



HAL
open science

Wireless sensor networks for active vibration control in automobile structures

F. Mieleville, Mohamed Ichchou, Gérard Scorletti, D. Navarro, Wan Du

► **To cite this version:**

F. Mieleville, Mohamed Ichchou, Gérard Scorletti, D. Navarro, Wan Du. Wireless sensor networks for active vibration control in automobile structures. *Smart Materials and Structures*, 2012, 21 (7), pp.Article number075009. 10.1088/0964-1726/21/7/075009 . hal-00793946

HAL Id: hal-00793946

<https://hal.science/hal-00793946>

Submitted on 2 May 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Wireless Sensor Networks for active vibration control in automobile structures.

Fabien Mieyeville¹†, Mohamed Ichchou², Gérard Scorletti³, David Navarro¹ and Wan Du¹

¹Ecole Centrale Lyon, INL UMR CNRS 5270, ECULLY, FRANCE

²Ecole Centrale Lyon, LTDS UMR CNRS 5513, ECULLY, FRANCE

³Ecole Centrale Lyon, AMPERE INL UMR CNRS 5005, ECULLY, FRANCE

† Corresponding author: fabien.mieyeville@ec-lyon.fr

Abstract. Wireless Sensor Network (WSN) are nowadays widely used in monitoring and tracking applications. This paper presents the feasibility of using Wireless Sensor Networks in active vibration control strategy. The active control method used is an active-structural acoustic control using piezoelectric sensors distributed on the car structure. This system aims at being merged in wireless sensor network whose head node collects data and process control law so as to command piezoelectric actuators wisely placed on the structure. We will study the feasibility of implementing WSN in active vibration control and introduce a complete design methodology to optimize hardware/software and control law synergy in mechatronic systems . A design space exploration will be conducted so as to identify the best Wireless Sensor Network platform and the resulting impact on control.

Keywords: Wireless Sensor Network, Active control, mechatronic, Design methodology

1. Introduction

Wireless monitoring has imposed during the last year as a promising technology due to its wireless connectivity removing many hardware constraints and paving the way for new implementation on mechanical elements. Furthermore collocating computational power with sensors opens new perspectives in mechanical domain where a centralized approach is often preferred.

Use of Wireless Sensor Networks (WSNs) in mechanical domain has already successfully been demonstrated, particularly in Structural Health Monitoring (SHM) [1] where realizations are numerous and distributed computation possibilities offered by WSN are beginning to be explored [2]. If numerous works put the emphasis on demonstrating the aptitude of WSN for active control with sometimes comparison between hardware platform [1] or deployment of decentralized algorithms, very few works put the focus on hardware platform analysis [3] or on design space exploration.

This work is embedded in Mécatronique@Lyon (M@L)‡ project focusing on intelligent structures and systems in mechatronic domain from the component to the integrated design methodologies. The main objectives of this project are the identification and the integration of new intelligent active technologies in automotive systems so as to improve internal comfort (noise and vibrations). For this article, we will focus on the top-down design approach of Wireless Sensor Networks for active vibration control that takes into account the hardware platform specificities at high level of abstraction. When most current works are based on physical deployment of one (sometimes more) hardware platform, we will present and demonstrate a design flow that enables, by simulation, the exploration of different WSN node's architecture choice for a mechatronic application. From this design space exploration, we are able to identify the best hardware and to precisely evaluate the pro and cons of every architecture so as to elaborate the control law in adequation.

‡ M@L is supported by Ingénierie@Lyon (I@L), an institute of Carnot Network.

Mechatronic is the domain where several separate domains are strongly implicated [4]. If technological solutions merging those domains are commonly developed, only synergy between those domains can enable significant technological breakthroughs. Design methodologies are the keystone of this synergy: every specificity of those separate domains must be considered at conjointly at high level of abstraction so as to perform a global optimization of the system and the optimal distribution of the specifications. This paper offers a new technological solution by removing the conventional wired sensors used in network for active control and replacing them by wireless connected smart sensors such as shown on figure 1. Mechatronics being a tight integration of mechanical, electronic and information-driven units a specific design approach will be adopted to integrate the choice and the evaluation of the deployed WSN in the core of the development of the control law.

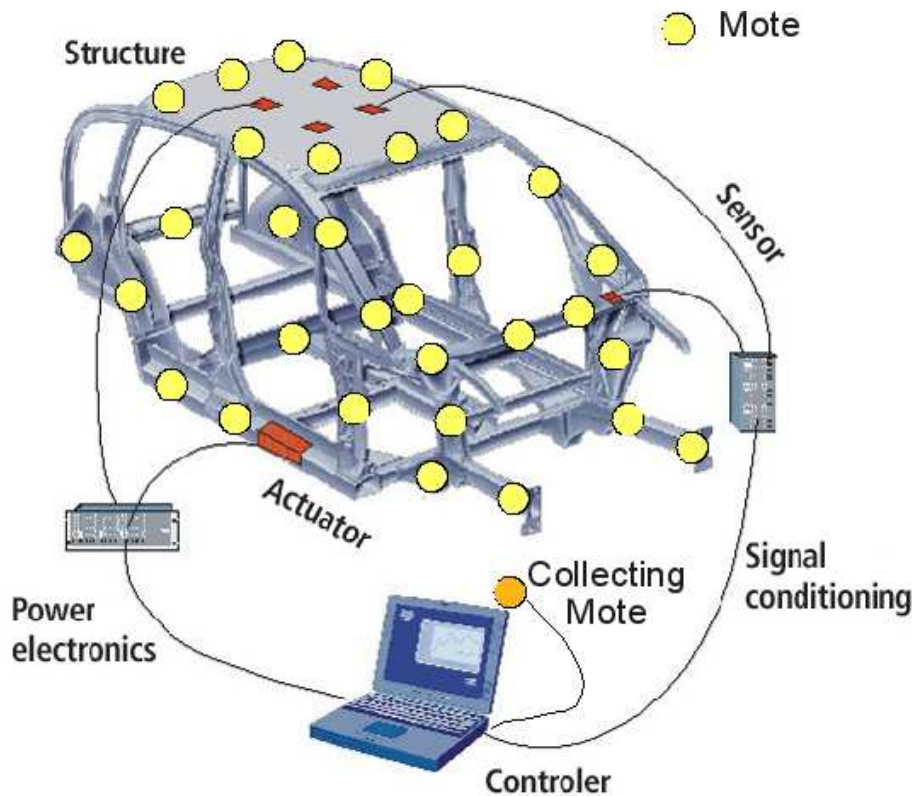


Figure 1: M@L wireless sensor network infrastructure.

WSN are now widely used [5] in industrial application areas such as traffic control, stock management, industrial process monitoring and control, environment and habitat surveillance (water pollution, ...), structure monitoring [6] as well as in civilian area where they are the keystone of ambient intelligence and home automation [7]. Commonly deployed WSN are used for application requiring low data rate exchange (sampling rate are often below the hertz). In this paper, we use WSN for active control where the data flow rate generated by each node of the network exceeds the kilohertz. We will establish the feasibility of deploying WSN for high data rate automotive application.

If feasibility is validated, new approaches could be used for active control. Indeed, beyond the removing of the wires of the network, WSNs, by providing distributed smart sensors embedding local computation power, could pave the way for new distributed approaches in the active control of vibrations. The feasibility of using WSN in active control is a two-steps approach: first we establish the specification of the mechatronic system to design a preliminary Wireless Sensor Network and then we used a WSN framework named IDEA1 [8] to explore the design space and identify the bottlenecks in the use of WSN for active control. From resulting conclusions and synthesis figures extracted from the analysis, we then propose guidelines for control laws § that suit the best to the WSN based active control network.

2. Wireless Sensor Networks

Wireless Sensor Networks are highly distributed self-organized systems [9]. A Wireless Sensor Network is made of a large number of scattered tiny low-cost devices featuring strong constraints in terms of processing, memory, communications and energy capabilities. Common applications of WSNs deployed on a given space are data collection from sensor nodes measurements that are transmitted (or not depending on the local decision process implemented in each node) to a specific node called the sink node. The sink node can be as simple as a node whose connectivity has been extended to enable the transmission of data to external or networked systems, or as complex as a PC motherboard able to achieve elaborated computations. The longevity of the deployed network (often inaccessible or barely accessible) being a strong constraint, power saving techniques are commonly implemented resulting in specific behaviour for every nodes. Indeed, nodes sleep most of the time, waking up only for RF transmission or data measurement and rely on low power communications mainly based on a multihop data transmission from sensor nodes to sink nodes and vice versa.

Typical deployment of wireless sensor network can be seen figure 2. In this work, we use a heterogeneous network composed of off-the-shelf nodes and home-made nodes based on Zigbee communication protocol. The following subsections will rapidly describe hardware architecture of the node and radio-frequency communication in WSNs so as to introduce design parameters before developing feasibility of WSN in active control.

