Steam-Powered Sensing *

Chengjie Zhang[1] Affan Syed[1,2] Young Cho[1] John Heidemann[1]
chengjie@isi.edu affan.syed@nu.edu.pk youngcho@isi.edu johnh@isi.edu
1: Information Sciences Institute
University of Southern California
Marina del Rey, California, USA
2: National University of Computer
& Emerging Sciences
Islamabad, Pakistan

Abstract

Sensornets promise to extend automated monitoring and control into industrial processes. In spite of great progress made in sensornet design, *installation and operational costs* can impede their widespread adoption—current practices of infrequent, manual observation are often seen as sufficient and more cost effective than automation, even for key business processes. In this paper we present two new approaches to reduce these costs, and we apply those approaches to rapidly detect blockages in steam pipelines of a production oilfield. First, we eliminate the high cost of bringing power to the field by *generating electricity from heat*, exploiting the high temperature of the very pipelines we monitor. We demonstrate that for temperature differences of 80 °C or more, we are able to sustain sensornet operation without grid electricity or batteries. Second, we show that *non-invasive sensing* can reduce the cost of sensing by avoiding sensors that pierce the pipeline and have high installation cost with interruption to production. Our system instead uses surface temperature to infer full or partial blockages in steam pipelines and full blockages in hot water pipelines. Finally, we evaluate our "steam-powered sensing" system to monitor potential blockages in steam pipeline chokes at a production oilfield. We also show the generality of our algorithm by applying it to detect water pipeline blockages in our lab. To our knowledge, this paper describes the first field-tested deployment of an industrial sensornet that employs non-solar energy harvesting.

Categories and Subject Descriptors: C.2.2 [Computer-Communication Networks]: Network Protocols

General Terms: Design, Algorithms, Performance, Experimentation, Measurement

Keywords: thermal energy harvesting, non-invasive sensing, blockage detection, industrial wireless sensor networks

* 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.

SenSys'11, November 1–4, 2011, Seattle, WA, USA.

1 Introduction

Automation of monitoring and control is essential in today's industrial processes. Since the 1960s, supervisory control and data acquisition (SCADA) systems have automated monitoring and control of industrial processes in applications including water management, power grids, chemical processing, and oil production [12]. Today, SCADA systems are a multi-billion-dollar-per-year industry with a growing need for wireless and distributed sensornet techniques.

One effective use of SCADA systems can be seen in oil industry. While a "gusher" produces oil from internal pressure, this kind of primary production can only extract a fraction (5–10%) of oil in the ground. Today, many older fields depend on *secondary* production techniques, where water, steam, or CO_2 is injected to force out oil, extracting up to 30–60% of reserves. While such techniques are essential to meet energy demands, their key limiting factor is *cost*, not technology. Although oil companies have great technical sophistication, solutions as simple as monthly human observation are often seen as sufficient and more cost effective than available forms of automation. Even with relatively inexpensive hardware, installation of a new pipeline pressure sensor can easily top US$10k when one considers the labor needed to run power and safely install sensors inside the pipeline.

In this paper we propose an *inexpensive* sensornet system to monitor steam injection in oilfields. We directly address the cost of current approaches through two contributions. First, we demonstrate a new approach to *harvest energy from temperature differential* inherent to the phenomena we are studying (§3). We exploit the Seebeck effect to generate electricity from heat inherent in a steam injection system, eliminating any need for external power or batteries. We are the first to use heat to power an industrial sensor network, and to show batteryless operation—valuable approaches for sensornets deployed where solar power is unavailable and maintenance is difficult (for example, northern Alaska).

Our second contribution is to employ *non-invasive sensing techniques* to detect problems in steam distribution (§4). Current approaches to detect blocking usually measure differential steam pressure with sensors that must pierce the pipeline, incurring a high cost of labor, equipment, and stopped production. Instead, we observe that external temperature observation is sufficient to detect problems such as blockage or flow constriction, *provided* we observe at multiple locations. We argue that pervasive industrial sensing requires this sort of non-invasive sensing to reduce deployment cost. We also show that our approach works with multiple types of pipeline networks.

Our final contribution is to demonstrate that our approach of non-invasive, "steam-powered" sensing works as a com-

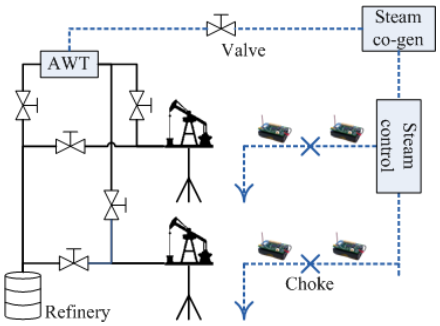


Figure 1. Steam injection (right) and oil production (left) in oilfield.

plete system, through both laboratory experiments and field tests (§5). Although low-power sensing and energy harvesting have been demonstrated before, we demonstrate an integrated system for this new application. Our system employs a custom thermoelectric energy harvesting/conditioning unit, and a custom amplification board with calibrated thermocouples, controlled by a standard Mica-2 mote running new detection algorithms. We also show the importance of relatively simple hardware and sensing to solve real-world problems: simple hardware makes operation on harvested energy feasible, and simple, non-invasive sensing, when taken at *multiple* locations can provide actionable decisions. Unlike some papers on sensor networking, we focus on the sensing and energy components of a complete system; we assume a traditional field network to communicate results externally.

Although we validate our approach with a specific steampipe oilfield deployment, laboratory tests show our non-invasive sensing also applies to water blockages, and in many environments waste heat (from engines or other industrial processes) could support thermal energy harvesting. We believe our two approaches are applicable to a wide range of industrial sensing, where they can bring sensing and deployment costs in line with inexpensive communications and computation.

2 Problem Statement and System Overview

We next summarize specific sensing needs of modern oilfields, our target sensing problem, and our general approach.

2.1 Sensing Needs for Secondary Production

Most modern oilfields employ *secondary production*, where water, steam, or CO_2 is injected into the ground to release otherwise difficult to extract oil. Injection helps release trapped oil either by pushing it out or raising temperatures to make it flow more freely; a higher reservoir pressure also avoids ground subsidence which can damage wells or property. Secondary production is essential in older fields where the natural pressure is insufficient for primary (unaided) production, but it greatly adds to the complexity of the field.

Figure 1 shows a simple producing field with steamflood-based secondary production. Steam is produced at a central site ("steam co-gen" in figure) , and distributed through the field at high temperature and pressure (250 °C and 5000 kPa or more) [5]. Injection lines (dotted in the figure) convey a mix of steam and water. Maintaining an exact ratio, or *steam quality*, throughout the distribution network is important to control distribution and injection. A special *steam control* device maintains steam quality at branches. A *choke*, a small, controlled-size hole (about 1 cm or more in diameter) just before an injection well, manages pressure at the injection well, making injection rates predictable and allowing operators to control the field.

Producing wells extract oil from the ground, where automatic well testing (AWT) systems allow per-well production measurements. The steam network is linked to the production network to allow producing wells to be flushed with steam to remove blockages.

This brief description shows how essential instrumentation is to an oilfield. Steam quality must be monitored in the steam distribution network; flow rates at injection wells and chokes must be observed; well monitoring is essential at the production side; the ability to inject steam in production systems means the injection and production sides are cross-linked and must be monitored for leaks. Yet sensing must be cost-effective, even for wells that produce only a few barrels of oil per day, and in fields that have hundreds or thousands of production and injection wells!

2.2 Target Problem: Blockage at the Steam Injection Choke

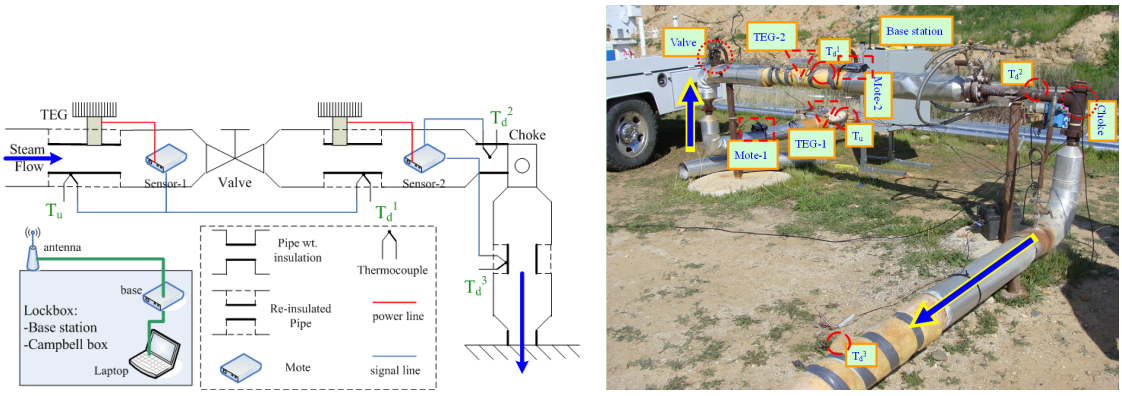
In this paper we focus specifically on the problem of *blockage at the injection-well choke in a steamflood field*. We define *blockage* as the decreasing of the choke's cross section due to obstructions. Field engineers report that choke blockage is a serious problem in field operation. Chokes are vulnerable to blockage because of their small bore sizes. Sources of blockage occur naturally in a steam distribution system due to scaling and corrosion in the pipe, buildup of any impurities or mineral content in the water, and aging of the network and choke. Partial or total blockage at a choke is a serious problem because it alters the steam injection rate, throwing off field management, reducing production, and potentially contributing to ground subsidence.

