

On-node Signal Processing to Reduce the Power Consumption of Wireless Sensor Nodes for Vibration Monitoring

Application in Rolling Element Bearing Monitoring

Vignesh Raja Karuppiah Ramachandran¹, Andrea Sánchez Ramírez², Berend Jan van der Zwaag³, and Paul Havinga¹

¹Pervasive Systems, Dept. of Computer Science

²Applied Mechanics, Dept. of Mechanical Engineering
University of Twente, Enschede, The Netherlands

v.r.karuppiahramachandran@utwente.nl, a.sanchezramirez@utwente.nl, b.j.vanderzwaag@alumnus.utwente.nl, p.j.m.havinga@utwente.nl

³Adaptive Systems

Hengelo (O), The Netherlands

Abstract—In recent years, the use of wireless sensor networks for vibration monitoring is emphasized, because of its capability to continuously monitor at hard-to-reach locations of complex machines. Low power consumption is one of the main requirements for the sensor nodes in continuous and long-term vibration monitoring. However, the power consumption of state-of-the-art wireless sensor nodes is significantly increased by wireless radio in continuously transmitting the raw vibration data to the base station. One of the ways to reduce the power consumption is to reduce the duty-cycle of wireless transmission. Accurately processing the vibration data on the sensor node and transmitting only the critical information, such as natural frequency, defective frequency and amplitude of the vibration, will not only reduce the amount of data transmitted but also the duty cycle of the wireless communication. It eventually leads to reduction of total power consumed by the sensor nodes. In this paper the capability of a sensor node to accurately process the real-time vibration data is analyzed and the corresponding power consumption is measured. In particular, impact-based analysis of real-time vibration data is performed by breaking complex signal-processing tasks into manageable segments on the sensor nodes to minimize algorithmic complexity while still meeting real-time deadlines of the algorithm. As a result, it is found that the accuracy of the on-node signal processing is comparable with conventional off-node monitoring methods, whilst reducing total power consumption.

Keywords—*signal processing; vibration monitoring; wireless sensor networks*

I. INTRODUCTION

Wireless sensor networks are widely being used in monitoring vibrations of complex machines such as gas turbines, trains, helicopters, and industrial manufacturing machines. The need of wireless communication is inevitable for monitoring at hard-to-reach locations in complex machines, such as rotor blades in helicopters and gas turbines, bearings in industrial manufacturing set-ups, and wheels of train wagons. Wired communication is not possible between embedded sensor nodes and the base station such as a computer, where the raw vibration data is analyzed with monitoring algorithms. Howev-

er, in the case of wireless communication from these miniaturized sensor nodes to the base station, more than 50% of the total power is required to continuously transmit the vibration data. It is evident that the state-of-the-art sensor nodes are embedded with sophisticated microprocessors capable of performing signal processing. The continuous transmission of raw vibration data can be reduced if the signal is analyzed locally on the sensor nodes using signal processing methods. It is obvious that vibration monitoring algorithms are complex and it is challenging to embed the algorithm on the sensor nodes. In order to reduce the algorithmic complexity, a vibration monitoring algorithm is split into signal processing segments and these segments are embedded on the sensor nodes. Results of these signal processing segments are transmitted to the base station only if the results are critical enough to predict the damage. Further analysis from these results is done on the base station to monitor the condition of the machine.

In this paper we focus on embedding impact-based analysis of vibration signals acquired from rolling element bearings. The implementation is done on the ARM cortex M4 processor, which can also be found in sensor nodes, and compare the results with off-node implementation. By embedding the algorithm on the node we show that the accuracy of the algorithm is still the same, yet with lower power consumption compared to the conventional method. The paper is presented as follows. Section 2 reviews literature of vibration monitoring in rolling element bearings. Section 3 presents the embeddability of vibration monitoring algorithms in an ARM processor with limited computational resources. In Section 4 we propose a methodology for on-node optimization of the vibration data. Hardware implementation is explained in Section 5, followed by comparative study in Section 6. Conclusions and future work are presented in Section 7.

II. VIBRATION MONITORING

Vibrations are the indicatory outcome of a dysfunction of any moving mechanical system. Monitoring these vibration signals with the help of wireless sensor networks will result in improved diagnosis of the mechanical system. Wireless sensor

This research is supported by the EU FP7-ICT project WiBRATE (<http://wibrates.eu>), under the Grant No. 289041.



