Energy and Quality of Service Management in Wireless Multimedia Sensor Networks

Paulo Rogério Pereira, José Gonçalves, António Grilo, Carlos Fortunato, Mário Serafim Nunes, Augusto Casaca

Abstract—**Sensor networks are composed of resource constrained nodes that capture data from the environment, pre-process it and then transmit it to a sink node. This paper presents a scenario for monitoring an electricity distribution network, an energy analysis of the used sensor nodes and an intelligent energy and quality of service (QoS) manager. This manager continuously adapts the provided QoS according to the energy level of the nodes.**

Index Terms—**Wireless Sensor Networks, Quality of Service, Energy Management, Wireless Multimedia Sensor Networks.**

I. Introduction

Wireless Multimedia Sensor Networks (WMSN) are networks of wirelessly interconnected devices that are able to ubiquitously retrieve multimedia content such as video or audio streams, still images, and scalar sensor data from the environment [1]. The reductions in hardware cost and increased miniaturization have increased the use of WMSN. Some typical applications of WMSN are remote video surveillance, industrial process control, traffic management systems, advanced health care systems and environment monitoring.

The delivery of multimedia contents introduces quality of service (QoS) requirements such as transmission bandwidth, end-to-end delay, energy consumption, security and reliability. As the energy and processing power in sensor nodes is usually scarce, the QoS to be provided needs to be adapted in real time, according to the energy available in the nodes.

In this paper we present an energy and QoS management system for WMSN that dynamically balances the QoS provided according to the energy level of the nodes. The presented case study consists of a WMSN that is being deployed to monitor the Portuguese electricity distribution network provided by EDP (Energias de Portugal), the main electricity provider in Portugal. The electrical energy distribution infrastructure is a critical infrastructure that requires protection for safety and security reasons.

Fig. 1 shows the network architecture [2]. Power substations convert very high voltage (HV) to medium voltage (MV) that is distributed up to urban and suburban Power Transformers that convert MV to low voltage (LV) and provide electricity to customers. The electricity distribution network is tree-shaped with the customers being the leaves. There are sensors in the substation to monitor the temperature of the transformers, sensors in the power lines to monitor the current and detect and locate line breaks, e.g. if some tower falls, and video sensors in the MV/LV power transformers to detect intrusions. There is usually line-of-sight between the towers, which facilitates the establishment of wireless links, but the distances between towers may be significant, requiring directional antennas. The sensor network interacts with the Supervision Control And Data Acquisition (SCADA) system of the electricity provider to allow a centralized monitoring and control of the protection system. A gateway node acts as a sink node to receive data from the sensors and provides the interface to the Supervisory Station.

The video streaming scenario is the most demanding in terms of sensor network resources. Video is streamed multihop from the MV/LV power transformer to the SCADA supervisory station when a passive infrared (PIR) motion detector detects an intrusion.

While in the substation and in the MV/LV power transformer locations, 230 V AC is available to provide energy to the sensor nodes, the nodes in the towers need to harvest power from the high voltage line itself. The nodes in the towers are hung to the high voltage line through a current transformer [3] that draws a small current to provide energy to the node. The laboratory test of one such node is shown in Fig. 2. If the current in the high voltage line is below a certain threshold, the current drawn is not enough to power the node. As this situation may occur during the night or weekends, when the electricity consumption is reduced, the nodes are equipped with batteries that charge when the current is high enough, providing energy when it is not.

Manuscript received September, 2011. This work was supported in part by the FP7 WSAN4CIP (Wireless Sensor and Actuator Networks for the Protection of Critical Infrastructures) project under grant agreement no. 225186 as well as by FCT (INESC-ID multiannual funding) through the PIDDAC program funds and MPSat project (FCT Ref. PTDC/EEA-TEL/099074/2008).

P. Pereira, A. Grilo, M. Nunes and A. Casaca are with Inesc-ID/Inesc INOV, Instituto Superior Técnico, Technical University of Lisbon, Rua Alves Redol, nº 9, 1000-029 Lisboa, Portugal. (phone: +351-213100345; fax: +351-213145843; e-mails: {prbp, antonio.grilo, mario.nunes, augusto.casaca} @inesc.pt).

J. Gonçalves is with Inesc INOV, Rua Alves Redol, nº 9, 1000-029 Lisboa, Portugal. (e-mail: jose.goncalves@inov.pt).

C. Fortunato is with EDP Distribuição, R. Camilo Castelo Branco, 43; 1050-044 Lisboa; Portugal (e-mail: carlos.fortunato@edp.pt).

