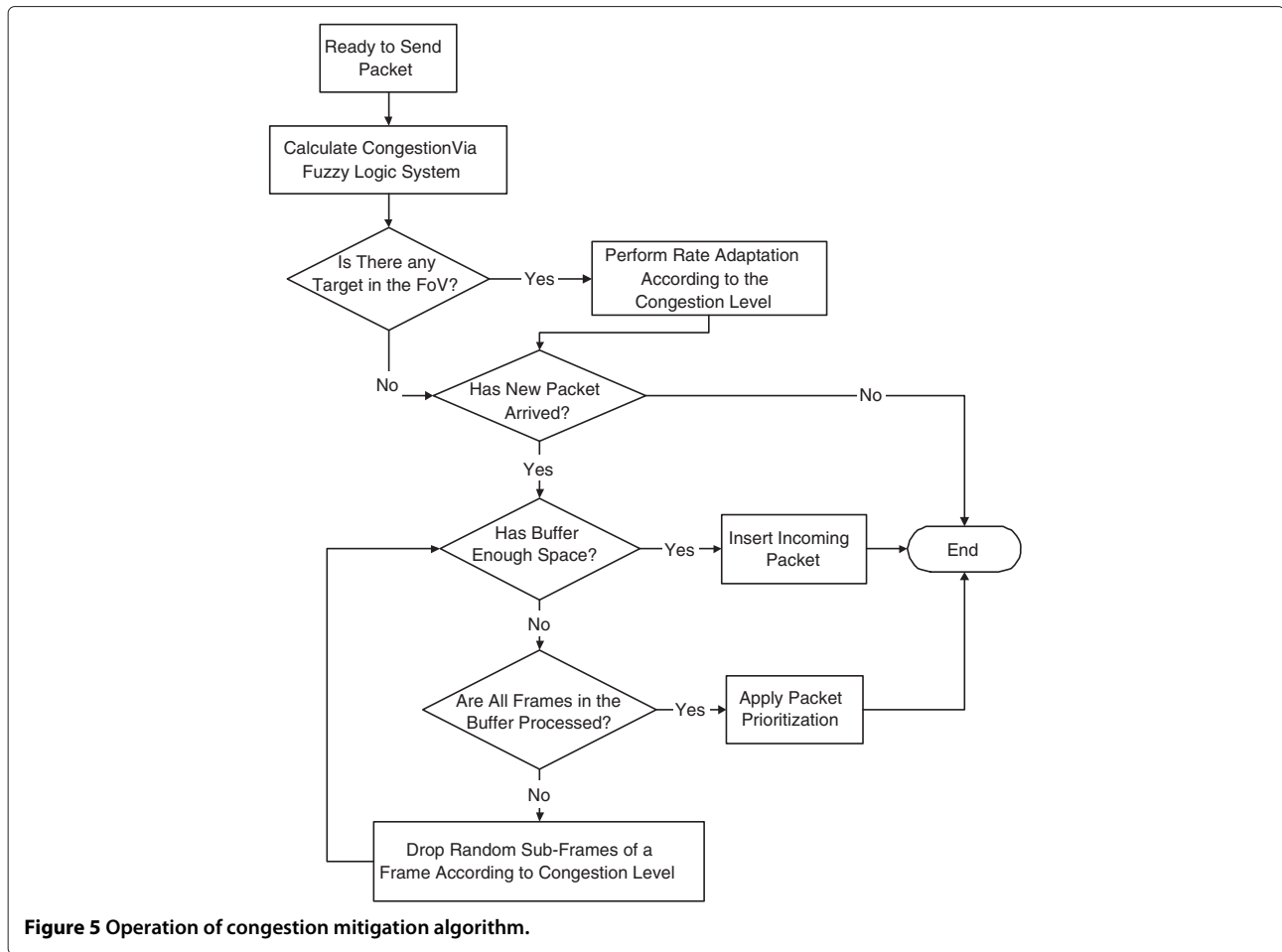
FCE [16] uses a fuzzy logic-based congestion estimation mechanism which has two fuzzy variables; *net packet arrival rate* and *buffer occupancy*. The result of FCE congestion estimation mechanism is a numeric value which can be 0, 0.5 or 1. We adapt the result of FCE congestion to SUIT's rate adaptation by assigning *Not Congested* for 0, *Slightly Congested* for 0.5 and *Fairly Congested* for 1.

