A Low-Cost Water Flow Meter on the Edge using Machine Learning

Baptiste Solioz[1], Pierre-André Mudry[2]

[1]School of Engineering, HES-SO Valais/Wallis, Sion, Switzerland

Abstract

In this paper, we present a novel non-invasive water flow metering technique that is cheap and exhibits decent performance. Targeting mainly irrigation monitoring, the technique has been applied to create a prototype measuring apparatus consisting of a small, battery operated board that includes both a vibration and an acceleration sensor. Data acquired from those sensors is then processed on-board via a neural-network that has been pre-trained and calibrated in the lab. The inferred water flow rate is then transmitted via LoRaWAN to a data back-end for further processing. With this device, we demonstrated that for a total cost of less than 18 €, our prototype communicating sensor could run for a complete irrigation season on 2 AAA batteries with data sent every 20 minutes. Regarding the performance of this AI-augmented sensor, the results exhibit less than 10% of error for most flow rates when compared to a fully calibrated, lab-grade water flow meter, with potential for improvement.

Keywords

Water Flow Meter, embedded systems, edge computing, machine learning, IoT, LoRaWAN, low-cost

1. Introduction

The technologies related to water flow metering considerably evolved during the last twenty years, increasing reliability and repeatability of measurements. In addition, several new measuring solutions have been developed, notably for industrial pipes [1, 2].

In the domain of low-pressure water flow metering (under 6 [bar]), several low-cost (less than 30 €) digital solutions are available off-the-shelf. However, as soon as medium pressures are concerned – between 10 and 40 bars – the prices of sensors and systems increase massively due, among others, to more complex mechanical constraints.

In this paper, we demonstrate a novel approach to develop very low-cost (less than 18 €), relatively precise, LoRaWAN connected and battery-powered non-intrusive sensors to measure water flows. The key idea implemented and validated here to fulfill both price and quality of sensing constraints was to rely on relatively cheap MEMS sensors (an accelerometer along with a 1-axis vibration sensor) and analyse the results in the embedded system with pre-trained neural-networks models calibrated offline. Experimental results proved the feasibility of a sensor fitting the budget, having an expected battery life of more than 2 years with an acceptable accuracy for the considered use case in irrigation monitoring.

Proceedings of FTAL 2021, October 28–-29, 2021, Lugano, Switzerland

baptiste.solioz@gmail.com (B. Solioz); pierre-andre.mudry@hevs.ch (P. Mudry)

0000-0002-6926-2274 (P. Mudry)

1.1. Paper outline

We start our article in Section 2, with a brief but representative state-of-the-art of the methods currently available for measuring water flows using non-intrusive methodologies. In the next section, we describe the sensors principles and choices that were made at the beginning of the project. In Section 4, we explain the lab setup we used to generate data for training the machine learning algorithms. Section 5 concludes this paper with a presentation of the embedded systems specifics difficulties related to our approach as well as the experimental results.

2. State of the art of water flow meters

In order to situate our research in the field, we briefly introduce here the most relevant types of water-flow measuring techniques [3] for our research: two using intrusive sensors and five based on non-intrusive techniques.

2.1. Hall Effect Sensor

The first intrusive methodology – which is also very popular thanks to its ease of use and relatively low-pricing – is based on the Hall effect, which can be used to measure the rotational speed of a turbine wheel placed through the water flow. Rotation speed being proportional to water flow rate, the latter can then be deduced. The major disadvantage of this technique lies in the fact that high pressures or large pipe sections tend to complexify the design of the wheel.

2.2. Electromagnetic Sensor

The second intrusive sensor relies on a coil placed on the top and the bottom of the pipe to generate a magnetic field through it. Thus, the liquid inside the pipe will act as as conductor which induces a voltage which is proportionate to the average flow velocity. The voltage is then measured using two electrodes placed inside the pipe, allowing to determine the flow rate of the liquid (which must be conductive).

2.3. Ultrasonic Flow Meter

The use of ultrasonic waves to determine the flow rate of a fluid is widely used for industrial sensors. Two main approaches coexist: time-of-flight of the wave or Doppler effect [3].

Time-Of-Flight The time-of-flight approach requires two ultrasonic transmitters/receivers. Each of them will transmit a pulse of a given frequency through the pipe towards the second transceiver. Because the propagation of the signal is influenced by the speed of the liquid flow, so is the time between emission and reception in different directions. Using this information, it is then possible to calculate the flow rate according to the time difference.

Doppler Effect Doppler effect is the change in the wave's frequency in relation to an observer who is moving relative to the wave source. To use this effect for water-flow measurement, one transmitter and one receiver are placed side by side on the pipe. The transmitter emits an ultrasonic frequency which is reflected by the particles contained in the flow (bubbles or suspended solid). The receiver can then compute the flow velocity thanks to the Doppler shift which appears on the received signal.