Additional relay nodes may be installed in street lights. In this case, power is available only during the night when the light is on, so the node has to be battery operated during the day.

For these reasons, it is very important to have an energy and QoS manager that balances the QoS provisioning and energy consumption in the nodes.

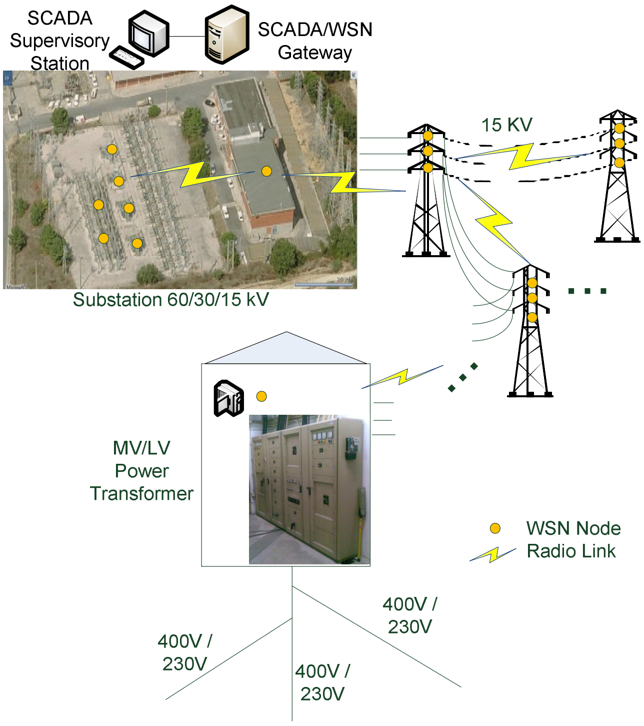

Fig. 1. Network Architecture.

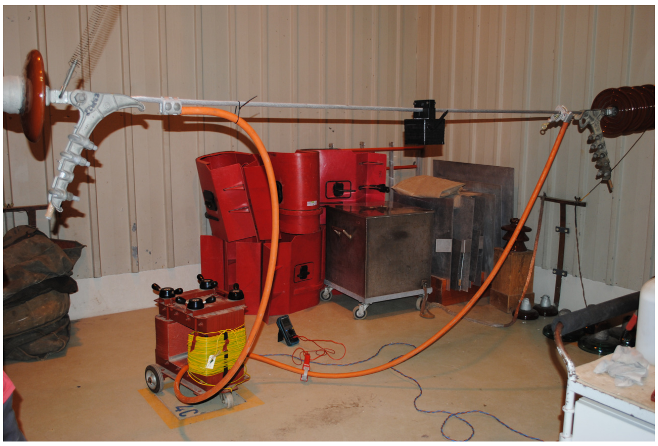

Fig. 2. Laboratory testing of a wireless sensor node (hanging on the line).

The next section presents an overview of related work. Section 3 presents an energy analysis of the sensor nodes used, based on real measurement with our setup. Section 4 presents an analysis of the video streaming operation and the energy and QoS management system. Section 5 presents conclusions and further work topics.

II. Related Work

Power conservation mechanisms can be classified into two main categories [4]: Active and Passive (Fig. 3). *Active* refers to mechanisms that achieve energy conservation by smartly utilizing energy-efficient protocols (by not turning off the radio interface), while *Passive* refers to mechanisms that save a node's power by turning off the radio (transceiver) interface module when there is no communication activity.

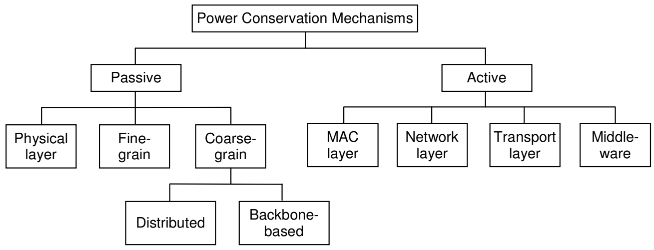

Fig. 3. Classification of power conservation mechanisms.

Periodic hibernation is a synonym to Passive power conservation that is frequently found in the literature. The idea of turning off the radio transceiver was firstly introduced in IEEE 802.11 [5]. According to this standard, a sensor node may switch to sleep mode in accordance with the Network Allocation Vector (NAV). Also, every sensor node in the network must wake up during the Announcement Traffic Indication Message (ATIM) period, during which transmitters inform their destinations not to turn to power save mode. If no notification is received, the sensor node can turn to power save mode and wake up in the next ATIM period. IEEE 802.11e [6] defines an enhancement of the IEEE 802.11 power save mode, automatic power save delivery (APSD), which takes advantage of some of its QoS mechanisms and reduces the required signaling load.

