Mapping Hidden Water Pipelines using a Mobile Sensor Droplet

TED TSUNG-TE LAI, National Taiwan University
WEI-JU CHEN, National Taiwan University
YU-HAN TIFFANY CHEN, National Taiwan University
POLLY HUANG, National Taiwan University
HAO-HAU CHU, National Taiwan University

This study presents several extensions to our previous work on the PipeProbe system, which is a mobile sensor system for identifying the spatial topology of hidden water pipelines (i.e., non-moldable pipes such as copper and PVC) behind walls or under floors [Lai et al., 2010]. The PipeProbe system works by dropping a tiny wireless sensor capsule into the source of a water pipeline. As the PipeProbe capsule traverses the pipelines, it gathers and transmits pressure and angular velocity readings. Through spatio-temporal analysis of these sensor readings, the proposed algorithm locates all turning points in the pipelines and maps their 3D spatial topology. This study expands upon previous research by developing new sensing techniques that identify variable-diameter pipes and differentiate 90-degree pipe turns from 45-degree pipe bends.

Categories and Subject Descriptors: C.3 [Special-Purpose and Application-Based Systems]: Real-time and Embedded Systems, Signal Processing System

General Terms: Design, Experimentation, {Performance

Additional Key Words and Phrases: Wireless sensor networks, Mapping Water Pipeline, Sensor Inference, Constraint Satisfaction

ACM Reference Format:

1. INTRODUCTION

All humans need water to live. Unfortunately, water is also a scarce resource in many parts of the world. Proper monitoring, maintenance, and protection of the water infrastructure system that delivers water to humans are critical to public health, economic development, and resource conservation. This paper focuses on residential water infrastructure. Houses are often equipped with an extensive water pipeline network that distributes water to different water-using fixtures and appliances throughout the home, including bathroom toilets, kitchen faucets, garden sprinklers, and washing machines. It is therefore unfortunate that plumbing is ranked as one of the ten most frequently occurring problems in homes [American Society of Home Inspectors]. A study conducted in 1999 [AWWARF 1999] shows that leaks ranked as one of the top five water usage items (13.7% of residential per capita water usage) in the U.S. Leaking pipes are one of the most common problems in plumbing [Dump 2003], and hidden leaking pipes often cause extensive damage to floors, walls, and belongings in a home.

The first step in fixing leaking pipes is to locate the leak for further inspection. When leaking water pipes are hidden inside walls and underneath floors, identifying their location without direct inspection is very difficult, especially when the original diagram of the pipeline layout is missing. In this case, finding pipeline locations becomes guesswork that often requires a brute-force method, such as knocking down walls and stripping floor coverings. This problem created an opportunity to develop PipeProbe, a mobile sensing probe that is dropped into the source of a water pipeline. As it traverses the pipeline, PipeProbe collects the sensor readings (i.e., pressure and angular velocity) necessary to reconstruct the 3D spatial topology of the pipeline. Unlike the tradi-

An earlier version of this study appeared in the Proceedings of the 8th International Conference on Embedded Networked Sensor Systems (SenSys’10).

Author's address: email: tedlai@csie.ntu.edu.tw

Permission to make digital or hardcopies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credits permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.

©2010 ACM 1544-3558/2010/05-ART1 $10.00
tional brute-force approach, the PipeProbe system is a non-destructive method of mapping and locating indoor water pipelines that requires no alteration to the water pipeline infrastructure. Since leaks often occur where pipe tubes join together (including pipe diameter changes, L-connection pipe, and T-connection pipe), mapping the locations of these pipeline connection points is a crucial step in pipe inspection.

Three previous projects used wireless sensor network technologies to monitor water pipes: the NAWMS project [Kim et al. 2008], the PIPENET project [Stoianov et al. 2007], and HydroSense [Froehlich et al. 2009]. The NAWMS project detected and located pipe leaks by attaching vibration sensors to the pipe surface. Similarly, the PIPENET project [Stoianov et al. 2007] monitored water flow and detected leaks by attaching acoustic and vibration sensors to large bulk-water pipelines and pressure sensors to normal pipelines. HydroSense [Froehlich et al. 2009] employed a single endpoint sensing solution in which the amount of water outflow from each water outlet could be uniquely estimated using pressure wave signatures. Unlike these projects, the PipeProbe system adopts a mobile sensing approach in which a tiny mobile sensor travels inside of the water pipeline infrastructure while remotely performing on-the-spot data collection near potentially problematic locations. Alvarado et al. [Alvarado and Youcef-Toumi 2005] developed a robotic fish under a foot long that closely mimics a real fish’s natural swimming motion. This robofish is equipped with sensors to detect environmental pollutants. Its one-foot size is considerably larger than that of the proposed PipeProbe capsule, and its motor requires a 2.5-5W external power source.

