Efficient Multimedia Transmission in Wireless Sensor Networks

Jose F. Mingorance-Puga, Gabriel Maciá-Fernández, António Grilo and Nestor M. C. Tiglao

***Abstract*—Real-time multimedia data such as video are usually loss-tolerant but require timely delivery in order to be useful to the application. Loss recovery through the retransmission of lost data may introduce unacceptable delays, which is the reason why these data types are usually delivered with no transport layer reliability, using erasure coding and similar techniques to maximize data recovery at the receiver. However, in Wireless Multimedia Sensor Networks (WMSNs), these mechanisms are not enough to provide an acceptable image quality and, thus, reliable transport protocols adapted to these requirements are needed. This paper presents some mechanisms to improve multimedia transmissions in WMSNs when reliable transport layer protocols are used. They consist of assigning a budget of time for the sending of certain amount of information and estimating if the channel conditions allow to complete the transmission or not. If it is not likely to complete it, then the transmission is stopped, thus saving important energy resources in the sensors.**

We evaluate this approach by modifying the behavior of a previously proposed reliable transport protocol (DTSN). Our proposal, M-DTSN, improves DTSN flexibility by managing the trade-off between media quality and timely delivery for real-time multimedia data with some degree of loss-tolerance. The simulation results demonstrate that the advantages of M-DTSN for the transmission of multimedia data are quite significant when compared with the original DTSN protocol.

***Index Terms*—Sensor networks, multimedia transmission, transport layer.**

I. Introduction

Multimedia wireless sensor networks (MWSN) are sensor networks which have capabilities for dealing with multimedia information. They are composed of multimedia sensors which are able to capture and transmit multimedia information, *e.g.*, sensors might typically be low-resolutions cameras in surveillance environments. MWSNs present some challenges that are common to wireless sensor networks, *i.e.*, the existence of limited resources, like sensors memory, energy consumption, CPU performance, etc. Due to these limitations, it is essential to maximize the lifetime of sensors by reducing the amount of information that traverses the network. In addition, MWSNs have special characteristics that make them different from traditional WSNs, *i.e.*, they must be able to handle special quality of service requirements for multimedia traffic. This means that sensors should be able to adapt their transmission capabilities to the particularities of multimedia information.

J. F. Mingorance-Puga (jose.mingorance@gmail.com) and G. Maciá-Fernández (gmacia@ugr.es) are with University of Granada, CITIC, Spain. António Grilo (antonio.grilo@inesc.pt) is with INESC-ID/INOV and IST, Lisbon, Portugal. Nestor M. C. Tiglao (nestor.tiglao@inesc-id.pt) is with EEEI, University of the Philippines, Diliman, Quezon City, Philippines and INESC-ID/IST, Lisbon, Portugal.

Multimedia information is typically composed of video and audio data. Here, we will consider only multimedia information that require certain timing restrictions for its transmission. Specifically, and without any lack of generality, we focus on video transmission, as it constitutes a typical real-time multimedia flow. Video information is composed of a sequence of image frames, normally coded following a specific standard, *e.g.*, MPEGv2. Scalable video coding techniques allow the sender to support multiple video resolutions by encoding a frame with a base layer and one or more enhancement layers. Thus, video codecs are able to separate the critical information of a frame, *i.e.*, a base layer, to render a low resolution image, and the complementary information, *i.e.*, enhancement layers, that allows it to build high resolution images. When a transmission is established between two points, the amount of multimedia information required by the receiver induce the adaptation of the specific features used in the transmission by the codecs, *e.g.*, frame size, maximum resolution, number of enhancement resolution layers, sampling period for frames, etc.

Despite the capabilities of the receiver, network capabilities in the case of WMSNs are also an important limitation factor with respect to the maximum video resolution allowed during a given transmission. Bad quality links mainly due to the reduced power equipment installed in the sensors pose a challenge for the design of transmission protocols in sensor networks. Furthermore, when multimedia information is involved, the existence of delay and jitter is a traditional problem considered by many researchers. Due to the existence of this error prone environments, recent research points out that the use of reliable transport protocols is recommendable for these transmissions in order to obtain acceptable quality [1]. In this direction, some recent proposals have been contributed [2], [3]. These contributions have mainly focused on the design of transport protocols optimized for achieving fast transmission rates and fast error recovery, offering the application layer a better throughput of information which allows using better resolution for the multimedia information. Notwithstanding this, to the best of our knowledge, all of the contributed approaches consider the transport layer as a monolithic system trying to behave as better as possible on its own.

In this paper, we explore how transport layers in WMSNs could make use of certain knowledge about the characteristics of the transmitted multimedia information to improve the network performance perceived by the application layer. Specifically, we analyze some useful features of multimedia information and discover that it is possible to exploit the knowledge about the sampling rate of this information to allow a considerable rise in the performance observed by the receivers. The basic idea for our approach is explained now. Let us consider a simple example consisting of a video transmission between two peers in a WMSN. Here, we suppose that the minimum information unit for a receiver is an image frame. The receiver is able to render a frame when it arrives before the rendering instant. Then, if a transmitter were able to assess if the frame is not able to reach the receiver in time, *i.e.*, before the instant of rendering the frame, it would be no worth in sending this information. Our approach exploits this idea and, therefore, we suggest some mechanisms for improving the performance of the multimedia transmissions.

Although our approach could be used with any other reliable transport protocol, in order to implement and evaluate it, we have chosen the distributed transport protocol for sensor networks, DTSN [4]. Then, our proposal is a modified version of this protocol, termed multimedia DTSN, M-DTSN. We extensively evaluate our approach in a simulated scenario, obtaining promising results.