2.1. Hardware architecture

The elementary part of a wireless sensor network is called a mote or a node and consists of one or more sensors associated with a small microcontroller, a radio transceiver and an energy source, usually a battery. A sensor node might vary in size from that of a card box down to one-cubic-millimeter sized box [10] although functioning “motes” of

§ Since this work does not focus on specific active control algorithms, generic guidelines will be given for active control laws in general : the offered strategy can be adapted for various kind of control algorithms.

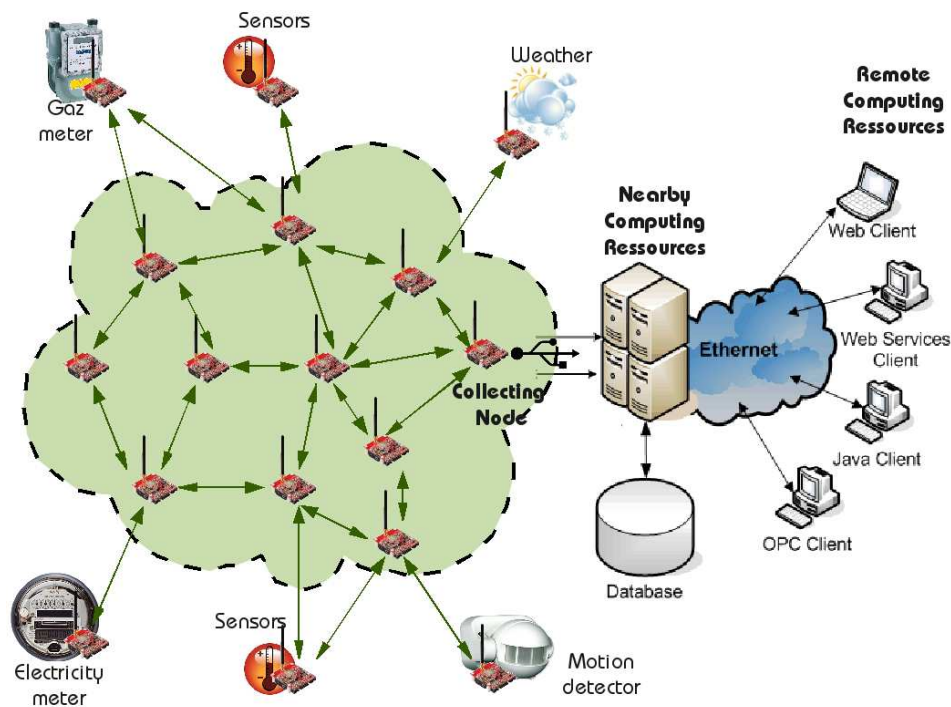


Figure 2: Classical deployment of wireless sensor network.

genuine microscopic dimensions need still some development. Sensor nodes aim at being deployed by thousands: then the cost of a node is generally quite reduced (tenths of dollars) yet expanding to hundreds of dollars for high performances specific nodes.

Constraints on nodes differ from commonly considered electronic system ones (even embedded domain) in two ways. First, the very limited amount of embedded energy coupled with the difficulty of accessing to nodes when deployed results in constrained resources in memory, computational speed and bandwidth [11]. Secondly, due to their structure, WSNs belong to distributed systems in which the main actor is not the element itself but the network that performs a task (data collecting and processing) in collaboration (data-centric approach).

In this work we used both commercial product from Crossbow and internally developed node that will be described later.

2.2. Software in WSN

In wireless sensor networks, software is at stake at two hierarchical levels: on the node itself and on the global network for the management of every node so as to achieve the targeted application. Software embedded in node results in operating system development whereas software in network is more related to middleware concerns [12]. An exhaustive and deep analysis of those aspects can be found in the following works [9, 13].

For our application, where local computation is limited (embedded software is

reduced to the following loop: data acquisition - sending data - sleep), the need for operating system is limited. But shall we implement local algorithms for active control, we should then use operating systems. Among several existing operating systems for WSN, TinyOS [14] is the most used and offers support for most existing commercial off-the-shelves WSN nodes. Because of the high energy cost of dynamic reconfiguration in TinyOS, our final solution will be funkOS [17] that enables modular description, is supported by many microcontroller architectures and reduces energy costs in dynamic reconfiguration. Furthermore, in the first stages of our approach where few nodes are used to demonstrate feasibility of WSN in active control, middleware is not needed. In further steps, the link will be created between the framework and the deployment of the application on hardware and the maintenance of this network: dynamic reconfiguration [18], extension of the network,

2.3. Zigbee, a Radio-frequency solution for mechatronic

Radio-frequency (RF) communications are the backbone of the wireless sensor network. Due to the limited available energy, commonly used RF solutions cannot be used for WSN. Many works have focused on developing specific protocols and medium access control (MAC) layers [9]. A low-cost, low-power, wireless mesh networking proprietary standard exists today for WSN based on the IEEE 802.15.4 standard: Zigbee [5, 9]. Among the wireless solutions (Bluetooth, Ultra Wide Band and Wifi) that can be used for automotive, Zigbee is considered as a good solution [19] despite its lack of real-time capacity. We adopted it because of his low power consumption that could enable us to deploy an autonomous vibration powered wireless sensor network in a long term approach . Many RF transceivers implement this standard in their hardware (no need to develop code on the microcontroller to manage the RF transmission, the transceiver takes in charge the whole communication automatically) enabling fast development of massive networks. While not being, by some aspects¶, the best solution for high performances wireless sensor network (no standard still exists in this domain), our WSN will be based on this standard protocol ||.

2.4. From IEEE 802.15.4 to Zigbee : brief overview of key parameters

With a data rate of 250kb/s, Zigbee limits the number of nodes that will be supported by the network and the latency (time elapsed between sensing operation on the node and data collection on the coordinator) of the network will directly impact on the control law. A global knowledge of its structure and the way it works is necessary to be able to establish the constraints. This section will present the basics of the IEEE 802.15.4 Zigbee tailored to WSN for mechatronic systems. For an exhaustive description of IEEE 802.15.4, we will report to the norm [20].

¶ Zigbee protocol is optimized for low duty-cycle applications (<0.1%).

|| Miwi protocol based on IEEE 802.15.4 developed by Microchip is another way of development while not being as widespread as Zigbee.

Since in this work, we use standard radio-frequency transceiver, the physical layer is set and cannot be a design parameter : we will then focus on the parameters of the MAC layer and on the overlayer that Zigbee adds to IEEE 802.15.4 at the network level: organisation of the node of the network, data circulation in the network and application in the network.

2.4.1. At the network level : architecture parameters So as to transfer data from a node to another, five important aspects are to be considered :

- Topology of the network: among the different topologies supported by Zigbee, the star topology is the best choice for our application so this aspect won't be considered as a parameter of design in our approach.
- Device class: in our application, with a star topology, the collecting node is a Full Function Device and the other nodes are Reduced Function Devices. This aspect being set, we will not developed further this aspect neither.
- Frame structure: to the data to be transmitted, the IEEE 802.15.4 MAC will add bytes so as to be able to fulfil safely the transfer. This supplementary bytes are the address of the receiving node and some bytes necessary to the synchronization of the data and to check the integrity of the data. Typical structure of a data frame will be developed later since offering several design parameters.
- Access mode to the medium: non-beacon mode and beacon-mode have both been explored in this work.
- Data transfer: the star topology sets, in our application, the status of every node. Furthermore, so as to achieve the best latency results, optional acknowledgements will not be used.

Every of these aspect will have its impact on the number of nodes that can be supported and on the average latency. Parameters associated to these aspects will be described in the following subsections.

2.5. Beaconed and non-beaconed networks

The main access method to the media in IEEE 802.15.4 is the channel access mode Carrier Sense, Multiple Access/ Collision Avoidance (CSMA/CA). In IEEE 802.15.4, this channel access mode can be used in two ways: unslotted mode and slotted mode. Both have been explored in this work. Since in beacon-mode, a superframe structure is used, we will briefly present its characteristics so as to introduce the design parameters for the MAC layer. The superframe structure divides time into different transmission periods: Beacon, Contention Access Period (CAP), Contention Free Period (CFP) and inactive period as shown figure 3. Beacon mode is selected when power-saving is a really hard constraint since every node can sleep between two beacons. Furthermore minimal latency can be guaranteed.