Physical layer power conservation techniques can produce substantial energy savings by reducing to the minimum the energy of the Central Processing Unit (CPU) in idle system states. Dynamic Voltage Scaling (DVS) and Dynamic Power Management are typical examples of this passive power conservation mechanism.

According to Fine-grain power conservation mechanisms, the Medium Access Control (MAC) layer is responsible for deciding if there is a frame transmission that is destined to it, and then turn off the radio for that transmission frame to save power if there is no frame transmission.

Coarse-grain power conservation mechanisms use higher layer application information to decide when to turn off the radio interface module. There are two implementation approaches: the distributed approach, and the backbone-based approach. In the distributed approach, sensor nodes independently schedule their sleeping intervals based on both internal information and/or neighbour information. The coordination of sleeping schedules is done implicitly through exchanged beacon or hello messages. In the backbone-based approach, a backbone infrastructure is required to be set up and the power conservation application resides on the backbone sensor nodes. The power conservation application can have a better view of their local network environment.

As for the active power conservation mechanisms, at the MAC layer, one approach is to adaptively adjust the transmission power to an appropriate level just enough to reach the next hop. An additional objective is to reduce the collision probability by reducing the radio range.

At the network layer, two basic active power saving schemes are used. Power-aware routing has the objective of finding routes that consume the least possible power. Maximum lifetime routing, in contrast, balances energy dissipation among sensor nodes to prolong the operational lifetime of the network.

At the transport layer, protocols can alter the packet retransmission behaviour by reducing unnecessary retransmissions to a minimum and thus achieving lower power consumption. An example is the Distributed Transport for Sensor Networks (DTSN) protocol that uses a cache in intermediate nodes and a configurable reliability level to save end-to-end packet retransmissions [7].

At the middleware level, protocols can perform data aggregation [8], reducing the total number of transmitted messages and thus saving energy. Messages can aggregate data from different source nodes or from different samples from the same node. Another possible action is changing the encryption algorithm in use to another that requires less energy, or even disabling encryption. [9] presents performance and energy use of different encryption algorithms.

Naturally, power conservation mechanisms can be combined. For instance, LEMMA (Latency-Energy Minimization Medium Access) [10] is a new MAC protocol that combines active and passive techniques. LEMMA uses carrier sense mechanisms to evaluate the real interference level between nodes and uses this information to schedule active and sleeping slots so as to minimize collisions between the different nodes' transmissions.

Another interesting research topic is combining energy conservation mechanisms with quality of service (QoS) constraints. [11] uses a DVS mechanism to reduce energy consumption while guaranteeing soft real-time constraints in job execution. LEMMA [10] is an example of MAC protocol that minimizes latency and energy use. [12] presents a routing protocol that meets QoS objectives of throughput and latency as well as energy conservation to maximize network lifetime. [13] and [14] are other examples of routing protocols for wireless sensor networks combining energy and QoS requirements.

In this paper, we present an intelligent energy and QoS management system that provides an energy-aware cross-layer solution for our scenario. The intelligent agent monitors the energy level of the nodes and dynamically adjusts several operating parameters of the nodes accordingly.

III. Energy Analysis of the SX-560

The video surveillance service is incompatible with a low rate radio technology such as the IEEE 802.15.4 [15]. It demands the use of a broadband wireless technology such as the IEEE 802.11 [5]. The latter also makes sense given the expected maximum distance between sensors in our scenario, which may even require high gain directional antennas when the distance between towers is large. The ensemble of requirements led to the choice of the Silex SX-560 Intelligent Programmable WLAN Module [16] as the platform for the WMSN nodes.

The SX-560 includes an IEEE 802.11a/b/g wireless card (Atheros Communications AR6001XL chipset connected as a SDIO daughtercard), a 200 MHz ARM926EJ-S CPU, 16 MB of RAM, 8 MB of flash, USB 1.1 host, GPIO and SPI interfaces, among other characteristics. It runs Linux version 2.6.29.

As this Linux version does not support dynamically adjusting the CPU frequency, we backported the CPU frequency driver from version 2.6.32, so that we are able to dynamically control the CPU frequency. With this driver, it is possible to change the CPU frequency to 50 MHz, 100 MHz and 200 MHz.