SUIT's congestion control provides fuzzy logic-based congestion detection and two congestion mitigation techniques. One of the mitigation techniques is data rate adaptation and the other one is quality adaptation while packets are being transmitted over the sensor nodes. SUIT's competitors use their own congestion detection mechanisms. As for congestion mitigation, they only use rate adaptation at source sensor which streams image frames.

4.1 Implementation considerations and simulation parameters

OPNET [34] network simulation tool is used for the implementation and simulation of the SUIT protocol. We set up a target monitoring application scenario for the simulations. The motivation for this scenario is monitoring movement patterns of moving targets by using the camera extension of the sensor nodes. In this scenario, number of transmitted frames is more important than the quality of the frames as long as they have an acceptable quality. In the simulations, targets are moving according to the random waypoint mobility model because of its wide-usage in simulating mobile network scenarios with mobile users. This target mobility model is designed in a way to allow targets to move in all parts of the area of interest so that the spatial distribution of the video data traffic sources are distributed uniformly over the surveillance area. Sensor nodes use binary detection model to detect target. According to the binary detection model, the probability of sensing a target is equal to 1 if the target is in the FoV of the sensor node.

Diff-MAC [35] protocol is used as the MAC protocol. Diff-MAC is a QoS-aware and priority-based MAC protocol for WMSNs. However, in the simulations, the QoS awareness of Diff-MAC is not used. Since it supports prioritization of three different types of packets (best effort, non real-time, real-time) and we use only a single type of



packet, we did not use the service differentiation mechanism of Diff-MAC. All SUIT packets are considered as video packets. In order to use Diff-MAC with SUIT, the following modifications are applied:

- An inter-layer communication module is implemented to exchange information with other layers.
- A module is implemented in order to construct neighbor buffer occupancy table (NBOT). This module piggybacks the buffer occupancy information into the outgoing packet header and reads this information from each packet that the sensor can hear regardless of the packet destination.
- A module is implemented in order to distinguish SUIT packets with respect to their importance. It chooses different number of retries according to the importance level.

IEEE 802.15.4 compliant radio has low cost and low power attributes. Thus, these radios are widely used for WSN implementation [36]. In the market, there are several radio modules which are compliant with IEEE

802.15.4 standard. Most of the radios of Atmel, Chipcon and Microchip provide 250-kbps data rate [37]. However, for multimedia transmission 250-kbps data rate is not adequate. Chulsung Park et al. design a high data-rate wireless sensor node named as eCAM [38]. Their sensor node provides 1 Mbps data rate and their camera module can operate as a JPEG compressed still camera. Also Atmel designs an IEEE 802.15.4 compliant single chip ATmega128RFA1 [39] which can provide up to 2 Mbps data rate. Since high data-rate transmission should be used for multimedia streaming applications, we determine the data rate as 2 Mbps in our simulations according to the capabilities of these example radios. In order to simulate a real application more accurately, a realistic link model proposed in [40] is used in the simulations. Zuniga et al. use the log-normal shadowing model proposed in [41] as the radio propagation model and they use the signal-to-noise ratio (SNR) to calculate the probability of successfully receiving a packet.

In the simulations, $400 \times 400 \text{ m}^2$ -shaped environment where 256 stationary video sensors are uniformly deployed is used. The communication range of the sensors is 80 m and these sensors have a camera module

whose detection range is 30 m and observation angle is 52°. GPSR [42] protocol is used as the routing protocol due to its simplicity and common usage. There are one to seven targets which are moving in the environment. The target is assumed to be a pedestrian whose velocity is 2 m/s. When the target enters the FoV, the sensor node's camera starts generating video. The images used in simulations are Sub Quarter Common Intermediate Format (SQCIF) progressively encoded JPEG images which have a resolution of 128 × 96. SQCIF can have a small resolution for the video streaming applications, but it is a proper format that can be used with the determined data rate which is 2 Mbps to maintain the desired frame rates. If the data rate of the sensor devices increases by technological improvements, other formats with higher resolution can be used. Other significant parameters used in our simulations are listed in Table 2.