Our current work focuses on choke blockage for steamflood fields, but it applies to several related problems as well. Other points of operational concern include steam control at pipeline branches and automatic well-test monitoring; both could use systems similar to ours, provided likely blockage sites are known. We show that our sensing algorithm applies to waterflood networks (§5.5), and we believe our energy harvesting system can adapt to different thermal conditions.

2.3 System Overview

The goal of our sensing system is to detect blockages at the choke of steam injection wells, and to do so at a cost much lower than current invasive sensing. We next briefly review the hardware and software we have taken into the field to evaluate solutions to this problem.

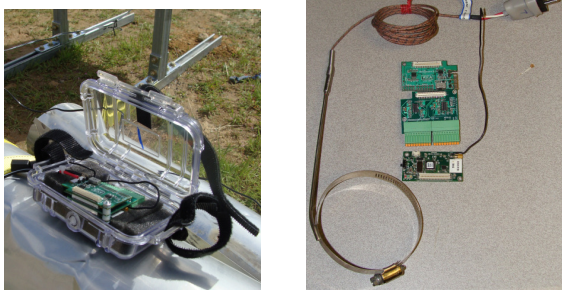
Figure 2 shows our system as deployed in the field in March 2010. The photograph in Figure 2(b) shows two sensor nodes (each a mote with two temperature sensors, a thermal energy harvester, and a wireless network connection) and a base station to connect to the field network, while Figure 2(a) shows the logical view. We review the hardware and software below, and discuss details of this field experiment



(a) Logical view of deployment.

(b) Physical view of deployment.

Figure 2. March 2010 field deployment of our sensing system.



(a) Deployed mount, in a pelican box with lid open.

(b) A Mica-2, a custom amplifier board, a Heliomote and a hose-clamp thermocouple.

Figure 3. Mote system hardware.

in §5.3. By comparison, Figure 13 shows a current invasive pressure sensor; we compare deployment costs of our approach to current approaches in §5.6.

Each sensor node consists of a computing platform based on a Mica-2 running TinyOS-1. We choose Mica-2 because its computing power and customizability are sufficient for us to solve the problem; because each sensing target requires two sensors and one or two motes, we must minimize sensor node cost. Figure 3(a) shows a mote packaged for field deployment, and 3(b) individual system components and sensors. The sensors themselves are NANMAC D6-60-J J-type thermocouples with hose clamps to attach to the pipeline; §5.2 discusses the care that must be taken to get accurate, calibrated temperature readings. Because the voltage output by thermocouples is quite small (less than 15 mV), we add a custom amplification board to boost this signal 100-fold. Each of our sensor packages is powered by a custom-built thermo-electric generator (TEG) described in §3. Our prototype system is packaged for short-term use and is suitable for use at injection wells; additional explosion-proof (Class I, Division 1) packaging would be required if the system were deployed near production wells.

Software on our sensor node includes our new blockage detection algorithm (§4), We run the sensing algorithm locally on the mote; the base station can relay alerts to the field SCADA system. In addition, we log temperature over the radio to the base station, and locally to flash memory for debugging and long-term analysis. In our field experiments we disable logging to flash as described in §5.1.3, but in operation, we would expect local logging to serve as backup in case of temporary network outages.

The base station bridges the sensor nodes to the field network and SCADA system. In principle, a mote with a wired network connection, or a multi-hop mote network could serve this purpose. We expect a large scale deployment would thus eliminate our base station, connecting directly to the field network. The structure of the field network (mesh or point-to-point, line, steam or solar powered, etc.) is outside the scope of this paper. We do not currently have permission to integrate with the field SCADA system, so for our experiments our base station is a mote that connects directly to a laptop that logs data to disk.

3 Steam Power: Harvesting Thermal Energy

We now describe our "steam-powered" energy generator: the TEG, power conditioning, and its physical mount.

3.1 The Opportunity

One reason steam is injected into the oilfield is to heat viscous oil. Surface temperature (underneath insulation) of steamflood pipes is around 260 °C, while the ambient temperature is typically 0–38 °C. Our insight is that we can convert this near constant 200 °C temperature differential (Δ_{HC}) into electricity through the *Seebeck effect* [37]. Seebeck effect is the thermoelectric phenomenon where the temperature difference between two dissimilar metals in a circuit generates electric current. To our knowledge we are the first to exploit this opportunity for industrial sensing (see §6.1 for full review of related work).

More than energy efficiency, our goal is *energy sufficiency*. Thus, it is more important that we provide enough energy to operate under all conditions, even if thermoelectric generation is not optimally tuned for all temperatures or loads. This goal drives our cost-conscious design (§3.2) to sufficiently power our sensing system (§5.1). In fact, §5.1.3 shows that we can go further to *batteryless operation*. We next describe the components of our energy harvester.

3.2 Thermo-electric harvester Design

We have several system requirements for our thermo-electric power harvesting. First, we need a thermo-electric module that works at the 250 °C plus temperature typical of the steam injection pipes. Secondly we want the TEG module to harvest *sufficient* energy to directly power mote-class devices. Lastly, we want the power-harvesting module to be low-cost to engender dense deployment.

To satisfy the first two requirement we choose the 1261G-7L31-04CQ thermal power generation module from Custom Thermoelectric [37]. This module has a maximum temperature rating of 260 °C and is rated to generate up to 5.9 W under ideal conditions. With an expected load of only 82 mW (a mote transmitting [6]), we expect ample headroom.

Physically, the cold side of our TE-harvester mates to a $5^3/_8 \times 5^3/_8 \times 1^3/_8$ inches aluminum heatsink. We use inexpensive thermal paste on the cold side/heatsink junction, but omit it on the hot side/harvester, since paste rated to higher temperatures is quite expensive. We avoid active cooling for simplicity, and discuss mounting and passive cooling below (§3.4). We do not use energy storage such as batteries to minimize maintenance costs, although we consider capacitor buffering in §5.1.3. We later show that our TEG design meets our application's needs (§5.1.1).

3.3 Power Conditioning the TEG Output

We chose the Heliomote to regulate the power output of our TEG [16]. The low-cost Heliomote provides an efficient charging circuit that uses solar panels to provide power near the 2.4 V to charge two NiMH batteries. We chose the Heliomote because of its commercial availability and our expectation that it could be easily adapted to thermal power.

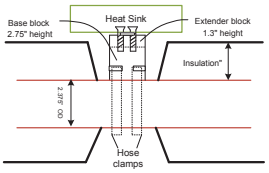
Although we hoped to use the Heliomote directly, differences in voltage and current from solar to thermal sources required modification. Power harvested from an energy source depends on the source's voltage and the load. Our TEG generates optimal power at 1.2 V (Figure 7(b)), much lower than the 2.4 V design point for a Heliomote with solar panel. Unfortunately, the Heliomote cannot charge batteries at 1.2 V, so we instead directly power the Mica-2 using Heliomote's power regulator. With advice from the Heliomote designers, we made minor modifications to directly regulate the Heliomote output voltage from the TEG to 3.0 V allowing the direct powering of a mote, as shown in §5.1.3. This modification removes Heliomote's support for battery backup at low power. To gracefully handle brownout, we enable the brown-out detection on the Atmega128 processor to proactively shutdown if the voltage drops below a user-specified threshold.

Although we adapted the Heliomote to our purposes, we expect a TEG-specific solution could be more flexible and are currently investigating alternatives.

3.4 Mounting Design

TEG effectiveness depends on temperature *differential*, so a complete design must consider coupling to the heat source and a heat sink to dissipate heat to the environment.

We designed a custom TEG mount to meet these goals under several constraints. First, outside pipe diameters in oilfields vary between $2^3/_8$ and $3^1/_2$ inches. Second, most steam pipes are insulated to prevent heat loss. While we deploy our TEG directly on the pipe, we expect it to be abutted by insulation with thickness of 2 to 4 inches; the heat sink must be *above* nearby insulation to get good airflow. Finally, we must securely attach the TEG assembly to the steam pipe.



(a) Design of the TEG pipe-mounting apparatus

(b) Deployed mount, without extender block

Figure 4. Mounting design for TEG

Our solution to the above constraints is shown in Figure 4: we use a rounded base-block with an optional extension to accommodate differing insulation heights. The base block is curved to match the pipe diameter at our target site ($2^3/_8$ inches), and has a height of $2^3/_4$ inches to extend past an expected 2 inches of insulation. An optional 1.3 inch extension can be added to accommodate deployment to locations with thicker insulation. Finally, the base block also has curved grooves going through its one side allowing two hose clamps to mate the apparatus securely to the pipe.

4 Non-invasive Sensing of Pipeline Blockages

We next describe how non-invasive, temperature-based sensing can detect blockages using pairs of upstream and downstream sensors. While the TEG reduces costs to power our application (§3), non-invasive sensing reduces costs of sensing. Below we briefly review the physical effects our sensing exploits, then describe our basic algorithm and extensions to avoid false positives.