This study makes three important contributions to research on this topic:

- Rather than fixing sensing points in the utility infrastructure, PipeProbe adopts a mobile sensing approach in which a mobile sensor travels and performs on-the-spot data collection at different places.
- This study develops a novel localization method to estimate the 3D spatial topology of the capsule-traversed pipelines based on the pressure and rotation graphs collected and computed by the PipeProbe system. Experimental results indicate that the proposed mobile sensing approach produces a high-precision 3D map of the pipeline with centimeter-level accuracy.
- Since the PipeProbe capsule was designed to model a water droplet, its physical movement leverages the force inside of the pipeline infrastructure for propulsion. This means that no motoring is necessary to power its movement, which increases the PipeProbe capsule’s energy-efficiency and allows it to operate on only 15 mA of current. A tiny lithium button cell battery can keep the PipeProbe capsule operating for over 1 kilometer at a water flow rate of 15 centimeters per second.

The rest of this paper is organized as follows. Section 2 presents the design principles for PipeProbe’s mobile sensing approach. Section 3 explains the design and implementation of PipeProbe’s sensing capsule and the data collection process. Section 4 details the PipeProbe’s operation and how data processing maps the spatial topology of the pipelines. Section 5 describes the experimental testbed and scenarios. Section 6 presents the experimental results. Section 7 discusses the application of PipeProbe to detect pipe diameter changes and 45-degree pipe bends. Section 8 discusses the limitations of this study, and their possible solutions. Section 9 reviews related work. Finally, Section 10 concludes the study and suggests directions for future research.

2. PIPELINE PROFILING

The PipeProbe system is a micro sensing hydro molecule that observes the wall-embedded pipelines from within by flowing along the pipeline, much like the myriad other hydro molecules in the fluid system. The current PipeProbe prototype consists of a tiny wireless sensor node packaged in a water-proof spherical shell measuring 4 centimeters in diameter. PipeProbe works in two stages: (1) data collection stage, and (2) data analysis stage. In the data collection stage, the PipeProbe capsule traverses a water pipeline and collects data from pressure and gyroscope sensors; in the data analysis stage, the system analyzes the sensor readings and derives the 3D spatial topology of the traversed pipeline.
PipeProbe operates as follows. First, the capsule is dropped into the main water inlet of a home or building. When an outlet (i.e., a faucet) is opened, the force of the resulting water flow pushes the capsule through the different possible paths for the connected water pipes. As the capsule flows through the water pipelines, it logs the sensed pressure and angular velocity data to an EEPROM (Electrically-Erasable Programmable Read-Only Memory). A radio within the PipeProbe capsule transmits the sensor data buffered in the EEPROM to a PC-connected base station. Alternatively, when the PipeProbe capsule flows out of a water outlet, users can manually transfer sensor data from the capsule’s EEPROM to a PC. Finally, the data analysis part of the PipeProbe system computes and maps the 3D spatial topology of the hidden pipeline.

If the PipeProbe capsule flows out of a water outlet during the data collection stage, it can be reinserted into any water inlet and reused for additional data collection. Multiple trips enable the discovery of diverse pipeline branches, which are then used to produce a full map. Multiple measurements of the same flow path make it possible to filter out noise in the data and increase the accuracy of the 3D spatial topology reconstruction.

### 2.1 Vertical Movement

The water pressure sensor is based on the Pressure Principle, which states that the static pressure at any sensing point in a confined liquid is produced by the weight of the liquid above that point. In other words, the pressure depends only on the height of the liquid above the sensing point and the liquid density. If a liquid is confined in a tank, the pressure at any sensing point in the tank is given by

\[ P = \rho gh \]  \hspace{1cm} (1)

where \( P \) is the pressure, \( \rho \) is the density of the liquid (in this case, water), \( g \) is the acceleration due to gravity, and \( h \) is the height of the sensing point. With constant gravity and density, pressure is proportional to the height of the sensing point.