In order to evaluate the performance of SUIT on a congested WMSN, we study the scenarios where congestion is generated on the relay nodes. In these scenarios, we make sure that the generated data traffic does not exceed the capacity of the source nodes. Therefore, the data load on the relay nodes is increased without dropping packets at the source node. To achieve this, we increase the number of intruders on the deployment area and create video traffic over relay nodes from multiple source nodes.

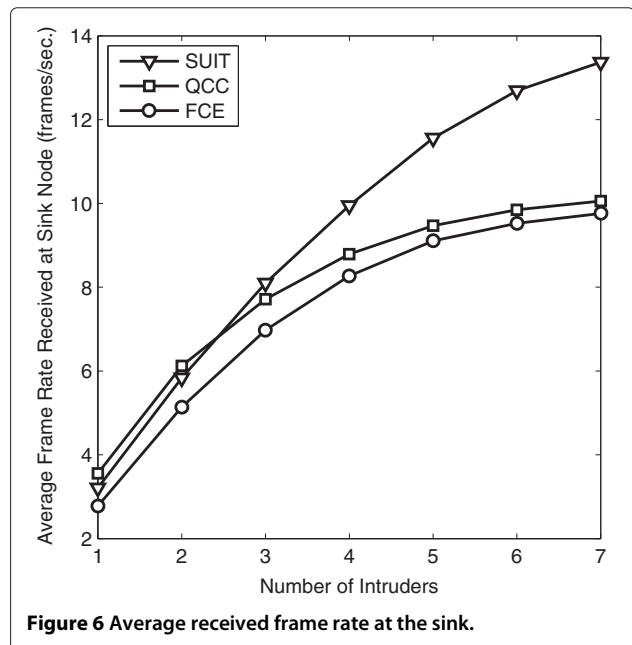
4.2 Performance evaluation of congestion control techniques

One of the most important metrics for application QoS is the average received frame rate at the sink. In Figure 6,

Table 2 Simulation parameters for the evaluation of SUIT

Parameter	Value
Number of sensors	256
Area	400 m × 400 m
Sink location	(0,400)
Number of targets	1 to 7
Target velocity	2 m/s
Camera observation angle	52°
Depth of field	30 m
Rate for NC case	2 fps
Rate for SC case	4 fps
Rate for FC case	6 fps
Average packet size	1 KB
Average JPEG image size	9.2 KB
Buffer size	100 KB
Channel rate	2 Mbps
Communication range	80 m

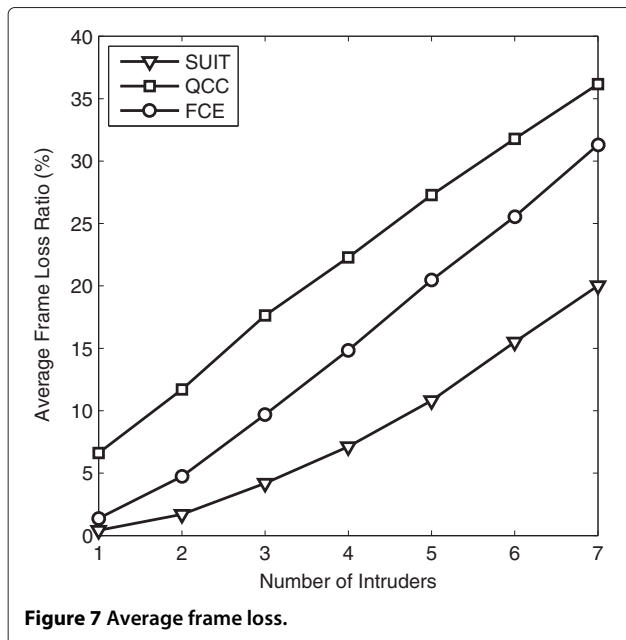
Target mobility model: random waypoint; detection model: binary FoV; MAC protocol: Diff-MAC [35]; routing protocol: GPSR [42].



comparison of the average received frame rate is depicted. The X-axis shows the number of targets which is varied to increase the load in the network. If the network is not congested, all protocols provide a similar frame rate, i.e. with 1 to 3 targets. However, when the network is getting congested, the SUIT protocol can transmit more frames than the other protocols. The reason for this result is the congestion mitigation technique of SUIT which performs quality adaptation on the fly. In a congested network, SUIT can decrease the buffer fullness by decreasing the quality of the images. However, other protocols have to drop incoming frames. As shown in Figure 6, without quality adaptation, the maximum frame rate that can be received by the sink is 10 fps, whereas SUIT allows the sink to receive more than 13 fps.

