Reducing False Alarms with Multi-modal Sensing for Pipeline Blockage (Extended) *

ISI Technical Report ISI-TR-2013-686b

June 2013 (revised 3 July 2013)†

Chengjie Zhang
Information Sciences Institute
University of Southern California
Marina del Rey, California, USA
chengjie@isi.edu

John Heidemann
Information Sciences Institute
University of Southern California
Marina del Rey, California, USA
johnh@isi.edu

Abstract

Industrial sensing applications place a premium on cost-effectiveness and accuracy. Traditional approaches often use expensive, invasive sensors, because inexpensive sensors suffer from false positive detections. Sensor cost means automation is sparse or avoided when the value of specific sites cannot be justified. In this paper, we show combining different types of sensors can allow low-cost sensors to avoid false positives, enable much greater levels of automation in some applications. We explore this problem by studying a specific application: blockages in oil flowline common in cold weather. We use pipe skin temperature to infer changes in fluid flow, and combine readings with acoustic data to avoid false positives and be robust to environmental changes. We demonstrate that our approach is effective with field experiments. Finally, suggest that this approach generalizes to other classes of problems where false positives from one sensing modality can be resolved by multi-modal sensing.

1 Introduction

Sensor networks are used to collect data, detect problems, and take actions in the physical world. Small and inexpensive, sensornets can be easily deployed to address many real-world problems, from sewage pipe leakage detection [37], milling machine wear-out prediction [51] and live stock health monitoring [9].

In spite of their effectiveness in some applications, sensornet uptake has been slow in many industrial applications. SCADA systems today often employ traditional dedicated and often expensive sensors, or fall back on manual observations where automated sensing is not seen as cost effective. A challenge in use of low-cost wireless sensors is that simple sensing methods often create many false alarms when they are confused by noise or changes in regular operation.

In this paper we propose to use different kinds of sensors to distinguish real anomalies from false alarms. We select a main sensor that detects the anomaly but may be confused by changes during regular operation. We then add additional sensors that can distinguish actual problems from false positives, although they cannot detect anomalies alone.

Our overall goal is to identify classes of industrial applications where multi-modal sensing can resolve sensing ambiguities. In this paper we prove this claim in the context of a specific example: *cold-oil blockages* in flowlines in producing oilfields. A typical oilfield has many kilometers of distribution flowlines that collect crude oil extracted from wellhead pumpjacks, gathers the oil for measurement and accounting, and ultimately sends it to refineries. Distribution systems near the wellhead are often small, particularly in older fields. In cold weather, oil thickens because oil viscosity has an inverse proportional relation with its temperature. Oil may then interact with sand or other contaminants in the fluid, and with pipe sags or narrow fittings, resulting in blockages in the lines. Blocks cause production loss, and if left unresolved they can result in pipe leaks, damage to the flowline, or even to the pumpjack. After pipe being fully closed, it takes only tens of seconds for pressure to build up before some parts in line rupture.

Although the oil industry has explored several stand-alone sensors, current approaches are either unreliable or too expensive to install and maintain (Section 2.1). Although some fields contain thousands of wells where production lines are vulnerable to blockage, manual inspection is the most commonly used technique today.

Our insight is that multi-modal sensing can not only reduce the cost of detection of cold-oil blockages while avoiding false alarms. Automating sensing can provide much more rapid detection than current approaches. Rapid feedback is important because a shorter gap between blockage reaches critical level and alarm is signaled can minimize different losses, including environmental and equipment. We detect blockage by sensing temperature and acoustic signals. We infer flow interruption from pipe skin temperature, but in addition to blockages, many regular events change temperature, including automatic pumpjack shut-ins and diurnal environmental effects. We avoid false positives by comparing multiple temperature readings and by using acoustic sensing to monitor pumpjack status. We define our sensing problem and summarize our approach in Section 2.1.

Our experimental results focus on cold-oil blockage, but the principle of multi-modal sensing to avoid false positives applies to many other sensing problems. For example, Girod and Estrin suggest using video evidence to correct problems from obstacles in acoustic ranging [10]. In human motion detection, Stiefmeier et al. cross-segment data stream between different sensors, including inertial, vibration and

* This research is partially supported by CiSoft (Center for Interactive Smart Oilfield Technologies), a Center of Research Excellence and Academic Training and a joint venture between the University of Southern California and Chevron Corporation.

† Revisions in July include correction of typos and small clarifications in Section 2.1, 2.3, 2.4, 4.2, 4.4, 4.6 and 5.1.2.

force sensitive sensors [36]. We discuss more on generalizing our approach to other applications in Section 4.10.

The first contribution of this paper is to identify the opportunity for multi-modal sensing to reduce error rates with low-cost sensors. While some prior sensors have explored multi-modal sensing with expensive sensors (for example, cameras [10]) and PC-level computation (including mobile phones or laptops [3,51]), we believe we are the first to show these approaches apply to low-cost embedded sensors.

Our second contribution is to prove this claim by exploring a specific application: we design an embedded sensing approach that detects cold-oil line blockages using a combination of inexpensive temperature and acoustic sensors (Section 2), then test our specific implementation (Section 3) in the field (Section 4).

2 Design of Cold-Oil Blockage Detection Algorithm

Here we define the problem we are solving, then explore how low-cost temperature sensors detect blockage, acoustic sensors detect equipment operation, and the two together provide reliable blockage detection with a low false positive rate.

2.1 Problem Statement

The goal of this paper is to understand how sensing can assist industrial applications, and how multi-modal sensing can help avoid false positives. While in the abstract, multi-modal sensing is straightforward, the key question is understanding how *real-world* sources of noise and false detections affect sensing system design. To that end, we focus on cold-oil blockage as a real-world application.

The Problem: Cold-oil blockage occurs when the return line from a producing oil well becomes blocked, typically due to changes in oil viscosity as a result of cold weather, sometimes compounded by buildup of sand in the pipe.

Blockage typically build up gradually over time. Producing wells often operate intermittently with on/off cycles of 5-15 minutes (to allow downhole pressure to build up for suitable operation); when the pump is not operational, oil can transition from flowing slowly to blocked. A blocked pipe can cause equipment damage and oil leaks, since if well production continues with a blocked flow line pressure in the line will cause flow line rupture or pumpjack damage. Recovery from equipment damage can easily amount to ten thousand dollars per event, in addition to reducing production.

Cold-oil blockage is a significant problem in some oilfields. Figure 1 shows eight consecutive years of production data of an oil field where cold-oil blockage is a concern. We normalize production values to remove long-term decreasing trend in field production and show seasonal variation in production. The first step of normalization is computing monthly index by applying exponential decay fitting over the whole dataset. The resulting fitting error is 1.3%, low enough to show we have a good fit. The fitting forecast is the monthly index—the baseline of monthly production which is unaffected by the overall trend. Next, we normalize the raw data by its ratio against its index for every month and the result is in the upper plot. The lower plot summarizes, for each month, how often that month's production is below its index. It further shows that winter months (November through February) witness consistently lower production rates. There are multiple factors that contribute to this trend (scheduled maintenance is often planed to avoid hot summer months), but field engineers confirm that a significant factor to reduced winter production are well problems due to cold-oil blockage.

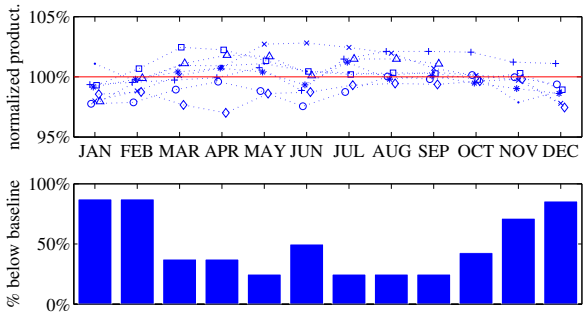


Figure 1. Seasonality analysis shows winter production loss.

Although we focus on cold-oil blockage so can evaluate real-world sources of error, in Section 4.10 we consider how multi-modal sensing applies in other applications.

Current Approaches: Aware of the problem, oil companies have explored current sensing approaches, including flowmeters, pressure sensors, leak detection, and of course manual inspection. Unfortunately, currently techniques for automation have high installation and maintenance costs. For example, pressure sensors cost US$1 000 or more to purchase and have annual costs of $300 or more to recalibrate; flow sensors are more costly. As a result, these sensors are used only on a few, very productive wells, while manual inspection remains common, in spite of its large delay.

These approaches suggest the importance of the problem, but sensor cost means they are deployed on only a few of the thousands of wells where there is concern. Field engineers confirm that in some cases, the alternative is simply to preemptively stop production on certain well that have no monitoring.

Degree of Blockage: Blockages build up over time, and one would like to detect them before they happen, or very quickly after they happen. Currently pressure sensors are deployed on only a few wells. Instead, manual inspection is done to identify equipment damage that follows a blocking, something that often occurs twelve hours after the fact.

Here we focus on *rapid detection of full and near-full blockages*. Detection of blockage allows well shut-in and recovery before damage; rapid detection after blockage may avoid equipment damage and will minimize leaks. Our evaluation (Section 4.7) shows we can detect blockage in 10 to 30 minutes, much shorter than 12 hours by current manual inspection. While not instantaneous, this detection is potentially able to save large production loss.

We emulate full blockages in field experiments (Section 4). We cannot test near-full blockages in the field due to safety concerns, but we do evaluate near-full blockages in

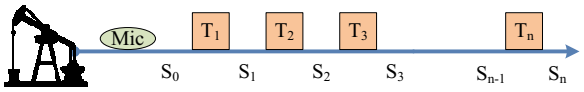


Figure 2. A diagram illustrates our problem statement. Square symbols are temperature sensors; the oval is an acoustic sensor. S_i denote different pipe sections.

laboratory tests (Section 4.9). The success of the both field and lab tests shows the generality of our approach on cold-oil blockage and even a broader range of applications.

Earlier detection of partial blockages (50% or less) would be helpful and is future work. However, our current temperature method is not enough to detect the subtlety. Possible future research could study how to implement sophisticated signal processing on sensor platform, and how to leverage the differential temperature due to pressure change before and after blockage.

2.2 Overview of approach

To detect blockages we use two sensing methods: acoustic sensing at the wellhead and temperature sensing at locations along the flow line (Figure 2). Typical flow lines are much hotter than ambient temperature (100 °C vs. 0–30 °C), particularly in fields that use secondary production techniques such as steam injection. We can therefore infer flow blockage by observing pipe skin temperature: the pipe downstream of a blockage will converge to ambient temperature. In operation we expect to place multiple temperature sensors along the flow line, near places where blockages are expected.

Unfortunately, pumpjacks often stop production ("shut in") periodically to allow downhole pressure to accumulate. Pumpjack shut-in causes drop in pipe temperature the same as a blockage, so temperature sensing alone will result in false alarms.

We therefore add a second acoustic sensor to detect pumpjack operation. And one acoustic sensor placed near the wellhead can provide pumpjack status for all temperature sensors on the production line. temperatures on the same line downstream. The acoustic sensor listens to flow in the pipe and the clanging of the pumpjack rods and tubing to detect pumpjack operation. (These sounds propagate well through the pipe, so acoustic sensor placement can be within 20 m of the wellhead.)

Our hypothesis is that our combination of temperature and acoustic sensing is both necessary and sufficient to detect cold-oil blockage.

Sources of noise: Although we focus on pumpjack operation as our main source of error, we must consider many sources of noise, from the environment, field, and measurement system.

Environmental noise contains *diurnal* and *seasonal* changes in weather and ambient temperature. Pipe skin temperature changes by a few °C over the course of a day due to changes in sunlight and wind or other weather. Our algorithm is insensitive to this change because the temperature difference between normal flow and ambient is much larger. Seasonal weather changes have a greater change, with temperatures that vary by 38 °C or more from the min in winter to the max in summer. However, this *long-term* change does not affect our algorithm because the detection threshold is hourly auto-retrained against recent pipe skin temperature, quickly adapting our algorithm in as short as hours.