4.1 Background: pipeline physics

Our hypothesis is that pipe surface temperature can indicate internal choke blockages. In this section we summarize the physics of fluid flow in the pipe to show how a blockage decreases downstream pressure, which in turn decreasing surface temperature, a phenomenon we can detect.

To understand what happens in the pipe, we must understand what happens when supersaturated steam passes through the choke (see [27] and [9] for general background). The choke has a narrow opening in the pipe (called the choke *bean*) designed to keep steam at critical flow, where the fluid reaches sonic velocity to isolate pressure upstream and downstream of the choke [9]. This isolation is essential for oilfield operation, since downhole conditions (downstream) change, and also for our algorithm, since we detect blockage by observing temperature differences upstream and downstream of the choke (T_u and T_d).

As described in §2.2, scaling inside the pipe, steam impurities, and device wear can all cause blockages, changing cross-sectional size (A) of the choke, written as:

$$A' < A \tag{1}$$

where A' indicates the value after a blockage occurs.

The volume of steam passing in a unit time ($\dot{m}$, the *mass flow-rate*) is determined by the choke aperture size, so a partial blockage reduces steam volume. The Thornhill-Craver choke rate equation shows mass flow for straight-bore chokes [8]:

$$\dot{m} = 73YA\left(1 - \frac{0.00625L}{\sqrt{A}}\right)\sqrt{\rho P_u} \quad (2)$$

The mass flow-rate depends on gas expansion factor (Y), aperture size (A), choke length (L), upstream pressure (P_u) and steam density (ρ), calculated by vapor-phase and liquid-phase specific volumes. Field operations keep ρ constant during normal operation. Choked flow is by definition flow at a fixed pressure, so P_u is constant as well.

From Equation 2, we see that a partial blockage ($A' < A$) reduces flow rate:

$$\dot{m}' < \dot{m} \quad (3)$$

Since steam is compressible, a decrease in mass flow decreases pressure [27]:

$$P'_d < P_d \quad (4)$$

A lower downstream pressure reduces internal steam temperature ($T_{d,i}$) and therefore pipe surface temperature. Experimental data shows this relationship with internal temperature [11], as shown by the the following empirical equation provided by field engineering:

$$T_{d,i} = \frac{7006.3}{9.48654 - \ln\frac{P_d}{144.9}} - 382.55 \quad (5)$$

We know that surface temperatures follow internal ($T_d \propto T_{d,i}$), and a drop in pressure implies a drop in temperature:

$$T'_d < T_d \quad (6)$$

and upstream temperature and pressure are not changed ($T'_u \approx T_u$, because it is choked flow), so we can therefore detect blockage by looking for relative temperature differences:

$$T'_u - T'_d > T_u - T_d \quad (7)$$

The above reasoning suggests why choke blockage is visible in our system. However, oilfields are complex, and choke blockage is not the *only* possible cause of pipe temperature changes. Weather changes on the surface, and downhole pressure changes are both potential sources of noise. Our detection algorithm (§4.2) triggers on *sudden* and *relative* temperature differences, so it should not trigger on surface changes that affect both sensors (such as weather, since the relative differences between the sensors are unchanged), or gradual downhole changes (such as reservoir changes, since they take place over days or weeks).

In this section we summarize how blockage eventually reduces downstream pipe skin temperature and we provide theoretical and empirical equations to prove that. Griston and Abate observe similar phenomenon that smaller choke bean size does not affect upstream temperature much while significantly reduces downstream temperature in their experiments [11]. These results are consistent to our hypothesis that we can use temperature to detect choke blockage remotely. This background is used in our algorithm design to provide good detection accuracy (§5.4).

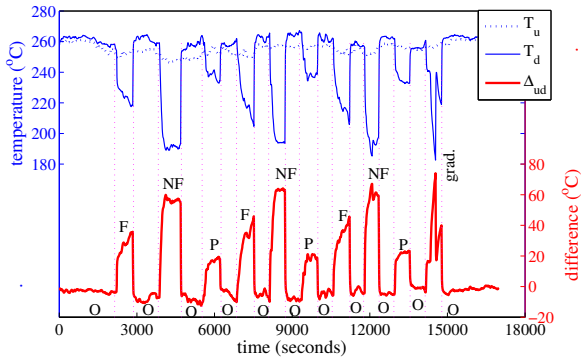


Figure 5. Up- and downstream temperatures (top lines and left scale) and Δ_{ud} in several controlled blockages. A valve emulates the following blockage levels: *F*: full, *NF*: for nearly full, *P*: partial, and *O*: open (no blockage).

4.2 Design of the base algorithm

We next apply our observation of pipe temperature to detect choke blockage (§2.2). Since blockages represent changes in flow behavior, the principle of our algorithm is to look for changes in temperature *upstream and downstream* of a possible point of blockage, then look for *short-term* changes in temperature. Our algorithm adaptively learns typical pipe temperatures from long-term averages.

We study two kinds of blockages, each with its own inference derived from §4.1: For a **partial blockage**, upstream pressure is unchanged while downstream pressure drops as the orifice size is reduced. For **total blockage**, upstream pressure drops as flow stagnates, and downstream pressure drops to ambient, in-pipeline pressure. In both cases, we measure temperature to infer pressure, detecting blockage because of a sudden drop in downstream temperature relative to upstream.

We next show experimental results to illustrate this hypothesis. We present a detailed description and discussion of our field experiments in §5.3. Letters (F, NF, O, P) in Figure 5 indicate the degree of emulated blockage, and vertical dotted lines in the figure show points when we change blockage level. We see that upstream temperature (the dotted line) is relatively stable, although it dips slightly upon full and nearly full blockage. The downstream temperature (the solid blue line) is much more sensitive to pipe status, and the temperature differential (Δ_{ud}, the wide red line) shows ten distinct peaks corresponding to ten controlled full or partial blockages. Full analysis of this experiment is §5.3.

Our algorithm (pseudocode in Algorithm 1, with notation in Table 1) works in two steps. We first compute the short- ($s(\Delta_{ud})$) and long-term ($l(\Delta_{ud})$) history of the difference between upstream (T_u) and downstream (T_d) temperature via exponential weighted moving average. We choose EWMA because it is light weight and easy to implement on our 8-bit mote platform; short- and long-term EWMAs may use separate gains (α_s and α_ℓ). Whenever the difference ($\delta_{s\ell}$) between two history exceed pre-defined threshold (*th_block*), the system declares pipe blockage.

After a blockage is detected, we expect responders to investigate the problem and reset the algorithm after it is cor-

Table 1. Notation used in the description and analysis of blockage detection algorithms.

T_u, T_d	Temperature at Up- and downstream
Δ_{ud}	Up- and downstream temperature difference
$s(\Delta_{ud}), l(\Delta_{ud})$	Short- and long-term Δ_{ud} history
th_norm	normal state threshold
th_block	blockage state threshold
th_maint	upstream maintenance state threshold
α_s, α_ℓ	Short- and long-term EWMA gain
$\delta_{s\ell}$	Short- and long-term history disparity
s	pipe status

rected. By far our system does not transmit alarms back to field surveillance room, which is out of scope of this work. Since in our field tests (§5.3) we artificially induce blockages rapidly (in tens of minutes), we employ two addition rules specifically to aid testing. First, after blockage detected, when two history series converge ($\delta_{s\ell} = 0$), we automatically reset pipe state in order to precede to follow-up tests. Second, we stop updating the long-term history when the pipe is in any non-normal state. While these rules were designed to allow our short-term tests to mimic long-term operation, they do not negatively affect normal operation.

Our algorithm successfully detects pipe blockage, as shown experimentally in §5.3. Our basic algorithm detects problems using two sensors on either side of the blockage site. These sensors could be network-connected, or directly connected to a common controller (as in our implementation). Although this algorithm is correct, upstream maintenance can cause false alarms when pressure for the entire steam system drops. The next section shows how we can employ networked sensors to avoid these false alarms.

Algorithm 1 Blockage detection algorithm.