The Pressure Principle makes it possible to estimate the movement of a pressure-sensing capsule on the vertical plane from the pressure difference between two sensing points. Consider a vertical pipe with length \( L \). The difference in pressure readings between the top and bottom of the vertical pipe is \( \Delta P \). From Eq. (1), the pressure difference is

\[ \Delta P = \rho g \Delta h \]  \hspace{1cm} (2)

Since \( \Delta h \) is the length of the vertical pipe, \( L \) can be derived as follows

\[ L = \frac{\Delta P}{\rho g} \]  \hspace{1cm} (3)

While Poiseuille’s law states that the pressure is not constant in a flowing liquid, we describe in Section 8 how this affects our calculations.

### 2.2 Horizontal Movement

To measure horizontal pipe length, PipeProbe uses another approach, based on angular velocity from a gyroscope sensor, to estimate the capsule’s movement on the horizontal plane. This combination of length and direction of horizontal movement allows full 2D horizontal mapping. This study first describes how to determine a pipe’s horizontal length from the capsule’s traversal time on a horizontal plane, and then describes how to identify horizontal turning points from the capsule’s angular and rotation velocity.

To determine the length of a horizontal tube (h-tube), the sensor is cased in a spherical shell whose density (i.e., weight over volume) equals water’s density. This allows the capsule to flow through the pipes as if it were part of the fluid system. As a result, the estimated water flow velocity approximates the capsule’s own velocity. Thus, measuring the duration that the pressure sensor’s readings remain constant reveals the length of the corresponding horizontal pipe \( L \), which is estimated by multiplying the capsule’s flow velocity \( v \) by the flow time \( t \).
This equation assumes that the pipe has a uniform diameter; thus, water flow velocity in the horizontal plane remains constant across all connecting pipes. To derive the water flow velocity, fix the valve at the water’s inlet and then divide the amount of water entering the inlet by the area of the pipe’s intake surface. Since home water pipes come in several sizes [PPFA 2006], Section 7.1 discusses how to relax this assumption using additional sensors on the PipeProbe capsule.

To locate a pipe’s horizontal turning points, a gyroscope on the PipeProbe capsule measures its angular velocity. By combining angular velocity with the rotation angle, the PipeProbe system can distinguish a left horizontal turn, i.e., with the positive 90-degree rotation angle, from a left horizontal turn, i.e., with a negative 90-degree rotation angle.

3. DATA COLLECTION

The PipeProbe capsule was prototyped with the Eco wireless sensor mote [Park and Chou 2006]. The Eco mote is an ultra-compact and low power wireless sensor node measuring only 13 mm (L) × 11 mm (W) × 7 mm (H) and weighing 3 grams (including battery). It consumes less than 10 mA in transmission mode (0 dBm) and 22 mA in receiving mode. Its maximum data rate and RF range are 1Mb/s and 10 meters, respectively. The Eco’s small form factor and low power consumption make it ideal for the PipeProbe capsule, which requires a tiny size to allow it to flow freely within a water pipeline. The Eco mote has a flexible-PCB type expansion port with 16 pins. This expansion port includes two digital I/O pins, two analog input lines, a serial peripheral interface (SPI), RS232, and voltage inputs for a regulator and battery charging. An Intersense MS5541C pressure sensor [MS5541C] is wired to the Eco mote via the SPI protocol. The MS5541C sensor measures a pressure range from 0 to 14 bars with an accuracy of ±0.2% FS and ±0.05% FS. Drawing less than 5 μA operating current, the MS5541C enables the Eco mote to sample frequently without draining too much battery power. The MS5541C requires an external oscillator at the frequency of 32.768 kHz for sensor ADC. A SG3030JC external oscillator was chosen to fulfill this requirement. The pressure sensor samples the water pressure at a rate of 33 Hz. Figure 1 illustrates the components in the PipeProbe capsule. After the pressure sensor and oscillator were integrated with the Eco mote, they were enclosed with a waterproof plastic casing. The pressure sensor was exposed outside of the casing to maintain contact with the water. This packaging went through 4 iterations of design. The first prototype (Fig. 2) used a cylindrical casing. However, the cylinder shape (which has non-uniform surfaces from different perspectives) proved problematic, incurring varying moving velocities as the capsule tumbled through the pipes. The 2nd iteration switched to a spherical shape to solve this problem.