Second, field conditions change: including *downhole* conditions, equipment *maintenance* and *main-line back pressure*. Downhole temperature and pressure changes as the field produces oil and due to changes in injection. These changes are generally slow (over days or weeks); our algorithm retrains hourly and so adapts to these. Valve close-up caused by maintenance indeed behaves similar to a real, sudden full blockage. We depend on field engineers to identify maintenance a possible source of false blockage detections. Finally, there will be some temperature propagation from the main line back to a blockage. We expect this effect to be minimal.

The last group is measurement noise, which is related to our deployment setting, including *sensor installation* and *random glitch*. If a sensor has a loose contact with the pipe, the readings are always a weighted average between ambient and pipe skin temperature. Poor connection will reduce our algorithm's sensitivity, but our tuning accounts for variations. We confirm in tests that our algorithm adapts to loose connections that cut the mid-point between ambient and normal pipe operation temperature in half, still finding the correct reference value and triggers on sub-20 °C drop.

From the discussion of three categories of noise—environmental, system and measurement, we conclude that our algorithm with parameter auto-tuning is robust enough.

2.3 Temperature Sensing for Flow Presence

Section 2.2 shows flow presence detection is the first part of our multi-modal cold-oil blockage detection. In this section, we talk about how to detect flow presence by temperature and how to automatically tune parameters.

According to our problem statement and hypothesis above, we need to measure pipe skin temperature to detect the presence of flow, or in another words, suggested blockage. Since the temperature usually drops gradually (about 20 °C in an hour), we need an algorithm to process streaming temperature trace and identify its trend of approximating ambient.

For the above reason, our algorithm employs one-sided CUSUM (or cumulative sum control chart [31]), originally a statistical technology developed for process quality control. The algorithm starts at low-pass filtering raw temperature observation (by EWMA) to filter transient noise. Next, it compares every observation to a reference value to calculate the deviation from it. Meanwhile, it maintains a running statistics, the cumulative sum of all the deviation in history as basic CUSUM does. In this paper, we call this cumulative sum of deviation *certainty of drop* (C_d). When observation is lower than k, C_d becomes larger and larger before it exceeds a threshold, which suggests a blockage because the temperature is too low for too long. We use one-sided CUSUM, resetting C_d when it is less than zero to respond quickly to temperature drops.

We must set two algorithm parameters: the threshold for certainty of drop, and the reference value (k). We set the threshold to 15 times normally observed temperature, in this

case 3 000, to be robust to transient temperature dips. The reference value, k is set as the mid-point between quality level—normal pipe temperature, μ_0 and anomaly level—flow stopped, μ_1 ($\mu_1 < \mu_0$).

Since k is important to the accuracy and responsiveness of the algorithm, we auto-tune it instead of hard-coding. However, due to different sources of noise we list in problem statement, we do not think predefined, fixed μ_0 and μ_1 estimation can best reflect an appropriate k. Hence, it is necessary to first auto-tune μ_0 and μ_1 for its dependency and we embed auto-tuning in our algorithm to adjust the estimation of the two levels. When pumpjack is operating (determined by acoustic node, introduced later in Section 2.4), we constantly update the quality level μ_0 by temperature observation. When pumpjack shuts in, we stop updating μ_0 but start anomaly level μ_1 updating as temperature drops. To generalize this, we are using a second sensory channel, to convert a false-alarm hazard into a helper of parameter tuning. By the time of the shut-in is over and pumpjack resumes operating, a new reference value k will be ready, based on auto-tuned μ_o and μ_1. Another k-tuning feature is that we do not update k at shut-in because during pump-off, temperature detection becomes less important. More importantly, we intend to avoid accidentally updating threshold to an inappropriate value.

2.4 Acoustic Sensing to Avoid False Alarms

Our discussion in Section 2.1 shows that temperature alone is not enough. Acoustic sensing on pumpjack status can avoid the false alarms caused by regular pump-off. In this section, we describe our acoustic algorithm design and next discuss how we automatically tune parameters in that algorithm.

We need to determine if pumpjack is *operating* for end pipe blockage detection. Since pumpjack stroke with engine rumbling generates wide band noise and propagates along pipe, we use microphone mounted on pipe surface to measure the sound pressure level (SPL), a high level of which suggests pumpjack operating. When pumpjack is off, microphones are expected to pick up much lower energy of environmental noise.

Our acoustic algorithm works as follows. First, for each stroke cycle C, we detect if pumpjack is on by comparing sound amplitude to a pre-configured threshold θ_p. If samples in C exceeds the threshold, mostly because of a significantly loud rod-tube clang noise associated with each stroke, we decide the pumpjack is on during the whole cycle (typically 7 s). However, simple pumpjack flip detection is not robust against transient error and hence we need to know if the pumpjack is *steady* on. In order to make that decision, we check a longer history to see if it *was being* on for a whole warm-up period W long, usually far longer than a single cycle.

Hence, to correctly detect pumpjack status, we need to properly configure three parameters: certainty of drop (C), warm-up period (W), and threshold (θ_p). We do tuning on base station because the training involves certain intensive computation as auto-correlation and memory storage complexity both beyond mote capacity; so we employ a PC in our experiments. (In principle a mobile-phone class processor could easily accommodate this work, although it is beyond 8-bit motes.) The on-site training step makes acoustic sensors robust against environment noise and mechanical difference across pumpjacks. We next describe our training algorithm for these three parameters, started by training data collection.

Before deployment, we collect a short period of acoustic training data containing both pump-on and -off. We next compute C by running auto-correlation over the pump-on trace. The lag yielding the largest coefficient represents pumpstroke cycle. To prevent from choosing harmonics, in implementation we search the highest coefficient in a possible-cycle range, say [5 s, 9 s]. Further, based on our prior study, W could be set as five times of C.

We consider both pump-on and -off to compute θ_p, because it needs to be able to properly denote the difference between those two status. We first compute the noise floor by averaging all the samples in pump-off trace. We next throw away all samples below noise floor in pump-on segmentation. θ_p equals the 86-percentile of amplitude among all the rest of the pump-on segmentation. The reason we choose this value for θ_p is that during a common 7 s pump cycle, our threshold should detect the single sample capturing the loudest rod-tube clang noise against other six under 1 Hz sampling rate. Therefore, the signature noise sample is likely to have a higher amplitude than the other 86% (six out of seven in one cycle) samples.

2.5 Sensor Fusion for Blockage Detection

We talk about the two algorithms in the above sections and next we describe how to fuse them to detect end blockage. If we interpret our basic hypothesis (Section 2.1) with technical details, we find that blockage could be detected as flow stops but pump is steady on. In another words, if pumpjack is off, our algorithm ignores all suggested blockage detection by temperature sensing, although the certainty of drop builds up due to stagnant flow.

On the contrary, if pumpjack is on, our algorithm can detect blockage, all in the following two different situations. If blockage occurs during pumpjack operation (*i.e.* pumpjack is steady on), we expect to witness a line temperature drop. As soon as line temperature stays below reference value long enough, blockage detection triggers. Besides, if blockage occurs during shut-in, after pipe cools off and pumpjack resumes, line temperature stays close to ambient and does not increase significantly. Hence, the certainty of drop can too build-up, followed by blockage detection. We evaluate the fusion result later in Section 4.7.

3 System Implementation

Before we review the details of our field experiment, we briefly talk about the implementation of our mote sensing platform with low-cost sensors. We first briefly summarize the hardware of our multi-modal sensing system. Next, we discuss the two challenges in acoustic node implementation and our software approaches to solve them.

3.1 System Hardware

Our sensor network consists of three types of nodes: base node for data collection, acoustic mote for pumpjack status detection and temperature mote for flow presence detection. In this section, we introduce the hardware of their parts.

Our base node is simply a Mica-2 mote [40] connected to PC through MIB520 programming board. It passively listens and logs all the packets transmitted from acoustic or temperature motes in the network.

Our acoustic mote is composed of a Mica-2 mote and an MTS310CA, "Mica Sensor Board" with an on-board electret condenser microphone, Panasonic WM-62A (Figure 3(a)). Figure 3(b) shows how we tape and clamp the extended microphone on the pipe with thermal insulation and we discuss how decoupling microphone benefits signal gain later.

We cannot directly mount Mica-2 microphone on pipe because the high pipe temperature may damage the equipment, or at least result in inaccurate measurement. Common electret condenser microphone has a sub-70 °C operation temperature, lower than the pipe skin temperature in operation. Although it is not mentioned in its specification [46], we believe this model on Mica sensor board, WM-62A is not designed for a higher temperature task. Even if it sustains the heat, electret condenser microphone has an unpredictable frequency response under high temperature (around 80 °C [49]). Therefore in deployment, we apply thermal insulation on top of the over-warm pipe to protect our microphone. The insulation is called Fire Blanket and is made of woven fiberglass. We are aware of some side effects of sandwiching insulation between the microphone and pipe, for example, signal attenuation. However, under the design principle of low-cost sensing, we decide to make this trade-off instead of employing expensive specially-customized microphones, say US$5 000-priced Brüel & Kjær 4949 automotive surface microphone.

Finally, the design principle of temperature motes inherits our prior work [52]. They each consists of a Mica-2 for control, a custom amplifier board to optimize thermocouple signal readings and a thermocouple sensor (NANMAC D6-60-J J-type) for pipe line and ambient temperature measurements. Figure 3(c) and 3(d) shows how we deploy them in our experiment.

During experiments, we were surprised to find that our custom amplifier boards are sensitive to their operation temperature, although all components are rated at a much higher range. Our initial field trials show if exposed under the sun directly, temperature sensors with the amplifiers sometimes return random readings, but sensors without the amplifiers work correctly. Hence in the latest test (Section 4.2), we covered the sensor motes in shade, but we are currently examining our design and seeking a more robust solution.

3.2 Hierarchical Sampling and Aggregation in Acoustic Mote

To obtain the sound pressure level of pipe, our acoustic sensor samples 2 000 times a second. This sampling rate is high for a mote, posing two challenges. First, although the sensor generate and transmit one packet per second, we cannot collectively stack 2 000 (one-second-long) samples in buffer due to the limited Mica-2 RAM size (4 kB for both program and data). The other challenge is that because the sensor samples at such short interval as 500 μs, hardware interrupts from other components (radio, flash logger, *etc*) are likely to cause large variation in sampling rate [14, 18]. For accurate sampling, we shut down all external components which might occupy the CPU for too long to hold up the timer. Hence, we design our software able to schedule and interleave processing, transmitting and flash logging among continual sampling. We do local flash logging because in operation, it could serve as backup in case of temporary network outages, although in a fully integrated system, data is always streamed back to a central server through field network.

We design a hierarchical sampling and aggregation scheme to overcome the two challenges above. Overall, we pause the high frequency sampling and schedule other operations, before next sampling cycle. The pause causes gaps in sampling, and in the worst case we may mis-detect interesting phenomenon. To minimize this sampling gap and coordinate data management, we make following design choices. At a high level, our sensor samples and computes the SPL within a one-second-long window (long window) before logging it to flash and transmitting it out. At an intermediate level, we divide each long window into ten 0.1-second-long short windows. In each short window, sensor samples for 0.06 s at 2 kHz rate, and uses the remaining 0.04 s to do SPL aggregation. The final 10^{th} short window does further aggregation by choosing the maximum SPL value among the past ten to represent the entire long window, before flash logging and radio transmission. Our lab testing shows 60%/40% duty cycle is optimal because a slightly more aggressive setting (*i.e.* short than 0.04 s gap) causes significantly more packet loss. Besides, the 0.04 s gap does not cause mis-detection on the 0.2-second-long signature rod-tube clanging noise.

3.3 Maximizing Acoustic Gain

To maximize the acoustic signal gain, we take three steps on software and hardware customization.