This paper is organized as follows. In Section II, we provide some fundamentals about the specification of the DTSN protocol, which are not essentials for this work but help the reader to understand the modifications suggested in our approach. Then, we describe our proposal in Section III. Next, it is extensively evaluated and discussed in Section IV. Finally, some related work is discussed in Section V and conclusions are drawn in Section VI.

II. DTSN Fundamentals

The basic DTSN specification [4] was thought for critical data transfer requiring end-to-end full reliability in the fashion of TCP. However, for sake of improving energy efficiency in WSNs, DTSN employs a Selective Repeat Automatic Repeat reQuest (ARQ) using negative acknowledgements (NACK). Positive acknowledgement packets (ACK) are also used to prevent the situations where all the message or its last packet were lost (which cannot be detected solely based on NACKs). Both NACKs and ACKs are to be sent by the receiver only upon request by the sender (Explicit Acknowledgement Request – EAR), which can be piggy-backed in data packets.

In DTSN, a session is a source/destination relationship univocally identified by the tuple `<source address, destination address, application identifier, session number>`, designated the session identifier. The session is soft-state by nature both at the source and the destination, being created when the first packet is processed and terminated upon the expiration of an activity timer (provided that no activity is detected and there are no pending delivery confirmations). A randomly chosen session number is appended in order to unambiguously distinguish between successive sessions sharing the endpoint addresses and application identifier. The processing of a packet with the same `<source address, destination address, application identifier>` and different session number causes the previous session to be terminated and a new one restarted with the new session number along the path between sender and receiver.

Within a session, packets are sequentially numbered. The Acknowledgement Window (AW) is defined as the number of packets that the source transmits before generating an EAR. The output buffer at the sender works as a sliding window, which can span more than one AW. The size of the output buffer and of the AW depend on the specific scenario, namely on the memory constraints of individual nodes.

In order to minimize end-to-end retransmissions, the intermediate nodes are able to cache a number of packets with a certain probability. Upon interception of a NACK message, in case they are able to find any of the requested packets in cache, they are able to retransmit them to the receiver. After deleting the respective sequence numbers from the NACK, the latter is forwarded towards the source. Since the sequence numbers for those retransmitted packets are not present in the NACK anymore, the intercepting nodes located closer to the sender (as well as the sender itself) will not retransmit those packets. As such, the mean number of hops that a retransmitted packet has to travel is lower than in a traditional end-to-end retransmission mechanism.

III. Multimedia Transmission Mechanisms

In this section, we introduce our approach for improving the efficiency obtained during the transmission of multimedia information in a wireless sensor network. We consider a scenario composed of a transmitter sending multimedia information to a receiver through a network of sensors. As previously discussed, our hypothesis is that we are using a reliable transport protocol for the transmission of the information. Although DTSN is chosen as a candidate for our study, our results are not constrained to it, but generalizable to others.

Three main factors are considered in our scenario of study: (*i*) multimedia flows are time-constrained, (*ii*) sensors are limited in the amount of energy allowed for the transmissions and (*iii*) data link layer / MAC reliability is usually not enough to assure the required delivery success ratios, and even FEC codes may not be effective in radio channels featuring bursty error patterns. We sustain that if the transport layer in a given sensor is informed about certain features regarding the needs of the application layer, it is possible to considerably improve the required transmissions. Next, we explain how this could be achieved.

Multimedia receivers present certain limitations due to the time-constrained nature of the received information. Some are related to the amount of tolerated delay and the maximum allowed jitter during the transmission. In addition, the receiver should also obtain the information coming from the transmitter with a required mean traffic rate, *e.g.* considering a video streaming, frames should arrive with a frame rate which is mainly determined by the sample ratio of the video signal.

Although many authors have focused their work on improving the network performance in term of the delay and jitter, we leverage the existence of a constraint in the receiver related to the need for a guaranteed mean traffic rate, for developing our mechanism. To make things more specific, and without lack of generality, we consider an scenario in which a video receiver is reproducing a video stream composed of a sequence

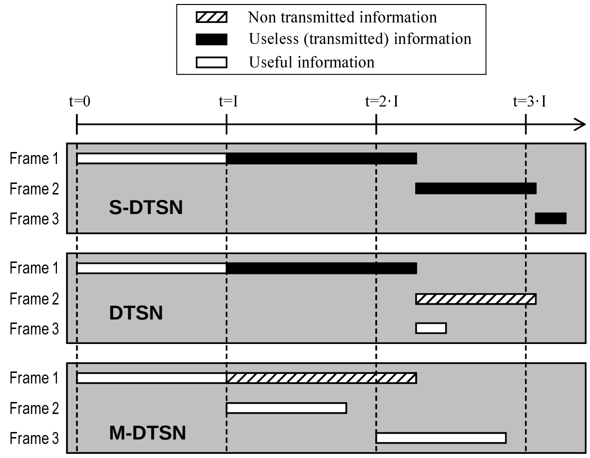


Figure 1: Diagram showing the reception of frames at the receiver and the number of useful/useless information from the receiver perspective, for three different approaches: S-DTSN, DTSN and M-DTSN.

of image frames. For the sake of simplicity, let us assume that the size of all frames is equal and let its value be noted as S. Furthermore, we consider that the receiver requires to obtain one frame every period of time termed *frame interval, I,* to be able to reproduce the video sequence. Both the transmitter and the receiver are located in a wireless sensor network separated by a number of H hops.

In this scenario, we are interested in the suggestion of a transmission mechanism that improves the effectiveness achieved by current transport protocols. In order to allow the comparison of our approach with the behavior of other protocols, we now focus on DTSN and clarify for this protocol which are the differences in the transmission of the information with our proposal.