Fig. 1. V-Mon sensor node of Inertia Technology

nodes such as V-Mon [7] are being used to capture the vibration signals and continuously transmit the signals to the base station for further processing and monitor the condition of the machine. Real-time transmission of vibration signals is possible in such sensor nodes. Complex signal-processing techniques such as time-domain averaging, continuous wavelet transform, empirical mode decomposition are performed on the base station, such as a personal computer.

A. Related Work

Many algorithms for off-node vibration monitoring exist. All these algorithms are complex and are not embeddable on the nodes due to the limitation of sensor nodes in terms of memory, computational complexity and power requirements. Simple and light-weight signal-processing methods such as filtering and pattern recognition are implemented on the sensor nodes to increase the quality of sensed vibration data. In [0] an embedded discrete wavelet transform and time-sample averaging technique is implemented on a microprocessor to monitor gear faults. Different implementations using a dedicated DSP or FPGA are presented in [2][3][4][5], however dedicated DSP and FPGA design is not the best solution for enabling on-node signal processing in a sensor network. Apart from [0], no other implementations of entire on-node vibration monitoring algorithms, i.e., in the microprocessor of sensor nodes, was found in literature. Most of the algorithms are developed to be processed off-node, i.e., on a dedicated base station. Whereas [0] presents a solution for gear fault detection, this paper provides a solution for fault detection in rolling element bearings, where the characteristics of the defects that may occur are different, as explained in the following.

B. Rolling Element Bearing Monitoring

A ball bearing is composed of three main parts: outer race, inner race and tiny metal balls composed with lubricants, as illustrated in Fig. 2. The load points are distributed mainly over these three structures. The damages on the outer race, inner race and the rolling elements are critical in the vibration moni-

toring. Each of these defects has a unique frequency spectrum. Analyzing the frequency spectrum of these defects will result in the fault prediction of the rolling element bearing [6]. Time-domain averaging followed by statistical analysis is an alternative approach for the rolling element fault detection. In scenarios where the specifications of the bearings are not known, blind identification of faults is required. In this case, a probabilistic approach is carried out in the time-frequency domain. All these approaches are too computationally complex for a sensor node to be embedded.

III. ALGORITHM FOR ON-NODE SIGNAL PROCESSING

Based on different conventional approaches, we implemented a simple algorithm using signal processing to predict faults in rolling element bearing. The algorithm was developed in MATLAB and validated using four data sets of real-time vibration data of rolling element bearings in industrial manufacturing machines. Three data sets contain traces of three different defects of the system and the fourth one is a trace of the system in pristine condition. The bearing specification is not known, except for the rotational frequency of the machine.

The frequency of the raw vibration signal will be in the order of kilohertz due to the mechanical stress provided by metal-metal contact of the bearing. This high-frequency vibration signal will be modulated by the defective frequencies. For the detection of bearing faults these defective frequencies have to be identified. In order to identify the defective frequency from the normal high-frequency vibration signal, an envelope extraction of the time-domain signal is performed. Applying a Fast Fourier Transform (FFT) to the extracted envelope will result in detection of the defective frequency. If there is no defect present, the rotational frequency of the machine will be clearly visible in the spectrum. The frequency peaks along with the energy information are transmitted to the base station instead of the raw data. This results in reduction in the amount of data transmitted to the base station and eventually reduces the power consumption of the sensor nodes.



Fig. 2. Composition of a ball bearing (top left); damaged bearing balls (top right); outer race defects (bottom left); and inner race defects (bottom right)

IV. PROPOSED METHODOLOGY FOR OPTIMIZATION

Conventionally, the time-domain signal is filtered by a high-pass filter to reduce noise and corrupt signals from sensor data and thus to eventually improve accuracy and reliability of monitoring process. In scenarios where the specifications of bearings are not known, it is not possible to predict the natural frequency and ball pass frequencies. As a result it is not possible to directly implement a high-pass filter with a specific cut-off range. As a work-around, we propose an optimized algorithm, a flowchart of which is shown in Fig. 3. An FFT is implemented before enveloping the time-domain signal and information regarding the highest peak in the frequency spectrum is transmitted to the base station. A band-pass filter is then designed based on 80% of the energy of the highest frequency peak. The algorithm is implemented on this band-filtered data which results in a reduction of noise and corrupt signals. The optimization should be repeated from time to time to increase the efficiency of the algorithm. In effect, the cut-off frequencies should be recalculated whenever the frequency of the highest peak changes by more than 20% (compared to the last stored peak frequency) due to progression of defects over time.