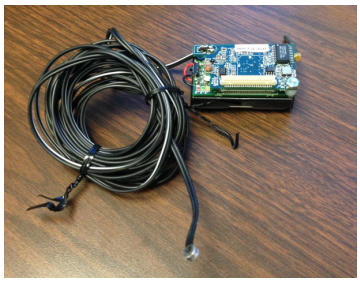
First, we optimally adjust digital current bias through calibration. The 10-bit ADC channel of Mica-2 returns values ranging from 0 to 1023 mapped to 0 to 3 V. As a result, it does not return negative voltage. To avoid losing the negative half of the waveform, MTS310CA is designed to elevate the center of the output acoustic waveform from 0 V to approximately 1.5 V, which corresponds to 512 in ADC value. We test this feature with our equipment and find that the new ADC waveform centers around 501, slightly off by the theoretical value of 512. We thus use our experimental result to offset mote ADC readings, removing DC bias.

Second, we decouple microphone unit from the board for better mounting. The flat Mica sensor board does not well match the curved pipe surface. Therefore, we desolder the microphone off and extend it out via wire, which enables us to simply tape it down to pipe in deployment for best contact and windscreen.

Finally, we use TinyOS to maximize the microphone analog gain. We use the OS service to tune an resistor in the amplification stage to its largest value, which is an on-board, digitally controlled, variable resistor [41].

4 Evaluation

We next describe the experiments we carried out to demonstrate we can detect flow blockage, and that multimodal sensing can avoid false positives. We first evaluate



(a) Acoustic mote with microphone extended

(b) Mote mic on a pipe

(c) Temperature mote packed in a box

(d) Thermocouple on a pipe.

Figure 3. Our temperature and acoustic sensor hardware and deployment.

our inexpensive sensors in the laboratory. We next describe our field experimental setup and evaluation metrics test how temperature and acoustic sensors can infer blockages and equipment operation, and finally show how their combination provides a robust system.

4.1 Calibrating Individual Sensors

Our premise is that low-cost sensors are sufficient to detect flow blockages. We next compare inexpensive mote-based temperature and acoustic sensors against high-quality PC-based sensors to confirm that inexpensive sensors are "good enough".

4.1.1 Temperature Sensor Measurement

We show our acoustic mote is close enough to ground truth in previous section. Next we compare temperature data by mote against USB data logger to verify if our low cost temperature collection solution performs well enough or not. The major differences between the two systems lies in hardware and calibration. The software processes are likely the same, although only limited information about USB data logger disclosed by its manufacturer.

In our prior work [53], we find that in relatively low temperature range (0–200 °C), it is unnecessary to calibrate J-type thermocouples before we deploy them in detection tasks. Hence, our mote reports raw ADC readings while USB data loggers are pre-calibrated by manufacturer.

Our mote is almost equivalent to USB data logger because of the strong correlation between them pairewisely. It is 0.91 on T_u, 0.80 on T_d^1 and 0.83 on T_d^2. We further visualize one data set, T_d^2 as an example to better demonstrate that this claim. Figure 4 clearly shows that the data by mote is merely off from ground truth by a constant but the fluctuation is almost the same. For clearer comparison, we post-facto'ly convert raw ADC reading by mote to Celsius scale under following equation [53]:

$$T = 18.259 \times \frac{ADC \times 3 \times 1000}{2^{10} \times \beta} + 2.852$$

where β is 367 as the gain of our pre-amplifier board. In our detection task, the algorithm is more sensitive to temperature drops instead of the absolute value, and hence a constant disparity is acceptable.

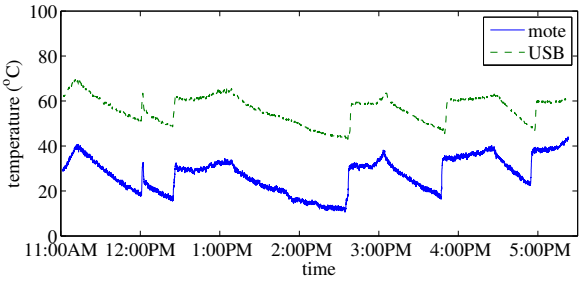


Figure 4. Temperature measured by mote and USB data logger at T_d^2.

4.1.2 Acoustic Sensor Measurement

We first look into our acoustic mote. Before we compare mote and PC acoustic sensors, we briefly describes their components and the difference in the data collection approaches. Our acoustic mote is consist of Mica-2 mote, an electret condenser microphone and a Mica sensor board. Whereas, our PC acoustic system is equipped with more powerful hardware—a laptop with sound card complying to Intel high definition audio architecture and a battery-powered lavalier microphone. The hardware superiority of PC system alone is enough to justify its cleaner data.

Other than the hardware difference, the second major differences is sensor installation. Although both sit on top of insulation, we tape down mote microphone by duct tape to pipe while we clamp on PC microphone by a customized clasp and hose-clamps, likely to produce larger force to press the microphone against the pipe for a better contact.

Finally, there is difference in sampling and aggregation mechanism in software after we abstracted out the OS differences, although the final packet rates of both for evaluation are the same, 1 Hz. As we described in Section 3, the raw sampling rate on motes is 2 kHz and we next use a hierarchical aggregation to assemble one packet every second from ten 0.1-second-long short packets. On the other hand, the raw sampling rate on PC is 16 kHz, much high than what we have with motes, mainly because we plan to keep high quality ground truth data in case we need to investigate the frequency domain of acoustic signal. Since the software we choose, Audacity, does not support a sampling rate as low as 1 Hz, and more importantly, we prefer to maintain the consistency between both systems, we re-sample our PC data in

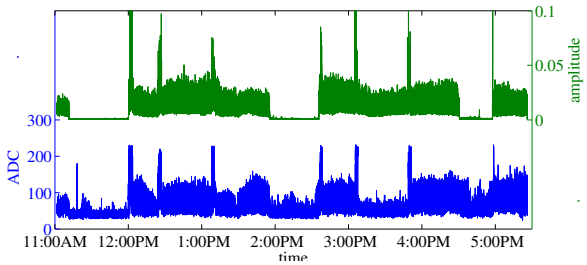


Figure 5. Acoustic measured by mote and PC microphone.

2 kHz by the same software before we further aggregate it by one-second-long window.

Despite the differences we listed above, Figure 5 shows that our mote data is close enough to the ground truth. It is difficult to directly convert their units; hence we keep both trace in their raw units and hand-scale them in the plot. The correlation coefficient between the two traces is 0.44, proving a strong positive correlation. The other observation that PC data has higher SNR, which is depicted by much higher state transition spikes and near-zero pump-off noise detection.

In all through the comparison, we show our mote data is close enough to PC data by a more expensive hardware suite. This result further supports our hypothesis above that low cost sensor is capable of reaching effective yet economical sensing.

4.2 Field Experiment Approach

We next evaluate our system in field tests. From 9:30am to 5:30pm, November 7th, 2012, we evaluated our system and a producing oilfield in the California Central Valley, working with field engineers from our research partners who operate that field. During the nearly seven-hour-long experiment, our system collected acoustic and temperature traces, did in-node processing and ran the full detection algorithm. We also collected ground truth data, concurrent with operation of our experimental system. Ground truth temperature and acoustic data employed USB thermocouple data loggers (EL-USB-TC [45]) and a laptop computer with a commodity microphone. Figure 6(b) shows the test site and the producing wellhead.

Our experiment emulates oil blockages by controlling valves. (We are not able to inject actual blockages, nor was it the time of year when they would form naturally.) Figure 6(a) illustrates the topology of sensors, pumpjack and valves. Oval shape represents acoustic mote and squares do temperature ones. T_u is located before the production-circulation branch-out and so upstream to both valves. T_d^1 and T_d^2 are both on production line and straddle production valve, downstream to T_u. To emulate blockages, we activate the production valve to close because it is not practical to create a real blockage in the field. When we close the production valve, oil stops flowing in the pipe and hence we observe a total blockage in line with the valve. In our experiment, we always leave open either the production or circulation valve, since closing both could cause high pressure at the wellhead that would damage the producing well or equipment.

Table 1. Experiment schedule and scheme.

start	pump	product. valve	purpose
11:01am	on	open	$T_d^{1,2}$ learn μ_0
11:11am	off		all learn μ_1
12:01pm	on	close	$T_d^{1,2}$ non-op
12:25pm		open	$T_d^{1,2}$ learn μ_0
1:09pm		close	$T_d^{1,2}$ in-op
1:54pm	off		all learn μ_1
2:35pm	on	open	$T_d^{1,2}$ learn μ_0
3:05pm		close	$T_d^{1,2}$ in-op
3:48pm		open	$T_d^{1,2}$ learn μ_0
4:30pm*	off		all learn μ_1
4:58pm	on		$T_d^{1,2}$ learn μ_0

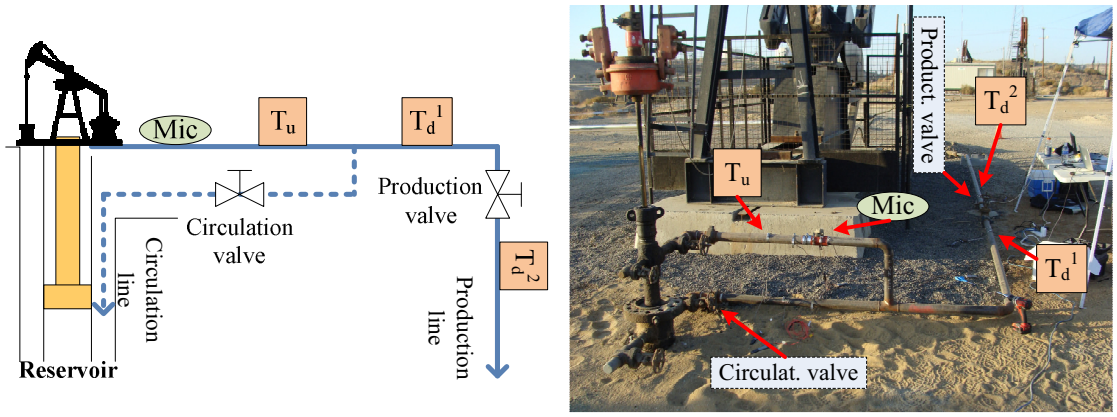
We conduct experiments on approximately half-hour intervals to allow the system time to stabilize between changes. Table 1 shows our schedule, with three pump-off periods for all four temperature motes to learn μ_1 and update k, with the last one (28-minute long) ran shorter than the first two (each about 50-minute-long) due to time constraint. According to the blockage introduction in Section 2.1, in reality we may generally categorize blockage in to two types regarding how it is formed. One is caused by a lump of viscous oil or sand clogging narrow fitting during pumpjack operation (*op*), an *in-op* blockage. The other is caused by residue oil in pipe cooling off and turning solid during shut-in before pumpjack resumes operation, a *non-op* blockage. To better evaluate the generality of our algorithm, we simulate both types in three instances over the course of the day, and each stage runs between 24 to 45 minutes. The simulations are interleaved with other two types of stages. One is valve-open and pipe temperature rebounce, so sensors can learn normal pipe temperature μ_0 during operation. The other is pumpjack shut-in, which configures the sensors' CUSUM anomaly level μ_1 (*i.e.*temperature on stagnant flow).

In addition to this field test, we carry out two prior field experiments where we evaluate components of our system and collect ground truth data for analysis in the lab. Prior tests were done at a different wellhead. We omit this data here due to space, but replay of this ground truth data in the lab shows our system works correctly on another well with different sensor locations.

4.3 Evaluation Metrics

Section 2 shows that we detects cold-oil blockage by filtering out irrelevant flow absence with acoustic pumpjack status detection. Before evaluation, we describe below the temperature and acoustic detection metrics due to their similarity.

We evaluate both temperature and acoustic sensing in an event-based manner, but with separate event definition. For temperature, one event is one interval between changes of equipment setting, because we care about if flow presence



(a) Logical view of deployment.

(b) Physical view of deployment.