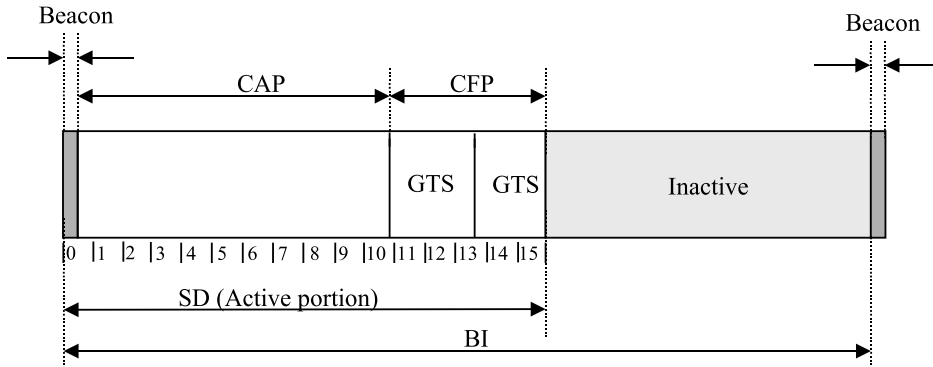


Figure 3: Superframe structure in beacon mode.

The duration of each period (active and inactive) will determine dynamic performance of the network: maximal data rate and latency. They can be calculated as follows :

- The Superframe Duration (SD) is the total time duration of the CAP, CFP (GTS) and a Beacon; it does not include the inactive period and is composed of 16 slots. This duration can be calculated by the following formula

$$SD = aBaseSuperframeDuration \cdot 2^{SO}$$

with $0 \leq SO \leq BO \leq 14$

(1)

The minimum duration of a superframe (*aBaseSuperframeDuration*) is fixed to 960 symbols (16 slots of 60 symbols) corresponding to 15.36 ms, assuming 250 kbps in the 2.4 GHz frequency band (*BO* and *SO* are respectively the Beacon Order and the Superframe Order).

- Beacon Interval (*BI*) is the time duration between two successive beacons: it is composed of the active period and the inactive period. It can be calculated as follows :

$$BI = aBaseSuperframeDuration \cdot 2^{BO}$$

with $0 \leq BO \leq 14$

(2)

When $BO = SO$, the inactive period is reduced to zero and then $BI = SD$.

For the superframe structure, *BO* and *BI* will be the two key parameters used in our design space exploration.

In this part, we presented the basis of the Zigbee and IEEE 802.15.4 protocol with emphasis on our mechatronic application. In addition each design parameter that can be used for optimal adaptation of Wireless Sensor Network to the targeted application has been presented. In the next section, we will establish the specifications of the Wireless Sensor Network that has to be deployed.

3. M@L: Wireless Sensor Network specification

The first part of our work was to design the Wireless Sensor Networks from the hardware specification of each part of the node to the size and topology of the global network. The starting point being the data that needs to be collected, we begin by extracting the mechanical specifications of our vibrating structure to establish the format of the data that have to be processed by the control law.

3.1. Mechanical specification extraction

Active vibration control in vehicles is a complex problem that often necessitate the isolation of particular transmission paths. In this work where the focus is not on the development of the control law itself but rather on the wireless aspect of the data sampling collection, we will limit ourselves to the hypothesis of the use of piezoelectric sensors and actuators. For active control of internal automotive vibrations, we will consider the first three modes resulting in a [10Hz,300Hz] range. The preliminary estimation of the needed sampling rate, based on the active control strategy to be deployed is about 1kHz. Those two elements will set both the sampling rate performances of every node of the WSN as well as the constraints on the wireless sensor networks that must provide the necessary bandwidth resulting from the sampling data rate that must be respected.

3.2. Control law in active control vibration

Specifications for active control vibration are naturally expressed in the frequency domain. Unfortunately, up to recently, most of the efficient active control design methods focus on time domain specifications. Nevertheless, the H_∞ control method [21] emerges as an efficient active control design method which directly handles frequency domain specifications. The active control of flexible systems using H_∞ control begun to be explored in the nineties [22], with a special emphasis on multi-actuators multi-sensors systems. Based on the know how developed on the application of this approach, the use of model reduction methods [23] allow to design active control systems of the lowest complexity. Note that the necessary complexity of the active control system dramatically depends on the performance specifications under consideration. The active vibration control strategy that will be first implemented in the computer connected to the collector node of our WSN will be a global semi-active strategy [24] chosen for its focus on reduced energy since the WSN deployed in this work aims at being energy sufficient (energy harvesting) in a long term view.

3.3. Hardware architecture of the node

It has been previously established that every node of the WSN must at least perform a sensing operation every 1ms. Furthermore, the latency should be kept to minimal.

In this section we will deduce from specification the hardware architecture of the node that will meet the specification for the global Wireless Sensor Network performance.

3.3.1. Microcontroller specifications From the previous sections, we can infer, as expected [25] [26], that an 8-bits architecture based microcontroller is sufficient. Furthermore, judging by the quantification specification for the analog to digital conversion, a microcontroller's integrated ADC (Analog-to-Digital Converter) solution can be considered: indeed current 8-bits microcontroller offer 10-bits based analog-to-digital conversion with a sampling rate up to 20 MHz with an acquisition time about 15-20 μ s. A brief overview of commonly used microcontroller for WSN make appear that ATMEGA128 family from ATMEL [27] or MSP430 family from Texas Instrument [28] meets the requirements. Microcontrollers from Microchip are also good candidates with PIC16F [29] family and their nanowatt technology.

3.3.2. Transceiver specifications We have chosen two devices commonly used in WSN community: the first one is the Chipcon CC2420 that can be found in many commercial platforms and particularly in the MicaZ node from Crossbow. The second is the MRF24J40 from Microchip chosen because of its hardware support of IEEE 802.15.4 specification. For the chipcon CC2420, the MAC layer must be realised by the microcontroller and then sent to CC2420 by SPI wire whereas for MRF24J40, microcontroller just has to send data to the transceiver by SPI wire and MAC layer support is done by the transceiver itself.

3.3.3. Sensor and signal conditioning From mechanical specifications, it has been deduced that a soft PZT would fit our needs. We have chosen a PIC255 from Picoceramic. Yet, the output voltage delivered by the piezoelectric sensor is about millivolts. A signal conditioning stage has then be realized using a classical charge mode amplifier circuit architecture (cf. figure 4) based on the TLV2772 MOS amplifier from Texas Instrument, a commonly used amplifier in piezoelectric conditioning circuit [30]. The TLV2772 is an operational amplifier with high slew rate and bandwidth, rail-to-rail output swing, a high input impedance, a high output drive and an excellent dc precision. This device provides 10.5 V/ μ s slew rate and 5.1 MHz gain bandwidth product. With a consumption of only 1 mA of supply current, this device is perfectly adapted for signal conditioning for WSN. Furthermore, the high output drive and the rail-to-rail output swing are perfect for driving the the analog-to-digital converter of the microcontroller. The amplifier exhibits typical value of 60 V input offset voltage, 17 nV/vHz input noise voltage, and 2 pA input bias current.

The signal conditioning system is a three stage band-pass amplifier: the first stage is a passive high-pass filter that rejects continuous value of the signal, the second stage is dedicated to the amplification of the signal and the last stage acts as a passive low-pass filter to reject high frequency signal.

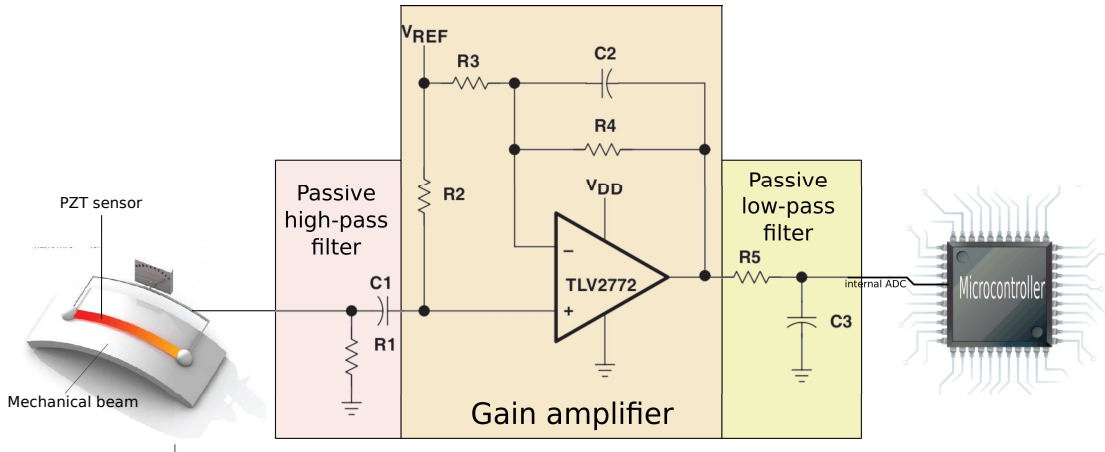


Figure 4: Piezoelectric signal conditioning circuit architecture.