2.4. Vibration Intensity Sensor

Vibration sensors like accelerometers or piezoelectric sensors can be used to determine flow rates. In fact, the contact of the fluid with the pipe's material generates vibrations which increase with the flow velocity. This methodology is based to the assumption that the turbulent regime generates higher vibrations, roughly proportionate to the flow velocity.

2.5. Impedance Sensing

This method uses the impedance variation of the flow in function of the velocity of the flow. Two electrodes are placed on the pipe and an AC voltage at a given frequency excites them. It is then possible to measure the current and then compute the impedance of the flow, which can be transformed to a velocity.

2.6. Heat Sensing

This method uses the fluids' thermal transfer principle to determine the flow rate. It is composed of two temperature sensors placed on both sides of a heater, the whole system being placed along the pipe. When there is no flow, a resistor heats the water locally, which causes an almost null temperature difference between the two sensors. With a higher flow rate, the heated fluid flow towards one of the sensors and the temperature difference is therefore higher.

3. Model building with accelerometers and vibrations sensors

In the search of a low-cost, battery-operated autonomous solution for water sensing, the analysis of the state-of-the-art pointed towards non-invasive technique, which have a lower bill-of-materials and easier installation on the field. Some previous works such as [4, 5] demonstrated that accelerometers could be used by a sensor to determine the flow rate, at least in laboratory. In addition, the works from [2] shown that vibrations sensors could also be used as vibration propagation measurements. Starting with the methods presented in those articles, our first goal was to replicate and improve the results obtained by acquiring more data. In a second phase, we developed a more elaborate model, coupling a vibration sensor and accelerometer as part as a complete module.

3.1. Sensors used, test bench

Some of the methodologies listed above do not fit the context of use in irrigation monitoring or are too expensive. Thus, the invasive methodologies are not interesting because the pipe must be modified, which is not feasible or too costly for a large number of pipes. Some invasive techniques are also difficult to apply to our use case: impedance sensing is highly correlated to the pipe's material – and therefore difficult to be widely used – and heat sensing consumes too much energy to be used on a battery-powered module. As a result, three candidate sensors remained to be explored experimentally and compared in order to choose the best candidate for our machine-learning approach.

Ultrasonic Sensor using Doppler effect Tests have been made with an evaluation module on a small pipe and the results were interesting. However, price estimations for such a solution could not fit the budget.

Accelerometers A small test bench has been developed using three accelerometers connected to a high-speed acquisition card. The data from three similar 3-axis accelerometers were analysed and several features were extracted. The most salient results are depicted in Fig. 1, which compares different flow rates with computed features from raw data provided by the accelerometers (with and without high-pass filter):

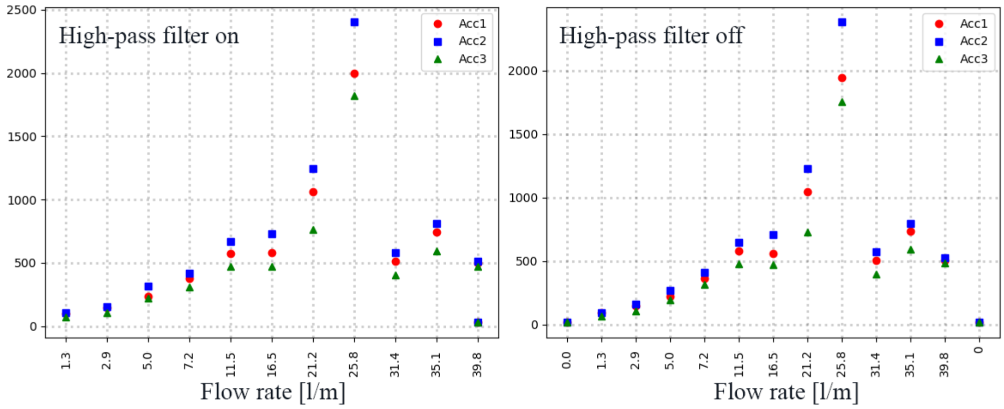


Figure 1: Arithmetic mean for 3 different accelerometers at various flow rates.

Piezo Sensing In a second phase, we replaced in our acquisition setup the accelerometer with a piezoelectric sensor, which exhibits faster acquisition speed and better resolution than accelerometers. In our case, the reference sensor we used (type *1005939-1* from *Measurement Specialities*), has a sensitivity of 1.1 [V/g] (6 [V/g] at the resonance frequency of 75 [Hz]).

Again, several features were extracted (raw data and FFT). We observed that the power intensity around the resonance frequency (75 Hz) varies proportionally to the flow rate, as depicted in Fig. 2, which shows a clear relationship between the RMS of the FFT for several flow rates. However, those results to not scale-up for larger flow-rates or larger pipes.