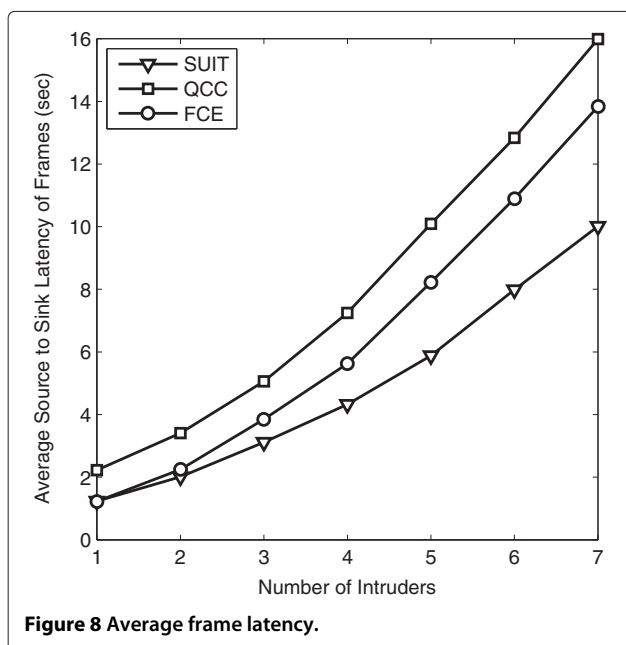
A protocol can transmit more frames if it generates excessive number of frames, but number of dropped frames should also be considered. So the frame loss ratio is another important QoS metric. In Figure 7, the frame loss performances of the protocols are given. For all of the cases, SUIT has better frame loss performance than QCC- and FCE-based congestion detection protocols. QCC has worse performance than FCE because it calculates the congestion level in an optimistic manner and generates more frames than FCE. Therefore, compared to FCE, QCC provides a better average received frame rate at the sink but worse frame loss performance.

We compare the performance of the protocols in terms of the mean delay of frames successfully delivered to the sink in Figure 8. The latency difference between QCC- and FCE-based protocols is nearly 1 s regardless of the congestion level. When the load is light, SUIT has a similar



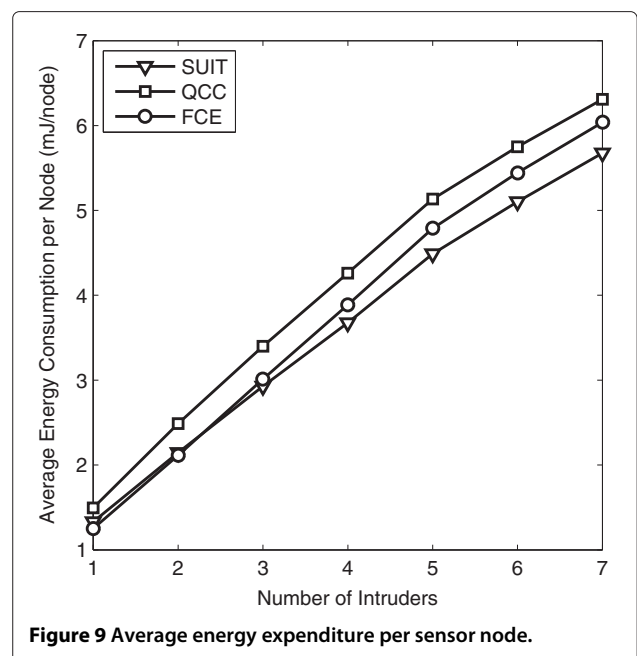
performance with FCE. However, as the load increases, the latency gap between SUIT and FCE widens up to 4 s, which means 28% performance gap. The main reason for the performance gap between SUIT and the other protocols is the average size of the frames. If the network congestion increases, SUIT decreases the frame quality, so the size of the frames decreases. Since small-sized frames are transmitted faster than the original frames, SUIT provides the best frame latency performance.