Component	Value
R_1	1 M Ω
R_2	1 M Ω
R_3	10 k Ω
R_4	1 M Ω
R_5	1 k Ω
C_1	0.22 μ F
C_2	2.2 nF
C_3	0.22 μ F

Table 1: Component values for signal conditioning interface circuit.

We used the spice electrical model of the TLV2772 to design our signal conditioning circuit. For the piezoelectric, we used a very simple charge model (classical inductance-capacitance model with values extracted from the datasheet): should it be necessary, we could use a more detailed 2D VHDL-AMS model [31]. The global transfer function of this system can easily be calculated:

$$G(j\omega) = \left(\frac{1}{1 + \frac{1}{jR_2C_1\omega}} \right) \cdot \left(1 + \frac{R_4}{R_3} \cdot \frac{1}{1 + jR_4C_2\omega} \right) \cdot \left(\frac{1}{1 + jR_5C_3\omega} \right) \quad (3)$$

The components values are summarized in table 1.

The Spice simulation results from figure 5 (bode diagram: gain and phase) show that we have a band-pass filter with low cut-off frequency equal to 720mHz and high cut-off frequency equal to 470Hz. The gain in the flat band is about 11.

The resulting circuit exhibits a 2V amplitude signal with adequate frequency bandwidth. This measurement was made on a beam whose parameters have been chosen identical with the car structure. The amplitude and responsivity fit at best with the full scale of the internal analog-to-digital converters of the two microcontroller we have

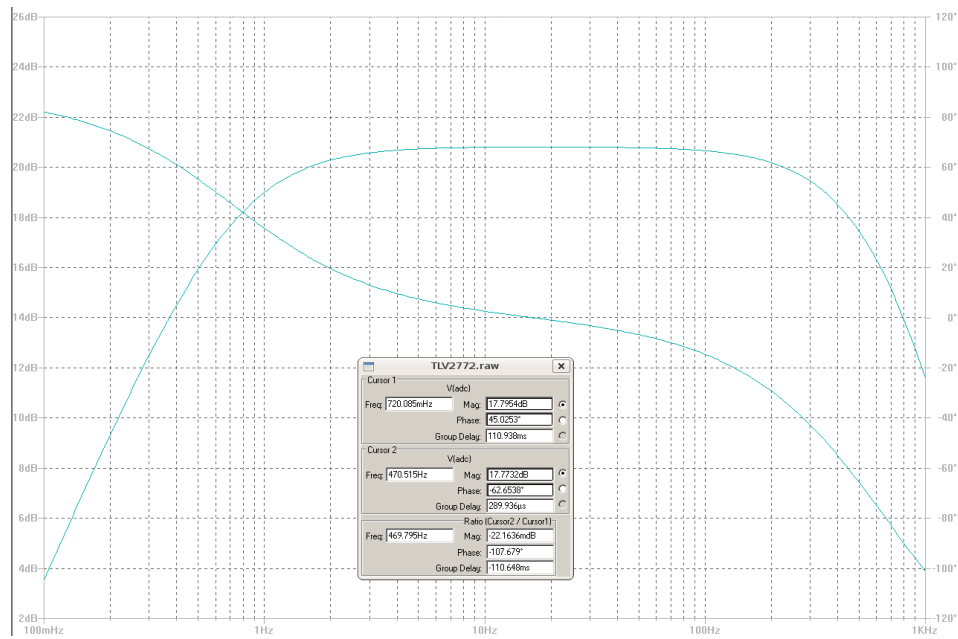


Figure 5: Spice simulation of the signal conditioning circuit.

chosen for this work. Furthermore, this results will be used to generate the sensor stimuli of our design framework IDEA1 [8].

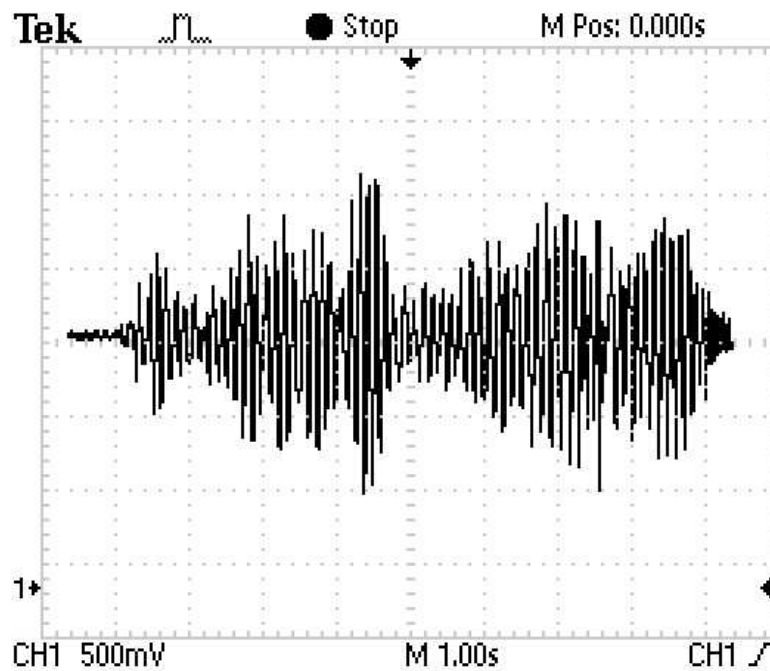


Figure 6: Piezoelectric signal conditioning on M@L mote.

The nature, number and properties of measured data are of relevance in the system performance. In our case, as low frequency vibration control is targeted, typically the control of first vibration modes of the given structure is considered. From a pure control

point of view one can consider displacement, velocity or acceleration for the vibration monitoring. It is expected that the data quality, if reasonable, will not strongly affect the algorithm performance in the low frequency range. Finally the issue of sensors locations can be treated considering the structural mode shapes of controlled modes.

3.4. Network specification

Previously, all hardware specifications have been established. Since the node will be deployed in a network, we must determine the number of node that can be supported by our application and the latency that it will present. The topology of our network will be a star composed of a Full Function Device (the coordinator) that has the ability to initiate communication and can authorize node to join the network and of Reduced Function Device that will collect the data and transmit them.

3.5. MAC layer parameters

The choice between beacons mode and non-beacons mode will have influence on the data rate and the latency. Depending on the communicating mode, collision will occur more or less. Collision that occurs can necessitate to send again data that were not received. The CSMA-CA access mode used in IEEE 802.15.4 offers mechanism to prevent collision from happening: each node must verify that the channel is free before sending data. This is achieved by sensing the idleness of the channel at random time duration. If the channel is busy, the node will wait a random time before retrying to access channel: the time unit of this random time is called backoff period (aUnitBackoffPeriod) equal to 20 symbols in IEEE 802.15.4. The random time is a random value of aUnitBackoffPeriod chosen in $[0, 2^{BE} - 1]$ with BE, the Beacon Exponent, an integer set to a minimal value and increased each consecutive time the node cannot access to the channel.

Hence, it is very difficult to estimate the average behaviour of the network with actual data rate and latency (integrating collisions and backoff). If some works exist on performances of IEEE 802.15.4 based network, they result in analytical study of particular case of the IEEE 802.15.4 [32], [33] or are linked to a specific node architecture [34]. In this work we want to design a wireless sensor network based on IEEE 802.15.4 (we will have to explore the different algorithms adequacy) and we want the best architecture for it. We will use a design framework to explore the design space at MAC layer level and hardware node level and to establish the best wireless sensor network adapted to our needs but we need a starting point for this exploration. If the models previously mentioned, can help us we will prefer more general case model to establish starting point configuration of our network.

3.5.1. Latency The only IEEE 802.15.4 mode that can guarantee a latency is the GTS mode. El Tamar et al. [35], in their work on Intra-Vehicle Wireless Automotive Sensor Networks, established a formula for the worst case latency (in ms) for beacons-mode

with GTS. If IEEE 802.15.4 could support up to 40 nodes with a latency that not exceeds 100 milliseconds, the IEEE 802.15.4 cannot offer a latency smaller than 15.9 milliseconds for any star of network of any size. The first order approximation means, in the frame of our work, that the coordinator will receive samples 16 sampling periods later compared with the real sampling time. It will have to be taken into account in the control law. The consequences of such a delay will be developed later in this paper.