Figure 6. November 2012 field deployment.

detection triggers or does not trigger *eventually* in certain conditions. Each event starts with setting change, including valve close-up/open and pump-on/-off and ends with another change, retaining the same setting across the entire event. According to the schedule in Table 1, we divide our experiment after 11:11am into ten stand-alone events. We discard events less than ten minutes (the period between 11:01am and 11:11am) because our algorithm requires 15 minutes to stabilize and our algorithm is still learning parameters. Hence, we first define metrics for temperature detection to denote the correctness of flow presence for each event: a *True Positive* (*tp*) is when flow stops, due to either pump is off or a valve in-line is closed, and the algorithm triggers during the whole event; a *True Negative* (*tn*) is when flow is normal and the algorithm does not triggers anytime during the event; a *False Positive* (*fp*) is when flow is normal but the algorithm incorrectly triggers; and a *False Negative* (*fn*) is when flow stops but the algorithm incorrectly keeps silent. And we define overall accuracy using terms from information retrieval [33]:

$$Accu_{all} = \frac{tp+tn}{tp+tn+fp+fn}$$

Contrary to temperature, the event in acoustic evaluation is defined by a sample (*i.e.*one-second-long sensor reading), because we care about instantaneous pumpjack detection. We use similar ways to define the four terms (*tp*, *tn*, *fp*, and *fn*) out of the pairwise combination between pump-on/-off and algorithm output-on/-off. For example, it is a *tp*, if pumpjack is on and the algorithm correctly declares it on. We inherit the same equation to compute overall accuracy as in forgoing temperature metrics. We can then define accuracy of pump-on and -off events using subsets of these measurements:

$$Accu_{on} = \frac{tp}{tp+fn}$$

$$Accu_{off} = \frac{tn}{tn+fp}$$

In our experiment, our acoustic node log 23 129 valid samples. Among them, 15 310 are *tp*, 4 893 are *tn*, 2 227 are *fp* and 699 are *fn*.

After we define metrics, we next evaluate our temperature flow presence detection, followed by acoustic pumpjack status detection. In addition to $Accu_{all}$, we care about $Accu_{on}$ and $Accu_{off}$ in acoustic sensing for future algorithm improvement.

4.4 Accuracy of Flow Presence Detection

We carry out full system deployment with algorithm online in the field for both testing and data collection. In order to incrementally test each component in the system, we ran a manual version of temperature algorithm in parallel with the fully automated version on each mote. The sole difference between the two versions lies in the process of parameter auto-configuration. we remotely re-program the temperature motes to inform the manual version of the perfect pumpjack status, while the automated obtains an imperfect update from their peer acoustic mote. We use the manual version to evaluate flow presence detection, while the fully-automated is for blockage detection evaluation in Section 4.7.

The three plots in Figure 7 shows that our CUSUM-based, flow presence detection algorithm works perfectly during our field test. We achieve 100% accuracy for the all ten events without any false positive or negative and we discuss more detailed observation below. Since T_u is upstream to both production and circulation valves, oil flows as long as pumpjack is on, regardless the status of the production valve. Figure 7(a) shows our algorithm remains silent while pump is on and temperature drops upon the first two pump-off events effectively triggers our algorithm. In addition, our algorithm successfully detects temperature drop caused by in-line valve close-up, showing in Figure 7(b) and 7(c), where C_d builds up at all "valve:closed" events.

We expect to see a trigger in the third pump-off event (4:30pm–4:58pm, marked "*"), but surprisingly C_d does not build up high enough in the algorithm, different from the prior two pump-off events. We still count that a true positive rather than a false negative, because it is a result of our compressed experimental schedule—we ran out of the time at the end of our experiment and hence we cut-off the third pump-off event prematurely. The trend shows we require 5

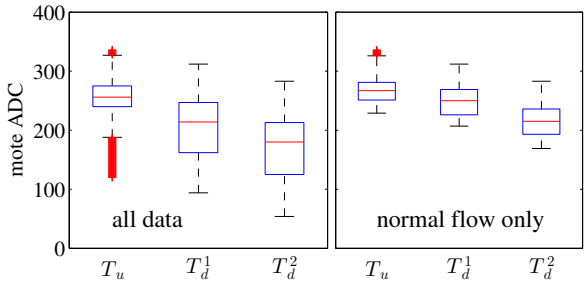


Figure 8. Two box-plots illustrate the difference in temperature between three location. The blue boxes cover both the upper and lower quartile with a red median mark in the center. The whiskers extend to 1.57 interquartile range, excluding outliers (red '+' marks).

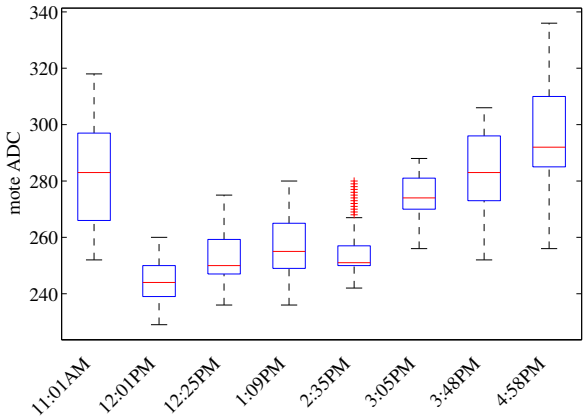


Figure 9. Box-plot illustrate the fluctuation of temperature when flow is normal at T_u during the experiment. The x-axis tick labels are the starting time of each event.

additional minutes to trigger, and since operational conditions do not have a 30-minute time limit, we count this event as correct. Two evidences are that the temperature still maintains a steep drop trend and C_d does *start* to build up at T_d^1 and T_d^2.

After comparing the results from the three temperature sensors, we make four observations. First, our approach achieves repeatable detection results, because the results are consistent across all three sensors. Second, one sensor is enough to cover a large pipe segment for detection, because it can detect blockage upstream *and* downstream to it. Temperature traces before and after the valve, T_d^1 and T_d^2 is almost equivalent, shown by a high positive correlation (0.98) and low standard deviation in differential (1.5 °C). Therefore, in our case either of them is enough for the pipe section after production-circulation branch, at least 20 m long to the next junction. Three, our sensor placement shows minimal recirculation from the main line since the temperature falls even downstream of the blockage (both T_d^1 and T_d^2 show similar temperatures). Finally, the different responses upon blockage between T_u and the other two shows multiple sensors can be used to locate blockages by distinguishing pipe locations with and without flow.

In this section, we conclude that flow presence detection with auto-configuration achieves perfect result in field tests. Therefore we next investigate the necessity of auto-configuration by cross-comparing between the three datasets.

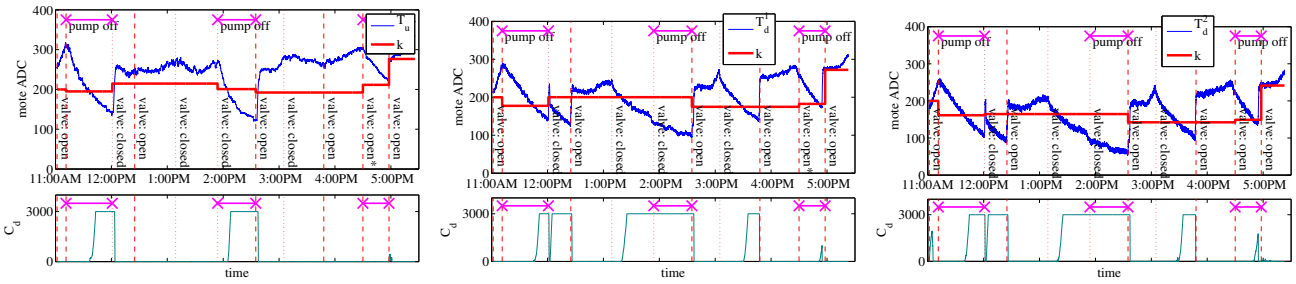
4.5 Auto-Configuration of Temperature Measurement

In the previous review, we demonstrate that our flow presence detection is accurate with parameter auto-configuration on quality and anomaly levels. We next show the need to auto-configure the parameters of our temperature algorithm, and that with auto-configuration deployment is robust to different wells and conditions.

We first show that baseline temperatures vary at different pipe locations and times, and therefore we require different tuning parameters for different locations. The left graph in Figure 8 shows the basic statistics over the entire 6.5-hour-long temperature data, while the right one focuses on the temperature under normal flow, excluding no-flow periods caused by either pump-off or valve-close. We know that temperatures of pipes vary, and they are affected by ambient temperature as well. With only one day for field experiments, we cannot allow the pipe to completely cool. In addition, this statistics is not necessary because the overall and the normal-flow statistics are enough to prove that temperatures vary. We find that the temperature upstream to production-circulation branch-out (T_u) is significantly different from the pair straddling the valve (T_d^1 and T_d^2), showing by the different quartiles (blue boxes) in either plot. Hence, auto-configuration is critical because one parameter setting works on one location does not necessarily work on another. For example, a reference value of 229 (ADC value) gives 100% accuracy on T_u, but would trigger three false positives on sensor downstream to production valve (T_d^2).

Additional motivation for auto-configuration is that temperature changes constantly at the same location. Therefore hand-tuning parameters on each sensor to cope with the location disparity is still insufficient. Figure 9 breaks down the T_u temperature trace and compares between eight events only when flow is normal. The temperature measurement fluctuation is significant, mostly caused by a noise combination of diurnal amplitude, downhole change and back pressure (Section 2.3). The first and last three boxes has no overlap with the other four, which suggests that maintaining a fixed threshold during the whole time is likely to cause misdetections. Our further study confirm with this observation and hence an adaptive temperature algorithm is necessary.

To address the above problems, our flow presence detection auto-configures the most important parameter—the CUSUM reference value based on the training temperature under normal flows and pump shut-ins (details in Section 2.3). Our 100% accuracy shows it is effective (Section 4.4). More importantly, we find it is *necessary* because if a uniform reference value were mis-configured above 229, sensor downstream to production valve (T_d^2) would start to trigger false positives.



(a) Upstream T_u. (b) Downstream before production valve T_d^1. (c) Downstream after production valve T_d^2.

Figure 7. Flow presence detection results.

4.6 Detecting Pumpjack Operation

Previous evaluation shows flow presence detection by temperature is effective. However, inferring blockage solely on flow absence is not enough because regular pumpjack shut-in too stops fluid flow (details in Section 2.1). To avoid false alarms caused by these irrelevant temperature drop, we use acoustic sensing to detect pumpjack status and later apply the result on top of temperature. Next, we evaluate acoustic algorithm accuracy and draw conclusions based on the results.

We first evaluate the overall, pump-on and -off detection accuracies. The overall accuracy is high, 20 203 out of 23 129 events (defined by samples in Section 4.3) are correct (87%) and the accuracy of detecting pump-on is even higher, 96%. However, the accuracy of detecting pump-off is 69%, which is low compared to the other two metrics. Next, we visualize the algorithm output trace to investigate why pump-off detection only works partially.

Pump-off detection is much less accurate than overall and pump-on detection. To understand the difference, we need more information about why many pump-off samples trigger pump-on detection. Figure 10 visualize the details by showing the acoustic amplification trace and algorithm outputs on mote. A red thick line in the upper plot indicates the threshold (θ_p) we used in our field test. The high spikes follow every valve status change is because the relatively loud noise generated by wrench-valve clanging is captured by the acoustic sensor. One reason pump-off detection is not effective is that our acoustic algorithm runs with fixed threshold. For example, the noise floor rises in the third pump-off period (4:30pm–4:58pm), triggering false positive under now-too-low threshold. We are currently working on making our algorithm adaptive to cope with this situation. In addition to an adaptive algorithm, this result suggests that filtering noise during pump-off may too improve pump-off accuracy (or overall). To verify if noise-filtering helps, we next apply our algorithm to a cleaner dataset by PC.