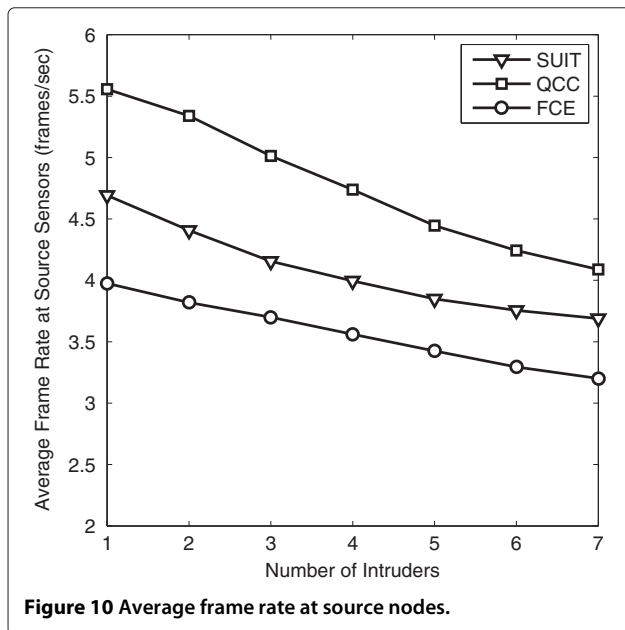
The performances of the protocols are compared in terms of average energy consumption per sensor node



in Figure 9. The average energy expenditure is related to the number of generated frames at source nodes. As shown in Figure 10, QCC always generates more frames than SUIT and FCE so it consumes more energy than the other protocols. SUIT produces more frames than FCE, and therefore, it consumes more energy than FCE if the load is light (if the number of targets is lower than three). However, when the network is getting congested, energy expenditure of SUIT provides better results and the performance gap between the SUIT and other protocols gets wider. Since the SUIT protocol can shrink the size of frames in congested networks, it transmits lower number of packets and exhibits better performance in terms of energy consumption in highly loaded networks.

The frame rates at source sensor nodes are calculated in a different manner for each protocol. SUIT and FCE use fuzzy logic-based approach whereas QCC uses local buffer occupancy for deciding on the congestion level. According to the congestion level, the frame rate is assigned dynamically while generating images. In Figure 10, the average calculated frame rate at the source sensor nodes is illustrated. All of the protocols decrease the frame rate as the network is getting congested. QCC estimates the congestion level optimistically and therefore calculates higher frame rates than the others. On the other hand, FCE estimates the congestion in a pessimistic manner and calculates lower frame rates than the others. SUIT provides a balanced congestion level *via* its efficient fuzzy logic-based congestion estimation technique. Therefore, SUIT provides better performance than its competitors in all QoS metrics, except of the average received frame quality.





The quality of PJPEG images are compared with respect to the peak signal-to-noise ratio (PSNR) values. PSNR is a well-known image quality metric and widely used [43]. For two grey-level (8 bits) images f and g whose dimensions are $X \times Y$, PSNR calculation is defined as

$$\text{PSNR}(f, g) = 10 \log_{10} \left(\frac{255^2}{\text{MSE}(f, g)} \right) \quad (12)$$

where

$$\text{MSE}(f, g) = \frac{1}{X \times Y} \sum_{i=1}^X \sum_{j=1}^Y (f_{ij} - g_{ij})^2 \quad (13)$$

In Figure 11, visual examples of PSNR values are given for PJPEG images. The images shown in the tables are selected randomly from the image set which is received by the sink sensor. For the image quality performance evaluation of the protocols, the average PSNR values of the received images are used and the PSNR values are calculated using Equation 12. Again, in these simulations, we varied the number of targets. Average quality of the received frames for QCC and FCE are the same and PSNR values for the received frames are computed as 35 dB, whereas the average PSNR values of the received frames for SUIT are a bit lower than the other protocols and for the worst case (when there are seven targets in the area) is observed as 32.5 dB. Since the main goal of SUIT is delivering more frames with lower delay and lower energy consumption by decreasing the image quality to an acceptable level, it is not surprising that the average PSNR values of the images transmitted with the SUIT protocol are lower. As discussed in [44,45], 32.5 dB is a sufficient quality for our target monitoring application and can be used for object detection algorithms at the sink side. Therefore, images with 32.5 dB still have fairly good quality.