3.5.2. Data rate The IEEE 802.15.4 supports a maximum over-the-air data rate of 250 kbps for the 2400-MHz band. In reality, due to the many mechanism implemented in a frame structure to ensure a robust data transmission, the effective data rate is lower. It will depend on the choice of the mode (beaconed or non-beaconed), the size of the payload, the number of nodes, In the hypothesis of a non-beacon enabled network where CSMA/CA algorithm never finds that the channel is busy and no retries are required (associated acknowledgement included), we obtain an effective data rate of 127kbps for a maximum payload of 114bytes. If we consider a more real case with a 25% probability of busy channel necessitating one retry, the effective data rate falls to 101kbps [36]. This effective rate, taken in the context of our work where we consider that the analog-to-digital conversion will be realised on one byte means that the maximum size of our network is about 16 nodes.

3.6. M@L Wireless Sensor Network: a summary

The Wireless Sensor Network that can be deployed for the active control of vibrations will be a star network composed of eight nodes based on a 8-bits microcontroller architecture and Zigbee-IEEE 802.15.4 compliant transceiver. Microcontroller of every node has internal analog-to-digital converter responsible for the digitalization of the data collected from the piezoelectric sensor after a signal conditioning based on a TLV2772 amplifier architecture. A first order analysis showed that a minimal 15.89ms latency is to be expected for a 1khz sampling rate of vibration data.

We have used our Wireless Sensor Network design platform to validate these results and to optimize the network. We used two different hardware platform for the node: a commercial platform MicaZ from Crossbow [37] and an internally designed node named N@L (acronym for Node@Lyon). Both platforms will be evaluated.

4. Wireless Sensor Network optimization

4.1. WSN platform: IDEA1

IDEA1 (hierarchical DEsign plAtform for sensOr Networks Exploration) is an internally developed framework for Wireless Sensor Networks [8]. It aims at simplifying the design of any wireless sensor network from global application specification downto hardware individual performance of every component of a node. Written in C++ and based on SystemC subset commonly used in electronic and network community [38], it enables

to take into account hardware characteristics at a network level. This framework allows designer to evaluate the impact of hardware and software architecture of a node (microcontroller/transceiver choice, Operating System implementation, sensor ...) on the performances of the global network.

4.1.1. *Architecture of IDEA1* IDEA1 is a component-based simulation framework. Every component is modeled as an individual SystemC module communicating with each other via channels. The architecture of IDEA1 is illustrated in fig. 7.

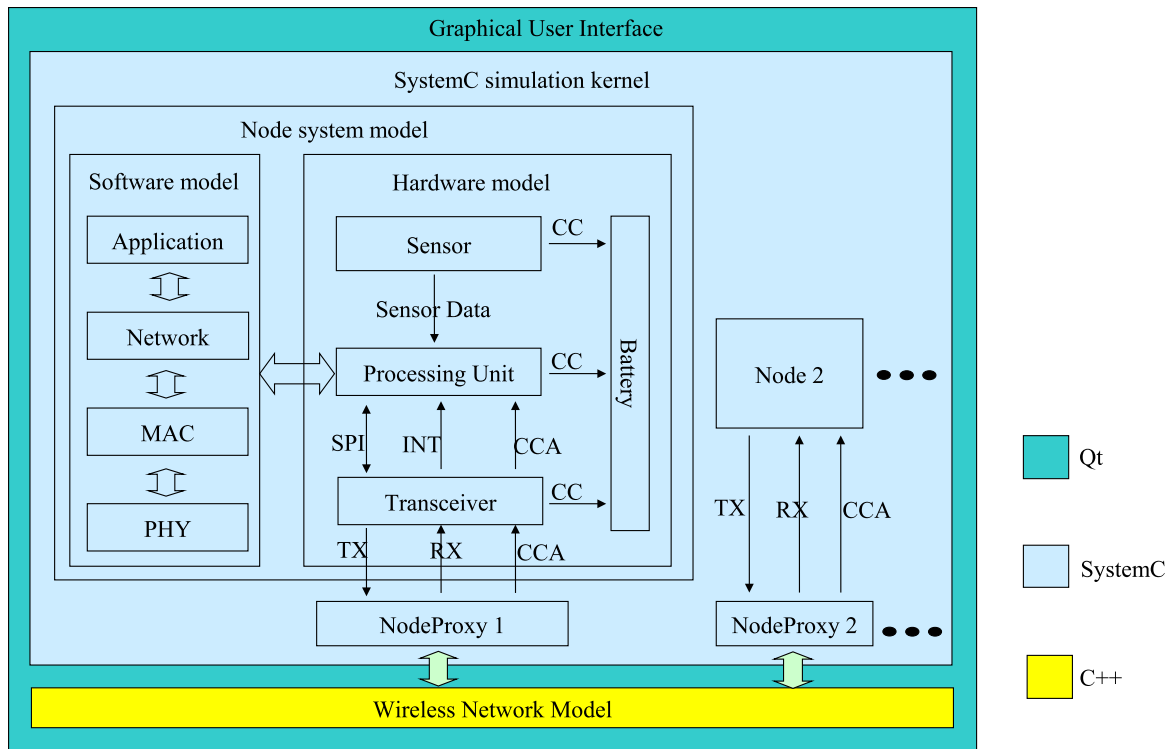


Figure 7: Architecture of IDEA1

The node system is a composite module divided into 2 sub-models, hardware model and software model. The hardware components of a sensor node generally include a processing unit, a RF transceiver, several sensors and a battery. The software model consists of operating system, middleware, protocol stack, and application implementation. We can then estimate performances impact at network level from two aspects: hardware modification or software modification. All the nodes are connected to a same network object via their proxy modules. At the initialization phase, every proxy registers its information in the network module such as position, TX power and RX sensitivity. During simulation, the network object reads the packet sent by nodes, calculates the distance between the source and its destination based on the parameters of relative nodes, and forwards the packet according to the radio propagation models. If two nodes in the radio range transmit at the same time, a collision will occur. The SystemC kernel acts as the simulation engine. It schedules the execution of processes

and updates the state of all modules at every simulation cycle. All active processes are invoked orderly at the same simulator time, which creates an illusion of concurrency.

4.1.2. *A Graphical User Interface* IDEA1 is targetting an audience composed of Wireless Sensor Network designer. We have then developped graphical user interface based on Qt platform to integrate all the parts, which can facilitate the system configuration, network topology visualization, simulation control and result analysis.

Users can use graphical interface to configure the network system and analyze the simulation results. The input parameters of IDEA1 are defined in an eXtensible Markup Language (XML) file, which is read by the executable simulation code during runtime. The input parameters and output results of IDEA1 are summarized in fig. 8.

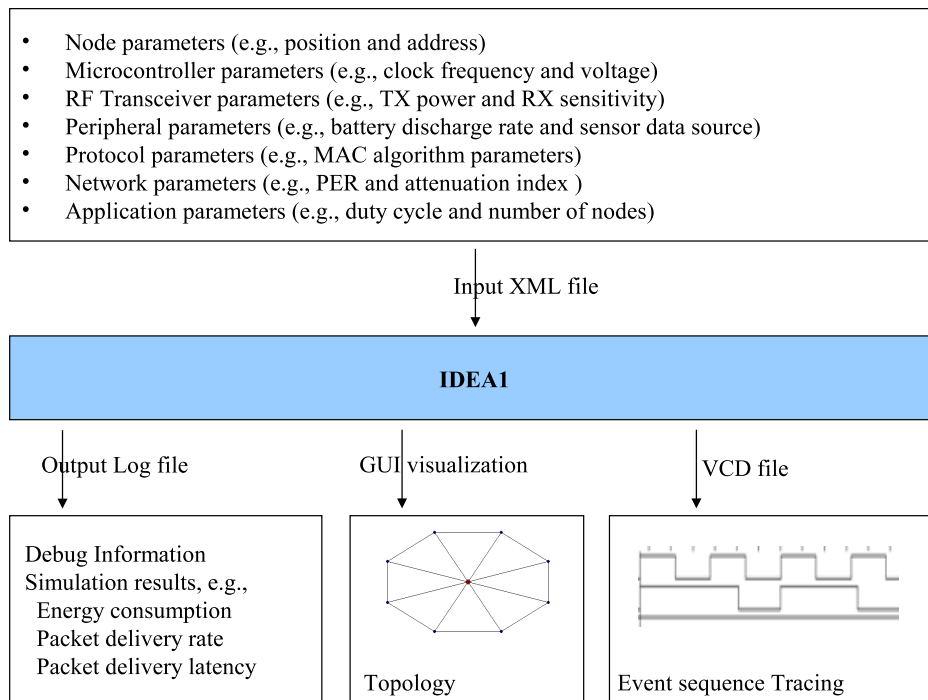


Figure 8: IDEA1 input parameters and output results

Many parameters of different-levels can be configured by users, including node level, protocol level, application level, etc.

4.1.3. *A library* Many commercial off-the-shelf (COTS) hardware components have been modeled. The current library of hardware supported by IDEA1 is summarized in table 2. We are currently developping TI MSP430 microcontroller so as to have a 95% coverage of existing commercial node for WSN.