We find the easiest way to improve the accuracy of pump-off detection is to upgrade the hardware for acoustic measurement, because the same algorithm works perfectly on PC acoustic dataset. Figure 11 shows the same experiment as forgoing but collected by PC microphones. We take six minutes of the data, covering the first pump-on/-off transition to configure the threshold, with the same auto-configuration algorithm. The end detection result is encouraging, with 100%

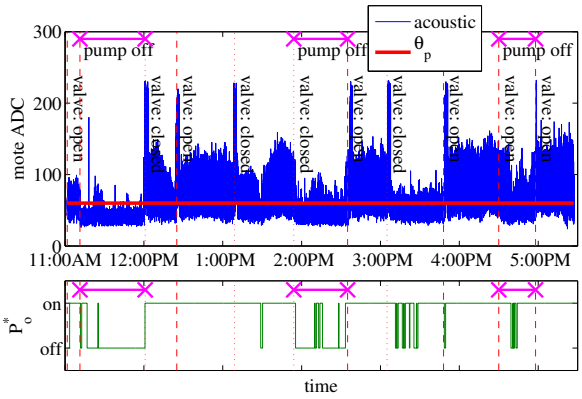


Figure 10. Acoustic pumpjack status detection result by mote. Pump status is denoted at the top of both plots. Dashed/dotted vertical lines indicate the production valve is open/closed. The tags "valve" means the production valve.

accuracy. We study the hardware difference helping PC microphone collect better data with higher SNR for future mote improvement guidance. We focus on the three most important stages in line with converting raw vibration to digital signal: microphone, preamplifier and analog/digital converter (ADC). After comparing the specifications, we find that PC beats mote microphone in all the three stages. First, although mote microphones have higher sensitivity (-45 dB, [46]) than the one with PC (-54 dB, [44]), but PC microphone has a much smaller resistance (1 kΩ against 2.2 kΩ, potentially able to produce larger current under the same sound pressure. Second, our PC has better preamplifier embedded in its sound card, SoundMAX AD1988A [43] than mote does [42]. For example, the former has higher input impedance but lower total harmonic distortion. Finally, the ADC on the microcontroller of our motes have a much lower resolution, 10-bit comparing to PC sound card's 24-bit resolution. In short, all these differences results in better quality of PC data with larger SNR than the ones collected by our acoustic mote system.

In this section, we evaluated our acoustic mote for pumpjack status detection. We further applied the same algorithm to PC acoustic data to provide more complete verification of our acoustic detection approach. Based on experiment results, we conclude that our mote acoustic detection system

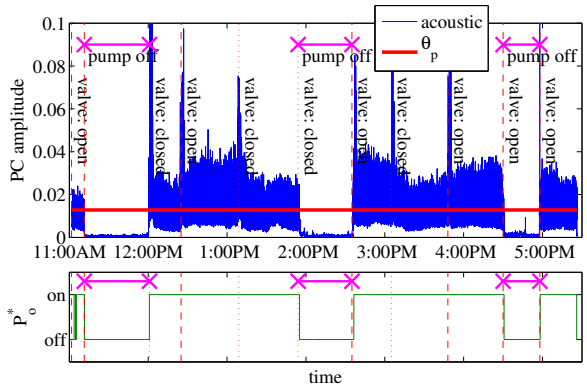


Figure 11. Acoustic pumpjack status detection result by PC.

Table 2. The accuracies of blockage detection.

location	correct *tp*	correct *tn*	incorrect *fp*	incorrect *fn*	$Accu_{all}$
T_u	–	8	2	0	80%
T_d^1	3	6	1	0	90%
T_d^2	3	6	1	0	90%
	total event#: 10				

performs reasonably well and it is easy for our algorithm to achieve perfect accuracy on a cleaner dataset. We next evaluate the end blockage detection on top of both temperature and acoustic sensing.

4.7 Blockage Detection Accuracy

In order to review our end blockage algorithm performance, we deploy a full system in our field test. The blockage detection fuses both pumpjack status and flow presence to determine if pipe is clogged or not. In the section, we evaluate its accuracy after defining the metrics below.

Like temperature and acoustic sensing (Section 4.3), we define the metrics of blockage detection in a similar event-based manner. Since blockage detection is mostly based on temperature sensing, event is likewise defined by experiment interval. The terms to denote the correctness is as follows: a *True Positive* is when the algorithm correctly declares an emulated blockage; a *True Negative* is when flow is normal *or* the pump is off, and the algorithm remains silent; a *False Positive* is when flow is normal *or* the pump is off, but the algorithm incorrectly declares a blockage; a *False Negative* is when the algorithm mis-detects an emulated blockage. We then accordingly defines the overall accuracy, $Accu_{all}$.

We first evaluate $Accu_{all}$ after fusing both temperature and acoustic results. Table 2 shows that overall accuracy of our fully-automated system is between 80% and 90%. This result further shows our blockage detection algorithm is very accurate. This table suggests two further observations. First, all temperature drops caused by blockages are correctly detected, because no false negative occurs across all three sites. This sensitivity of our blockage detection algorithm to temperature drop is consistent with the result we have in evaluating our flow presence detection in Section 4.4. In addition, no false positive further indicates that our algorithm is general to different situations, because we emulate different blockages which forms either during pump shut-in or during pump operation.

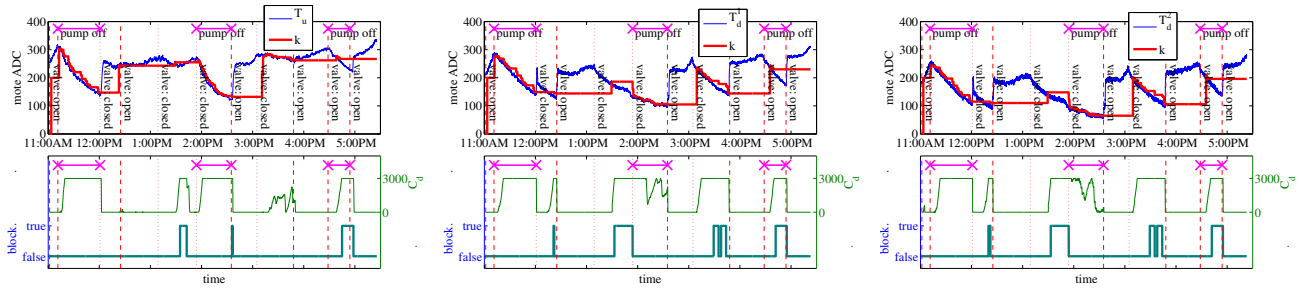
The second observation is the detection period is short, meaning our system is able to give rapid feedback. In problem statement (Section 2.1), we explain why rapid feedback is important to mitigate the loss. We find it generally takes between 10 to 30 minutes before our algorithm triggers.

The third observation is that some false positives are raised. We next evaluate why a perfect flow presence detection does not lead to a perfect blockage detection. To answer this question, we need to investigate the result on each event, particularly on false positives. The three figures in Figure 12 visualize our fully-automated system outputs and show why false positives exist. The lower plot in each figure contains both the certainty of drop (C_d) and the ultimate blockage indication. We see that there is transients after the pumpjack resumes operation. A blockage signal raised at 2:35pm in Figure 12(a) because the pipe skin temperature resumes to normal sightly later than the temperature sensor first receives pump-on signal. We expect our base algorithm to remain silent, although false positive is triggered. However, this can be easily fixed in an extended algorithm which suppresses anomaly outputs a short while after temperature sensor receives pump-on signal. Therefore, we still count it a true negative in later evaluation. One major cause of false positives on all three motes is incorrectly reporting pump-on during the third pump-off period (4:30pm–4:58pm), under effective temperature drop detection. However, the blockage detection algorithm successfully suppress the suggested blockages (*i.e.*temperature drops) in the first two pump-off period because of a build-in anti-false-alarm feature which ignores sporadic mis-detection (*fp*) in pumpjack status. The other cause is parameter mis-configuration. Due to the imperfect pumpjack detection (an overall accuracy of 87%) in Section 4.6, reported pumpjack status often incorrectly flips in the middle of an event, causing mis-configuration on anomaly and quality levels. However surprisingly, a relative chaotic parameter auto-configuration scheme does not throw off our entire blockage detection. Our algorithm exhibits robustness against the configuration errors, which at best cause only one false positive on T_u alone (after 1:09pm in Figure 12(a)).

In all, our blockage detection algorithm has a high accuracy, 80% in the worst case. A close look on algorithm output plots shows what causes mis-detection. However, the interesting results raises one further question about the robustness of our algorithm against the jittering in pumpjack detection. We next investigate this issue in Section 4.8.

4.8 Robustness of Blockage Detection

The results in evaluating our blockage detection surprise us because blockage detection is often insensitive to the error in pumpjack status detection, after fusing it with flow presence detection. One commonality in these errors is that the threshold to detect pump-on is such mis-configured that detection frequently (in minutes) flips between pump-on and -off. In order to verify if this feature is systematic or random,



(a) Upstream T_u. (b) Downstream before production valve T_d^1. (c) Downstream after production valve T_d^2.

Figure 12. Blockage detection results.

we take a three-fold approach in this section. We first formalize three simplified detection cases to theoretically prove our hypothesis. Second, we run simulation over these cases, and finally we further support the simulation result by experimental data. The simplified detection cases consists of both acoustic and temperature modalities. We employ the same acoustic model across the three cases that pumpjack status detection oscillates continuously at every observation time, starting from t_0:

$$P_i^o = \begin{cases} on & ,i = 2m-1 \\ off & ,i = 2m \end{cases}$$

We next introduce the three different temperature models—monotonically decreasing, stable with fluctuation, and monotonically increasing.

The first case we look into is when the pipe skin temperature monotonically decrease. The physical meaning behind this model is that fluid flow stopped either because of pump-off or blockage. Under this situation, we expected that our flow presence algorithm based on one-sided CUSUM should trigger, because observed temperature signal should stay below the reference value significantly long enough. We denote the temperature readings over time as:

$$s_i > s_{i+1}$$

According to the pumpjack status above and without losing generality, we assume pumpjack switches from on to off at t_{2m}. At the same time, we update the quality level, μ_{2m}^0 correspondingly:

$$\mu_i^0 = \begin{cases} s_{i-1} & ,i = 2m-1 \\ s_i & ,i = 2m \end{cases}$$

Please see Section 2.3 for more details about parameter tuning. Likewise, at t_{2m+1} we update the anomaly level, μ_{2m+1}:

$$\mu_i^1 = \begin{cases} s_i & ,i = 2m-1 \\ s_{i-1} & ,i = 2m \end{cases}$$

Hence, the reference value, k_i between $[t_i, t_{i+1}]$ is:

$$k_i = \frac{\mu_i^0 + \mu_i^1}{2} = \frac{s_{i-1} + s_i}{2} > s_i > s_{i+1}$$

This result clearly show that the reference is always larger than immediate temperature observation and therefore certainty of drop (C_d) builds up, triggering algorithm. The left

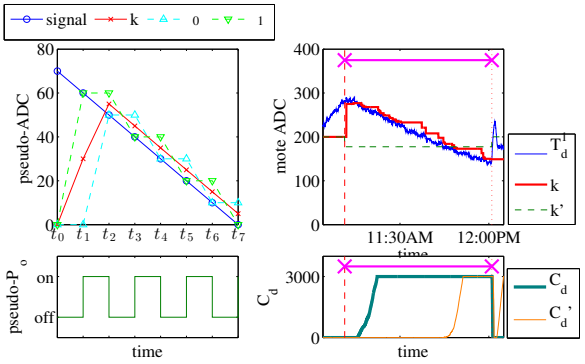


Figure 13. A monotonically decreasing example illustrates the robustness of our blockage detection algorithm.

half of Figure 13 illustrates this process. The right half is an excerpt of our experiment data rendering similar behavior. The k (solid thick line) in the top right plot comes from the result of our automated algorithm. On the contrary, the k' (dashed line) comes from the hand-set version to show what we expected to see if parameter setting is perfect, which yields C_d' below.