Let us consider an example scenario shown in Figure 1. Here, three consecutive frames are sent by a transmitter to a receiver. Although all of them have a same size, S, the amount of time employed for their transmission is not equal, as it heavily depends on the network conditions, *i.e.*, link quality, number of hops, number of retransmissions, etc. The time employed for the reception of each of the three frames is shown in horizontal bars in Figure 1. The same representation is done for three different strategies in the transmission of the information: S-DTSN, DTSN and M-DTSN. These strategies are explained next. Among them, M-DTSN is our proposal. The other two are presented to allow the comparison of M-DTSN with other simpler approaches.

Simple DTSN, S-DTSN, is the simplest strategy in the use of DTSN. Here, the transport layer receives, every frame interval, I, a new frame to deliver to the receiver. These frames are sent sequentially, in such a way that a given frame is not sent before the transmission of the previous one is completed. We see in Figure 1 that frame 1 lasts more than I seconds. This implies that the receiver obtains some information considered useful (white bar), as it arrives during the interval of time reserved for the reception of that frame, $t \in [0, I]$. Nevertheless, the information received for frame 1 after the instant $t = I$ is useless, as the frame should have been rendered just at $t = I$. This useless information, at the same time that does not contribute to the video stream, makes other frames be delayed, thus increasing their probability of arriving late also. In our example, once that frame 1 has been completely sent, frame 2 is also sent. The delay introduced by the useless part of frame 1 makes frame 2 arrive after its rendering instant, $t = 2 \cdot I$, thus becoming also another useless frame. This also happens for frame 3. In sum, using S-DTSN none of the transmitted frames in our example are finally rendered by the receiver, due to the fact that they arrive with a higher delay than that allowed by the receiver.

DTSN is the second approach presented for our comparison. The behavior in DTSN is similar to that of S-DTSN, but in this case the transmitter is slightly modified. At the beginning of the transmission for a frame, the transmitter is able to select the frame which is being expected by the receiver and, thus, discard the previously queued frames if they have not been sent. Let us clarify this point by observing the diagram in Figure 1 for DTSN. The receiver experiences the same evolution as in S-DTSN for frame 1. However, when the transmitter has to send the second frame, it notices that the instant for rendering that frame has already occurred and, thus, decides to send frame 3. This frame now arrives in time to the receiver to be rendered at $t = 3 \cdot I$. This modified behavior with respect to S-DTSN present two benefits in this example: first, the transmission of frame 2 is not done, thus saving energy and, second, the receiver is able to render at least one of the frames.

M-DTSN is our contribution for improving the efficiency of the transmission. Here, the transmitter assumes that the sending of a frame should not last more than I seconds. On the contrary, the receiver would not be able to receive the complete frame in time. Consequently, the transmitter stops the transmission of a frame when I seconds have elapsed since the beginning of its transmission. Looking at our example in Figure 1, we see that the transmission of frame 1 is stopped at $t = I$. Then, the transmission of frame 2 begins due to a change in the network conditions, this frame now arrives in time to the receiver. Finally, frame 3 also arrives successfully and it is rendered. We see in our example that, with this strategy, two of the frames are completely received, while the reception of the first frame is only partial and, thus, it would be useless.

In sum, our proposal for improving the transmission of multimedia information in sensor networks consist of an adapted transport layer, which is able to receive a primitive from the application layer requesting the sending of a frame, and the amount of time (deadline) for the transmission of that frame.

Although the benefits of M-DTSN over the other two approaches are clear in the scenario shown in Figure 1, some issues should be clarified. First, the fact that many receivers are able to recover the complete information of a frame from partial information. As an example, in [5], the authors propose a codec which is able to recover the whole frame from 5/7 (72%) of its information. In this scenario, it is not clear if the benefit of M-DTSN keeps over that of the other two approaches, as we stop the transmission when the deadline given by the frame interval is reached. Second, receivers usually implement buffers to allow some jitter in the transmission. This means that receivers will be able to receive some information after the deadline for the transmission is reached. For these reasons, we thoroughly evaluate the evolution of M-DTSN and compare it with that of S-DTSN and DTSN in Section IV.

IV. Evaluation of the Approach

Here, we evaluate our approach by means of simulation. Our intention is to check if M-DTSN outperforms DTSN or S-DTSN as expected, and to analyze if these results are valid for a wide range of configurations. For this purpose, we have implemented M-DTSN in the TOSSIM simulator using TinyOS 2.1.0 version, and we have used a previous implementation of DTSN, as presented in [4].

We have made an extensive set of experiments in which the variation of the following variables is considered: frame size, S, channel attenuation of all the links in the network, A, number of intermediate hops between receiver and transmitter, H, and frame interval for the multimedia transmission, I. We have thoroughly explored the combination of these variables by selecting multiple ranges of operation. For all these scenarios, simulations have been carried out to obtain the efficiency of the transmission in the three approaches presented in Section III. In the simulations, fixed configuration values have been selected for the rest of the parameters needed to setup the studied scenarios. Specifically, DTSN layer parameters have been properly configured in such a way that undesired effects, like window congestion or excessive delays, are eliminated in every simulation scenario. This has been done by following the guidelines presented in [4]. We have also setup a network of 20 sensor nodes where all the bidirectional links have a variable attenuation (variable of study) and a well known noise pattern example included in the TOSSIM package (`heavy-meyer.txt`). Regarding the application layer, the receiver implements a buffer that supports certain jitter with a maximum value of 100 ms.

Measuring the efficiency of a multimedia transmission

We are now interested in evaluating the results yielded by M-DTSN in terms of the efficiency of the multimedia transmission. This efficiency is understood here as the perception that the application layer obtain from the service given by the transport layer. Therefore, we are interested in discovering how much useful information the application layer is going to obtain when M-DTSN is used. With this purpose, we define certain indicators that allow measuring this effectiveness.