The weight of the capsule was the next problem encountered: the electronics and the case were too light. The density difference between the capsule and water resulted in a constantly floating capsule, whose traveling velocity was particularly unstable. The 2nd prototype failed to behave like a water droplet, i.e., traveling at the same velocity as the current. Thus, a counterweight was added to the third prototype so that the capsule would neither float to the surface nor sink to the bottom while sitting in water. Given a target density at 1g/cm³, the ideal weight was 33.51 grams for a 2-cm radius sphere.

However, significant variations in pressure readings indicated that the 3rd prototype still required further modification. Specifically, the pressure sensor would turn arbitrarily as the capsule rotated through the pipelines. To minimize jitter in the pressure sensor readings, the 4th prototype (Fig. 3) fixed a counterweight to the bottom hemisphere of the capsule. This design minimized the amount of capsule rotation around the z-axis, and mimicked a roly-poly toy, or a tumbler, which has a heavier hemisphere below its center. When the tumbler is pushed down, it quickly rights itself. Simple tests confirmed that creating a heavier hemisphere in the capsule significantly reduced the amount of flipping and stabilized the pressure sensor’s readings.

Figure 4 shows the 5th and final prototype, which uses a gyroscope module (Fig. 1(f)) to detect horizontal turns, thus giving it 3D pipeline mapping capability. The gyroscope module includes a STMicroelectronics LISY300AL chip [LISY300AL] that measures the rotational motion along the
yaw (z) axis with a $\pm 300^\circ/s$ range and outputs an analog voltage. The gyroscope module is fixed at the very top of one of the capsule’s hemispheres. This allows the gyroscope to lie flat on the horizontal plane to obtain an accurate z-axis measurement. The final prototype also includes a tail-like fin to stabilize the capsule’s movement on the horizontal plane and to re-align the capsule’s heading in the presence of turbulent water flow within the pipeline.

Fig. 1. The PipeProbe capsule and its parts: (a) waterproof plastic casing, (b) hemisphere used to stabilize flow velocity, (c) Eco mote, (d) a pressure sensor, (e) an oscillator, (f) a gyroscope sensor, and (g) counter weight.

Fig. 2. The initial capsule prototype.

Fig. 3. The 4th capsule prototype.
4. DATA PROCESSING

Figure 5 illustrates the four steps in the analysis of the data collected from the PipeProbe capsule. (1) A median filter smoothes out and remove noises from the pressure time-series data. (2) Turn detection performs a spatial-temporal analysis of the pressure and gyroscope time-series data to detect all vertical and horizontal turning points along the capsule’s flow path. (3) Sensor data alone cannot determine the precise location of all turns. Thus, layout mapping determines the unknown coordinates of these turns by modeling them as a constraint satisfaction problem in which the constraint specifies that the coordinates of these intermediate turns fall on a path between the known coordinates of the inlet and outlet. Repeated measurements from multiple mapping trips are aggregated to remove noisy outliers and enable more accurate reconstruction of the 3D spatial topology of pipelines. (4) Solving the constraint satisfaction problem may generate multiple topological solutions. To find the correct spatial topology, the PipeProbe system first uses the spatial constraints within a home’s walls to eliminate unreachable positions. Beacon listeners were placed on walls to clarify path ambiguity as necessary. The listener on the wall closest to where the PipeProbe capsule actually flows would record the highest received packet rate. These four steps are elaborated below.

![Data analysis diagram]

4.1 APPLYING A MEDIAN FILTER TO PRESSURE READINGS

Median filtering is a common technique for removing noise in image processing, and is applied here to smooth the pressure signal. This process first divides the pressure signal into windows of ten pressure samples, and computes the median of the pressure values within each window. These
medians form the skeleton of the smoothed signal. The true pressure signal is often segment-wise linear. Thus, it is possible to reconstruct the intermediate data points of the smoothed signal by linear interpolation of the consecutive medians. To illustrate, Fig. 6 shows a raw pressure signal where the x-axis represents the time the pressure sensor is sampled and the y-axis represents the pressure reading at the time. Applying the median filter produces a smooth pressure signal (Fig. 7).

Fig. 6. A raw pressure signal before applying the median filter.

Fig. 7. A smoothed pressure signal before applying the median filter.
Fig. 8. Three types of turns detected by the PipeProbe system. (a)(b) contain two h-v-turns, where the pipeline turns from a horizontal plane to a vertical plane and the vertical turning angle is restricted to a 90-degree upward or downward angle. (c) shows a v-h-turn, where a pipeline turns from a vertical plane to a horizontal plane and the horizontal turning angle has unrestricted freedom from 1 to 360-degrees. (d) shows an h-h-turn, where a pipeline turns and stays on the same horizontal plane and the horizontal turning angle is restricted to a right or left 90-degree angle.