The second case is when the pipe skin temperature generally stabilized, but with natural fluctuation. The physical meaning is that fluid flow normally. We expected that our algorithm should remain silent because temperature should stay high enough. Although this model can represent the situation when pipe completely cools off, we do not discuss it because of no equivalent data. We denote the temperature readings over time as:

$$s_i = s_{i-2}$$

and without losing generality:

$$s_i > s_{i+1}, i = 2m+1$$

The quality and anomaly level update schema is the same as the last case, which leads to the reference value, k_i between $[t_i, t_{i+1}]$:

$$k_i = \frac{\mu_i^0 + \mu_i^1}{2} = \frac{s_{i-1} + s_i}{2} \begin{cases} < s_i & ,i = 2m+1 \\ > s_i & ,i = 2m \end{cases}$$

The above relationship between reference value and signal indicates that the temperature is likely to envelope the ref-

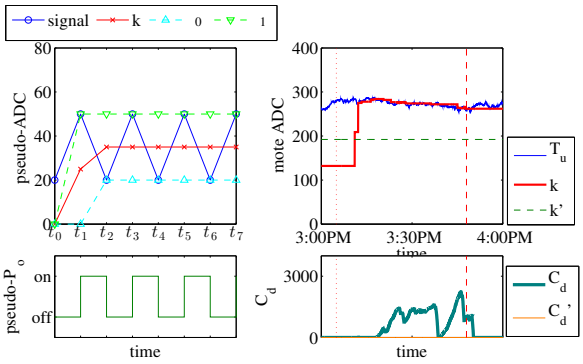


Figure 14. A stable with fluctuation example for robustness analysis.

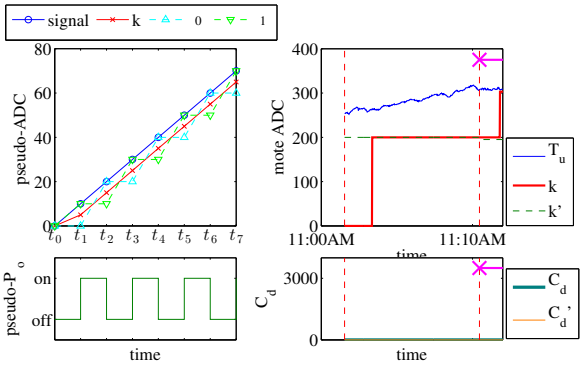


Figure 15. A monotonically increasing example for robustness analysis.

erence value and hence C_d cannot easily build up. Our algorithm is capable of suppressing false alarms in this case, showing in Figure 14

The final case is when the pipe skin temperature monotonically increase. The physical meaning is that fluid flow resumes after temporary pump shut-in. We expected that our algorithm fall back to silent quickly after temperature becomes high enough. Similar to the stable case, we denote the temperature readings over time as:

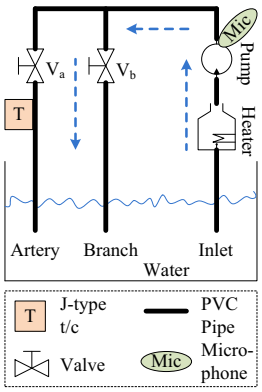
$$s_i = s_{i-2} \text{ and } s_i > s_{i+1}, i = 2m+1$$

Likewise the reference value, k_i between $[t_i, t_{i+1}]$:

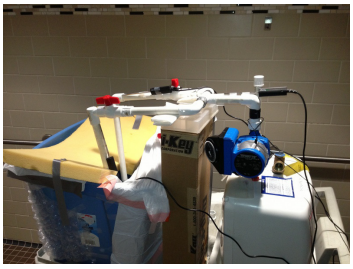
$$k_i = \frac{\mu_i^0 + \mu_i^1}{2} = \frac{s_{i-1} + s_i}{2} < s_i < s_{i+1}$$

Since k is always smaller than the temperature, C_d will not build and hence our algorithm achieves true negative. This process reversely mirrors what we observe in the decrease case above, depicted in Figure 15.

Summarizing how it response to all the three models, we conclude that our blockage algorithm is systematically robust against the negative effect of jittering pumpjack status detection on parameter setting. The most significant reason is that under those circumstances, the reference value correctly stay above or below the temperature by tracking it as low-pass-filtered signal.



(a) Logical view of lab test.



(b) Physical view of lab test.

Figure 16. In-lab, near-full blockage test on water.

4.9 In-lab Near-Full Blockage Detection

In prior sections, we show that our multi-modal detection correctly detects full blockages in the field. However, full blockages can quickly result in damage to equipment, so we would like to detect blockages *before* they fully close the line. We therefore next extend our work to detect *near-full blockage*. We verify this extension with laboratory tests. Unlike full blockage, we do not evaluate near-full blockage in oil field because it is not safe to emulate a realistic one—opening production valve slightly but with circulation valve closed would cause over-high pressure at the wellhead. Instead we use a mock-up flowline system in the laboratory. We choose hot water as the fluid because it has some similar properties as oil—both are incompressible and moderately warmer than ambient. We next introduce our lab experiment and show the results of our multi-modal sensing on near-full blockage.

To evaluate partial-blockaged detection we set up a testbed in our lab. We constructed a recirculating network of hot water similar to that used in our prior work [53]. It consists of a tankless water heater; a recirculation pump; a plastic, lidless tank; and a small network of PVC pipes and valves (Figure 16). Because we are evaluating multi-modal sensing, we deploy both temperature and acoustic sensors. To detect pipe skin temperature, we tape down USB-based Go!Temp temperature sensor [47] on the artery line after a valve (Figure 16(a)). Our acoustic sensor is a lavalier microphone, the same as the one used in our field test to collect ground truth data.

For acoustic detection, we must account for differences between the signal of pumpjack operation in the field, and the water recirculation pump in the lab. The major difference between pump-on and -off is average amplitude, and our algorithm is still able to distinguish the two when we adjust parameters. Please see detailed explanation on signal difference and evaluation result in later this section. In deployment, we tape down the microphone on the recirculation pump to detect the pump on/off status by picking up pump operating noise. Similar to the field test (Section 4.1), we collect raw pump noise with a sampling rate of 8 kHz, and down-sample to 2 kHz before aggregating into one-second-

Table 3. In-lab Experiment schedule.

start	pump	Artery valve	purpose
1:00pm	on	open	T learns μ_0
1:15pm	off		T learns μ_1
1:56pm	on	90% off	non-op
2:29pm		open	T learns μ_0
2:50pm		90% off	in-op
3:31pm	off		T learns μ_1
4:04pm	on		non-op
4:35pm		open	T learns μ_0
5:15pm		90% off	in-op
5:55pm		open	T learns μ_0
6:25pm		90% off	in-op

long samples measuring average noise amplitude. We later run our detection algorithm over these aggregated samples.

Finally, we do not use fieldable hardware in our lab near-full blockage test, contrary to our field test for full blockage. The hardware difference is because the field-test hardware are designed and assembled for oil field environment, and hence it takes extra amount of work on tuning them for the lab environment. Fortunately, the key properties of the two different fluids (oil/gas/water mix in the field network, and water in our laboratory network) are similar enough that the success of this lab test demonstrates that our multi-modal sensing can detect near-full blockages in oil field. In addition, this lab test generalizes our algorithm to applications other than those in oil industry.

Our laboratory tests follow the same procedure as our field tests (Section 4.2), however here we emulate near-full blockage by closing about 90% the valve rather than closing it complete. Following Figure 16(a), we close valve *V.a* on the artery line in, and open the branch valve to keep water flow and the heater running. Table 3 shows our lab test schedule, with two pump-off periods and five near-full blockages. In the six-hour-long test, we collect both temperature and acoustic traces by PC. We run our detection algorithm on a PC and do analysis of the data after collection. In principle we can integrate our algorithm with the sensors and run with exactly the same hardware and similar software to the field, however, here our goal is to test the generality of the algorithm, so we do analysis off-line to allow us to study a range of algorithm parameters post-facto. We use the same evaluation metrics as in field tests (Section 4.3).

Figure 17 shows that our flow presence detection algorithm, with adjusted parameters, gives rapid and accurate response on abnormal flow (pump-off and near-full blockage). Certainty of drop (C_d) correctly builds up on all seven abnormal flow events within 20 minutes (the lower plot of Figure 17). In addition, no false positive is raised when we open up the valve to learn μ_0.

To get these results, we must adapt the parameters to detect near-full blockages of the flow. We found that the base algorithm takes too long (about an hour) to trigger on blockages at 3:00pm, 4:00pm and 6:30pm, when the the reference value (k) is set at the mid point of anomaly (μ_1) and quality (μ_0) levels. This delay occurs because learning μ_1 out of pump-off temperature underestimates the temperature behavior at near-full blockage. The different temperature behavior is manifested by the two facts—temperature drop at *in-op* is more gradual than pump-off drop and temperature rebounce a little at *non-op*. In another words, the suboptimal μ_1 makes the algorithm insensitive to the temperature drops, because it tunes k too low and the lower k, the longer before temperature drops below k and triggers detection. Therefore, to adjust the parameters of our algorithm for near-full blockage detection, we increase μ_1 by 20% every time before computing a new k.

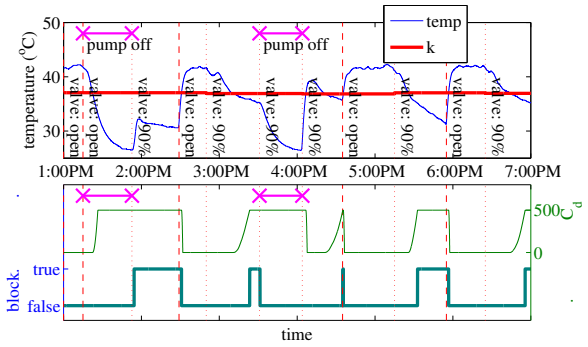


Figure 17. Temperature flow presence detection in water flowline.

We next evaluate our acoustic pump status detection and find the overall accuracy is 100%, shown in Figure 18. More specifically, all nine pump-on and two pump-off periods are correctly detected.

Like temperature detection above, we adjust the parameters in acoustic detection because the signal from the water recirculation pump is different from the pumpjack in the field test. The water pump-on noise is a wide-band signal from the motor and fluid flow, and unlike the oil pumpjack, there is no periodic cycle and bursty rod-tubing clang. However, the major difference between pump-on and -off signal lies in the average amplitude—pump-off is much quieter relative to ambient noise, while pumpjack flow is relatively loud. The result shows, with proper parameters, our algorithm is capable of distinguishing average amplitude difference between pump status. In this test, we use a longer cycle (18 s) and lower pump-on threshold (60-percentile of amplitude among pump-on training samples), comparing to our field test. On the other hand, the success of our acoustic detection on signal with different properties than pumpjack generalize our approach to a broader range of applications.

Combining the perfect temperature and acoustic detections, the end near-full blockage detection accuracy is 100% over the total of eleven events. The lower plot of Figure 17 shows that our algorithm triggers on all five blockages and the two pump-off periods are correctly suppressed.

In all, our lab test shows our multi-modal sensing works with parameter changes on near-full blockage detection in a hot water network. We believe this result generalizes to

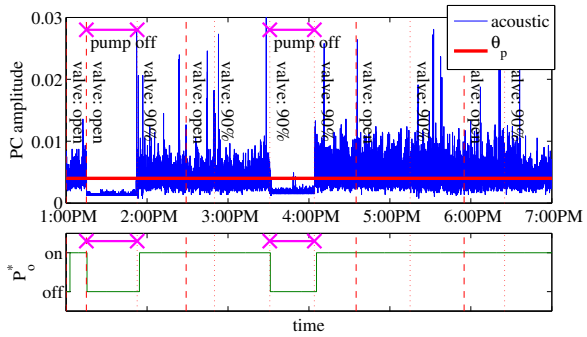


Figure 18. Acoustic pump status detection in water flowline.

real-world near-full blockage in oil flowlines.

4.10 Evaluation Summary and Algorithm Generalization

We have shown our multi-modal approach works well to detect cold-oil and hot water blockages. We next consider related problems and possible future work.

We believe our general approach—use of two different types of sensors, one accurate but noisy, and the second to filter the noise—generalizes to other sensing problems. Our algorithm may apply to other applications where blockages occur, in addition to cold oil lines. For example, vehicle or building cooling systems may have similar blockage problems in systems that duty cycle, as do hot water distribution systems. These systems both have fluid moving through a pipe network, where flow can be detected by temperature variation from ambient, yet one must also coordinate with driving machinery that operates intermittently and would otherwise cause incorrect outage detection.