Let us first define, for a given frame sent during a multimedia transmission, its *frame completeness* as the percentage of the frame bytes that arrive before the instant at which the receiver should render the frame. We also define, for a specific multimedia transmission, the *average frame completeness, AFC*, as the mean value of the frame completeness for all the frames that should be sent during this multimedia transmission.

In a first approach, receivers would only be capable of rendering those frames that completely arrive before the rendering instant, *i.e.*, those with a frame completeness equal to 100%. Nevertheless, as previously indicated, many codecs recently proposed are able to recover the information for the whole frame when only a portion of the frame is received. As an example, the codification proposed in [5] by using erasure coding over lossy networks is able to recover a complete frame if more of 72% of its transmission is reached. For this reason, we are not only interested in exploring the amount of frames that completely arrive in time at the receiver, but also in the percentage of the frame information that has reached the destination at the instant for rendering the frame. This is measured, precisely, by the previously defined indicator: *frame completeness*.

Results from the simulations

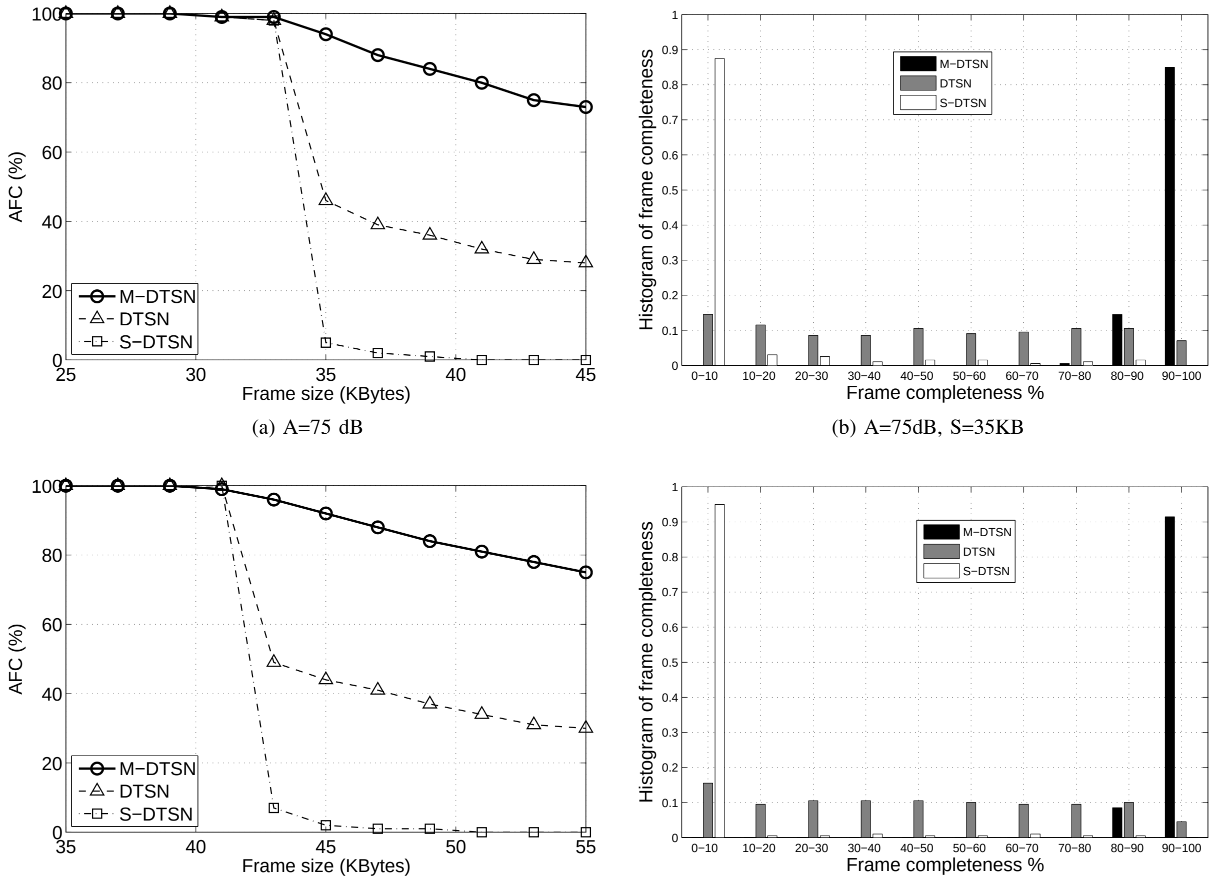
Here, we show some of the results obtained from the simulated experiments. Specifically, for every of the four studied variables, *i.e.*, S, A, H, I, we give details for two different scenarios. In each of the results, we show the average frame completeness obtained for the three strategies explained in Section III, *i.e.*, M-DTSN, DTSN and S-DTSN. Furthermore, for a better visibility of the results, we study certain interesting points of the simulation and obtain the distribution of the frame completeness.

In all the results provided next, the values for the studied variables are well indicated. Sizes for the frames have been adjusted to ranges of the typical values for SQCIF or compressed CIF/QCIF frames. When not explicitly specified, the default values shown in Table I apply. We assume an underlying ZigBee network.

Performance dependence with the frame size. Figure 2 shows the results for two different set of simulations. In the first one, an attenuation A=75dB is considered for the channel, while A=0dB is configured for the second. In both, the frame size has been varied and the average frame completeness obtained -Figures 2(a) and 2(c)-. We can check in these figures that, for sizes higher than S=33KB (A=75dB case) and S=42KB (A=0dB case), the value of AFC decreases as the size increases. This behavior is due to the fact that the

Table I: Default configuration values for the experimental simulation environment.