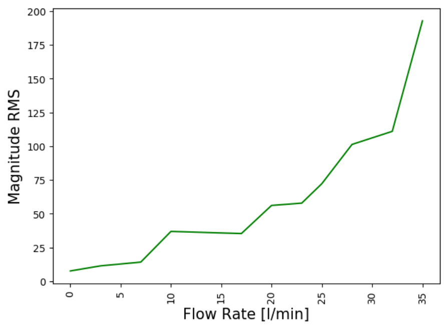


Figure 2: RMS of FFT transform, vibration sensor.

3.2. Developed hardware

As the preliminary results using piezo sensing were encouraging, a complete electronic module for vibration acquisition was developed. The processing unit used is a STM32WLE5xx module from *STMicroelectronics*, which provides the required peripherals like ADC or SPI but also an integrated analog front-end for LoRa modulation, enabling connectivity of the sensors to a network using LoRaWAN. In addition, an accelerometer has been added to handle the low-power side of the system, which is used to detect the large acceleration occurring at the beginning of the water flow through the pipe. This event starts the measuring phase.

3.3. Data acquired samples

In order to acquire many calibrated water flow samples as well as raw data, we integrated our sensor on a large hydraulic test bed. The setup consists of a big tank of water connected to a pump with a maximal flow rate of 50 $[m^3/h]$. The water flows through a reference irrigation pipe of 4.8 mm diameter where we mounted four of our developed sensors. After passing through the nozzle, water refills the tank in a closed loop. The final part of the setup is shown in Fig. 3.

As shown, four different sensors were placed on the pipe during data acquisition and the first and last modules were swapped at half of the acquisition process to avoid the influence of geometry (such as the bend in the pipe). With this setup, 112 calibrated measures were made for each type of acquisition and the four modules. Therefore, a total amount of **896 samples** of FFT and accelerometer raw data were acquired.

4. Machine learning methodology

4.1. Features and techniques tested offline

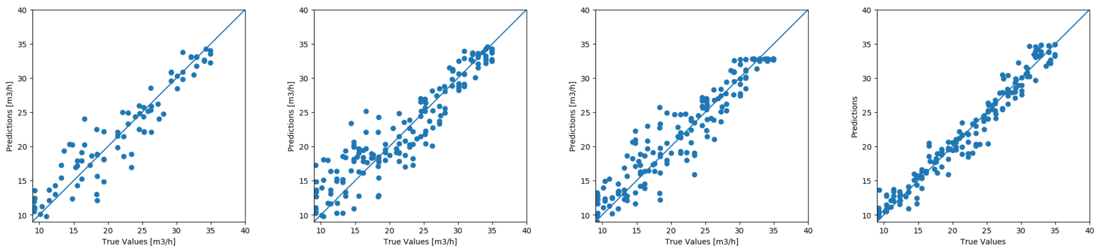
The objective of the machine learning phase is to produce a model to infer a flow rate given vibration characteristics. To develop this model, like in the sensor validation phase, the features of the acquired data set have been extracted, using signals converted to the frequency domain. That done, we performed a dimensionality reduction of the dataset using *Principal Component*

Figure 3: Experimental setup for vibration data acquisition.

Analysis (PCA) to retain only the more relevant components of the features for machine learning training. In addition, we also performed a normalization of the dataset (using zero-normalization techniques) because the features are differently scaled. The samples were split in several folds and 20 % of them were kept to test and validate the model.

4.2. Results on off-line machine learning

Our machine learning problem is clearly a regression due to the output values, a number which corresponds to a flow rate. According to the features previously extracted, three linear regression algorithms were tested. The hyper parameters of each of them were found using a grid search algorithm. Fig. 4a and 4b hereafter depict the results obtained with the two best algorithms.



(a) Random forest MSE 6.84 $[m^3/h]$

(b) KNN MSE 8.68 $[m^3/h]$

(c) Neural network MSE 6.56 $[m^3/h]$

(d) Convolutional NN MSE 2.41 $[m^3/h]$

Figure 4: Comparison of various ML regressors.

As the results with the first three approaches were not satisfactory, the usage of a neural network with features used as inputs to predict the flow rate was tried. For this neural network, multiple configurations have been tested, varying the number of layers, number of neurons, activation function, etc. The best result were obtained with two hidden layers of respectively

10 and 5 neurons, a *tanh* activation function for the hidden layer and *ReLU* for the output layer (Fig. 4c).

Considering the improvement obtained by using features as input, we also tested using the power spectrum provided by the FFT as input for learning. In addition, several convolution layers and multi-layer perceptron layers were added, using a kernel size of 6 for each convolution layer (which is the window size of the convolution's product). This architecture demonstrated better results than other models, with a mean squared error of only 2.41 $[m^3/h]$ (Fig. 4d).