A. *No Communication*

We measured the power consumption of the SX-560 module without any active communication, in different configurations. The results are shown in Table 1. The results include the consumption of the printed circuit board (PCB) that was developed to manage the battery, provide energy to the module from the different possible sources and offer interfaces to the sensors and web camera. The web camera was not included in the measurements, as in our scenario the nodes with web camera have AC power supply available. The developed PCB is much smaller than the Silex Evaluation Daugthercard and includes a switched power supply for maximum efficiency.

CPU use	CPU @50MHz	CPU @200MHz
0%	846 mW	888 mW
100%	874 mW	1011 mW

Table 1. Power consumption with different CPU uses and frequencies.

If the Linux system is halted, we get a consumption of 270 mW. However, as there is no real-time clock in the SX-560 module, it is not possible to resume operation after a halt, except by pressing the reset button and, consequently, this mode cannot be used to save energy.

Changing from an idle CPU to a CPU use of 100% increases the power consumption by about 14%. Decreasing the CPU operating frequency from 200 MHz to 50 MHz decreases the power consumption for about 5% at the cost of a reduction of processing power.

B. *Short Range Communication without Encryption*

To evaluate the energy consumption and the wireless communications performance, we established an ad hoc connection, at 1 meter distance, between an SX-560 and an Asus Eee PC 1001P laptop with an Intel Atom N450

1.666 GHz CPU running the Ubuntu Netbook Edition 10.04 operating system. The 802.11g protocol was used as communication medium and the frequency channel 10 (2.457 GHz) was selected. The criterion for channel selection was to use one that was not used in the vicinity of the test location.

The network performance was evaluated with the 'iperf' [17] and the Linux 'ping' software tools. To measure communication *Latency* we used 'ping'. To measure *Throughput* we used 'iperf' with TCP data streams. To measure *Jitter* and *Packet Error Rate* (*PER)* we also used 'iperf', but with UDP data streams.

Table 2 shows the results of the TCP tests, made with an 85.3 KB TCP window size, in a 20 seconds period. We tested both communication directions separately. To test the sensor module reception we called the 'iperf' utility in server mode at the module and in client mode at the test PC. To test the sensor module transmission we reversed the roles.

Table 3 shows the results of the UDP tests, made with a 32KB buffer size and 1470 bytes datagrams, in a 20 seconds period.

Table 4 shows the results of the ICMP (ping) tests, made with 56 bytes data packets, in a 60 seconds period, with a 1 second interval between each transmission. The 'ping' test was also made by issuing the command alternatively in the sensor module (Tx) and in the test PC (Rx).

Direction (Sensor)	Throughput [Mbit/s]	Power [mW]	Throughput [Mbit/s]	Power [mW]
	50 MHz		200 MHz	
Rx	5.06	1056	7.25	1248
Tx	1.43	978	4.76	1260

Table 2. Iperf TCP test results.

Direc-tion (Sensor)	Band-width [kbit/s]	PER [errors /total]	Jitter [ms]	power [mW]	PER [errors /total]	Jitter [ms]	power [mW]
		50 MHz			200 MHz		
Rx	1000	0/1702	5.76	912	0/1702	5.70	960
Tx	1000	0/1702	1.75	966	0/1702	3.37	1014
Rx	56	0/97	5.00	870	0/97	5.75	912
Tx	56	0/97	1.62	942	0/97	2.63	942

Table 3. Iperf UDP test results.

Direction (Sensor)	RTT min [ms]	RTT avg [ms]	RTT max [ms]	RTT min [ms]	RTT avg [ms]	RTT max [ms]
	50 MHz			200 MHz		
Rx	6.65	13.62	60.29	5.02	10.76	17.03
Tx	8.50	14.64	18.42	5.88	9.72	36.11

Table 4. Ping (ICMP) test results.

Reducing the CPU frequency causes a significant degradation of the maximum achievable throughput, particularly significant when the sensor is transmitting data (reduction to 30%). It also causes degradation in the RTT and jitter.

Receiving a data flow at 56 Kbit/s increases the power consumption for about 3% as compared with the idle case. Transmitting a data flow at 56 Kbit/s increases the power consumption for about 6% as compared with the idle case. Naturally, using higher data rates results in higher power consumption. The ping (ICMP) tests have such a low data rate (1 packet per second), that the difference in power consumption is not measurable and thus is not shown in the previous tables. The scenarios foreseen to monitor the electrical energy distribution infrastructure produce a data rate below 100 bit/s, except for video streaming scenarios. At such low data rates, energy savings from sensor data aggregation mechanisms are negligible for this particular sensor.