Parameter	***Value***
Application parameters	
Frame interval	5s
Jitter allowed in the receiver	100ms
Network parameters	
Channel noise	defined by heavy-meyer.txt
Number of hops	1
DTSN parameters	
DTSN window size	50 packets
DTSN ack window size	12 packets
Maximum number of EAR retries	10
Sender DTSN activity timeout	30*250ms
Receiver DTSN activity timeout	40*250ms



(a) A=75 dB

(b) A=75dB, S=35KB

(c) A=0dB

(d) A=0dB, S=45KB

Figure 2: Performance comparison between M-DTSN and DTSN, S-DTSN when the frame size, S, varies: (a) Average frame completeness for an attenuation of 75 dB; (b) histogram of the frame completeness for attenuation 75 dB and S=35KB; (c) average frame completeness for an attenuation of 0 dB; (d) histogram of the frame completeness for attenuation 0 dB and S=45KB.

bigger the size of the frames, the higher the amount of time to deliver them, thus reducing their *frame completeness*.

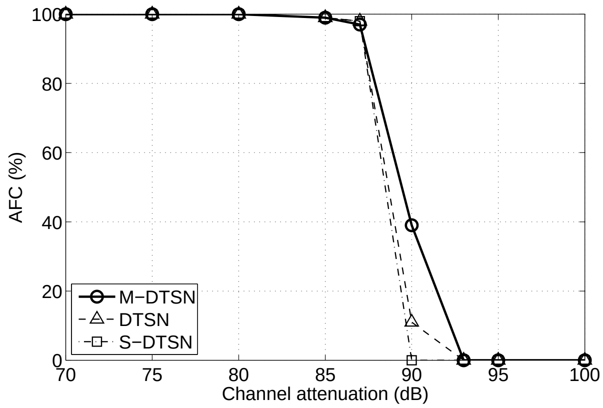
It is also interesting to determine the amount of frames that would successfully be rendered by the receiver. For this purpose, it is first necessary to establish a threshold in the amount of received information needed for each frame to be rendered. Here, we will consider that our multimedia transmission utilizes the codec proposed in [5], the threshold being in this case 72% of the total size of a frame. With the purpose of allowing this kind of evaluation, we obtain the distribution of the frame completeness for two significative points and using the method of the histogram. These distributions are shown in Figures 2(b) and 2(d). Here, we see that all the frames received when M-DTSN is used are above AFC=70% in both scenarios. This implies that the selected codec would be able to recover all of the frames. On the contrary, by using DTSN, only 24% of the frames arrives with AFC>72%, while this percentage is only around 2% for S-DTSN.

It is clear from these figures that the performance of M-DTSN is considerably superior to that of DTSN and S-DTSN. As expected, the results obtained for DTSN are also better when compared with S-DTSN, as the latter applies less intelligence to the transmission of the frames.

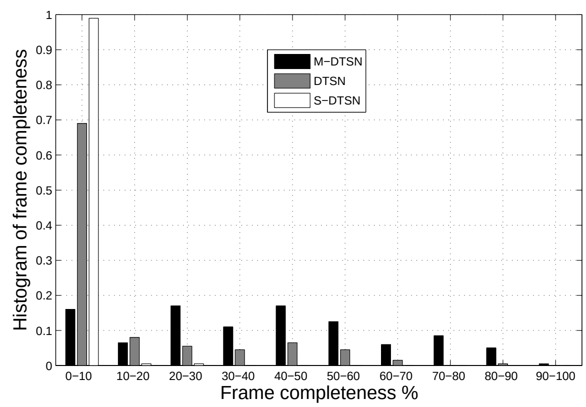
Performance dependence with the link quality. Figure 3 shows the results obtained when the link attenuation is modified. Figures 3(a) and 3(b) show the results for a frame size S=10KB, while Figures 3(c) and 3(d) for a size S=30KB. We see that the evolution of AFC is the same in both scenarios, although in the case S=10KB the values for AFC drop more abruptly than for S=30KB. This is because big frames are sent during more time, thus experiencing a higher overall variance in the transmission due to channel conditions. For this reason, the range of attenuation values along which AFC decreases is longer than when small frames are considered.

Performance dependence with the frame interval. The results corresponding with the variations on the value of the frame interval, I, are represented in Figure 4. They clearly show that, as expected, the average frame completeness increases as the frame interval does. This is because a longer time is devoted to the reception of each frame, thus increasing the received percentage of the frame.

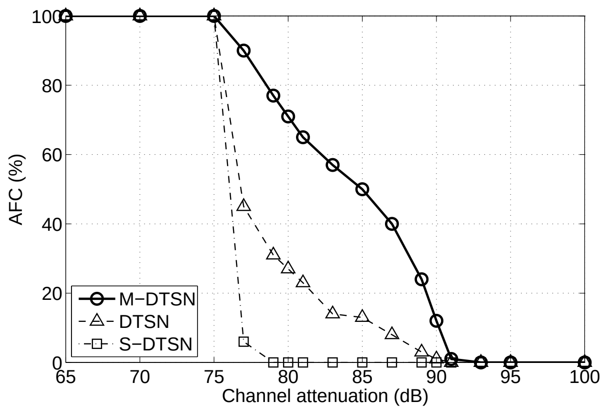
In these results, we confirm that the behavior of M-DTSN is much better than that obtained by DTSN and S-DTSN. In addition, we can see how M-DTSN is able to concentrate the histogram to higher values of the distribution of frame