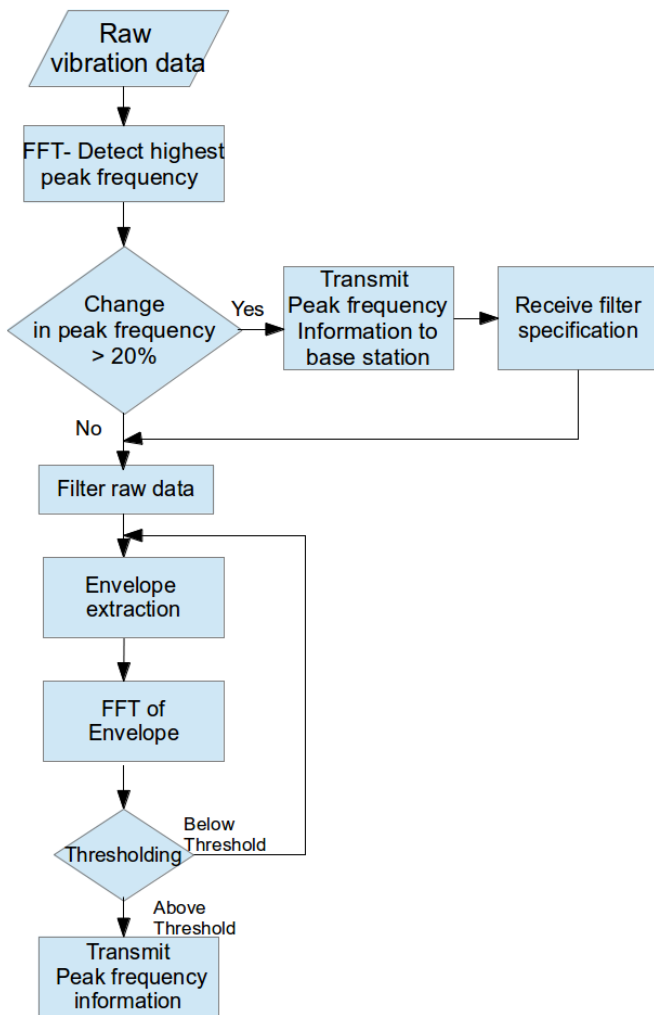


Fig. 3. Flowchart of the proposed optimized algorithm

V. ON-NODE IMPLEMENTATION

The V-Mon node has a DSP-enabled ARM cortex M4 processor [9]; however, the available memory and computational power are limited when compared to a desktop computer (PC). Signal processing blocks should be implemented on the ARM processor within the available memory and computational power.

Fig. 4 shows a flowchart of the algorithm that we implemented on a development board for the ARM cortex M4 processor, such as used in the V-Mon sensor node. This implementation was tested using the same data used for the MATLAB implementation on the PC. The data was transferred from the PC to the sensor node through an RS232 interface at 115200 baud to match the order of sampling frequency of 12 KHz with a 12bit ADC. A CMSIS signal processing library [10] provided by ARM Ltd. was used. An envelope extractor was implemented based on overlapping RMS. A buffer of 16384 samples is filled, and an RMS value is obtained for 32 samples with 50 % overlap. An FFT is applied over the 1024 samples acquired from the RMS-based envelope extraction. Only peaks above 60% of the maximum energy are buffered for transmission.

VI. COMPARATIVE STUDY

In order to test the accuracy and the power efficiency of the algorithm for on-node bearing fault detection, a comparative study was performed. Using the same dataset for both off-node and on-node signal processing, the accuracy of the algorithms and their power consumption were measured.

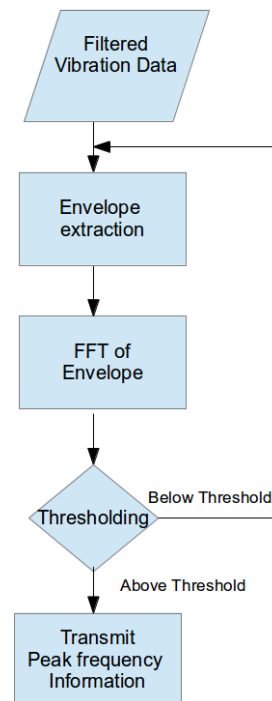


Fig. 4. Flowchart of the partial signal processing algorithm that was implemented on the node

A. Expected Outcome

The implementation of local DSP on sensor nodes is expected to effectively reduce the power consumption of sensor nodes.