Require: $T_u, T_d, th\_block, th\_maint, th\_norm, \alpha_s$ and α_ℓ.
Ensure: Pipe state s.

```
 1: s ← NORMAL
 2: while system on do
 3:     Δ_ud ← T_u − T_d
 4:     s(Δ_ud) ← s(Δ_ud) + α_s × (s(Δ_ud) − Δ_ud)
 5:     l(Δ_ud) ← l(Δ_ud) + α_ℓ × (l(Δ_ud) − Δ_ud)
 6:     δ_sℓ ← s(Δ_ud) − l(Δ_ud)
 7:     if (δ_sℓ ≥ th_block) ∧ (s = NORMAL) then
 8:         s ← BLOCKAGE
 9:         print "Pipe blocked"
            {*** below are extensions from §4.3.}
10:     else if (δ_sℓ ≤ th_maint) ∧ (s = NORMAL) then
11:         s ← MAINTENANCE
12:     else if (δ_sℓ ≥ th_norm) ∧ (s = MAINTENANCE) then
13:         s ← STABILIZATION
14:         start timer
15:     else if (timer fired) ∧(s = STABILIZATION) then
16:         s ← NORMAL
17:     end if
18: end while
```

4.3 Avoiding false positives

The above algorithm detects blockages around the target, but regular steam distribution maintenance also changes system pressure and temperature. Our base algorithm is unable to distinguish maintenance from choke blockage, thus incurring false positives. We avoid false positives by employing networked sensor readings from more distant parts

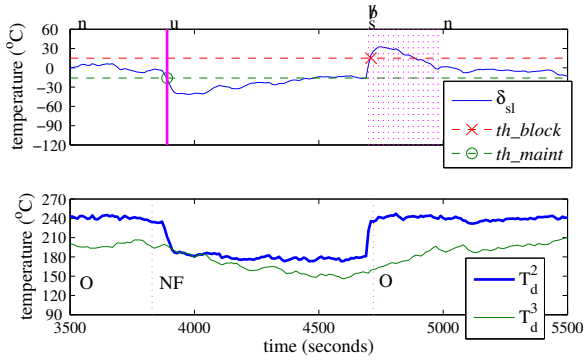


Figure 6. A cause of false positives and our solution. *n, u* or *s* means the pipe is in *normal, upstream maintenance* or *stabilization* state respectively. A ✌ shows a false blockage detection suppressed.

of the steam distribution system. While we could, in principle, record scheduled maintenance events and explicitly disable blockage detection during those times, we strongly prefer a sensor-based solution. If we can infer maintenance at the sensors, we avoid dependencies on manually logged events and the system integration and additional error conditions that such coupling entails. We can also adapt to environmental changes such as seasonal temperature drift. We next describe how we extend our base detection algorithm to distinguish system-wide changes from local blockages.

The lower plot of Figure 6 shows the detail from the second trial of Figure 5 to illustrate how our extension solves a potential false alarm. We distinguish upstream maintenance from blockage by detecting both the start and completion of a maintenance period. We decide maintenance starts if $\delta_{s\ell} < 0$ because of the inertia at the choke making the upstream temperature drop before the downstream (solid vertical line in the upper plot at 3800 s). Likewise, we detect the stop at the next $\delta_{s\ell} > 0$ since the same inertia causes a reverse process (time $\approx$ 4700 s). Our extended algorithm then gives the pipe stabilization time, depicted in the upper plot as dotted shading area, suppressing any following potential $\delta_{s\ell}$ peaks that would otherwise be misinterpreted as blockage (for example, the crosses at 4710 s). We detect blockage by any $\delta_{s\ell}$ that larger than *th_block* but falls out of the stabilization period (i.e. not preceded by a sign of maintenance start). Lines 10 to 17 in Algorithm 1 implement this process.

This kind of false alarm also shows why we require multiple sensors. Our downstream sensor alone can not distinguish the temperature drop incurred by blockage and maintenance. A pair of sensors, on the other hand, can distinguish these events, and also can adapt to environmental changes. In §5.4 we show that this false-positive elimination algorithm successfully distinguishes upstream maintenance from blockage.

4.4 Tuning for different environments

Our algorithm has several parameters that require tuning. We next discuss the parameters and how we can tune them to support not only steam blockages, but also blockages in hot-water distribution networks. In §5.5 we show that, with proper parameters, our work generalizes to blockage detec-

tion in other types of pipeline networks. Currently we configure our system manually; automatic configuration is an area of future work.

The detection thresholds (*th_block*, *th_maint* and *th_norm*) are critical to trade between accuracy, responsiveness and reliability. We assume Δ_{ud} follows a normal distribution $N(\mu, \sigma^2)$. Usually *th_block* is set higher than 3σ, according to 3-sigma rule [40]. The water in PVC pipe has lower temperature and hence we observe less significant Δ_{ud} variance upon anomaly. To make accurate detection, we set both *th_block* and *th_maint* closer to 0. The *th_norm* parameter should be set to a small enough value to ensure hysteresis in our algorithm; we set it by default to 0.

Short and long-term gains (α_s and α_ℓ) determine how our algorithm reacts to noise and blockage. Long-term history should be relatively stable while short-term agile. Pipe material (metal or PVC), fluid type (steam or water), and the ambient environment all affect how quick the temperature reacts to changes in pipe status. With PVC and water, the pipe has better heat insulation than metal, and hence we want to keep long-term history more stable because of the sluggish short-term change. We therefore we propose $1/2$ for short-term EWMA gain, and $1/64$ for long-term in steam blockage detection, while $1/4096$ for water pipelines.

Finally, the stabilization time period helps avoid false positives. We find that a 360 s timer suppresses most noise due to upstream maintenance. The reason is that it takes $\delta_{s\ell}$ about 950 s to subside, and with α_ℓ of $1/64$ and peak usually occurs within the first half of the period. For water pipelines, with a smaller α_ℓ and non-metallic pipe, we set a longer timer because it takes accordingly longer to stabilize.

5 Evaluation

We next evaluate our key claims: that steam-power can support our application, even without batteries; that our sensors are accurate; and the effectiveness of our basic and extended algorithms in the lab, with lower-temperature water pipelines, and in the field. Finally, we show that our system has lower cost than today's systems. Although long-term deployments and full integration with field network are future work, we believe our laboratory testing and field experiments prove our approach is effective.

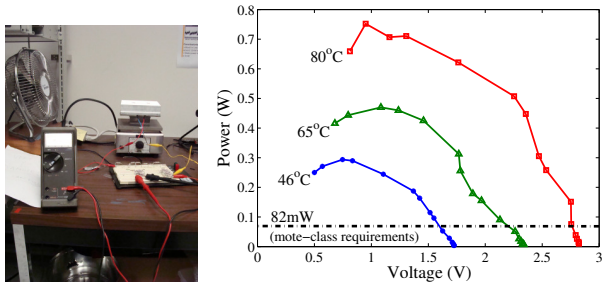
5.1 Long-term energy harvesting and consumption

To evaluate self-sufficient energy harvesting, we first evaluate the capability of the TEG to generate energy compared to the power consumption of our sensing system, concluding that batteryless operation is possible.

5.1.1 Energy Production

In the lab: To measure the energy production of our TEG sandwich, we built a prototype TEG with a flat surface that is placed on a laboratory hotplate (Thermolyne 900, rated to 260 °C). We place the TEG on a small aluminum block about 3.8 cm above the hotplate and use a small external fan to maintain a constant airflow over the heatsink.

To measure power generation we vary the load offered to TEG using a high-wattage resistor network. We then measure the output voltage and current sourced by the TEG as we vary both the load and hotplate temperature (Figure 7(a)).



(a) TEG power measurement harness (b) TEG power curves at different temperature differentials.

Figure 7. Power measurement of the TEG: With optimal loads, but depending on temperature differential, our TEG can provide between 0.3 to 0.8 W.

(We cannot directly measure current because measurement is difficult with its large dynamic range, from 3 to 800 mA.)

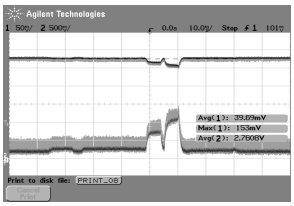
Figure 7(b) shows power characteristics of our prototype TEG in the lab. First, we observe that under most operational conditions (temperature and load), our prototype provides more than the the 82 mW needed to power mote-class devices [6]. Even after a blockage when pipeline temperature falls, the 180 °C temperature (Figure 5) provides sufficient power. Second, we verify that TEG delivers 0.3–0.8 W at the optimal load corresponding to 1.2 V. However, since our actual system does not necessarily operate at optimal load for a thermal harvesting source, we next confirm system operation with in-field experiments.

In the field: We perform in-field tests to confirm that our fielded TEG mount (§3.4) reproduces laboratory experiments, and to evaluate real oilfield conditions, where ambient temperature can reach 50 °C in summer.

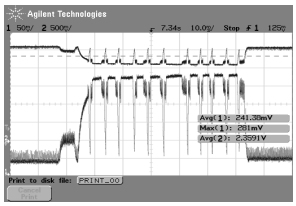
Our sensing platform was under development when we carried out our initial harvesting experiments (Dec. 2009), so we log the open-circuit voltage of our TEG and spot-measure temperature. We find a nearly constant temperature differential of 100 °C and open circuit voltage of 3.8 V. This observation verifies that our pipe-mount TEG can harvest energy comparable or even more than the lab prototype, prompting full system tests described next.

5.1.2 Energy Consumption of Sensing

With our understanding of energy generation, we next turn to energy consumption. Our steam-sensing systems employs a Mica-2 mote with an additional hardware for power conditioning (Heliomote) and thermocouple signal amplification. We therefore measure the power draw of our system running the blockage detection algorithm. Our results show that the average power draw is 70 mW, similar to previous measurements of motes that include radio transmission (82 mW [6]). These values suggest that our TEG can easily cover long-term energy requirements and power our system. Given the large headroom of harvested power, one possible future work is to reduce the size of harvesting unit and hence the total cost of our system. We next evaluate the instantaneous power requirements of our system to understand if batteryless operation is feasible.



(a) Logging to flash (erase-write) causes two load spikes.



(b) At temperature differentials below 80 °C, spikes cause the mote to fail and reboot.

Figure 8. Instantaneous load can cause failure.