4.1.4. *Design space exploration framework* If commonly accepted, use of simulation tools in the WSN is a subject of controversy. It is now established that these tools are effective in the comparison of solutions but not for validation before deployment

Processing unit	RF Transceiver
ATMEL ATmega128	TI CC2420
Microchip PIC16LF88	TI CC1000
	Microchip MRF24J40

Table 2: IDEA1 supported devices library

physique. According to analytical works of Stuart Kurkowski [39], [40], WSN simulations tools results are indeed not representative of reality unless some precautions should be taken. IDEA1 has been developed according to these aspects and has been validated through experimental results [41] thus enabling us to use it for design space exploration for WSN pre-deployment analysis.

4.2. Node target architecture

For this work we use two nodes architecture: MicaZ and N@L. MicaZ is a reference mote architecture in the WSN community and is often implemented in mechanical systems such as structural health monitoring systems [42]. It is composed of an Atmel ATmega128 microcontroller [27] and a CC2420 transceiver from Texas Instrument [43], a monopole antenna and a number of peripherals such as leds, flash memory and a 51-pin connector. The communication between the microcontroller unit and the transceiver is realized by a Serial Peripheral Interface (SPI) and control pins such as Clear Channel Assessment (CCA) and Start of Frame Delimiter (SFD). The SPI is used for both packets and control signal since the CC2420 does not support fully the IEEE 802.15.4. The microcontroller unit is then responsible for the IEEE 802.15.4 MAC layer support. Both the fact that IEEE 802.15.4 must be software implemented and the limited throughput of the SPI link can be bottleneck to global network performances.

N@L is an internally developed node: it is composed of a PIC16LF88 microcontroller unit [29] and a the MRF24J40 transceiver [29] (both from Microchip), a monopole PCB antenna and peripherals such as leds and a piezoelectric sensing circuit interface. The communication between the microcontroller unit and the transceiver is realized by a Serial Peripheral Interface (SPI) and control pins such as the wake-up trigger (WAKE) and the interrupt pin (INT). The microcontroller was chosen for its nanowatt technology that ensure low power consumption and the MRF24J40 because of its full hardware support of IEEE 802.15.4 MAC layer. At first view, N@L node architecture is superior to MicaZ's architecture and should exhibit better performances when deployed in a network.

4.3. Definition of the metrics.

To evaluate the performances of the Wireless Sensor Network, we need metrics. According to our application, the metrics are defined as follows

- Packet Delivery Rate (PDR): PDR is used to evaluate the network throughput. It is the ratio of the number of packets successfully received by coordinator to the number of packets need to be sent by nodes. A single communications failure occurs when a device transaction fails to reach the coordinator, i.e., channel access failure after `macMaxCSMABackoffs` attempts and an acknowledgment is not received after `macMaxFrameRetries` attempts,
- Average Latency (AL): Latency of a packet is the duration elapsed between the time the sensor from the node reads data and the time when the data is received by the coordinator. AL is an average latency of all packets successfully received by the coordinator,
- Power consumption: due to the battery of every node, the lifetime of the network is limited. It is very important to estimate the power consumption at the node level. We will differentiate global power consumption, microcontroller power consumption (data measurement and processing cost) and transceiver power consumption (radiofrequency communication cost).
- Energy Consumption per Sample (ECSple): ECSple is the average energy consumed for successfully transmitting one sample: with packet delivery rate and average latency, this metric fully characterizes the performance of the network.

All these metrics will be used to evaluate the performance and the lifetime of the wireless sensor network.

4.4. M@L Wireless Sensor Network: design space exploration

The three MAC algorithms in IEEE 802.15.4 standard are implemented, including unslotted CSMA-CA, slotted CSMA-CA and GTS. Unfortunately, the maximum number of GTS slots in a superframe defined in the IEEE 802.15.4 standard is 7. Since our current application consists of 8 nodes, we used the TDMA-based GTS algorithm proposed in [44] : it is more suitable than the original IEEE 802.15.4 GTS algorithm for industrial applications which require low packet delivery latency.

For each algorithm, many cases with different configurations of parameters (e.g., payload, superframe length, the maximum number of the retries allowed after a transmission failure, ...) have been simulated. Payload presents the number of samples in a packet. A sample occupies one byte. The node sends the sensor data out if the data in the buffer is more than payload. In an ideal case, the payload should be reduced to one sample but due to the high sampling rate (compared with the effective IEEE 802.15.4 data rate) such a solution is impossible. A small payload will result in more packets to be sent, causing more collisions and thus lower PDR. Furthermore, when payload is small, the short packet causes a short latency and the big number of packets results in more energy consumption due to more attempts of channel access. In contrast, a big payload takes a longer time for transmitting a packet, which will increase the channel access failures and cause lower PDR too. The best PDR, hence, occurs in the case with a moderate payload as short as possible so as to minimize the latency in the network.

Algorithm	Unslotted CSMA-CA		Slotted CSMA-CA		IEEE802154 GTS		TDMA-based GTS	
	MICAz	N@L	MICAz	N@L	MICAz	N@L	MICAz	N@L
Hardware platform	MICAz	N@L	MICAz	N@L	MICAz	N@L	MICAz	N@L
sizePayload (byte)	30	30	30	30	30	15	10	19
BO	n/a	n/a	1	1	1	0	n/a	n/a
BI (μ s)	n/a	n/a	30720	30720	30720	15360	10000	19000
PDR (%)	36.5	54.4	39.7	67.4	97.4	97.4	100	100
AL (μ s)	11583	15841	22426	24250	53854	42777	6953	12508
ECpkt (μ J/pkt)	3811	1924	3784	1576	1283	1001	425	408
APC (μ W)	46693	35155	50397	35684	41071	64630	42300	21264
APC of microcontroller (μ W)	29916	4576	29916	4576	29915	4448	29928	4573
APC of transceiver (μ W)	16777	30579	20481	31108	11157	60182	12371	16691

Table 3: Simulation results of MicaZ and N@L node.

We have proceeded to simulations based on the starting point previously established and taking into account the aforementioned parameters. For every MAC algorithms, we have kept only the best results with the biggest PDR (or lowest AL if two or more cases achieve the biggest PDRs): they are presented in table 3. Each case is the average over 2500 samples and is simulated 100 times with random seeds : each metric is the average of the corresponding metrics over the 100 simulations so as to take into account the statistical nature of the RF communication. The two different hardware platforms previously presented are exploited.

Beyond this global results giving the performances at the network level, it is interesting to extract performances at node level so as to be able to identify the bottleneck of every node. In figure 9, energy consumption of node is represented split over the architecture: processing unit is separated from the communication unit so as to evaluate the most consuming part. For the processing unit, we have three main processes: analog to digital conversion (EnergyADCPeNode), SPI communication (EnergySPIPeNode) and data processing (EnergyCPUPeNode). For the communication unit, we have chosen to separate the energy consumption in two parts: consumption in active mode (EnergyTransActive) and consumption in sleeping mode (EnergyTransSleep).

5. Discussions

5.1. MAC algorithm best choice

The CSMA-CA algorithms are not appropriate for this industrial application due to low PDRs. The sample rate is so high that the system is overloaded and every node keeps trying to transmit the sensor data during whole application and they compete for channel usage. The PDRs are small as a result of large number of collisions.

Due to the constraints of the maximum GTS slots number of the IEEE 802.15.4 standard, the number of nodes in this simulation is set to 7. Furthermore the original IEEE 802.15.4 GTS algorithm requires a minimum length of CAP period of a superframe (440 symbols – 7040 μ s for 250 kbps data rate): in our application, this period is not used resulting in a loss of performances. Each node can only be allocated for one slot. This algorithm is implemented by software in MicaZ mote and by hardware in N@L mote. If from a global point view, N@L is superior to MicaZ, for this configuration, the