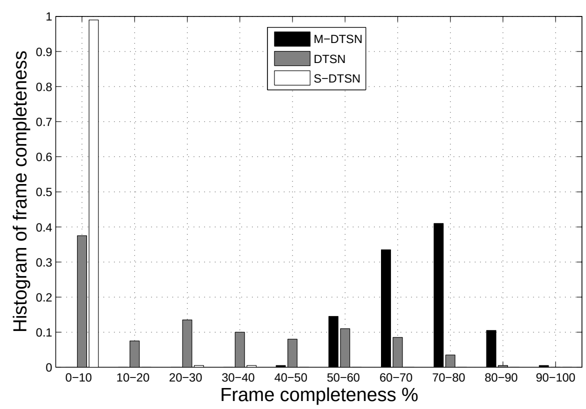
(a) S=10KB



(b) S=10KB, A=90dB



(c) S=30KB



(d) S=30KB, A=80dB

Figure 3: Performance comparison between M-DTSN and DTSN, S-DTSN when the link quality (attenuation, A) varies: (a) Average frame completeness for S=10KB; (b) histogram of the frame completeness for S=10KB and A=90dB; (c) average frame completeness for S=30KB; (d) histogram of the frame completeness for S=30KB and A=80dB.

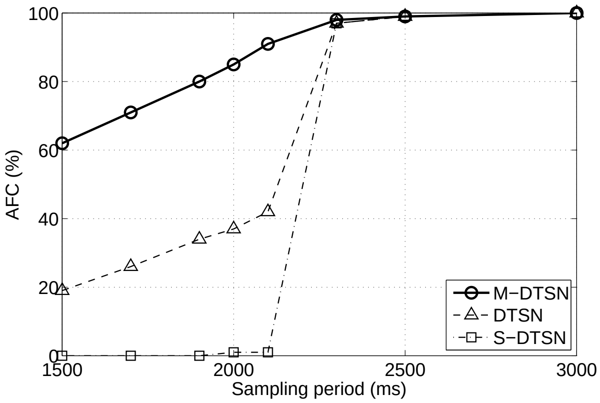
completeness -see Figure 4(b)-, while DTSN for example generates a quasi-flat distribution. This implies that, in cases where values around AFC=70% are acceptable for the receiver, the percentage of valid frames would be higher than 90% for M-DTSN, while being 16% for DTSN -see Figure 4(b)-.

Performance dependence with the number of hops. The results obtained when we consider a variation in the number of hops between the origin and the destination of a multimedia transmission are shown in Figure 5. Here, we confirm that M-DTSN is still better in these cases when the number of hops is low enough to allow the arrival in time of a considerable part of the multimedia traffic. Note that the histograms for the three strategies clearly indicate that the amount of frames potentially valid for being rendered is much advantageous for M-DTSN.

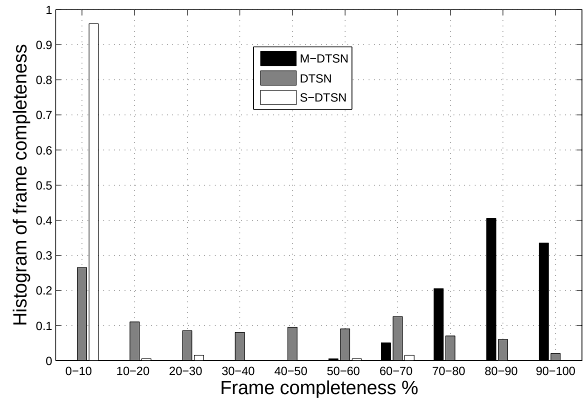
V. Related Work

Traditionally, the primary function of the transport layer has been to provide congestion control and guaranteed end-to-end delivery. Furthermore, the transport layer also provides network and quality of service (QoS) abstraction where the application layer specifies the QoS parameters and configures the service to meet the desired QoS guarantees through the transport layer API. In addition, it is desirable for the transport layer to reduce latency and maximize throughput. Transmitting multimedia streaming data in WSNs is a relatively new research topic as identified by the comprehensive survey in [6]. Transport protocols used for multimedia transmission over the Internet, such as RTP/RTCP [7] and DCCP [8] cannot be applied directly to wireless sensor networks because they were mainly designed to work in wired networks. As such, the underlying assumptions do not hold in wireless networks. In particular, wired networks do not suffer from interference and packet losses arise mainly from congestion. Wireless networks on the other hand suffer from physical layer impairments which lead to high bit error rates. As a result, the use of Internet/wired network transport protocols lead to throughput degradation and huge energy inefficiencies [9].

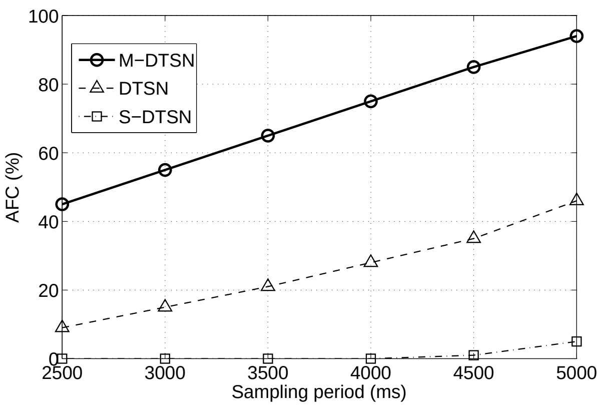
Most of the well-known transport protocols developed for wireless sensor networks like ESRT [10], CODA [11], SenTCP [12], and RMST [13] provide poor multimedia delivery performance because these protocols do not consider end-to-end delay and, thus, frame delays may be very high [14]. On the other hand, other recent works such as DTSN [15], CTCP [16], and RCRT [17] focus on reliably deliver the information