The accuracy of on-node DSP is expected to be equal to that of the conventional vibration monitoring methods. The number of peaks detected in software analysis should be reflected in the on-node implementation. Thresholding will be effective if the accuracy of the signal processing results are similar to that of the conventional method.

B. Comparison Parameters

The comparison of different signal processing methods is judged on the time required for processing, and memory used. Furthermore, the off-node and on-node implementations are judged on their (estimated or measured) power consumption. MATLAB profiling is performed to estimate the time and memory required for the signal processing blocks. Uvision 4 IDE is used to identify the time and memory required for the signal processing blocks in the ARM processor.

In order to determine the reliability, the number of peaks that occurred in the MATLAB simulation is compared to that of the on-node implementation.

C. Dataset

The given vibration dataset contains four labeled data traces of a bearing setup, see Fig. 5. No information on the type of bearing and the mechanical setup was given; instead the required parameters such as fundamental number of rotations per minute (rpm), sampling rate, and specifications of the measurement device are given. The four different types of data include non-defective (pristine) bearing model, outer-race defective model, inner-race defective model, and bearing-ball defective model.

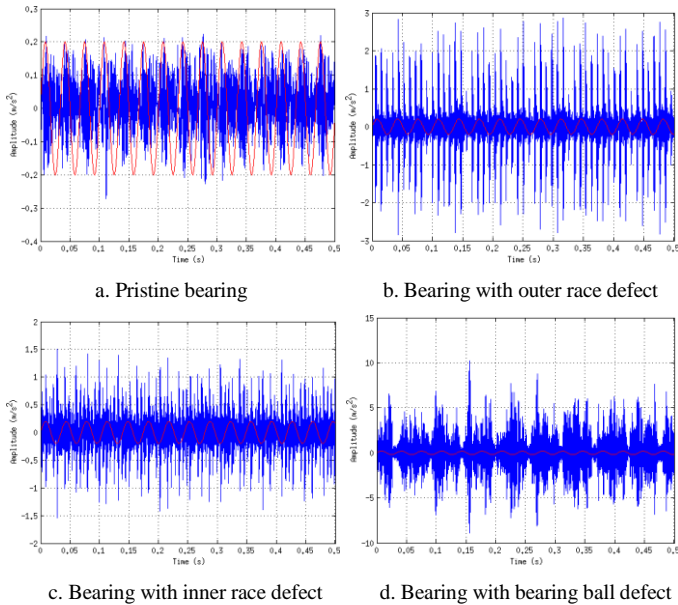


Fig. 5. Extracts from the data set. For easier comparison of the amplitudes of the time signals, a reference sine wave is shown in red.

D. Off-node implementation

Off-node implementation of the signal processing algorithm was done by means of simulating the monitoring algorithm in MATLAB running on a 3GHz CPU with 4 GB of RAM. The simulation provides an overview of the condition monitoring algorithm on implementation. Four different datasets, pristine, inner-race defect, outer-race defect, and rolling element defect are used for simulation. From simulation the defects have unique frequency peaks. The pristine condition of the bearing mainly has low frequency components and the high energy peak at 29.3 Hz represents the rotational frequency (1758 rotations per minute, RPM) of the machine. Due to the impulsive behavior of inner-race and outer-race defects, distinct high energy peaks are visible in the respective spectra. Peaks at three times and nine times the RPM frequency are visible in the race defect signals. If the harmonics of the RPM are present alongside with the race defective peak, it represents the inner race defect. If the harmonics are absent, it represents the outer race defect [7]. The rolling element defect can be clearly detected from the presence of high-energy equally-spaced harmonics of the RPM frequency. Progression of the frequency of the detected peaks with the same mechanical load indicates that a defect may soon develop. Progression in the energy of the frequency peaks represents an increase of the magnitude of the defect.

E. Measurement Setup

In order to measure the power consumed by the processor in implementing the algorithm, the setup shown in Fig. 6 was made. It is important to measure the power consumed by the processor alone excluding the overheads of the development board. A simple voltage regulator is developed to supply power to the processor and the voltage drop across the resistor R_1 is used to measure the current through resistor R_1 and the load resistance of the processor, R_L .