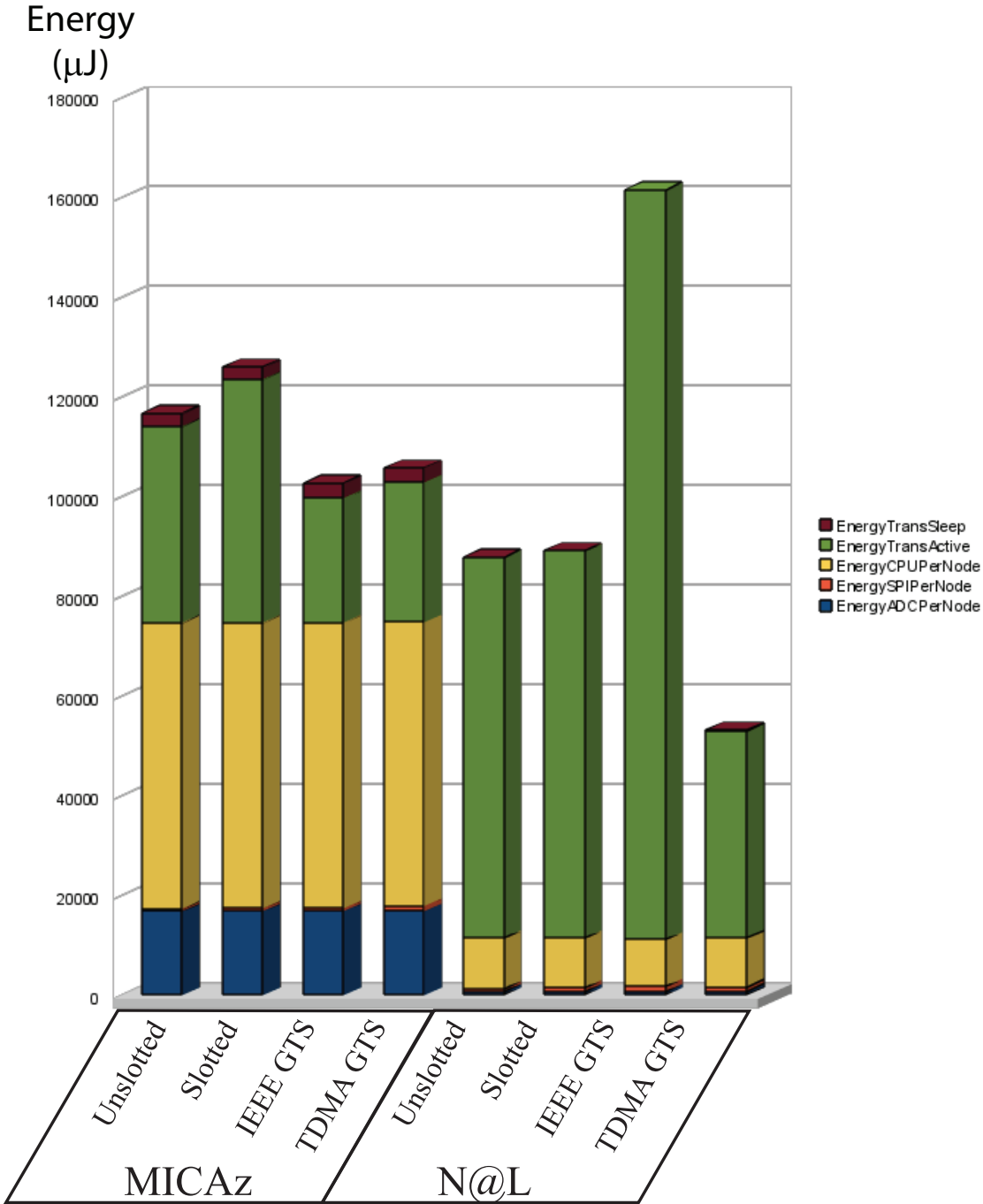


Figure 9: Power consumption anatomy of the node.

power consumption of the N@L node is surprisingly 50% higher than the MicaZ's power consumption. At the node level, it appears that for IEEE 802.15.4 GTS, the MRF24J40 transceiver is hugely consuming. This peak of consumption comes from the hardware implementation of the IEEE 802.15.4 standard. Indeed, for MicaZ mote, after receiving a beacon packet, the microcontroller can set the transceiver to sleep mode until its GTS slot; however for N@L mote, the transceiver acts automatically and stays in active mode during the CAP portion of a superframe. Yet, N@L stays the best candidate because of its payload of 15 compared with the payload of 30 for the MicaZ. This IEEE 802.15.4 algorithm is not suited for application since it authorizes only 7 nodes but we have taken it into account to demonstrate the feasibility and the limits of a WSN with standard IEEE 802.15.4. It has to be noticed that it is the first algorithm that can ensure a nearly 100% Packet Delivery Rate.

For the TDMA-based GTS algorithm, the PDRs can attain 100%, which prove the reliability of TDMA-based GTS algorithm based on the MicaZ and N@L hardware platforms in transmitting the sensor data to the coordinator. Yet, this IEEE 802.15.4 based sensor network fails to meet the real-time requirement of this application (1 kHz sample rate) due to the large average latency of packet. Although the average latency of packets can be reduced to 7.0 ms (in contrast with the theoretical 15.3 ms mentioned in El Tamar's work [35]), minimal payload is 10 samples which means that sensor node must collect 10 samples before sending them. Thus the collector node receives 10 samples at the same time, the first being 17ms old. It is interesting to notice that for the TDMA-based GTS algorithm, the power consumption of the N@L's transceiver is reduced since the microcontroller is taking in charge the communication and then can set the RF transceiver in sleeping mode.

From there, mechatronic system designers have two obvious choices : first they consider that the 9 first samples are useless and process only the last sample (this is equivalent to downgrade the effective sampling rate to 100Hz) or on the other hand they consider that they must establish a control law robust to a 17ms latency with the advantage of knowing the following coming samples. A third solution can be found: a distributed control law enabling local calculation on every node will reduce the data rate between the node and could be an interesting alternative.

5.2. Choice of an architecture node

Figure 9 allows a straightforward classification of pro and cons of each architecture from a consumption point of view. MicaZ has a low consuming transceiver but its processing unit is consuming in two aspects: data processing and analog-to-digital conversion. N@L has a low consuming processing unit but exhibits average to bad performances in RF communication. The full hardware support of the IEEE 802.15.4 standard that should be an advantage results in a consuming electronic circuit. Yet, the Packet Delivery Rates (PDR) of N@L are better than those of MicaZ (at same parameters they are better or they offer same results for lower payload), because of the hardware implementation

of the MAC algorithms in MRF24J40. Furthermore, the average latencies of N@L and MicaZ are in the same order with a slight advantage to the MicaZ. This slight advantage comes from the SPI communication between ATMEL ATmega128 and TI CC2420 that is faster than its counterpart between PIC16LF88 and MRF24J40. In order to transmit one packet of several bytes from PIC16LF88 to MRF24J40, the address needs to be sent before each byte. However, ATmega128 only has to transmit one address before the transmission of packet. For example, PIC16LF88 needs 410 μ s to transmit a packet of 10 bytes to MRF24J40, but it only takes 47.47 μ s for ATmega128.

The N@L node platform is the better choice since the energy per sample (ie. the energy efficiency) is 25% to 50% smaller than its counterpart (except for the IEEE 802.15.4 GTS algorithm that supports only 7 nodes).

The two node architectures studied in this work (MicaZ and N@L) permit the constitution of two networks presenting similar results with a slight advantage for the N@L platform for its reduced consumption. Furthermore, we could infer from the table 3 and figure 9 that a third architecture based on a PIC16LF88 and a CC2420 node won't be more performing because of the lack of hardware support for IEEE 802.15.4 standard. We can expect an equivalent power consumption because of the increased consumption from the microcontroller's software support of IEEE 802.15.4 standard and a resulting degraded latency. Indeed, if the PIC16LF88 realizes the communication process, the data that will be transmitted through the SPI wires between PIC16LF8 and CC2420 will be much larger and we saw previously that the SPI communication is one of the main drawback of the PIC16LF88 microcontroller.

6. Conclusion and perspectives

In this paper we have presented the design of a wireless sensor network for automotive active control, from system specifications down to hardware implementation of node. From mechanical specifications of the physical phenomena, we have established the specifications of the wireless sensor network to be deployed. Then, we have used our WSN design platform to validate the WSN network structure established based on two different node architecture: MicaZ and N@L. From the simulation results, we have demonstrated that the IEEE 802.15.4 standard used is hardly suited for our application: if the samplings can be routed to the collecting node at a nearly 100% rate, the average latency generated is really too high to establish an efficient real-time control law. We have also demonstrated that the two node architectures are surprisingly similar in terms of performance with an advantage for the N@L platform for the energy efficiency.

The wireless sensor network based on N@L platform is currently being deployed on an automobile structure so as to explore the possibilities of embedding distributed control law in the network. In parallel of this validation, the following scientific fields are under investigation as a consequence of the results of this work : distributed computation, energy harvesting and structural health monitoring applications. By offering a limited communication bandwidth, WSNs cannot offer support for centralized

control law but with an embedded computation unit at every node, local computation at sensor level is possible. Then active control laws could take benefit from distributed computation that may be used with centralized approach : the design bottlenecks are then on the distributed control laws and on the balance to establish between local control and distributed control. In an extended vision, dynamic control could be introduced by acting on this balance in a real-time approach. Energy that can be collected from vibration is high enough to enable the possibility of using sensor both as sensor and energy collector in a similar way that is currently used in structural health monitoring.