5 Conclusion

In this paper, we proposed a new protocol, SUIT, based on the necessities of efficient multimedia transmission over WMSNs. SUIT is an image transport protocol which provides a fuzzy logic approach for estimating congestion efficiently and then reacting to mitigate the congestion. SUIT has an efficient congestion detection mechanism. Instead of using a specific congestion indicator, it proposes a fuzzy logic-based congestion detection technique which gives more accurate result. For congestion

Original Image	Received Image	PSNR
		34.13 dB
		31.43 dB
		23.77 dB

Figure 11 PSNR values for PJPEG images.

mitigation, SUIE provides two different techniques. The first technique is adapting video frame rate at source sensor nodes. The second one is a novel congestion mitigation technique which can adapt the quality of images on-the-fly. Main contributions of the SUIE protocol are using a new combination of three congestion indicator in fuzzy logic-based congestion estimation approach, proposing a cross-layer information exchange method among different layers, and applying a quality adaptation technique which decreases the image quality while they are being transmitted. To the best of our knowledge, mitigating congestion by decreasing image quality on-the-fly is a new technique proposed by SUIE.

A surveillance application scenario is implemented and extensive simulations are conducted in OPNET to verify our approach. SUIE protocol is compared to QCC-based and FCE-based congestion control protocols. Especially for congested network cases, SUIE outperforms other protocols in all aspects except of image quality. As we mentioned before, the main goal of SUIE protocol is providing better performance in terms of energy consumption, frame delivery, frame loss and frame latency by decreasing the quality without sacrificing image quality too much. So, the quality of received image performance of SUIE is lower than the other protocols as expected. However, the average image quality of the SUIE protocol is still satisfactory.

As a future work, we plan to apply fuzzy logic-based packet prioritization while dropping packets from the local queue. In our current implementation, packets are dropped according to their weights. Weights are calculated based on the predefined rules. However, a fuzzy logic-based approach can provide more fairness on queue management. In addition, we want to improve the target detection process by implementing a communication module between neighboring sensor nodes. The sensor nodes which monitor the same area, can communicate with each other in order to prevent transmitting the same image of the target for better application performance.

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

This work is supported by the European Community's Seventh Framework Programme (FP7-ENV-2009-1) under the grant agreement FP7-ENV-244088 FIRESENSE, by COST Action COST IC0906 WiNeMO Wireless Networking for Moving Objects and by the Turkish State Planning Organization (DPT) under the TAM Project, number 2007K120610. A shorter version of this manuscript was accepted as an invited paper in the Fifth IFIP International Conference on New Technologies, Mobility and Security (NTMS'2012).

Author details

¹Computer Networks Research Laboratory, Netlab. Department of Computer Engineering, Bogazici University, Bebek 34342, Istanbul, Turkey. ²Department of Computer Engineering, Galatasaray University, Besiktas 34349, Istanbul, Turkey. ³Department of Mathematics, Bogazici University, Bebek 34342, Istanbul, Turkey. ⁴Information Technologies R&D, NETAS, Istanbul, Turkey.