F. Measurement Results

As shown in Table I, the time required for signal processing matches the real-time requirement of the vibration analysis. One iteration takes around 4 ms while the CPU operates at 168 MHz.

Our measurements show that the power consumption of the processor is 157 mW for executing the signal processing algorithm. An overview of the processes in the sensor node and their respective (estimated) power consumption figures is given in Table II.

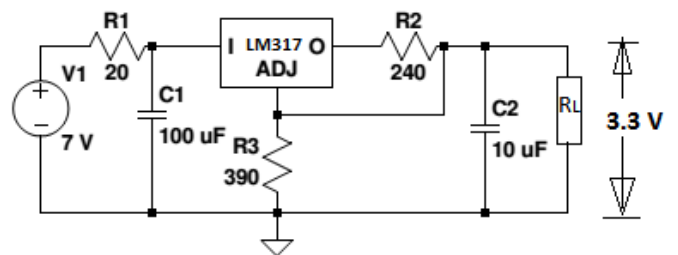


Fig. 6. Power consumption measurement setup

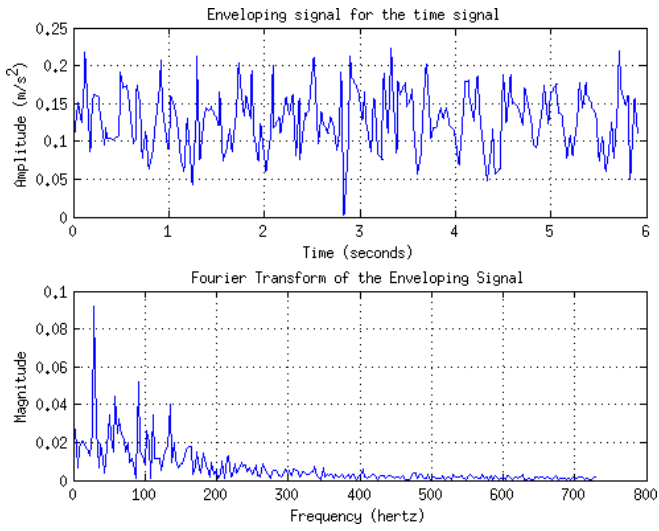


Fig. 7. Pristine condition

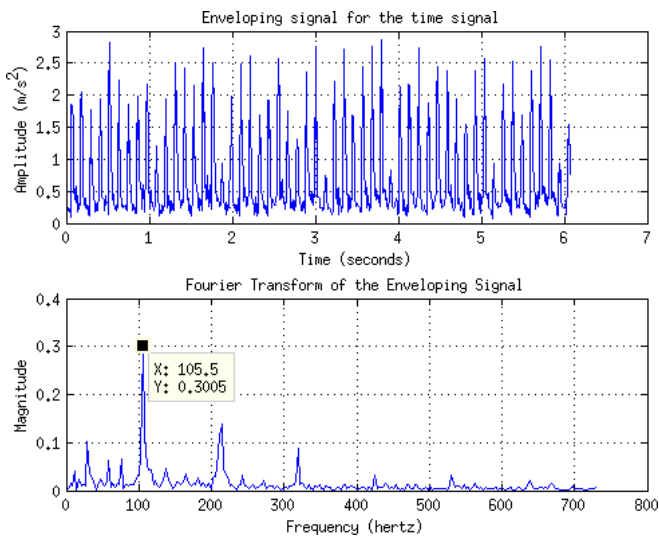


Fig. 8. Outer race defect

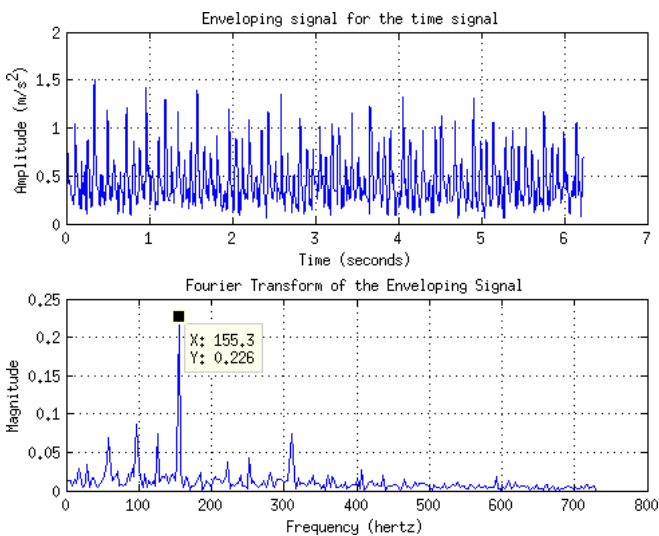


Fig. 9. Inner race defect

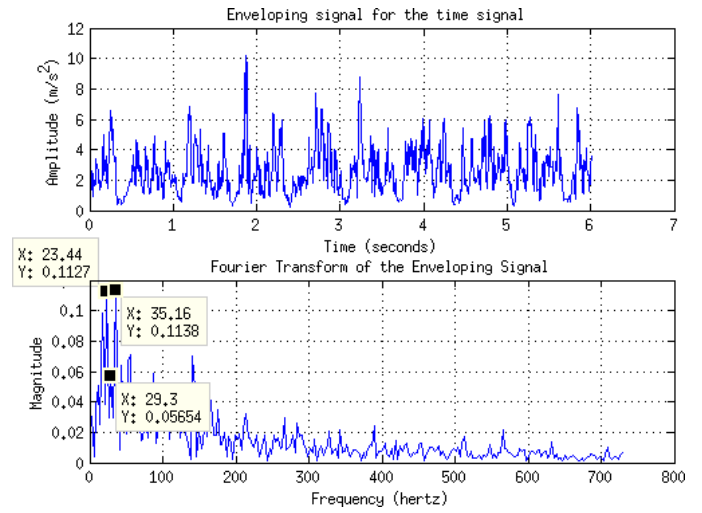


Fig. 10. Bearing-ball defect

The number of peaks detected resembles the software simulation results. Memory limitation of the microprocessor limited the number of data points to 16384. The CPU operating at 168 MHz was able to perform analysis in 4 ms, which is much less than the time required (1370 ms) to fill the data buffer by sampling at 12 KHz in 12 bit ADC, and thereby fulfills the real-time requirement of the rolling element bearing monitoring.

G. Discussion

The power consumption of 2.4 GHz wireless radio for continuously transmitting the raw data at 4 Mbps is around 100 mW operating at 3.3 volts [8]. Wireless radio in most of the sensor nodes do not have a transmission scheme and transmission and reception are just based on the availability of the wireless channel. If a channel is free, the sensor node begins the transmission. In combination with the processor