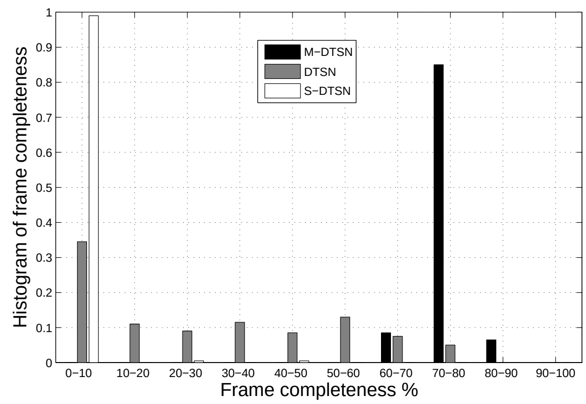
(a) S=15KB, A=75dB



(b) S=15KB, A=75dB, I=2000ms



(c) S=35KB, A=75dB



(d) S=35KB, A=75dB, I=4000ms

Figure 4: Performance comparison between M-DTSN and DTSN, S-DTSN when the frame interval, I, varies: (a) Average frame completeness for S=15KB and A=75dB; (b) histogram of the frame completeness for S=15KB, A=75dB and I=2000ms; (c) average frame completeness for S=35KB and A=75dB; (d) histogram of the frame completeness for S=35KB, A=75 dB and I=4000ms

without a high delay.

Despite the effort in this field, few protocols have been developed to address the specific requirements of multimedia traffic. For example, although it was not designed specifically for multimedia applications, the DART protocol [18] couples real-time delay bounds with event-to-sink reliability. Another approach is the use of multipath routing, which can potentially improve the delivery of multimedia data by sending traffic over multiple paths. MRTP [2] is one of the first protocols using multipath transport and shows that data partitioning through a few flows is enough to improve performance. MPMPS [3] supports multiple traffic priorities to differentiate multimedia streams from other traffic and chooses the maximum number of paths from all found node-disjoint routing paths in order to maximize the throughput of streaming data transmission.

Lastly, network coding has emerged as an alternative to traditional packet routing, and has opened a new research area in which novel protocols can be developed. A recent work [19] provides an overview of network coding principles and some recent work of its application to media streaming.

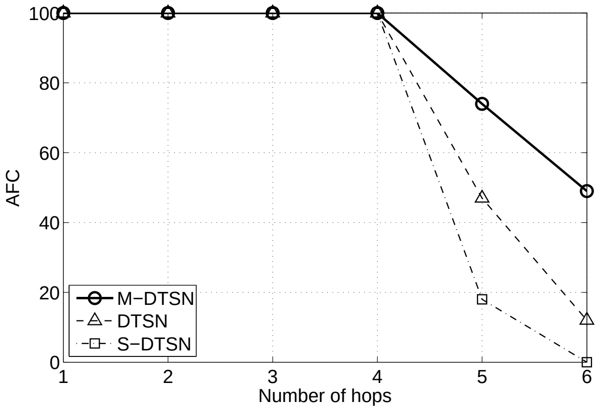
VI. Conclusions and Future Work

In this paper we have studied if it is possible to develop reliable transport protocols aimed to optimize the trade-off between timeliness and reliability of partially loss-tolerant delay-sensitive multimedia data in WMSNs. We have contributed a mechanism to be implemented in the transmitters which is able to considerably augment the efficiency of multimedia transmissions. This mechanism has been evaluated by adapting the reliable transport protocol DTSN. The adapted version, M-DTSN, has proven to be much more effective in the transmission of information. Moreover, we have evaluated if this approach is valid when coding network techniques are used jointly with reliable transport protocols, reaching the conclusion that our approach is highly beneficial in these environments.

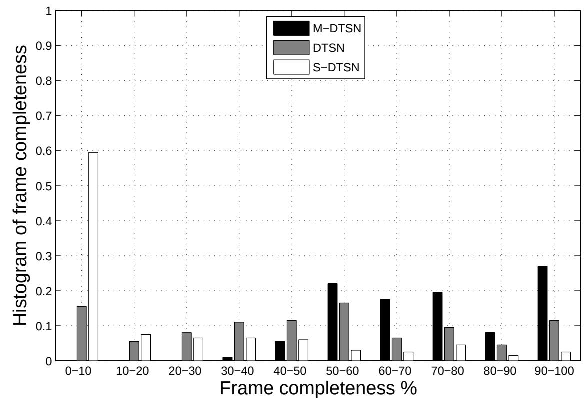
As future work, we plan to optimize this mechanism by developing adaptive algorithms able to select the appropriate configuration values for our approach as a function of the network and the receiver evolution along the time.

Acknowledgments

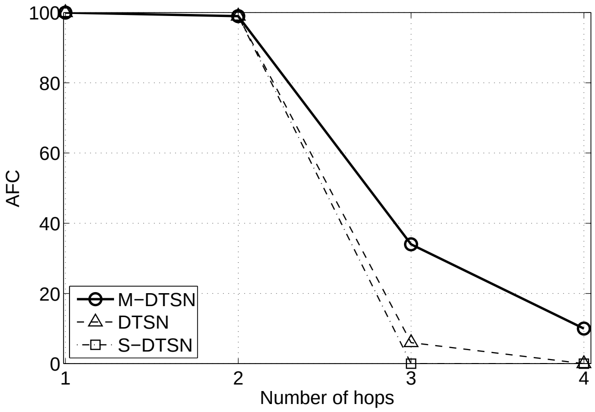
This work has been partially supported by the Spanish MICINN under project TEC2008-06663-C03-02 (70% FEDER funds), the European Community Seventh Framework Programme under grant agreement no. 225186, project WSAN4CIP, and the Department of Science and Technology and the University of the Philippines through the ERDT Grant.