Our short-term field experiment for cold oil blockage was successful, but additional work remains. A next step is longer-term testing to evaluate the robustness of the hardware system and the tolerance of the detection algorithm against environment change. Second, a full integration with field network is a stepping stone to transform this research into a practical field equipment. Third, we are actively seeking other sensing modality for better cost-efficiency. For example, from our survey we find some potential alternatives are infrared imaging and vibration. Finally, detecting partial blockages is future work.

5 Related Work

Our work builds on prior research results in flowline monitoring, change-point detection algorithms, and multi-modal sensing.

5.1 Pipeline Monitoring Systems

Our work first builds on pipeline monitoring related works, especially in oil line blockage detection.

5.1.1 Oil Line Blockage Detection Applications

Oil line blockage is not a new problem to the oil industry. We study how to use multi-modality with low cost sensors to detect line blockages, but prior work has looked at alternative methods.

Liu et al. shows it is practical to locate blockage in a long oil line by measuring the travel time of pressure decompression wave bouncing back from the blockage point [24]. They test their method by emulated blockage by an intermittently operating oil line, and we did similar blockage emulation for testing. However, their sensing pressure modality is invasive, which is different from our non-invasive sensing.

Liu and Scott show a theoretical work to localize blockage in subsea flowlines by comparing the inlet and outlet line pressure and other factors [25] They use invasive pressure sensing, while we are non-invasive. Our work suggests that we can locate blockage by segmenting pipe with pairwise temperature sensors, which is our future work. In addition, we have a real deployment beyond theoretical models.

In short, our major difference from prior work is our use of non-invasive sensing. Further, we demonstrate a successful wireless sensornet deployment in the field.

5.1.2 Other Pipeline Monitoring Work

Besides oil line blockage, there are many other industrial pipeline monitoring applications. SCADA systems have long been used for pipeline monitoring. Traditional SCADA systems use in-situ sensors (mostly single-modal) and centralized decision making [8, 28], while our work instead focuses on detecting problems with heterogeneous, intelligent, low-cost sensor in a network.

On the other hand, prior sensornet research in pipeline monitoring usually assume low-temperature, single-phase fluid [17,19,35,37]. Many researchers choose vibration sensing, effectively equivalent to our acoustic sensing, for low-temperature fluid monitoring. NAWMS focuses on personal water usage [19, 26]. Our pumpjack status detection hardware is similar to theirs in Mica series motes and MTS310 sensor boards. However, they use accelerometer while we use the microphone embedded on the same sensor board. They infer flow rate by pipe vibration frequency and linear-programming-based algorithms. We do not measure vibration in our flow inference because operating pumpjack generates wide-band noise which overwhelms flow vibration signal. Instead, we use vibration as a secondary modality to detect pumpjack operation status. Stoianov et al. prove in PIPENET the feasibility of measuring vibration to detect small leak on water sewage pipe [37]. However, they do not present a completely integrated system; no detection algorithm is implemented. Their field test only shows that their sensors deployed under urban sewage are capable of collecting certain types of data and relay them back. We instead focus on a complete multi-modal system running oil-line blockage-detection algorithm online. Jin and Eydgahi [17] and Sinha [35] utilize acoustic wave propagation for pipe defects detection. Jin and Eydgahi propose a general sensor network frame, while focusing on specific signal processing analytical technique. Sinha's work is mainly about transducer instrumentation and calibration for natural gas inspection. Instead of pipe defects, we focus on oil line blockage and real system development and deployment.

Contrary to forgoing low-temperature fluid works, we focuses on high-temperature, multi-phase fluid (oil-gas-water mix). This fluid property change allows us to use temperature for fluid presence detection [52, 55]. Zhu's work

shows the feasibility of temperature monitoring for blockage detection of pulverized coal injection system [55]. Our fluid is mostly liquid, while in his work fluid is actually fine powder. His detection algorithm first measures temperature by thermometers mounted on branch pipes, differentiates pipe skin temperature and compares the result against pre-configured thresholds. We design our flow presence detection with a similar idea. However, we use inexpensive and portable hardware (US$500), while his centralized system is likely expensive because one of its component is an industrial PC, which usually costs US$1 500. Our prior work studies steam choke blockage detection with wireless sensor networks [52]. We prove the feasibility of detecting steam choke blockage by non-invasive temperature sensing. We show similar sensing technique generalizes to a hot water distribution network, similar to the oil retrieval line network in the paper. In this prior work we develop a complete system with thermal energy harvesting, evaluated in a field test. In this work, all sensor motes are battery powered; energy harvesting is our future work. Unlike this prior work, our current work adds multi-modal sensing to address the false positives common to cold-oil blockage. In temperature sensing, the prior work takes differential temperature around the blockage point, but here we use absolute temperature from one sensor because contrary to choke blockage, the cold-oil blockage location is unpredictable on a long line. In addition, our current work auto-configures all parameters for detection during sensornet deployment.

We have previously explored the potential of sensor networks in oilfield production systems [50]. While that work suggests the potential, this paper demonstrates a field-tested system and evaluates specific sensing algorithms with multi-modality.

5.2 Change Point Detection Algorithms

Many real-time monitoring systems use abrupt detection [5] or change-point detection [1] to detect problems in observed data. In change-point detection on flow presence, we focus on cumulative sum control chart (CUSUM) and exponential-weighted moving average (EWMA) for lightweight implementations on mote-class platforms. Our flow presence detection algorithms are inspired by several prior works [7, 39].

Some researchers use CUSUM to analyze time series and we implement it to detect flow presence. Chamber et al. develop a CUSUM-based algorithm to detect vegetation changes in forests [7]. Similar to their work, we choose CUSUM for its capability in identifying small and gradual change and we too make our algorithm adaptive to noise. They post-process year-long time series for small change while we in-node process streaming data. Another difference is their algorithm is designed to identifying multiple decreasing segments in a series, but our algorithm focuses on immediate decision.

Several sensornets build on the simple EWMA algorithm from TCP [16]. Trifa et al. develop an adaptive alarm call detection system for yellow-bellied marmots, using EWMA to update threshold by noise distribution estimation [39]. Our work uses similar concepts to detect significant change in oil pipe temperature using EWMA.

5.3 Multi-Modal Sensing Applications

We use cold-oil blockage detection as a case study to learn multi-modal sensing in industrial monitoring. Next we review related multi-modality work in the same domain, followed by a broader review in other application domains.

5.3.1 Multi-Modal Sensing in Industrial Monitoring

Our cold-oil blockage detection shows complementary collaboration with multi-modal sensing has potentially great feasibility to industrial monitoring applications. A few other multi-modal sensing applications are targeted to the same field [9, 12, 51].

Zeng et al. show using vibration, force and acoustic emission sensors to monitor health of high-speed milling machine and predict wear-out [51]. Our pumpjack status detection is similar to their idea—rumbling machines emit detectable acoustic pattern. However, we are not doing oil pipe blockage prediction and it is part of our future work. Futagawa et al. design an integrated electrical conductivity temperature sensor for cattle health monitoring [9]. Their sensing is invasive since they embed sensors into rumens of cattle, while part of the requirement of our oil-line blockage-detection system is non-invasive to lower cost. Gupta et al. use microwave and eddy current image to evaluate corrosion under aircraft paint and in lap joints [12]. They collaborate two modalities competitively, meaning either on can fulfill the task but fusion achieves better results, but our sensor collaboration is complementary.

5.3.2 Multi-Modal Sensing in Academic Projects

Many academic studies explored multi-modal sensing for different applications, including general sensor fusion [13, 20], target classification [48], target tracking [2, 10, 21–23, 54, 56], human activity or health monitoring [3, 6, 15, 29, 30, 36], Robotic navigation [4, 34] and human-computer multi interaction [11, 27, 32, 38].

Some work uses multivariate statistics modeling for multiple sensor data fusion. Goecke uses coinertia analysis to find a mathematical compromise between the correlation of audio and video (3D lip tracking features) [11]. Kushwaha et al. uses separate non-parametric model for audio sensors and parametric mixture-of-gaussian model for video sensor in their vehicle tracking application [22]. Annavaram et al. use bivariate model for ECG and accelerometer data to monitor sensor bearer's activity [3]. Our oil-line blockage-detection algorithm has a similar way to do change-point detection. We build separate models for both acoustic and temperature data and configure thresholds for each. In addition, we plan leverage the correlation between the two modality when pipe is normal.

Other work specifically uses Bayesian networks. Tamura et al. use triphone HMM to model audio/video data for speech recognition. They find that a training set of audio-visual data achieves better recognition accuracy than audio-only data [38]. McGuire et al. leverage Bayesian networks to integrate spoken instruction, visual memory and gesture-based region bias to determine the object to be grasped by robotic arms [27]. Zou and Bhanu evaluate both time-delay neural network method and Bayesian network method for walking human detection from audio-video data. They conclude Bayesian network is better because of ease

to train, higher accuracy and clearer graphical model [56]. Huang et al. propose a coupled HMM method for audio-visual joint modeling, especially to solve asynchronization problem in a office activity monitoring application [15]. Oliver and Horvitz use layered HMM, a modular and hierarchical HMM method for office activity inference [30]. Singhal and Brown uses Bayesian network to joint model audio and video data to predict obstacle in navigation [34]. These works model multiple modality jointly but we instead focus on separate modeling, since we do not find significant inter-modality correlation during blockage.

Rather than direct fuse multiple sensor channels, a few works utilize a secondary/orthogonal sensory channel to assist the main channel for better perception or sensing. Girod and Estrin suggest using video evidence to solve the obstacle problem in their acoustic ranging application [10]. Qu et al. add vision (Pan-tilt-zoom camera) and actuator (pan-tilt-unit) to help the LDV automatically select the best reflective surfaces, point and focus the laser beam, in order to remotely pickup voice signal [32]. Stiefmeier et al. integrate several sensors into one wearable sensors to monitor worker's activity. They propose use one kind of sensing result to do automatic data segmentation for other sensing streams [36]. Barakova and Lourens propose event-based data fusion, contrary to fixed time interval based fusion. They use gyroscope data to segment visual data for robotic navigation [4]. Our method employs this idea because we are actually using acoustic date to assist temperature date for blockage detection. Our hypothesis indicates if pumpjack is not operating, we have no way to tell if the pipe is blocked. In other words, we use acoustic date to segment temperature date and ignore those when pumpjack is off.

6 Conclusion

We have described a system for multi-modal sensing to detect cold-oil blockages. We show combining different types of low-cost, non-invasive sensors can avoid false positives and provide rapid blockage detection. To achieve this, we developed an algorithm first uses pipe skin temperature to infer changes in fluid flow for suggested blockages. Later we suppress false alarms caused by some regular operation with acoustic sensing. We have demonstrated the effectiveness of this algorithm and our implementation through field experiments. Although we have developed this system to solve cold-oil blockage problem in oil field, the principle of multi-modal, low-cost sensor collaboration generalizes to other industrial sensing applications where false positives can be resolved by a different sensing modality.

Acknowledgments

We would like to thank Andrew Goodney for his input on acoustic sensing.

We thank Greg LaFramboise, Charlie Webb, Mohammad Heidari for their input on Chevron's business requirements and their assistance in our field experiments; Iraj Erhagi and Mike Hauser for their guidance as co-directors of CiSoft.

7 References

[1] V. V. A. Tartakovsky. Change-point detection in multichannel and distributed systems with applications. In *Applications of Sequential Methodologies*, pages 331–363, 2004.

[2] B. R. Abidi, N. R. Aragam, Y. Yao, and M. A. Abidi. Survey and analysis of multimodal sensor planning and integration for wide area surveillance. *ACM Computer Survey*, 41:7:1–7:36, Jan. 2009.

[3] M. Annavaram, N. Medvidovic, U. Mitra, S. Narayanan, G. Sukhatme, Z. Meng, S. Qiu, R. Kumar, G. Thatte, and D. Spruijt-Metz. Multimodal sensing for pediatric obesity applications. In *Proceedings of the International Workshop on Urban, Community, and Social Applications of Networked Sensing Systems (UrbanSense)*, Raleigh, NC, USA, Nov. 2008. ACM.