the wireless radio results in 340 mW power consumption for a sensor node at maximum transmission power. In this implementation, instead of transmitting the raw data, only peaks and energy are being transmitted. If the processor is enabled with the same sensing process, but with reduced data resulting from signal processing to be transmitted at the same bandwidth, a power reduction of around 100 mW can be achieved. This gives a worst case of 29% power reduction when compared to conventional methods. A comparison of total power consumption of a sensor node with and without on-node signal processing is shown in Fig. 11. The third bar in this figure represents a best case of 38% power reduction, when further optimization is performed by reducing the number of peripherals of the ARM processor used for data logging and by avoiding overheads of the sensor node.

TABLE I. TIME AND MEMORY REQUIREMENTS FOR OFF-NODE AND ON-NODE SIGNAL PROCESSING

DSP functions used (16384 16-bit data points)	Time required in PC: off-node impl. [ms]	Time required in ARM: on-node impl. [ms]	Sensing specs of V-Mon sensor node	Node memory required (code+data+buffer) [kB]
Enveloping	0.001	1.4	12 kHz, i.e., 0.084	55
1024-point FFT	0.2	2.5	ms per sample,	124
Thresholding	0.00005	0.005	totalling ~1.37 s for	38
Overhead for function calls	0.09	0.09	16384 data points	N/A
Total routine	0.3	4	1370 ms	217

TABLE II. POWER CONSUMPTION OF PROCESSES IN A SENSOR NODE

Process	Power (conventional)	Power (on-node)	Power (best case)
	[mW]	[mW]	[mW]
Sensing	20	20	20
Radio	100	10	10
On-board voltage regulators	30	30	30
ARM processor with standard I/O peripherals	162	157	120
Overhead	30	30	30
Total	342	247	210