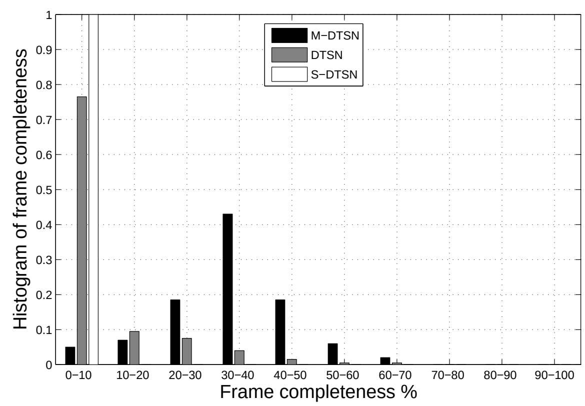
(a) S=10KB, A=0dB



(b) S=10KB, A=0dB, hops=5



(c) S=15KB, A=75dB



(d) S=15KB, A=75dB, hops=3

Figure 5: Performance comparison between M-DTSN and DTSN, S-DTSN when the number of hops varies: (a) Average frame completeness for S=10KB and A=0dB; (b) histogram of the frame completeness for S=10KB, A=0dB and 5 hops; (c) average frame completeness for S=15KB and A=75dB; (d) histogram of the frame completeness for S=15KB, A=75 dB and 3 hops.

REFERENCES

[1] Chonggang Wang, Sohraby K., Bo Li, Daneshmand M., Yueming Hu, A Survey of Transport Protocols for Wireless Sensor Networks. IEEE Network, Vol. 20, 2006, pp. 34–40.

[2] Shiwen Mao, D. Bushmitch, S. Narayanan, and S. Panwar, “MRTP: a multiflow real-time transport protocol for ad hoc networks,” IEEE Transactions on Multimedia, vol. 8, 2006, pp. 356-369.

[3] L. Zhang, M. Hauswirth, L. Shu, Z. Zhou, V. Reynolds, and G. Han, “Multi-priority Multi-path Selection for Video Streaming in Wireless Multimedia Sensor Networks,” Ubiquitous Intelligence and Computing, 2009, pp. 439-452

[4] F. Rocha, A. Grilo, P. Pereira, “Performance Evaluation of DTSN in Wireless Sensor Networks”, Proceedings of the 4th EuroNGI Workshop on Wireless and Mobility, Barcelona, Spain, January 2008. In Springer-Verlag Lecture Notes in Computer Science, vol. 5122, 2008.

[5] X. Zhang, and X.-H. Peng, “A testbed of erasure coding on video streaming system over lossy networks,” in Proc. IEEE 7th International Symposium on Communications and Information Technologies (ISCIT), Oct. 2007

[6] S. Misra, M. Reisslein, and Guoliang Xue, “A survey of multimedia streaming in wireless sensor networks,” Communications Surveys & Tutorials, IEEE, vol. 10, 2008, pp. 18-39.

[7] “RTP: A Transport Protocol for Real-Time Applications,” http://tools.ietf.org/rfc/rfc3550.txt.

[8] “Datagram Congestion Control Protocol (DCCP),” http://tools.ietf.org/rfc/rfc4340.txt.

[9] T. Braun, T. Voigt, and A. Dunkels, “TCP support for sensor networks,” 2007 Fourth Annual Conference on Wireless on Demand Network Systems and Services, Obergurgl, Tyrol, Austria: 2007, pp. 162-169.

[10] O. Akan and I. Akyildiz, “Event-to-sink reliable transport in wireless sensor networks,” IEEE/ACM Transactions on Networking, vol. 13, 2005, pp. 1003-1016.

[11] C. Wan, S.B. Eisenman, and A.T. Campbell, “CODA,” Proceedings of the first international conference on Embedded networked sensor systems - SenSys ’03, Los Angeles, California, USA: 2003, p. 266.

[12] C. Wang, K. Sohraby, and B. Li, “SenTCP: a hop-by-hop congestion control protocol for wireless sensor networks,” Proc. IEEE INFOCOM 2005, March 2005.

[13] F. Stann and J. Heidemann, “RMST: reliable data transport in sensor networks,” Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications, 2003., Anchorage, AK, USA: , pp. 102-112.

[14] O. Akan, “Performance of Transport Protocols for Multimedia Communications in Wireless Sensor Networks,” IEEE Communications Letters, vol. 11, 2007, pp. 826-828.

[15] B. Marchi, A. Grilo, and M. Nunes, “DTSN: Distributed Transport for Sensor Networks,” 2007 IEEE Symposium on Computers and Communications, Santiago, Portugal: 2007, pp. 165-172.

[16] E. Giancoli, F. Jabour, and A. Pedroza, “CTCP: Reliable Transport Control Protocol for sensor networks,” 2008 International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Sydney, Australia: 2008, pp. 493-498.

[17] M.H. Yaghmaee and D. Adjeroh, “A reliable transport protocol for Wireless Sensor Networks,” 2008 International Symposium on Telecommunications, Tehran: 2008, pp. 440-445.

[18] V.C. Gungor and Ö.B. Akan, “Delay aware reliable transport in wireless sensor networks,” International Journal of Communication Systems, vol. 20, 2007, pp. 1155-1177.

[19] E. Magli and P. Frossard, “An overview of network coding for multimedia streaming,” 2009 IEEE International Conference on Multimedia and Expo, New York, NY, USA: 2009, pp. 1488-1491.