4.2 Turn Detection

A water distribution pipeline infrastructure consists of multiple rigid tubes and joints (i.e., turning points). Section 2 describes the general approach to determine tube length. This section first defines different types of turning points in the PipeProbe system and then describes the corresponding turn detection algorithms.

The PipeProbe system detects three types of turning points in a pipeline:

- h-v-turn \((t, p, \theta_z)\) or horizontal-to-vertical turn: Figure 8(a)(b) shows two examples of horizontal tubes making an \(\theta_z\) vertical turn, either upward or downward, along the z-axis. At this vertical turning point, PipeProbe measures the pressure reading \(p\) at time \(t\). \(\theta_z\) is restricted to either a negative 90-degree depression angle or a positive 90-degree elevation angle, i.e., \(\theta_z = \{-90^\circ, 90^\circ\}\). From consultation with a master plumber, this 90-degree vertical turning restriction follows the conventional residential pipeline layout guide. This convention is also consistent with the example piping layouts recommended by PPFA [PPFA 2006] for the four most common house types.

- v-h-turn \((t, p, \theta_{xy})\) or vertical-to-horizontal turn: Figure 8(c) shows an example of a vertical tube making an \(\theta_{xy}\) horizontal turn. At this horizontal turn, PipeProbe measures the pressure reading of \(p\) at time \(t\). \(\theta_{xy}\) has an unrestricted 360-degree freedom of movement on the horizontal plane, i.e., \(\theta_{xy} = \{1^\circ, \ldots, 360^\circ\}\).

- h-h-turn \((t, \theta_{xy})\) or horizontal-to-horizontal turn: Figure 8(d) shows an example of a horizontal tube making an \(\theta_{xy}\) horizontal turn at time \(t\). \(\theta_{xy}\) is restricted to either a positive 90-degree left angle or a negative 90-degree right angle, i.e., \(\theta_{xy} = \{90^\circ, -90^\circ\}\). Since 90-degree pipe joints are the most commonly found (or only available) joints in water pipeline supply stores, this study focuses on mapping pipelines that make 90-degree turns.

This study proposes v-turn and h-turn detection algorithms to identify and locate the three turn types above. The v-turn detection algorithm locates (1) v-h-turns and (2) h-v-turns by analyzing...
changes in pressure readings. The h-turn detection algorithm identifies (3) h-h-turns by processing and integrating angular velocities from a gyroscope to determine horizontal rotation. The following subsections describe the details of these two turn detection algorithms.

4.2.1 V-Turn Detection

The v-turn detection algorithm identifies v-h-turns and h-v-turns from the smoothed pressure signals obtained in the previous step. This algorithm also computes pipe lengths and turn directions. When the capsule is moving vertically, the pressure increases or decreases linearly over the distance traveled. In contrast, when the capsule is moving horizontally, the pressure level remains constant regardless of the distance traveled. (1) Each turning point on the pressure graph marks a v-turn in the physical pipeline topology. (2) The distance traveled between two adjacent v-turns is the length of a vertical tube (v-tube). These two mapping rules are explained in detail as follows.

H-v-turns, v-h-turns and their connected tubes are often hidden behind a vertical 2D wall space. Changes in the pressure signal occur as the PipeProbe capsule traverses an h-v-turn or a v-h-turn. The red flow path in Fig. 9(a) shows a pipeline containing a downward h-v-turn, followed by a v-h-turn and another downward h-v-turn. Figure 9(b) illustrates the pressure graph recorded as the capsule traverses this pipeline structure. This pressure graph shows a steady rise in pressure readings after the capsule completes its first downward h-v-turn, i.e., turns from horizontal movement (constant pressure) to downward movement (increasing pressure). The rise in pressure readings stops when the capsule makes a v-h-turn, i.e., turns from downward movement (increasing pressure) to horizontal movement (constant pressure). Finally, the pipeline structure makes a downward h-v-turn, i.e., turns from horizontal movement (constant pressure) to downward movement (increasing pressure). By recognizing different pressure changing shapes in the pressure graph, the v-turn detection algorithm locates the turning points and determines the upward/downward direction of v-h-turns and h-v-turns.