C. *Data Link Layer Encryption*

Tests were performed to measure power consumption for short range communication with data link layer encryption. The data link layer encryption method selected was WEP-128bit. The results are shown in Table 5-Table 7.

The data link encryption is performed in the hardware. So, it has minimal influence in the power consumption (between -0.6% and 4%), causing a small degradation in the throughput (between 0 and 9%), a small degradation in the RTT (between 5 and 24%) and a minimal influence in the jitter.

Direction (Sensor)	Throughput [Mbit/s]	Power [mW]	Throughput [Mbit/s]	Power [mW]
	50 MHz		200 MHz	
Rx	5.01	1068	6.57	1302
Tx	1.43	996	4.75	1308

Table 5. Iperf TCP test results with data link encryption.

Direc-tion (Sensor)	Band-width [kbit/s]	PER [errors /total]	Jitter [ms]	power [mW]	PER [errors /total]	Jitter [ms]	power [mW]
		50 MHz			200 MHz		
Rx	1000	0/1702	5.68	924	0/1702	5.68	966
Tx	1000	0/1702	1.02	954	0/1702	0.92	1032
Rx	56	0/97	4.62	876	0/97	4.66	918
Tx	56	0/97	1.66	888	0/97	1.21	936

Table 6. Iperf UDP test results with data link encryption.

Direction (Sensor)	RTT min [ms]	RTT avg [ms]	RTT max [ms]	RTT min [ms]	RTT avg [ms]	RTT max [ms]
	50 MHz			200 MHz		
Rx	5.82	16.87	144.44	4.85	11.32	31.05
Tx	7.27	17.45	156.80	5.94	11.36	50.73

Table 7. Ping (ICMP) test results with data link encryption.

D. *Transport Layer Encryption*

Tests were performed to measure power consumption for short range communication with transport layer encryption.

The Stunnel/OpenSSL [18] encryption software was used. As this software only supports TCP encryption, only these results are presented in Table 8.

Direction (Sensor)	Encryption Method	Through-put [Mbit/s]	Power [mW]	Through-put [Mbit/s]	Power [mW]
		50 MHz		200 MHz	
Rx	AES 256	1.26	1038	3.75	1212
Tx	AES 256	0.75	996	2.69	1242
Rx	AES 128	1.41	1002	4.18	1206
Tx	AES 128	0.81	1032	2.91	1230
Rx	DES 56	1.38	1002	4.11	1182
Tx	DES 56	0.77	1002	2.78	1218

Table 8. Iperf TCP test results with transport layer encryption.

The use of transport layer encryption lowers the throughput by about 40%, resulting in a small power saving, as part of what is saved in the radio due to the reduced throughput is spent in CPU processing in the encryption process. The stronger encryption method (AES-256) has the smaller performance (throughput), as expected.

E. Medium Range Communication

Tests were performed to measure power consumption for medium range communication without encryption. The node under test was separated 7 meters from the test PC, with several metallic obstacles (PCs/Monitors) in between. The results are shown in Table 9-Table 11.

Direction (Sensor)	Throughput [Mbit/s]	Power [mW]	Throughput [Mbit/s]	Power [mW]
	50 MHz		200 MHz	
Rx	2.69	1218	3.18	1326
Tx	1.43	1248	3.75	1620

Table 9. Iperf TCP test results for medium range communication.

Direc-tion (Sensor)	Band-width [Kbit/s]	PER [errors /total]	Jitter [ms]	power [mW]	PER [errors /total]	Jitter [ms]	power [mW]
		50 MHz			200 MHz		
Rx	1000	0/1702	5.72	930	0/1702	5.45	966
Tx	1000	0/1702	1.69	1086	0/1702	2.20	1188
Rx	56	1/97	6.09	900	4/97	4.82	918
Tx	56	0/97	5.69	1050	0/97	4.47	1110

Table 10. Iperf UDP test results for medium range communication.

Direction (Sensor)	RTT min [ms]	RTT avg [ms]	RTT max [ms]	RTT min [ms]	RTT avg [ms]	RTT avg [ms]
	50 MHz			200 MHz		
Rx	5.86	16.28	79.58	4.96	12.44	64.02
Tx	7.29	16.24	51.77	5.90	13.86	40.70

Table 11. Ping (ICMP) test results for medium range communication.

When obstacles exist in the transmission path, there is a significant degradation of the throughput, with a similar or higher power consumption, that can reach a 30% increase. In this situation, the data link retransmissions are significant, due to the transmission errors, resulting in an increased radio energy usage. Most of the transmission errors are solved in the data link layer, so the application gets a very low error rate.