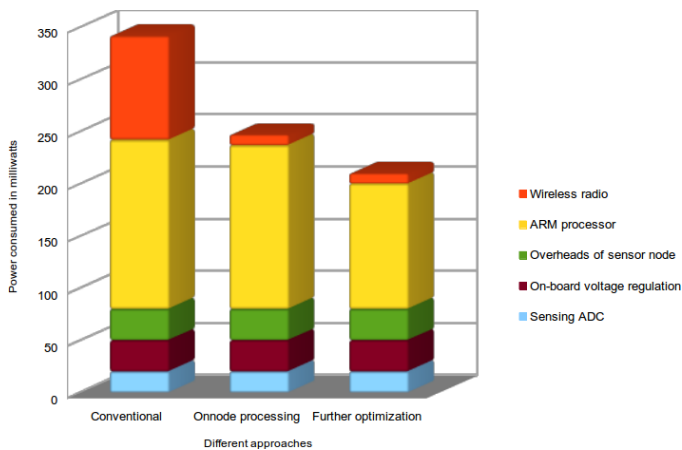


Fig. 11. Power consumption with and without on-node signal processing

VII. CONCLUSIONS AND FUTURE WORK

We implemented a partial algorithm for the analysis of vibration data on a sensor node. An algorithm based on conventional methods was adapted, implemented and validated. Based on the results from the hardware implementation, it is clear that on-node signal processing will result in reduction of power consumption by a sensor node. This work concentrates on rolling element bearing analysis, however, different vibration monitoring applications have different use cases ranging from bearings to rotor blades of a helicopter. Algorithms differ for different use-cases. This work implies that at least a part of such an algorithm being implemented in the sensor node will reduce the overall power consumption of the sensor node. In complex monitoring processes such as helicopter rotor blades, distributed sensor networks are required to monitor from different locations of the helicopter. In such cases, wireless radio cannot transmit raw data which would result in an ineffi-

cient sensor network. Instead critical parameters of the vibration should be analyzed locally on the node and distributed along the sensor nodes. Energy efficiency in such cases also depends on the wireless communication, and this can be improved by reducing the bandwidth usage of the radio.

ACKNOWLEDGMENTS

The authors wish to thank Mark Olieman of Inertia-Technology B.V and Niels Moseley of the Pervasive Systems research group for their valuable support in understanding the specifications of the sensor node and measurement of power consumption in the ARM processor.

REFERENCES

- [1] G. Merendino, A. Pieracci, M. Lanzoni, and B. Ricco, "An embedded system for real time vibration analysis," 4th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI), 28–29 June 2011, pp. 6–11.
- [2] G. Betta, C. Liguori, A. Paolillo, and A. Pietrosanto, "A DSP-based FFT-analyzer for the fault diagnosis of rotating machine based on vibration analysis," IEEE Transactions on Instrumentation and Measurement, vol. 51, no. 6, pp. 1316–1322, 2002.
- [3] C. Rodriguez-Donate, R.J. Romero-Troncoso, A. Garcia-Perez, D.A. Razo-Montes, "FPGA based embedded system for induction motor failure monitoring at the start-up transient vibrations with wavelets," International Symposium on Industrial Embedded Systems (SIES), 11–13 June 2008, pp. 208–214.
- [4] J.J. Rangel-Magdaleno, R.J. Romero-Troncoso, R.A. Osornio-Rios, E. Cabal-Yepez, and A. Dominguez-Gonzalez, "FPGA-based vibration analyzer for continuous CNC machinery monitoring with fused FFT-DWT signal processing," IEEE Transactions on Instrumentation and Measurement, vol. 59, no. 12, pp. 3184–3194, 2010.
- [5] C. da Costa, M.H. Mathias, and M. Kashiwagi, "Development of an instrumentation system embedded on FPGA for real time measurement of mechanical vibrations in rotating machinery," International Symposium on Instrumentation & Measurement, Sensor Network and Automation (IMSNA), pp. 60–64, 2012.
- [6] M. Kunli and W. Yunxin, "Fault diagnosis of rolling element bearing based on vibration frequency analysis," Third Int. Conf. on Measuring Technology and Mechatronics Automation (ICMTMA), 6–7 Jan 2011, pp. 198–201.
- [7] J. Tranter, "Detecting bearing faults (Part two)," Uptime, April/May 2011, pp. 58–61.
- [8] Inertia Technology, "V-Mon 4000 series product datasheet (preliminary)," 2013.
- [9] STMicroelectronics, "STM32Fx ARM Cortex M4-based microcontroller reference manual," available at <http://www.st.com>.
- [10] ARM, "CMSIS DSP library," available at <http://www.keil.com>.