With this model chosen and trained, a last phase was required for real-time measurements. As the chosen communication protocol (LoRa) can not handle a large transmission of real-time data, we had to be able to run the trained machine learning directly on the embedded system hosting the sensor, a method known as *edge-computing*, as we will describe in the next section.

5. Embedded application of machine learning on a prototype system

5.1. Transcription of the neural model on embedded system using CMSIS-NN

Given the power and cost constraints, the chosen embedded system had to host a LoRa front-end in a single chip. This implies that resources in terms of memory (RAM and Flash) and of CPU power are limited. For instance in the chosen SoC, no FPU is available but, in contrast, the system provides DSP instructions. In addition, because the embedded system is from the ARM™ Cortex family, we were able to use the CMSIS-NN ARM library [6] which is optimized to use DSP instructions with a fixed-point representation of numbers.

In order to translate the machine learning model (which was generated with Keras framework (https://keras.io/)) to a CMSIS-NN compatible C-code, a Python script has been implemented. This script quantizes weights, makes the appropriate conversions from Keras floating-point to the `Qm.n` representation and also performs several transformations related to number representations, overflow handling and accuracy verifications. The output of this script is is to generate the code usable on the embedded system according to the CMSIS-NN library.

5.2. Performance assessment in real-world experiment

Once the model has been implemented on the sensor itself, real-time and real-world performance tests were conducted. With this setup, measures with local inference of fluid velocity have been performed every 6 seconds and sent every 30 seconds through the LoRa network. A total of 94 measures were performed on the four modules, with 15 different flow rates (Fig. 5). Those results are considered good enough given the initial constraints, with an absolute error of 1,79 $[m^3/h]$ which are less than 10% (6,88%) of the measuring range (from 9 to 35 $[m^3/h]$). Only 25% of the measures have an error larger than 10%, mainly on low flow rates.

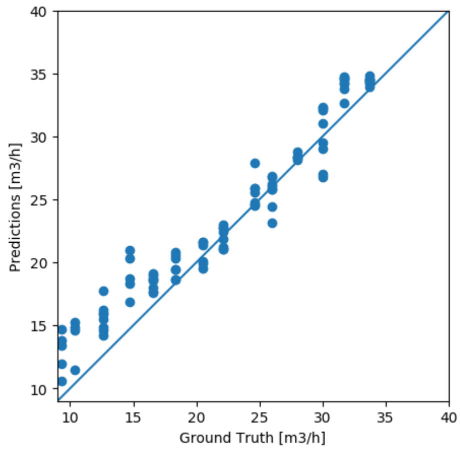


Figure 5: Real-world results with measure process on embedded system. MSE 5.31 $[m^3/h]$, mean absolute error 1.79 $[m^3/h]$

6. Conclusion

The approach explained in this paper is still in its infancy and the results were only demonstrated in the lab with some restrictions concerning the precisions of the measures as well as unknowns related to the genericity of the approach when it is applied to different hydraulic geometries.

Despite those limitations, we demonstrated in this paper the feasibility of a low-cost sensor for water-flow metering based on features extraction with machine learning on vibration data. In further work, we will study the generalization of the approach on different piping topologies and adapt the learning model accordingly. In addition, we will leverage the potential of improvement of the embedded machine learning models to attain better measurement accuracy.

7. Open-data access

The source-code of this work (hardware and software) as well as all the results and data is available on https://gitlab.com/baptiste.solioz/flowmeter.

References

[1] Lee S. H., Lee J., Chun S., Kang, W. "In situ measurement of micro flow rate using near infrared absorption method"'. Optics express, 26(13), 17078-17091, 2018.

[2] Flexim product leaflet, "Non-Invasive Hydrocarbon Products Mass Flow Measurement, Media Identification and Leak Detection", 2018.

[3] Zuzunaga A., Maron B., "A survey of non-invasive and semiinvasive flow meters for mining applications: Understanding and selecting the right technology for the application"'. CiDRA Minerals Processing, Wallingford, USA, 2013.

[4] Venkata S. K., Navada B. R. "Estimation of flow rate through analysis of pipe vibration". acta mechanica et automatica, 12(4), 2018.

[5] Dinardo G., Fabbiano L., Vacca G. “Fluid flow rate estimation using acceleration sensors“. In 2013 Seventh International Conference on Sensing Technology (ICST) (pp. 221-225). IEEE, 2013.

[6] Lai L., Suda N., Chandra V., “CMSIS-NN: Efficient Neural Network Kernels For ARM Cortex-M cpus. arXiv preprint arXiv:1801.06601, 2018