[4] E. I. Barakova and T. Lourens. Event based self-supervised temporal integration for multimodal sensor data. *Journal of Integrative Neuroscience*, 4(2):265–282, 2005.

[5] M. Basseville and I. V. Nikiforov. *Detection of abrupt changes: theory and application*. Prentice-Hall, Inc., 1993.

[6] M. Benning, A. Kapur, B. Till, and G. Tzanetakis. Multimodal sensor analysis of sitar performance: Where is the beat? In *Multimedia Signal Processing, 2007. MMSP 2007. IEEE 9th Workshop on*, pages 74 –77, Oct. 2007.

[7] Y. Chamber, A. Garg, V. Mithal, I. Brugere, M. Lau, V. Krishna, S. Boriah, M. Steinbach, V. Kumar, C. Potter, and S. Klooster. A novel time series based approach to detect gradual vegetation changes in forests. In *Proceedings of the 2011 NASA Conference on Intelligent Data Understanding*, pages 248–262, Mountainview, CA, USA, Oct. 2011. NASA.

[8] A. Daneels and W. Salter. What is SCADA? In *Proceedings of the International Conference on Accelerator and Large Experimental Physics Control System*, ICALEPCS '99, Trieste, Italy, Oct. 1999.

[9] M. Futagawa, M. Ishida, M. Ishida, and K. Sawada. Study of a wireless multimodal sensing system integrated with an electrical conductivity sensor and a temperature sensor for the health control of cows. *IEEE Transactions on Electrical and Electronic Engineering*, 6(2):93–96, 2011.

[10] L. Girod and D. Estrin. Robust range estimation using acoustic and multimodal sensing. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Maui, Hawaii, USA, Oct. 2001. IEEE.

[11] R. Goecke. Audio-video automatic speech recognition: an example of improved performance through multimodal sensor input. In *Proceedings of the 2005 NICTA-HCSNet Multimodal User Interaction Workshop - Volume 57*, MMUI '05, pages 25–32, Sydney, Australia, 2006. Australian Computer Society, Inc.

[12] K. Gupta, M. Ghasr, S. Kharkovsky, R. Zoughi, R. Stanley, A. Padwal, M. O'Keefe, D. Palmer, J. Blackshire, G. Steffes, and N. Wood. Fusion of microwave and eddy current data for a multi-modal approach in evaluating corrosion under paint and in lap joints. *Review of Quantitative Nondestructive Evaluation*, 26:611–618, 2007.

[13] K. Harrington and H. Siegelmann. Adaptive multi-modal sensors. In M. Lungarella, F. Iida, J. Bongard, and R. Pfeifer, editors, *50 Years of Artificial Intelligence*, volume 4850 of *Lecture Notes in Computer Science*, pages 164–173. Springer Berlin / Heidelberg, 2007.

[14] W. Hu, N. Bulusu, C. T. Chou, S. Jha, A. Taylor, and V. N. Tran. Design and evaluation of a hybrid sensor network for cane toad monitoring. *ACM Transactions on Sensor Networks*, 5(1):4:1–4:28, Feb. 2009.

[15] P.-S. Huang, X. Zhuang, and M. Hasegawa-Johnson. Improv-

ing acoustic event detection using generalizable visual features and multi-modality modeling. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 349 –352, may 2011.

[16] V. Jacobson. Congestion avoidance and control. In *Symposium proceedings on Communications Architectures and Protocols*, SIGCOMM '88, pages 314–329, Stanford, California, United States, Aug. 1988. ACM.

[17] Y. Jin and A. Eydgahi. Monitoring of distributed pipeline systems by wireless sensor networks. In *Proceedings of The 2008 IAJC-IJME International Conference*, Nashville, TN, USA, Nov. 2008.

[18] S. Kim, D. Culler, and J. Demmel. Structural health monitoring using wireless sensor networks. *Berkeley Deeply Embedded Network System Course Report*, 2004.

[19] Y. Kim, T. Schmid, Z. M. Charbiwala, J. Friedman, and M. B. Srivastava. NAWMS: nonintrusive autonomous water monitoring system. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, SenSys '08, pages 309–322, Raleigh, NC, USA, Nov. 2008. ACM.

[20] F. Koushanfar, S. Slijepcevic, M. Potkonjak, and A. Sangiovanni-Vincentelli. Error-tolerant multi-modal sensor fusion. In *IEEE CAS Workshop on Wireless Communications and Networking*, Pasadena, California, USA, Sept. 2002.

[21] C. Kreucher, D. Blatt, A. Hero, and K. Kastella. Adaptive multi-modality sensor scheduling for detection and tracking of smart targets. *Digital Signal Processing*, 16(5):546 – 567, 2006.

[22] M. Kushwaha, S. Oh, I. Amundson, X. Koutsoukos, and A. Ledeczi. Target tracking in heterogeneous sensor networks using audio and video sensor fusion. In *Multisensor Fusion and Integration for Intelligent Systems, 2008. MFI 2008. IEEE International Conference on*, pages 14 –19, Aug. 2008.

[23] L. Lazos, R. Poovendran, and J. A. Ritcey. Probabilistic detection of mobile targets in heterogeneous sensor networks. In *Proceedings of the 6th international conference on Information processing in sensor networks*, IPSN '07, pages 519–528, Cambridge, Massachusetts, USA, 2007. ACM.

[24] E. Liu, C. Li, S. Peng, and X. Wu. Detection technology for oil pipeline plugging based on decompression wave method. In *Computational and Information Sciences (ICCIS), 2010 International Conference on*, pages 820 –823, Dec. 2010.

[25] L. Liu and S. L. Scott. A new method to locate partial blockages in subsea flowlines. In *SPE Annual Technical Conference and Exhibition*, New Orleans, Louisiana, USA, Sept. 2001. Society of Petroleum Engineers. SPE 71548.

[26] P. Martin, Z. Charbiwala, and M. Srivastava. Doubledip: leveraging thermoelectric harvesting for low power monitoring of sporadic water use. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, SenSys '12, pages 225–238, Toronto, Ontario, Canada, 2012. ACM.

[27] P. McGuire, J. Fritsch, J. Steil, F. Rothling, G. Fink, S. Wachsmuth, G. Sagerer, and H. Ritter. Multi-modal human-machine communication for instructing robot grasping tasks. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, volume 2, pages 1082–1088, 2002.

[28] F. Molina, J. Barbancho, and J. Luque. Automated meter reading and SCADA application for wireless se nsor network. In *Ad-Hoc, Mobile, and Wireless Networks*, volume 2865 of *LNCS*, pages 223–234. Springer Berlin/Heidelberg, 2003.

[29] S. Moncrieff, S. Venkatesh, and G. West. Dynamic privacy assessment in a smart house environment using multimodal sensing. *ACM Trans. Multimedia Comput. Commun. Appl.*, 5:10:1–10:29, Nov. 2008.

[30] N. Oliver and E. Horvitz. S-seer: Selective perception in a multimodal office activity recognition system. In S. Bengio and H. Bourlard, editors, *Machine Learning for Multimodal Interaction*, volume 3361 of *Lecture Notes in Computer Science*, pages 122–135. Springer Berlin / Heidelberg, 2005.

[31] E. S. Page. Continuous Inspection Schemes. *Biometrika*, 41(1/2):100–115, 1954.

[32] Y. Qu, T. Wang, and Z. Zhu. An active multimodal sensing platform for remote voice detection. In *Advanced Intelligent Mechatronics (AIM), 2010 IEEE/ASME International Conference on*, pages 627–632, july 2010.

[33] C. J. V. Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 2nd edition, 1979.

[34] A. Singhal and C. Brown. Dynamic bayes net approach to multimodal sensor fusion. In *Proceedings of the SPIE - The International Society for Optical Engineering*, pages 2–10, 1997.

[35] D. Sinha. Acoustic sensor for pipeline monitoring. Technical Report LA-UR-05-6025, Los Alamos National Laboratory, July 2005.

[36] T. Stiefmeier, D. Roggen, G. Troster, G. Ogris, and P. Lukowicz. Wearable activity tracking in car manufacturing. *Pervasive Computing, IEEE*, 7(2):42 –50, april-june 2008.

[37] I. Stoianov, L. Nachman, S. Madden, and T. Tokmouline. PIPENET: a wireless sensor network for pipeline monitoring. In *Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, IPSN '07, pages 264–273, Cambridge, Massachusetts, USA, 2007. ACM.

[38] S. Tamura, K. Iwano, and S. Furui. A robust multimodal speech recognition method using optical flow analysis. In W. Minker, D. Bhler, L. Dybkjr, and N. Ide, editors, *Spoken Multimodal Human-Computer Dialogue in Mobile Environments*, volume 28 of *Text, Speech and Language Technology*, pages 37–53. Springer Netherlands, 2005.

[39] V. Trifa, L. Girod, T. Collier, D. T. Blumstein, and C. E. Taylor. Automated wildlife monitoring using self-configuring sensor networks deployed in natural habitats. In *Proceedings of the 12th International Symposium on Artificial Life and Robotics*, AROB '07, Beppu, Japan, Jan. 2007.

[40] `http://bullseye.xbow.com:81/Support/Support_pdf_files/MPR-MIB_Series_Users_Manual.pdf`. Crossbow mica-2 user's manual, 2007.

[41] `http://bullseye.xbow.com:81/Support/Support_pdf_files/MTS-MDA_Series_Users_Manual.pdf`. MTS/MDA sensor board user's manual, 2007.

[42] `http://datasheets.maximintegrated.com/en/ds/MAX4465-MAX4469.pdf`. MAX4465–MAX4469 data sheets, 2012.

[43] `http://www.analog.com/static/imported-files/data_sheets/AD1988A_1988B.pdf`. AD1988A/AD1988B data sheets, 2006.

[44] `http://www.audio-technica.com/cms/wired_mics/9c6eca17168eef6f/index.html`. ATR3350 omnidirectional condenser lavalier microphone.

[45] http://www.dataq.com/support/documentation/pdf/datasheets/el-usb-tc-data-logger.pdf. EL-USB-TC thermocouple data logger datasheet, 2012.

[46] http://www.panasonic.com/industrial/components/pdf/em05_wm62_a_c_cc_k_b_dne.pdf. WM-62A datasheet.

[47] http://www.vernier.com/go/gotemp.html. Vernier Go!Temp thermometer.

[48] X. Wang, H. Qi, and S. Iyengar. Collaborative multi-modality target classification in distributed sensor networks. In *Information Fusion, 2002. Proceedings of the Fifth International Conference on*, volume 1, pages 285–290, 2002.

[49] Y. Yasuno and J. Ohga. Temperature characteristics of electret condenser microphones. In *Electrets, 2005. ISE-12. 2005 12th International Symposium on*, pages 412–415, Sept. 2005.

[50] S. Yoon, W. Ye, J. Heidemann, B. Littlefield, and C. Shahabi. SWATS: Wireless sensor networks for steamflood and waterflood pipeline monitoring. *IEEE Network Magazine*, 2009. accepted in 2009 to appear in 2010 or later.

[51] H. Zeng, T. B. Thoe, X. Li, and J. Zhou. Multi-modal sensing for machine health monitoring in high speed machining. In *Industrial Informatics, 2006 IEEE International Conference on*, pages 1217 –1222, Aug. 2006.

[52] C. Zhang, A. Syed, Y. H. Cho, and J. Heidemann. Steam-powered sensing. In *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, SenSys '11, pages 204–217, Seattle, Washington, USA, Nov. 2011. ACM.

[53] C. Zhang, A. Syed, Y. H. Cho, and J. Heidemann. Steam-powered sensing: Extended design and evaluation. Technical Report ISI-TR-2011-670, USC/Information Sciences Institute, Feb. 2011.

[54] F. Zhao, J. Shin, and J. Reich. Information-driven dynamic sensor collaboration for tracking applications. *IEEE Signal Processing Magazine*, 19(2), Mar. 2002.

[55] J. Zhu. Application of temperature method for jam detection in BF coal injection. *Baosteel Technology*, 6:15–18, 2005.

[56] X. Zou and B. Bhanu. Tracking humans using multi-modal fusion. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, page 4, june 2005.