Table 2. Energy buffering test at TEG Δ_{HC} = 83.1 °C

capacitor	System operation status
none	sensing and radio fine, but always reboots upon flash logging
1000 μF	sensing and radio fine, but reboots after 2 packets flash logging
3300 μF	sensing and radio fine, but reboots after 12 packets flash logging
4300 μF	sensing, radio, and flash logging always correct
9900 μF	sensing, radio, and flash logging always correct

5.1.3 Batteryless operation?

We have shown that the TEG should provide sufficient energy for long-term operation (§5.1.1). Given the large amount of headroom shown there, we expected that batteryless operation would be straightforward. In fact, we have confirmed that *our system can successfully sense and communicate batteryless*, both in the lab and in the field.

However, we found that some debugging modes of our system require high *short-term* power that cannot be provided by harvested energy alone. Specifically, writing to flash has peak current draws that starve the CPU (peak power of 260 mW, as shown in Figure 8(a)), causing the mote to reboot. TEG power is a function of Δ_{HC} and the absolute temperature; we observe power shortages only for smaller Δ_{HC} and at lower absolute temperatures. We found brown-outs occur at Δ_{HC} around 80 °C (dynamic temperatures are difficult to measure, we estimate measurement accuracy at around $\pm$5 °C). Thus we conclude that short-term, power-intensive operations like flash require significantly additional energy generation headroom for batteryless operation.

Adding minimal energy buffering: A small energy reserve can bridge brief peak power requirements. We therefore evaluate traditional capacitors (1000 to 9000 μF) to support flash logging, while avoiding the maintenance problems of batteries and the cost of supercapacitors. We can use the capacitor both to tolerate flash logging in the field, and provide sensing-and-transmit operation at lower, in both cases operating at smaller Δ_{HC}.

We first add a 1000 μF capacitor and lower Δ_{HC} until the mote stops. Without a capacitor our system reboots at around 80 °C Δ_{HC}, and the mote will not boot at all at 50 °C Δ_{HC}. With the capacitor we are able to sense-and-transmit as low as 60 °C Δ_{HC}. Thus a small capacitor supports operation at about a 20% lower temperature differential.

We next vary the capacitor size to see how much energy buffering is required at around 83 °C. Table 2 shows the correlation between capacitor and system robustness. The scan interval is 10 s, the same as that of our field deployment and we do one radio transmitting and flash logging at the end of every cycle. We find that larger capacitors support power-intensive flash logging, with a 4300 μF capacitor sufficient to support continuous operation at this Δ_{HC}. (There is ample time for the capacitor recharge between 10 s cycles, much larger than the sub-second capacitor recharge time.)

We conclude that while batteryless operation is possible, a small energy buffer is important to support peak loads.

5.2 Sensor and system calibration

Our non-invasive sensing uses custom thermocouples with a custom amplification board. We therefore must calibrate these sensors, as read by the mote, to understand their accuracy over their range of operation (from 0 to 100 °C). We tested each step of the process (temperature to thermocouple output voltage, output to amplified voltage, and voltage to measured ADC values) in both the field and laboratory. We omit validation details here due to space (full details are in our technical report [43]), but report the two key conclusions. First, although there are small non-linearities in each step, our system is linear enough over the target range of operation that we can treat ADC measurements directly as scaled temperature without correction. Second, if we wish to measure absolute temperature, we can apply post-facto calibration.

In addition to sensor calibration, we collected preliminary data in the field in December 2009 to understand operational conditions. These initial measurements at the production oilfield allowed us to confirm thermocouple and TEG operation in the field. They also provided data to parameterize our algorithm and adjust the range and precision of our thermocouple amplification boards.

5.3 Does our detection algorithm work?

In §4, we described our blockage detection algorithm using low-cost, non-invasive sensing, and Figure 5 showed preliminary experimental evidence that it works. We next review the methodology behind that experiment and present additional data to show that our algorithm accurately detects full, nearly full, and partial blockages.

We conducted field experiments to evaluate our algorithm with the deployment shown in Figure 2. Since it is difficult to non-invasively induce controlled blocks in a real choke, and such actions might interfere with production, field engineers manually control a valve to emulate blockages. We believe this emulation models a small orifice restricting steam flow, the key similar physical property, and so we expect our results here to apply to real choke blockages.

We emulate four blockages levels: full (flow rate $\dot{m} = 0$), open ($\dot{m} = 100\%$), nearly full blocked ($\dot{m} \approx 10\%$), open, partially blocked ($\dot{m} \approx 50\%$) and open again; the blockage approximate are best estimates by the field engineers. We repeat the procedure three times and then slowly but continuously shut off the pipe over 9 minutes to observe a gradual blockage. These events are shown by letters and vertical lines in the lower plot of Figure 9.

We collect data with one mote controlling two thermocouples straddling the valve to measure T_u and T_d, as indicated in Figure 2(a) and seen in the top left of Figure 2(b). We also deploy a second mote and pair of thermocouples

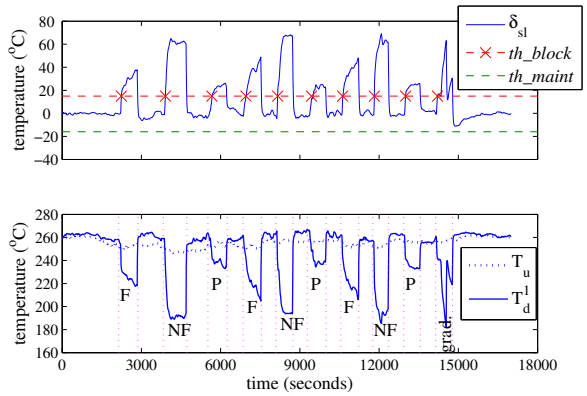


Figure 9. Similar base and extended algorithm results on the thermocouple-pair straddling the valve. The bottom plot shows the raw up- and downstream temperatures with pipe status mapping.

around the actual choke for use in our extended algorithm. A nearby laptop with a mote receiver archives transmissions from both motes. For ground truth, we use a Campbell Scientific CR1k datalogger with eight NANMAC D60-60-J hose-clamp thermocouples, each thermocouple placed adjacent to a mote thermocouple. We operated our system from 12:30pm March 4 to 8:30am March 5, 2010, collecting data for the first twelve hours as described next.

Our field experiment is not problem-free. Our experiment ran for 20 hours, however we collected data for only the first 12 hours. At this time our sensors did not have capacitor assist, so we disabled on-mote flash logging and depended on transmission and logging at the central laptop. Unfortunately, external power for the laptop failed overnight, so we lack data for the last 8 hours. Our stored data includes all controlled experiments and is sufficient to validate our algorithms. Also, analysis of timestamps the next morning confirms our batteryless prototype operated continuously. Finally, although we ran our algorithms live, an electrical coupling error between the thermocouples at the valve spoiled our live run of the algorithm. Fortunately, our ground-truth temperature sensors in the same location recorded the complete data. We therefore replay this data post-facto for the analysis presented in this section. Our sensor testing and calibration (§5.2) suggests that this substitution does not alter our conclusions. Due to the limited field experiment time, we use the same setup for training and evaluation. However, our evaluation is reliable because our water experiment uses a different setup and reports consistent result (§5.5).

Figure 9 repeats the raw data from Figure 5, but separates $\delta_{s\ell}$ and shows where it crosses the thresholds to indicate detections ($th\_block = 15, th\_maint = -16$, $\alpha_s = 1/2$ and $\alpha_\ell = 1/64$). It shows that our algorithm correctly detects *all* three levels of the blockages, capturing all nine $\delta_{s\ell}$ peaks. It correctly detects the final gradual blockage as well.

We draw four further observations from the bottom plot. First, consistent to our hypothesis, the upstream temperature is relatively constant while the downstream one is sensitive to pipe status. Second, surprisingly, nearly-full blockages yield the larger downstream temperature drops, even more than full blockage, while partial blockage has the smallest difference. We believe this behavior is because a nearly closed valve starts choking flow, reducing downstream pressure, but a full blockage completely isolate downstream pipe from the upstream steam network, leaving it occupied by back pressure from reservoir through wellbore. Field engineers confirm this intuition. Third, our algorithm is highly responsive. For example, it takes averagely around 6 samples (60 s) to correctly detect problems when $th\_block = 15$. Finally, carelessly configured parameters would trigger false alarms. An over-aggressive th_block value (≤ 5 °C) would confuse normal temperature fluctuation for pipe anomaly. In principle, one could measure typical temperature variation, for example by making sure that the threshold well outside typical variance.

In all, we conclude that our base algorithm is capable of detecting three different degrees of blockage, detecting nearly-full and full-blockage most easily, with the same threshold setting. According to field engineers, partial blockage is exceptionally rare in real field. However, when we run our algorithm over the thermocouple pair straddling the choke (data omitted due to space), our base algorithm triggers false alarms as we discussed in §4.3. The same blockage threshold captures seven out of ten positive a $\delta_{s\ell}$ peaks in Figure 10. We next evaluate how our extended algorithm can avoid the choke false positives.

5.4 Evaluating avoidance of false positives

In §5.3, we show that our base algorithm has good accuracy on emulated blockages. However, application of our basic algorithm to readings of our second sensor-pair (T_d^2 and T_d^3 around the choke in Figure 2(a); downstream of the sensor-pair T_u and T_d^1 around the valve) shows a number of false alarms, even though there were no blockages at that location. In effect, our experiments at the sensor-pair around the valve emulate maintenance on the steamflood network, changing the pressure for all downstream sensors.