F. Summary

Although the radio chipset of the SX-560 supports IEEE 802.11e QoS and power saving features, its energy consumption is significant (56 mW in sleep, 462 mW in idle, 1152 mW maximum, according to the manufacturer), as compared with radio chipsets that only support IEEE 802.15.4.

The use of a 200 MHz CPU also results in a significant power consumption (124 mW in idle, 259 mW in normal mode, according to the maker).

However, the combination of a faster data rate in the radio link and a more powerful CPU enable the use of multimedia sensors and the Linux operating system enables the use of advanced multimedia applications and IP protocols.

IV. Energy and QoS Management

A. Possible Management Actions and Analysis

We intended to develop an intelligent energy and QoS management middleware that based on the current state of the network would optimize the energy use and the QoS provided to the applications. Naturally, such a distributed QoS management implies additional energy and traffic overhead. Consequently, there is a need to investigate the possibilities, examine their costs and then, specify their application areas.

Table 12 shows some identified parameters that may be modified by the actions of the intelligent QoS middleware agent and their range of influence. They are discussed in the following paragraphs.

Parameter	Range of influence
Transmission power level	node and neighbour
CPU frequency	single node
Routing path	end-to-end path
Packet retransmission behaviour	end-to-end path
Data encryption algorithm	end-to-end path
Sensor sample rate	end-to-end path
Number of sensor samples aggregated in one message	end-to-end path
Video stream quality	end-to-end path

Table 12. Possible actuator parameters.

A first possible action is reducing the transmission power level to save energy as long as there is no packet loss and the received signal quality is sufficient. This action combines MAC and transport layer information. Reducing the transmission power, involves a high risk that transmission errors are produced, contradicting the purpose of saving energy. Since in our scenario, for most links where energy is scarce, the distances between towers are quite large, which may even imply the use of high-gain directional antennas, we decided not to manage transmission power. For this reason, we did not do an analysis of the power savings achievable with this action.

Every node can control its own CPU frequency independently of other nodes, thus without generating additional traffic. The implemented algorithm is very simple. When a node is involved in video streaming, the CPU frequency should be the highest (200 MHz) to minimize processing delays and improve throughput. When a node is not involved in a video streaming monitoring scenario and is operating from its battery, the CPU frequency should be the lowest (50 MHz) to save energy.

As the network is tree-shaped, there is a single path from the sensor nodes to the sink node. In our trial deployment, we will only place sensors in a single phase of the power lines, so routing cannot help much in power saving. As future work, it is expected that nodes will be placed in all three phases, so a maximum lifetime routing protocol can be implemented to select the nodes with more energy available to route the packets to the sink node.

We will use the DTSN transport protocol to optimize the packet retransmission behavior and improve energy efficiency. Packet caches at intermediate nodes and a budget of time for the transmissions can manage the balance between media quality and timely delivery of data, thus saving energy in a multimedia transmission [19].

As for the data encryption, we decided to use data link encryption, which results in very little energy consumption or QoS degradation. For sensitive control data, which results in a limited data rate, we will use additional security techniques at the routing and transport level [20].

As the scalar sensor data to be retrieved is of a very limited data rate (below 100 bit/s), and the corresponding energy requirements are too small to be measurable, we judged it was not worth implementing adaptive sample rates or aggregation of sampled data from different nodes. Only the different data from a single node (measured current in the high voltage line, temperature and battery status) are aggregated in a single packet.

The biggest energy consumption happens when video streaming is active, since both the CPU and radio are heavily used, thus requiring specific actions to manage its energy and QoS.

B. *Video Streaming Management*

The sensor node to be deployed in MV/LV power transformer locations is shown in Fig. 4. The Webcam is a Logitech Quickcam Sphere AF that supports pan and tilt. A passive infrared motion detector sensor detects intrusions in the premises and triggers an alarm to start the video. LED lighting may be used during the night. An additional infra-red camera provides a thermal image of the power transformer to detect hotspots and prevent problems. An external outside antenna is used to improve radio link quality. Permanent 230V power is available at this kind of location, so energy is not a limitation for this particular node.

Fig. 4. Sensor node with web camera, light, motion detector and infra-red camera.

We tested the streaming of video using the `ffmpeg` software [21] to capture, convert and stream the video. The VideoLAN Client (VLC) [22] media player software was used to receive the stream and show the video. As the camera does not support frame rates below 5 frames per second, the `ffmpeg` software had to be modified to allow dropping some frames to enable video modes with lower data transmission rates.