In conclusion, WSN nodes equipped with piezoelectric sensor and actuators (providing energy through harvesting and enabling local control through dissipation) should offer a performant alternative to current solutions with dynamic strategies based on the local control/global control ratio and methodologies for optimal dynamic spatial control.

Acknowledgment

The authors would like to thank I@L Carnot Institute for his financial support on this work.

Bibliography

- [1] J.P. Lynch and K.J. Loh. A summary review of wireless sensors and sensor networks for structural health monitoring. *Shock and Vibration Digest*, 38(2):91–130, 2006.
- [2] Y. Wang, R.A. Swartz, J.P. Lynch, K.H. Law, K.C. Lu, and C.H. Loh. Decentralized civil structural control using real-time wireless sensing and embedded computing. *Smart Structures and Systems*, 3(3):321–340, 2007.
- [3] Shamim N. Pakzad, Gregory L. Fenves, Sukun Kim, and David E. Culler. Design and implementation of scalable wireless sensor network for structural monitoring. *Journal of Infrastructure Systems*, 14(1):89–101, 2008.
- [4] Rolf Isermann. *Mechatronic Systems: Fundamentals*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.
- [5] Nitaigour P. Mahalik. *Sensor Networks and Configuration: Fundamentals, Standards, Platforms, and Applications*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.
- [6] K. Chintalapudi, T. Fu, J. Paek, N. Kothari, S. Rangwala, J. Caffrey, R. Govindan, E. Johnson, and S. Masri. Monitoring civil structures with a wireless sensor network. *Internet Computing, IEEE*, 10(2):26–34, 2006.
- [7] Byoung-Kug Kim, Sung-Kwa Hong, Young-Sik Jeong, and Doo-Seop Eom. The study of applying sensor networks to a smart home. In *NCM '08: Proceedings of the 2008 Fourth International Conference on Networked Computing and Advanced Information Management*, pages 676–681, 2008.
- [8] Wan Du, Fabien Mieyeville, and D. Navarro. IDEA1: A SystemC-based system-level simulator for wireless sensor networks. In Wen Chen and Shaozi Li, editors, *IEEE International Conference WCNIS2010*, volume 2, pages 618–623. IEEE, June 2010.
- [9] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Comput. Netw.*, 52:2292–2330, August 2008.
- [10] R.C. Johnson. Millimeter-scale computer targets environmental, medical monitoring. *EE Times*, (1597):26–26, 2011.

- [11] D. Culler, D. Estrin, and M. Srivastava. Guest editors' introduction: Overview of sensor networks. *Computer*, 37(8):41–49, 2004.
- [12] Ioannis Chatzigiannakis, Georgios Mylonas, and Sotiris Nikolettseas. 50 ways to build your application: A survey of middleware and systems for wireless sensor networks. In *Proceedings of the 12th IEEE Conference on Emerging Technologies and Factory Automation*, 2007.
- [13] Muhammad O. Farooq and Thomas Kunz. Operating Systems for Wireless Sensor Networks: A Survey. *Sensors*, 11(6):5900–5930, May 2011.
- [14] Jason Lester Hill. *System architecture for wireless sensor networks*. PhD thesis, 2003. Adviser-Culler, David E.
- [15] Funkenstein Software Consulting. *FunkOS : a fully preemptive multi-tasking real-time environment*, 2010.
- [16] Mihai Galos, Fabien Mieyeville, and David Navarro. Dynamic reconfiguration in wireless sensor networks. In *17th IEEE International Conference on Electronics, Circuits, and Systems: ICECS 2010, Athens*, december 2010.
- [17] Thomas Nolte, Hans Hansson, and Lucia Lo Bello. Wireless automotive communications. In *Euromicro Conference on Real-Time Systems, Palma de Mayorca*, pages 35–38, July 2005.
- [18] *Zigbee Specifications*.
- [19] G. Zames. Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses. *IEEE Trans. Aut. Control*, AC-26(2):301–320, April 1981.
- [20] S. Font, G. Duc, and F. Carrère. H_∞ control of a magnetic bearing. In *Proc. Control Conference Applications*, 1994.
- [21] G. Obinata and B.D.O Anderson. *Model reduction for control system design*. ISBN: 1-85233-371-5. Springer-Verlag, 2000.
- [22] M N Ichchou, T Loukil, O Bareille, G Chamberland, and J Qiu. A reduced energy supply strategy in active vibration control. *Smart Materials and Structures*, 20(12):125008, 2011.
- [23] J. P. Lynch, K. H. Partridge, A. and Law, T. W. Kenny, A. S. Kiremidjian, and E. Carryer. Design of a piezoresistive mems-based accelerometer for integration with a wireless sensing unit for structural monitoring. *J. Aerospace Eng.*, 16:108–114, 2003.
- [24] L Liu and F G Yuan. Active damage localization for plate-like structures using wireless sensors and a distributed algorithm. *Smart Materials and Structures*, 17(5):055022, 2008.
- [25] ATMEL. *ATmega128(L) Data Sheet. Rev. 2467SAVR07/09*, July 2009.
- [26] Texas Instrument. *Ultra-Low-Power MSP430 Microcontrollers*, 2011.
- [27] Microchip. *DS30487C - PIC16F87/88 - 18/20/28-Pin Enhanced Flash MCUs with nanoWatt Technology*, 2005.
- [28] Jim Karki. 3-v accelerometer featuring tlv2772 - slva040. Technical report, Texas Instrument, 1998.
- [29] Jean-Marc Galliere, Philippe Papet, and Laurent Latorre. A 2-D VHDL-AMS Model for Disk-Shape Piezoelectric Transducers. In *IEEE Behavioral Modeling and Simulation Conference*, page 5, San Jose, USA, 09 2008.
- [30] Chiara Buratti. A mathematical model for performance of IEEE 802.15.4 beacon-enabled mode. In Mohsen Guizani, Peter Müller 0002, Klaus-Peter Fähnrich, Athanasios V. Vasilakos, Yan Zhang, and Jun Zhang, editors, *IWCMC*, pages 1184–1190. ACM, 2009.
- [31] Chiara Buratti and Roberto Verdone. Performance analysis of IEEE 802.15.4 non beacon-enabled mode. *IEEE transactions on vehicular technology*, 58(7):3480–3493, 2009.
- [32] Nelson I. Dopico, Carlos Gil-Soriano, Inigo Arrazola, and Santiago Zazo. Analysis of IEEE 802.15.4 throughput in beaconless mode on MicaZ under TinyOS 2. In *Proceedings of the 72th IEEE Vehicular Technology Conference, VTC Fall 2010, 6-9 September 2010, Ottawa, Canada*, pages 1–5. IEEE, 2010.
- [33] Tamer ElBatt, Cem Saraydar, Michael Ames, and Timothy Talty. Potential for intra-vehicle wireless automotive sensor networks. In *In 2006 IEEE Sarnoff Symposium*, pages 27–28, 2006.

- [34] Application note JN-AN-1035 - calculating 802.15.4 data rates. Technical report, Jennic, August 2006.
- [35] XBow. *MICAz datasheet*, 2010.
- [36] Elvinia Riccobene, Patrizia Scandurra, Sara Bocchio, Alberto Rosti, Luigi Lavazza, and Luigi Mantellini. SystemC/C-based model-driven design for embedded systems. *ACM Trans. Embed. Comput. Syst.*, 8(4):1–37, 2009.
- [37] Stuart Kurkowski, Tracy Camp, and Michael Colagrosso. MANET simulation studies: the incredibles. *SIGMOBILE Mob. Comput. Commun. Rev.*, 9(4):50–61, October 2005.
- [38] Todd R. Andel and Alec Yasinsac. On the credibility of manet simulations. *IEEE Computer*, 39(7):48–54, 2006.
- [39] Wan Du, Fabien Mieyeville, David Navarro, and Ian O’Connor. Idea1: A validated systemc-based system-level design and simulation environment for wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2011(1):143, 2011.
- [40] Jeongyeup Paek, K. Chintalapudi, R. Govindan, J. Caffrey, and S. Masri. A wireless sensor network for structural health monitoring: performance and experience. In *Proceedings of the 2nd IEEE workshop on Embedded Networked Sensors*, pages 1–9, Washington, DC, USA, 2005. IEEE Computer Society.
- [41] Texas Instrument. *CC2420 Data Sheet, Rev. B*, March 2007.
- [42] Feng Chen, Thomas Talanis, Reinhard German, and Falko Dressler. Real-time Enabled IEEE 802.15.4 Sensor Networks in Industrial Automation. In *IEEE Symposium on Industrial Embedded Systems (SIES 2009)*, pages 136–139, Lausanne, Switzerland, July 2009. IEEE.