We next re-analyze the data from both sensor pairs sensor to show that our extended algorithm in §4.3 prevents these false alarms. We expect the extension works at both locations, successfully detecting target blockage at the valve and suppressing our the maintenance-like effects at the choke.

The extended algorithm yields exactly the same result as the base one over the valve-straddling pair with the same configuration. This proves that the extension does not impair our algorithm performance.

Figure 10 shows how the extended version avoids false positives at the choke-straddling sensor pair (T_d^2 and T_d^3). The base algorithm with the same threshold as before ($th\_block = 15$) captures seven out of ten positive $\delta_{s\ell}$ peaks upon upstream valve operation because of the asynchronous temperature drop on both sides (§4.3). However, the extended version triggers *no* false alarms any more (seven ɓ tags). With th_maint, the system detects all ten upstream maintenance events (tagged as "u"). The stabilization period (dotted stripes shading tagged as "s" in the upper plot) successfully suppresses all positive $\delta_{s\ell}$ peaks following upstream maintenance and $\delta_{s\ell}$ rises back above 0.

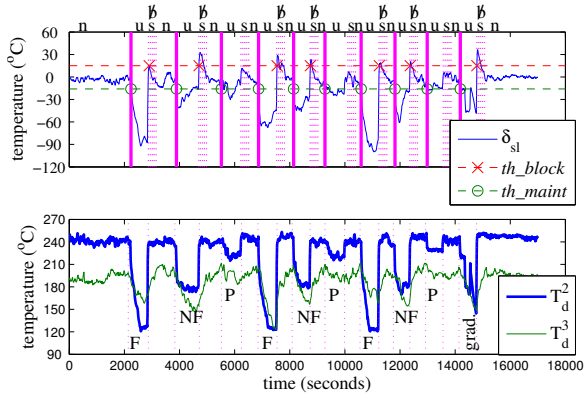
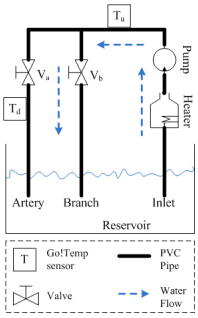
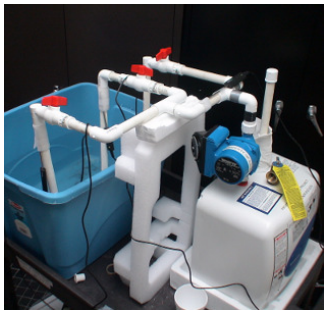


Figure 10. Extended algorithm result on the sensor pair straddling the choke with *th_block* = 15, *th_maint* = −16, $\alpha_s = 1/2$ and $\alpha_\ell = 1/64$. The bottom plot shows the raw up- and downstream temperature with pipe status mapping.



(a) Logical view of the prototype.



(b) Physical view of the prototype.

Figure 11. In-lab, water-based pipeline prototype.

Our field tests carefully evaluate our extended algorithm in two different scenarios—an emulated blockage, and an emulated, upstream maintenance event. These experiments demonstrate that our extended algorithm yields good accuracy and triggers minimum false alarms.

5.5 Generalizing to water pipelines

We have shown our algorithm detects blockages in steamflood pipelines. Our algorithm and the concept of detecting sudden temperature drops is not specific to steam networks, but can apply to other kinds of fluid flow such as water pipelines. However, water has very different physical properties than steam—it is incompressible and at operates at much lower pressure and temperature. These differences require re-tuning our algorithm and make detection of partial blockages difficult. We next show that, after adjusting parameters, our algorithm can detect full blockages in a hot-water distribution system, and therefore can generalize to applications in other domains and industries.

To evaluate a second network, we constructed a simple hot water network. Shown in Figure 11, our small testbed consists of a tankless water heater with a recirculation pump; a plastic, lidless tank; and a small network of PVC pipes and valves. This experiment precedes our mote implementation, so it uses the same algorithms but running on a PC, with

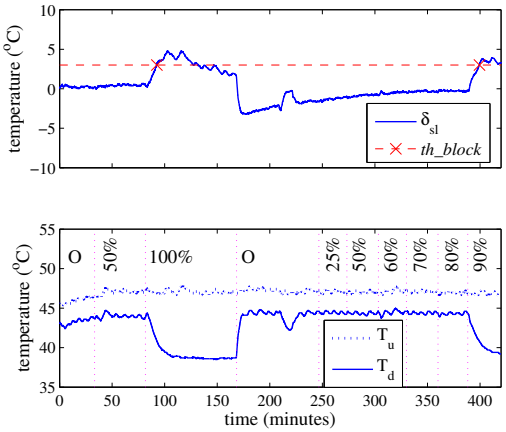


Figure 12. Blockage detection in a water pipeline: raw up- and downstream temperatures (bottom), with emulated blockage shown by annotations (vertical lines): *100%*: full blockage, *X%*: partial blockage by turning V_a X% off, and *O*: open (no blockage).

data from Go!Temp, USB-based temperature sensors [38]. As in the steamflood pipeline, we emulate blockages by controlling valves (V_a in Figure 11). While our steam experiments use fieldable hardware in an industrial pipeline network, our hot water experiments are laboratory-based with experimental hardware and a simple pipeline. However, they demonstrate the generality of our algorithms in a very different medium, showing our approach can apply to other cases where blockage points can be anticipated.

The lower plot of Figure 12 shows raw up- and downstream pipe temperatures change during a full blockage between 82 and 168 minutes. Vertical lines in the figure indicate when we change the emulated blockage to the approximate percentage shown. The upper plot in Figure 12 shows that our algorithm, with parameter (*th_block* = 3), captures both $\delta_{s\ell}$ peaks (minutes 93 and 400) caused by full and 90% blockages. We show one representative example of three consistent experimental runs.

From this experiment, we see that our algorithm successfully detects full and nearly full blockage in this very different water network. The downstream temperature drops significantly, converging to ambient temperature after full blockage, from 44 °C to 39 °C in about 25 minutes, while upstream temperature remains constant.

Second, we observe that it is difficult to detect partial blockage in this water pipeline, unlike the steam network. The temperature change upon partial blockage (minutes 34–82 and 247–389) is insignificant, except for nearly full (90%) blockage at the end of the experiment. We believe partial blockages are difficult to detect in water networks because water is incompressible and hence partial blockage has little affect on fluid pressure. Another reason is that PVC pipes in our testbed have a much lower thermal conductivity (0.19 W/mK) than copper pipes do (401 W/mK), and therefore this network shows much greater hysteresis.

Figure 13. Invasive flow sensing with solar panel for power.

5.6 System cost

Our goal is to reduce cost, so we next show the savings from small, non-invasive sensors with a thermal energy harvesting. We consider capital and operational costs of sensor system installation. Many industrial sites, including our test site, have existing field wireless networks, so we do not consider costs to deploy the field network (such network hardware is typically US$200 or less per node).

Currently deployed systems: The circular pipe fitting with two large protruding taps on the left of Figure 13 is a typical invasive pressure sensor currently in use. This system requires installation of pressure sensor taps between adjoining pipelines and a data acquisition unit tethered to a large solar panel and battery. Cost of data acquisition is high enough that measurement units are transported from site to site by field technicians to manually collect limited samples of data. Installation costs of differential pressure sensing taps and power to the sensor can be as high as US$20,000 per site, with mechanical costs from $8,000 to $16,000 and calibration an additional $3,000 to $4,000. Cost of the hardware itself, including pressure sensors, a datalogger, a solar panel, and battery packs can range from $13,000 to $20,000.

Given such high costs, a great deal of oilfield monitoring today is manual. For example, many surface oilfields contain hundreds or thousands of oil producing wells, and steamflood secondary production may employ half as many injection wells. However, not all the steam injection sites are equipped with the monitoring points due to high cost of installation. In addition, only a limited number of sites can be monitored at any time, since expensive instrumentation is shared over many wellheads rather than permanently installed. Today it is typical for a field technician to visit each site monthly, attaching a sensor and datalogger to the sensing points to collect data for several minutes. The technician uploads the data on return to the office, or via the field wireless network. With travel time, one data collection requires multiple person-hours per site.

Human-driven sensing reduces capital costs, with the sensing points installed when the line is built and the cost of the pressure sensor is amortized over many measurement points. However, human-in-the-loop makes the operational costs quite high: easily hundreds of dollars per measurement.

Table 3. Pipe temperature variation along time.

	T_u (°C)		T_d^1 (°C)		ambient (°C)	
time (s)	μ	σ	μ	σ	μ	σ
Noon	258	2.7	261	2.1	19	1.4
Evening	261	0.8	262	1.0	17	0.7
Midnight	258	1.0	263	1.0	10	0.23

This high cost for each measurement discourages frequent measurements and so prevents easy detection of problems before they occur. In addition, recurring costs will rise with upward trends in future labor costs.

Our sensor-network-based system: By comparison, the capital cost of our system is quite modest. Our prototype unit consists of a Mica-2 (US$100) for control, a modified Heliomote ($125) for power conditioning, a custom amplifier board ($50) and two thermocouple sensors ($70 in total) for sensing, the TEG ($50) and a custom heatsink and mount assembly ($200). Despite the system being a research prototype and so not benefiting from economies of scale, component costs are less than $600. Volume would reduce these costs, although technical support would add to them.