Table 13 shows a set of available video modes that have been tested with the web camera and the SX-560 sensor node. For each mode, the table lists the codec that is used to encode the image, the transport protocol used to transmit the data through the network, the image size in pixels, the frame rate in frames per second, the approximate resulting data transmission rate in Kbit/s and the CPU power used in the node in percentage of the available CPU time.

The Motion Joint Photographic Experts Group (M-JPEG) codec is built into the camera hardware, resulting in less CPU time consumption in the node. However, the M-JPEG codec compresses each frame individually as a JPEG image, without interframe compression, resulting in less efficient video compression ratios and a resulting higher data transmission rate.

The other codecs tested achieve a much higher compression ratio, resulting in lower bit rates. However, the larger image sizes and frame rates cannot be used with these codecs, as the 12 Mbit/s USB 1.1 rate limit for the connection between the web camera and the SX-560 node does not allow capturing raw video at a resolution higher than QCIF (Quarter Common

International Format) and the SX-560 CPU is not powerful enough to compress the video stream in real time at more than 10 frames per second.

The video stream management is implemented near the SCADA supervisory station. The user can select a low resolution with higher frame rate or a higher resolution with lower frame rate video mode. From the available video modes, we selected three video modes for low resolution use and three for high resolution use. The corresponding codec, image size and frame rates are listed in Table 14.

CODEC	*Transport Protocol*	*Image Size*	*Frame Rate [fps]*	*Bit Rate [Kbit/s]*	*CPU used [%]*
MJPEG	UDP	SVGA (800 x 600)	5	1455	15
			2.5	715	7
			1	288	3
		VGA (640 x 480)	10	1964	18
			5	975	9
			2.5	495	5
			1	200	2
		CIF (352 x 288)	15	1925	21
			10	1281	14
			5	360	4
			1	75	1
		QCIF (176 x 144)	25	1645	13
			20	1320	11
			15	930	8
			10	623	5
MPEG4 - Part 2	RTP	QCIF (176 x 144)	10	95	68
			5	65	34
H.263+ (H.263-1998)		QCIF (176 x 144)	10	115	68
			5	59	31
H.263-1996		QCIF (176 x 144)	10	120	61
			5	57	34

Table 13. Available video modes.

Video mode	*Resolution*	*video format*
1	Low	MJPEG:QCIF:10
	High	MJPEG:VGA:5
2	Low	MPEG4:QCIF:10
	High	MJPEG:VGA:1
3	Low	MPEG4:QCIF:5
	High	MJPEG:CIF:1

Table 14. Video formats to use (codec, image size, frame rate).

The rules for selecting the video mode to be used are as follows. The battery level of all nodes involved in the transmission of the video stream is sampled every minute. The minimum of the sampled battery level values of the different nodes is used to decide the appropriate video mode to use according to Table 15. The video mode is changed at most once in a sampling interval. If the video is turned off due to low battery, a warning is generated to the user, as listed in Table 15.

Minimum Battery Level	*Video mode*	*Warn user of change*
≥ 95%	1	No
≥ 50% & < 95%	2	No
≥ 25% & < 50%	3	No
< 25%	turn off video	Yes

Table 15. Video mode to select according to the energy available.

V. Conclusion

The use of a multimedia sensor node such as the SX-560 offers the broadband wireless communication of IEEE 802.11 and processing power enough for video streaming. The use of the Linux operating system and IP protocols simplifies the development of an integrated sensor solution.

However, the energy spent by a multimedia sensor node is much higher than for other simpler scalar sensor nodes. In addition, the USB 1.1 connection between the web camera and the sensor node limits the maximum resolution of uncompressed video that can be captured and the CPU has not enough processing power to compress more than 10 frames per second.

We presented a laboratory energy and performance analysis of the SX-560 sensor node. Based on this analysis, several energy management actions were identified and studied. An intelligent energy and QoS management agent was proposed and described. This agent dynamically controls the CPU frequency of the nodes and the video mode to be used, according to the current energy resources of the nodes. Reducing the CPU frequency results in 5% energy savings when not streaming video. Dynamically controlling the video mode saves energy and prevents battery exhaustion. The use of data link layer encryption does not impact energy, so it is being used permanently. On the other hand, transport layer encryption has a significant impact on performance and energy, so it is not being used for video streaming.

The system has been successfully tested in laboratory and will be deployed in Autumn 2011.