Received: 4 November 2013 Accepted: 4 April 2014

Published: 23 April 2014

References

1. S Poojary, M Pai, Multipath data transfer in wireless multimedia sensor network, in *Proceedings of the International Conference on Broadband, Wireless Computing, Communication and Applications (BWCCA)*, (2010), pp. 379–383
2. S Misra, M Reisslein, G Xue, A survey of multimedia streaming in wireless sensor networks. *IEEE Commun. Surv. Tutorials.* **10**(4), 18–39 (2008)
3. M Maimour, C Pham, D Hoang, A congestion control framework for handling video surveillance traffics on WSN, in *Proceedings of the 2009, International Conference on Computational Science and Engineering - Volume 02 (CSE '09 2009)*, pp. 943–948
4. W Yu, Z Sahinoglu, A Vetro, Energy efficient, JPEG 2000 image transmission over wireless sensor networks, in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, Volume 5, (2004), pp. 2738–2743
5. A Boukerche, Y Du, J Feng, R Pazzi, A reliable synchronous transport protocol for wireless image sensor networks, in *IEEE Symposium on Computers and Communications (ISCC)*, (2008), pp. 1083–1089
6. A Mammeri, A Khoumsi, D Ziou, B Hadjou, Energy-efficient transmission scheme of JPEG images over visual sensor networks, in *Proceedings of the 33rd IEEE Conference on Local Computer Networks LCN*, (2008), pp. 639–647
7. C Wang, K Sohraby, B Li, M Daneshmand, Y Hu, A survey of transport protocols for wireless sensor networks. *IEEE Netw.* **20**(3), 34–40 (2006)
8. Y Liu, N Xiong, Y Zhao, A Vasilakos, J Gao, Y Jia, Multi-layer clustering routing algorithm for wireless vehicular sensor networks. *Commun. IET.* **4**(7), 810–816 (2010)
9. L Xiang, J Luo, A Vasilakos, Compressed data aggregation for energy efficient wireless sensor networks, in *Sensor, Mesh and Ad Hoc Communications and Networks (SECON) 2011 8th Annual IEEE Communications Society Conference on*, (2011), pp. 46–54
10. Y Sankarasubramaniam, O Akan, I Akyildiz, ESRT: event-to-sink reliable transport in wireless sensor networks, in *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, (2003), pp. 177–188
11. B Hull, K Jamieson, H Balakrishnan, Mitigating congestion in wireless sensor networks, in *Proceedings of the International Conference on Embedded Networked Sensor Systems (SenSys)*, (2004), pp. 134–147
12. CT Ee, R Bajcsy, Congestion control and fairness for many-to-one routing in sensor networks, in *Proceedings of the International Conference on Embedded Networked Sensor Systems (SenSys)*, (2004), pp. 148–161
13. J Paek, R Govindan, RCRT: Rate-controlled reliable transport protocol for wireless sensor networks. *ACM Trans. Sensor Netw (TOSN).* **7**(3), 1–45 (2010)
14. CY Wan, SB Eisenman, AT Campbell, CODA: congestion detection and avoidance in sensor networks, in *Proceedings of the International Conference on Embedded Networked Sensor Systems (SenSys)*, (2003), pp. 266–279
15. M Zawodniok, S Jagannathan, Predictive congestion control protocol for wireless sensor networks. *IEEE Trans. Wireless Commun.* **6**(11), 3955–3963 (2007)
16. S Munir, YW Bin, R Biao, M Man, Fuzzy logic-based congestion estimation for QoS in wireless sensor network, in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, (2007), pp. 4336–4341
17. C Wang, K Sohraby, B Li, SenTCP: a hop-by-hop congestion control protocol for wireless sensor networks, in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, (2005)
18. MH Yaghmaee, DA Adjeroh, Priority-based rate control for service differentiation and congestion control in wireless multimedia sensor networks. *Comput. Netw.* **53**(11), 1798–1811 (2009)
19. JY Teo, Y Ha, CK Tham, Interference-minimized multipath routing with congestion control in wireless sensor network for high-rate streaming. *Mobile Comput. IEEE Trans.* **7**(9), 1124–1137 (2008)
20. M Maimour, C Pham, J Amelot, Load repartition for congestion control in multimedia wireless sensor networks with multipath routing, in *3rd International Symposium on Wireless Pervasive Computing 2008. ISWPC, 2008.* (2008), pp. 11–15
21. CY Wan, SB Eisenman, AT Campbell, J Crowcroft, Siphon: overload traffic management using multi-radio virtual sinks in sensor networks, in

- Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems* (ACM New York, NY, USA, 2005), pp. 116–129
22. T He, J Stankovic, C Lu, T Abdelzaher, SPEED: a stateless protocol for real-time communication in sensor networks, in *Proceedings of the 23rd International Conference on Distributed Computing Systems*, (2003), pp. 46–55
 23. Z Wan, N Xiong, N Ghani, A Vasilakos, L Zhou, Adaptive unequal protection for wireless video transmission over IEEE 802.11e networks. *Multimedia Tools Appl.* **62**(3), 1–31 (2013)
 24. C Basaran, KD Kang, M Suzer, Hop-by-hop congestion control and load balancing in wireless sensor networks, in *2010 IEEE 35th Conference on Local Computer Networks (LCN)*, (2010), pp. 448–455
 25. UU Alias, Swathiga, C Chandrasekar, An efficient fuzzy based congestion control technique for wireless sensor networks. *Int. J. Comput. Appl.* **40**(14), 47–55 (2012)
 26. M Zarei, A Rahmani, A Sasan, M Teshnehlab, Fuzzy based trust estimation for congestion control in wireless sensor networks, in *Proceedings of the International Conference on Intelligent Networking and Collaborative Systems (INCOS)*, (2009), pp. 233–236
 27. M Li, Z Li, A Vasilakos, A survey on topology control in wireless sensor networks: taxonomy, comparative study, and open issues. *Proc. IEEE.* **101**(12), 2538–2557 (2013)
 28. Y Durmus, A Ozgovde, C Ersoy, Distributed and online fair resource management in video surveillance sensor networks. *IEEE Trans Mobile Comput.* **11**, 835–848 (2012)
 29. V Kawadia, P Kumar, A cautionary perspective on cross layer design. *IEEE Wireless Commun.* **12**, 3–11 (2005)
 30. W Su, T Lim, Cross-layer design and optimization for wireless sensor networks, in *Seventh ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD)*, (2006), pp. 278–284
 31. G Shah, W Liang, X Shen, Cross-layer design for, QoS support in wireless multimedia sensor networks, in *2010 IEEE Proceedings of the Global Telecommunications Conference (GLOBECOM 2010)*, (2010), pp. 1–5
 32. A Vasilakos, K Zikidis, ASAFES.2: a novel, neuro-fuzzy architecture for fuzzy computing based on functional reasoning, in *Fuzzy Systems, 1995. International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium., Proceedings of 1995 IEEE Int*, Volume 2, (1995), pp. 671–678
 33. JM Mendel, Fuzzy logic systems for engineering: a tutorial, in *Proceedings of the IEEE*, Volume 83, (1995), pp. 345–377
 34. OPNET Modeler. https://www.opnet.com/solutions/network_rd/modeler.html. [accessed at May 2012]
 35. MA Yigitel, OD Incel, C Ersoy, Design and implementation of a QoS-Aware MAC protocol for wireless multimedia sensor networks. *Comput. Commun.* **34**(16), 1991–2001 (2011)
 36. CF Garcia-Hernandez, PH Ibarquengoytia-Gonzalez, J Garcia-Hernandez, JA Perez Diaz, Wireless sensor networks and applications: a survey, in *International Journal of Computer Science and Network Security (IJCNS)*, Volume 7, (2007), pp. 264–273
 37. AHF Abdul Abdul Hamid, RA Rashid, N Faisal, SK Syed Yusof, Development of IEEE802.15.4 based wireless sensor network platform for image transmission, in *International Journal of Engineering and Technology (IJET)*, Volume 9, (2009), pp. 112–118
 38. C Park, PH Chou, eCAM: ultra compact, high data-rate wireless sensor node with a miniature camera, in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys)*, (2006), pp. 359–360
 39. Atmel Corporation, *8-bit AVR Microcontroller with Low Power 2.4GHz Transceiver for ZigBee and IEEE 802.15.4*. Rev. 8266CS-MCU Wireless-08/11 2011
 40. M Zuniga, B Krishnamachari, Analyzing the transitional region in low power wireless links, in *Proceedings of the First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, (2004), pp. 517–526
 41. K Sohrabi, B Manriquez, G Pottie, Near ground wideband channel measurement in 800-1000 MHz, in *Proceedings of the IEEE 49th Vehicular Technology Conference*, Volume 1, (1999), pp. 571–574
 42. B Karp, HT Kung, GPSR: greedy perimeter stateless routing for wireless networks, in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, (2000), pp. 243–254
 43. A Hore, D Ziou, Image quality metrics: PSNR vs. SSIM, in *Proceedings of the 20th International Conference on Pattern Recognition (ICPR)*, (2010), pp. 2366–2369
 44. T Liu, Y Wang, J Boyce, H Yang, Z Wu, A novel video quality metric for low bit-rate video considering both coding and packet-loss artifacts. *Selected Topics Signal Process. IEEE J.* **3**(2), 280–293 (2009)
 45. A Chan, K Zeng, P Mohapatra, SJ Lee, S Banerjee, Metrics for evaluating video streaming quality in lossy IEEE 802.11 wireless networks, in *2010 Proceedings IEEE INFOCOM*, (2010), pp. 1–9

doi:10.1186/1687-1499-2014-63

Cite this article as: Sonmez et al.: Fuzzy-based congestion control for wireless multimedia sensor networks. *EURASIP Journal on Wireless Communications and Networking* 2014 **2014**:63.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com