More importantly, our approach can dramatically lower deployment and operational cost. Deployment can be done by a technician in an hour or two (deployment time for our field experiment was two hours, and we expect future deployments to be half that). Since deployment is non-invasive, steam flow need not to be interrupted and new plumbing is not required; since it is self-powered, electrical expertise is not required. The primary technical skills are SCADA integration and standard oilfield and steamflood safety training. We estimate deployment cost at around $300. Besides, we see *no* recurring operational costs for sensing.

We believe these significant reductions in both acquisition and operation will allow much greater deployment of sensing with systems such as ours than are possible today.

5.7 System robustness

Although we showed our system works in the lab and for short-term field deployments, a long-term, real-world deployment raises a number of questions about system robustness. We next look at three questions related to system reliability under different conditions. Do environmental changes or sensor location affect our algorithm accuracy? Is our radio communication reliable enough?

We first look at our system to evaluate if environmental temperature changes affect TEG and sensing performance. During our overnight field deployment the TEG-powered motes operated continuously, even while ambient temperature ranged from 9 to 22 °C. This observation confirms diurnal temperature changes are small compared the potential energy in the steam network.

We also see no diurnal affects on sensed pipe temperature. Table 3 records that pipe temperature remains relatively constant in spite of changes in ambient temperature. In fact, upstream co-generation power cycle or distribution branch configuration is likely to have more impact. In all, both our algorithm and hardware platform should work independently from diurnal amplitude.

Second, the sensor location can effect system operation because of choked fluid flow (§4.1). Figure 14 compares

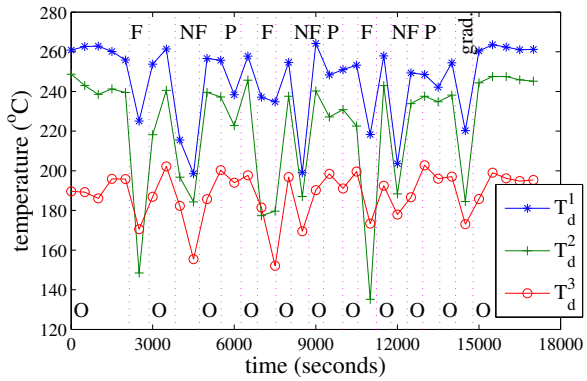


Figure 14. Aggregated all three downstream temperature.

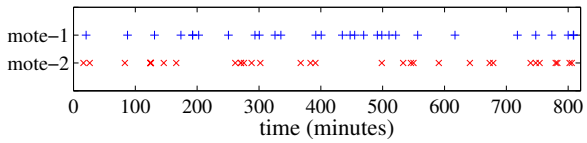


Figure 15. Mote radio packet loss distribution. Each marker represents one sample missing from data set.

temperature fluctuation at different downstream spots (after the valve) upon blockage. Letter labels and vertical lines are defined in Figure 5. Here we show running 50-sample means to smooth short-term variation. We see distinct temperature changes at all three locations for all levels of blockage, but the temperature change varies depending on proximity to the choke. However, contrary to T_d^1 and T_d^3, T_d^2 shows that temperature upon full blockage is lower than that of nearly full blockage. The reason is that T_d^2 is farther downstream than T_d^1 to the valve but still before the actual choke, experiencing a smaller transient due to choked flow. This analysis suggests that sensor placement can affect results. Since our system can adapt to both steam and water pipelines, we are confident parameter adjustments can accommodate such variation, but automating the process is future work.

Finally, we want to verify that the communication between our system and the base station is reliable. Since we hide the mote antenna inside the pelican box to prevent potential environmental hazard to our device, we expect packet loss, despite a short base-to-sensor distance of 4 m with build-in B-MAC [28]. In general, the total packet loss rate is low, 0.63% (31 losses in 4914 transmissions) at the valve mote and 0.67% (33 losses in 4914) at choke mote. Figure 15 shows loss is generally uniform and uncorrelated across motes, if slightly burst in time. If we consider the low data rate of the system – one tens-of-byte data packet per 10 s for current deployment and occasional blockage alarm packets for full system, we conclude our wireless communication channel is adequate and robust enough.

6 Related Work

Our work builds on prior research results in energy harvesting, change-point detection algorithms, and sensor networks for industrial monitoring.

6.1 Energy Harvesting Systems

Energy harvesting for sensor network has been an active area of research. Systems have considered sources of energy including light, wind, vibration, heat, magnetic, and radio [7]. Several efforts have focused on sustainably powering low-power sensor nodes by scavenging ambient energy from both traditional sources such as sunlight, vibration, and mechanical [29, 41, 25], and more exotic methods as body heat, radio fields, and multiple energy sources [21, 30, 32]. Our work builds on these approaches, but is the first, to our knowledge, sensornet system that sustains itself entirely from the very phenomena it senses (industrial heat).

Solar is a good source of energy, and several projects explore solar-powered sensornets [17, 39]. Heliomote was the first system to integrated solar-power and power conditioning to drive mote-class hardware [29]. We use a modified Heliomote in our work to condition the output of our TEG.

Other work considers battery alternatives. Prometheus replaces rechargeable batteries with supercapacitors to reduce conversion loss [14]. The AmbiMax platform further increases efficiency by matching source- and load-impedance with maximum-power-point tracking, and added multi-modal energy harvesting with solar and wind [26]. These platforms employ battery or supercapacitor to isolate energy harvesting from consumption. We show that this buffer can be eliminated or replaced with a standard capacitor.

Several systems explored large-scale thermal energy harvesting. Researchers have considered radioisotope thermal generators (10–300 W for spacecraft), waste heat in central heating systems (25 W, [31]), to exploiting automobile waste-heat (4 W, [22]), and even micro-generators for the soil-to-air thermal gradient (0.35 W, [20]). Our work provides a cost-efficient and energy-sufficient solution for powering an embedded system. Other researchers have investigated harvesting thermal energy for storage in energy buffers. Mateu et al. harvest about 5 mW using the thermal gradient between human body and ambient temperature and store it in an NiMH battery [21]. Sodano et al. argue that TEG modules can generate more power and charge more quickly than piezo-electric system under typical conditions [34]. Our work explores batteryless or standard capacitor energy buffering, and integrates thermal harvesting with sensing to provide energy sufficiency for a specific application.

In lower-power thermal energy harvesting, the Micropelt TE-node is most closely related to our work. The Micropelt platform low-power (sub-10 mW) sensor node [23] with an internal 100 μF capacitor for energy storage, with harvesting from a custom thermo-electrical generator [4, 3]. Our work differs in that we use a general purpose sensor platform, evaluating the potential and trade-offs for batteryless operation. We also explore a general purpose platform (running TinyOS), allowing exploration of a variety of sensor fusion algorithms, and we demonstrate that our application can be energy-sufficient.

6.2 Algorithms for Change-point Detection

Many real-time monitoring systems use abrupt detection [2] or change-point detection [1] to detect problems in

observed data. Several change-point detection algorithms exist. In our work we focus on exponential-weighted moving average (EWMA) for change-point detection because it admits very lightweight implementations, making it well suited to mote-class platforms. Our detection algorithms are inspired by several prior systems built on EWMA [13, 36, 18].

Several sensornets build on the simple EWMA algorithm from TCP [13]. Trifa et al. develop an adaptive alarm call detection system for yellow-bellied marmots, using EWMA to estimate environment noise [36]. Our work uses similar concepts to detect significant change in pipe skin temperature using EWMA. Kim and Noble propose EWMA-based algorithms to optimize streaming estimation of network capacity [18]. One of their algorithm, called flip-flop filter, keeps both agile and stable EWMA and switches between the two to find the best baseline. Although our algorithm also maintains two EWMAs, we directly compare these two traces to detect sudden changes in pipeline temperature.

6.3 Pipeline monitoring systems

SCADA systems have long been used for pipeline monitoring. Traditional SCADA systems use simple in-situ sensors and centralized decision making [24, 10], while our approach instead shifts detection algorithms into intelligent, communicating sensor nodes.

Prior work in sensornet pipeline monitoring usually assumes low-temperature, incompressible fluid [35, 19, 15, 33]; our work instead focuses on high-temperature, high pressure, compressible steam. This change in fluid allows us to use temperature, instead of vibration or acoustic sensing. PIPNET prototypes an urban sewage monitoring system based on wireless sensor and demonstrates that they can detect water leakage by vibration frequency analysis [35]. NAWMS instead focusing on personal water usage [19]. Their hardware is similar to ours, but we differ in sensing modality and use of EWMA instead of their linear-programming-based algorithm. Jin and Eydgahi [15] and Sinha [33] both explore acoustic pipeline monitoring. Jin and Eydgahi focus on a general sensor network platform while Sinha's work is mainly about instrumentation and calibration; we instead focus on blockage detection.

Zhu's work is closet to ours, showing the feasibility of temperature monitoring for blockage detection of pulverized coal injection system [44]. He uses temperature observation from thermometers mounted on branch pipes in his detection algorithms. Similar to our approach, his algorithm differentiates pipe skin temperature and compares the resulting Δ_{ud} against pre-configured thresholds. Unlike his work, we use EWMA and multiple sensors to adapt to changes, avoiding most hard-coded thresholds. Finally, we use inexpensive and portable hardware (less than US$600), while his system is centralized and likely to be much more expensive.