Some future work possibilities are using a maximum lifetime routing protocol if redundant paths are added to the sensor network, the dynamic selection of encryption protocols according to the available energy, the integration with the DTSN transport protocol, and the use of more powerful sensor nodes, such as the BeagleBoard [23], particularly for nodes that generate multimedia streams, so that video streams can be compressed in real-time at a higher frame rate.

REFERENCES

[1] Ian F. Akyildiz, Tommaso Melodia, Kaushik R. Chowdhury: "A survey on wireless multimedia sensor networks". Computer Networks, **51**(4):921-960, March 2007.

[2] A. Grilo, A. Casaca, M. Nunes, C. Fortunato, "Wireless Sensor Networks for the Protection of an Electrical Energy Distribution Infrastructure", IFIP WCC 2010 Critical Infrastructure Protection Conference, Australia, Sep. 2010.

[3] IEEE Std C57.13-1993, "IEEE Standard Requirements for Instrument Transformers", IEEE Power Engineering Society, June 1993.

[4] N. Pantazis, D. Vergados, "A Survey on Power Control Issues in Wireless Sensor Networks", IEEE Communications Surveys & Tutorials, **9**(4):86-107, 4th Quarter 2007.

[5] IEEE Std. 802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", 1999.

[6] X. Perez-Costa, D. Camps-Mur, "IEEE 802.11e QoS and Power Saving Features Overview and Analysis of Combined Performance", *IEEE Wireless Communications*, **17**(4):88-96, August 2010.

[7] Duarte Meneses, António Grilo, Paulo Rogério Pereira, "A Transport Protocol for Real-time Streaming in Wireless Multimedia Sensor Networks", 7th Euro-NF Conference on Next Generation Networks (NGI 2011), Kaiserslautern, Germany, 27-29 June 2011.

[8] Ramesh Rajagopalan, Pramod Varshney, "Data-Aggregation Techniques in Sensor Networks: A Survey", IEEE Communications Surveys & Tutorials, **8**(4):48-63, 4th Quarter 2006.

[9] D. Elminaam, H. Kader and M. Hadhoud, "Performance Evaluation of Symmetric Encryption Algorithms on Power Consumption for Wireless Devices", International Journal of Computer Theory and Engineering, **1**(4):343-351, October, 2009.

[10] Mário Macedo, António Grilo, Mário Nunes, "Distributed Latency-Energy Minimization and interference avoidance in TDMA Wireless Sensor Networks", Computer Networks, **53**(2009):569-582, April 2009.

[11] Linwei Niu, Gang Quan, "Energy-Aware Scheduling for Practical Mode Real-Time Systems with QoS Guarantee", 2009 WRI World Congress on Computer Science and Information Engineering, Volume 3, March 31 2009-April 2 2009, pp.428-432.

[12] Shanghong Peng, S. Yang, S. Gregori, Fengchun Tian, "An adaptive QoS and energy-aware routing algorithm for wireless sensor networks". International Conference on Information and Automation, ICIA 2008, 20-23 June 2008, pp.578-583.

[13] A. Pourkabirian, A. Haghighat, "Energy-aware, delay-constrained routing in wireless sensor networks through genetic algorithm", 15th International Conference on Software, Telecommunications and Computer Networks, SoftCOM 2007. 27-29 September 2007, pp.1-5.

[14] Yao Lan, Wen Wenjing, Gao Fuxiang, "A real-time and energy aware QoS routing protocol for Multimedia Wireless Sensor Networks", 7th World Congress on Intelligent Control and Automation, WCICA 2008. 25-27 June 2008, pp.3321–3326.

[15] IEEE Std. 802.15.4, "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)", 2003.

[16] Silex Technology, "SX-560 Intelligent Programmable WLAN Module", [online] http://www.silexeurope.com/en/home/products/wireless-modules/sx-560.html

[17] Iperf [online] http://iperf.sourceforge.net/

[18] Stunnel [online] http://stunnel.mirt.net/

[19] Jose Mingorance-Puga, Gabriel Maciá-Fernández, António Grilo, Nestor Tiglao. "Efficient Multimedia Transmission in Wireless Sensor Networks", 6th EuroNGI Conference on Next Generation Networks, NGI'2010, Paris, France, 2-4 June 2010.

[20] Levente Buttyán, António Grilo, "A Secure Distributed Transport Protocol for Wireless Sensor Networks", IEEE ICC'2011, Kyoto, Japan, June 2011.

[21] FFmpeg [online] http://ffmpeg.org/

[22] VideoLAN Client (VLC) [online] http://www.videolan.org/

[23] BeagleBoard [online] http://beagleboard.org/