We have previously explored the potential of sensor networks in oilfield production systems [42]. While that work suggests the potential, this paper demonstrates a field-tested system, evaluates specific sensing algorithms, and demonstrates that the whole system can operate on steam-power.

7 Conclusions

We have described a system for steam-powered sensing to detect pipeline blockages. Our system is designed to dramatically reduce deployment costs, allowing instrumentation of industrial processes that are today considered cost-prohibitive. We developed two novel approaches to reduce cost: "steam-powered" thermal energy harvesting, where the sensornet is powered by the phenomena being sensed and can operate batteryless or with a standard capacitor. Second, we developed an algorithm to detect problems in pipelines using non-invasive sensing. We have demonstrated the effectiveness of this system and its components through laboratory tests and field experiments. Although we have developed this system to match the needs of sensing blockage in steam distribution networks, the principles of thermal energy harvesting and non-invasive sensing apply to a range of industrial sensing applications.

Acknowledgments

We would like to thank Jonathan Friedman for his help using ATLA Labs' Heliomote. We thank Greg LaFramboise, Charlie Webb, Mohammad Heidari for their input on Chevron's business requirements and their assistance in our field experiments; David Menda for input into the project, and Iraj Erhagi, Mike Hauser for their guidance as co-directors of CiSoft.

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

[3] H. Bottner. Micropelt miniaturized thermoelectric devices: small size, high cooling power densities, short response time. In *Proc. of the 24th International Conference on Thermoelectrics*, pages 1 – 8, Clemson, SC, USA, June 2005.

[4] H. Bottner, J. Nurnus, M. Braun, J. Wollenstein, F. Volkert, and A. Schubert. MicroPelt: state of the art, road map and applications. Technical report, Infineon Technologies AG, Munchen, Germany, 2004.

[5] R. M. Butler. *Thermal Recovery of Oil and Bitumen*. Prentice-Hall, 1997.

[6] M. Calle and J. Kabara. Measuring energy consumption in wireless sensor networks using GSP. In *Proc. of the IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 1–5, Sept. 2006.

[7] A. Chandrakasan, R. Amirtharajah, S. Cho, J. Goodman, G. Konduri, J. Kulik, W. Rabiner, and A. Wang. Design considerations for distributed microsensor systems. In *Proc. of the IEEE 1999 CICC*, pages 279–286, May 1999.

[8] H. L. Cook and H. F. Dotterweich. *Report on calibration of positive flow beans manufactured by Thornhill-Craver Company, Inc.* Texas College of Arts and Industries, 1946.

[9] Crane Company. *Flow of Fluids through Valves, Fittings, and Pipe*. Technical Paper No 410, Crane Co., 1988.

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

[11] S. Griston and T. A. Abate. Field test of tapered-bore for steam flow control. In *SPE Western Regional Meeting*, pages 269–283, Anchorage, AK, USA, May 1996. SPE.

[12] H. Hayashi, Y. Takabayashi, H. Tsuji, and M. Oka. Rapidly increasing application of intranet technologies for SCADA (supervisory control and data acquisition system). In *IEEE/PES T&D 2002: Asia Pacific*, volume 1, 2002.

[13] V. Jacobson. Congestion avoidance and control. In *Symposium proceedings on Communications Architectures and Protocols*, pages 314–329, Stanford, CA, USA, Aug. 1988.

[14] X. Jiang, J. Polastre, and D. Culler. Perpetual environmentally powered sensor networks. In *Proc. of the 4th International Symposium on Information Processing in Sensor Networks*, pages 463–468, Los Angeles, CA, USA, Apr. 2005.

[15] Y. Jin and A. Eydgahi. Monitoring of distributed pipeline systems by wireless sensor networks. In *Proc. of the 2008 IAJC-IJME int'l Conference*, Nashville, TN, USA, Nov. 2008.

[16] A. Kansal, J. Hsu, S. Zahedi, and M. Srivastava. Power management in energy harvesting sensor networks. *ACM Transactions on Embedded Computing Systems*, 6(4), Sept. 2007.

[17] A. Kansal and M. B. Srivastava. An environmental energy harvesting framework for sensor networks. In *Proc. of the 2003 International Symposium on Low Power Electronics and Design*, pages 481–486, Seoul, Korea, Aug. 2003. ACM.

[18] M. Kim and B. Noble. Mobile network estimation. In *Proc. of the 7th Annual International Conference on Mobile Computing and Networking*, pages 298–309, Rome, Italy, July 2001.

[19] Y. Kim, T. Schmid, Z. M. Charbiwala, J. Friedman, and M. B. Srivastava. NAWMS: nonintrusive autonomous water monitoring system. In *Proc. of the 6th ACM Conf. on Embedded Network Sensor Systems*, pages 309–322, Nov. 2008.

[20] E. E. Lawrence and G. J. Snyder. A study of heat sink performance in air and soil for use in a thermoelectric energy harvesting device. In *Proc. of the 21st Int'l Conf. on Thermoelectrics*, pages 446–449, Long Beach, CA, USA, Aug. 2002.

[21] L. Mateu, C. Codrea, N. Lucas, M. Pollak, and P. Spies. Human body energy harvesting thermogenerator for sensing applications. In *Proc. of the 2007 Int'l Conf. on Sensor Technologies and Applications*, pages 366–372, Oct. 2007.

[22] K. Matsubara. Development of a high efficient thermoelectric stack for a waste exhaust heat recovery of vehicles. In *Proc. of the 21st International Conference on Thermoelectrics*, pages 418–423, Long Beach, CA, USA, Aug. 2002.

[23] Micropelt GmbH. TE-power node: Self-sufficient wireless sensor system thermoharvesting explorer. Micropelt product datasheet, Mar. 2010.

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

[25] J. A. Paradiso and T. Starner. Energy scavenging for mobile and wireless electronics. *Pervasive Computing, IEEE*, 4(1):18–27, 2005.

[26] C. Park and P. H. Chou. AmbiMax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes. In *Proceedings of the 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, pages 168–177, Reston, VA, USA, Sept. 2006.

[27] R. Perry and D. Green. *Perry's Chemical Engineers' Handbook, Sixth Edition*. McGraw-Hill, 1984.

[28] J. Polastre, J. Hill, and D. Culler. Versatile low power media access for wireless sensor networks. In *Proc. of the 2nd international conference on Embedded networked sensor systems*, pages 95–107, Baltimore, MD, USA, 2004.

[29] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava. Design considerations for solar energy harvesting wireless embedded systems. In *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*, IPSN '05, Los Angeles, CA, USA, 2005.

[30] S. Roundy, B. P. Otis, Y. Chee, J. M. Rabaey, and P. Wright. A 1.9GHz RF transmit beacon using environmentally scavenged energy. In *Digest IEEE Int'l Symposium on Low Power Electronics and Devices*, Seoul, Korea, 2003.

[31] M. D. Rowe, G. Min, S. G. K. Williams, A. Aoune, K. Matsuura, V. L. Kuznetsov, and W. Li. Thermoelectric recovery of waste heat-case studies. In *Proc. of the 32nd Intersociety Energy Conversion Engineering Conference*, volume 2, pages 1075–1079, Honolulu, HI, USA, July 1997.

[32] A. Sample and J. R. Smith. Experimental results with two wireless power transfer systems. In *Proc. of Radio and Wireless Symp.*, pages 16–18, San Diego, CA, USA, Jan. 2009.

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

[34] H. A. Sodano, G. E. Simmers, R. Dereux, and D. J. Inman. Recharging batteries using energy harvested from thermal gradients. *Journal of Intelligent Material Systems and Structures*, 18(1):3–10, 2007.

[35] I. Stoianov, L. Nachman, S. Madden, and T. Tokmouline. PIPENET: a wireless sensor network for pipeline monitoring. In *Proc. of the 6th International Conference on Information Processing in Sensor Networks*, pages 264–273, Apr. 2007.

[36] 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 *Proc. of the 12th International Symposium on Artificial Life and Robotics*, AROB '07, Beppu, Japan, Jan. 2007.

[37] `http://www.customthermoelectric.com/`. Custom Thermoelectric.

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

[39] T. Voigt, H. Ritter, and J. Schiller. Utilizing solar power in wireless sensor networks. In *Proc. of the 28th Annual IEEE International Conference on Local Computer Networks*, pages 416–422, Bonn/Königswinter, Germany, Oct. 2003.

[40] Western Electric. *Statistical Quality Control Handbook*. Western Electric Corporation, Indianapolis, Inc., 1956.

[41] C. B. Williams and R. B. Yates. Analysis of a micro-electric generator for microsystems. In *Proc. of the 8th Int'l Conf. on Solid-State Sensors and Actuators and Eurosensors IX.*, volume 1, pages 369–372, Stockholm, Sweden, June 1995.

[42] S. Yoon, W. Ye, J. Heidemann, B. Littlefield, and C. Shahabi. SWATS: Wireless sensor networks for steamflood and waterflood pipeline monitoring. *IEEE Network Magazine*, 25(1):50–56, Jan. 2011.

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

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