Fig. 9. This figure demonstrates how the v-turn detection algorithm works. (a) shows an example pipeline (only the red part) containing two h-v-turns and one v-h turn. (b) shows the corresponding pressure graph collected by the PipeProbe capsule.
The v-turn detection algorithm includes the following two steps. (1) Compute the derivatives from the smoothed pressure graph using a sliding window. The horizontal path corresponds to the derivative of the smoothed data equal to zero. The derivative of the vertical path may be either larger or smaller than zero, depending on the direction of the vertical path. (2) After segmenting the horizontal and vertical paths and identifying the direction of vertical movement, pinpoint the exact turning point at the intersection of the zero and non-zero derivative segments.

There are a number of candidate points for the intersections. A simple solution is to identify the data point exhibiting the greatest change in the derivatives. However, this method is sensitive to pressure sensing noise. To minimize estimation error, it is necessary to define a set of candidate points for each intersection. When the derivative shows a vertical-horizontal movement, include the data point that gives the largest derivative change and the four preceding data points in the candidate set. When the derivative shows a horizontal-vertical movement, include the largest change point and the four subsequent data points. The experiments in this study revealed that a window size of 5 samples provided the best results.

There are $K$ candidate sets for $K$ intersections. Each intersection is an h-v-turn or a v-h-turn in the physical pipelines. With 5 data points for each candidate set, there are $5^K$ possible combinations for the best solution. Derive the best linear fit for each combination by regression each of the segments. By testing all combinations and computing the mean square error of the individual data points for the best linear fit, it is possible to identify the combination that minimizes the sum of the mean square error over all segments. Figure 10 shows the turn detection results obtained from Fig. 9(b).

The following discussion provides an example of the v-turn detection algorithm detecting the blue-colored pipe segment depicted in Fig. 9. First, the turning point notation in Section 4.2 specifies the sensor data collected at the two turning points: h-v-turn$_2$ and v-h-turn$_3$. For example, h-v-turn$_2$(8.27 sec, 1,012 mbar, -90º) means that the PipeProbe capsule senses a pressure reading of 1,012 bar at the time point 8.27 seconds, with a turning angle of -90º inferred from changes in the pressure signals.

\[ \text{h-v-turn}_2(8.27 \text{ sec, 1,012 mbar, } -90^\circ) \]
\[ \rightarrow \text{v-h-turn}_3(11.46 \text{ sec, 1,046 mbar, } \theta_{y_3}) \]

Equation (3) determines the length of the blue v-tube between h-v-turn$_2$ and v-h-turn$_3$: the 34 mbar pressure difference (1,046 mbar − 1,012 mbar) between the two turning points approximates to a 40.34 cm drop in vertical height. Therefore, the corresponding v-tube is denoted as follows:

\[ \text{h-v-turn}_2(-90^\circ) \rightarrow \text{v-tube}_2(40.34 \text{ cm}) \rightarrow \text{v-h-turn}_3(\theta_{y_3}) \]
4.2.2 H-Turn Detection

The h-turn detection algorithm identifies h-h-turns and their horizontal turning angles ($\theta_{xy}$) based on pressure and gyroscope sensor readings. H-h-turns and their connected tubes are often hidden under floors and above dropped ceilings. Since 90-degree pipe joints are the most commonly found (or the only available) joints in water pipeline supply stores, the h-turn detection algorithm focuses on detecting 90-degree right and left h-h-turns, i.e., $\theta_{xy} = \{90^\circ, -90^\circ\}$. The h-turn algorithm includes the following three steps: (1) identifying horizontal tubes, (2) applying a threshold-based filter to remove noise from the gyroscope’s angular velocity readings, and (3) calculating the capsule’s rotation rate and identifying the h-h-turn. Figure 11(a) shows an example pipeline structure, i.e., the red flow path, containing a v-h-turn followed by two h-h-turns. The following discussion illustrates how the h-turn detection algorithm detects these h-h-turns.

The first step of the h-turn detection algorithm identifies h-tubes from the pressure readings (Fig. 11(b)), as described in Section 4.2.1. When the PipeProbe capsule travels along a horizontal plane, it detects little or no pressure difference because the height of the capsule remains unchanged.

The second step filters out noise in the raw angular velocity data and keeps only those angular velocity readings in which the capsule passes through an h-h-turn, yielding the correct rotation rate. This filtering of the raw angular velocity graph (Fig. 11(c)) occurs in two stages. First, any high angular velocity readings recorded during PipeProbe’s traversal of v-tubes are considered noise because h-turns by definition do not occur in v-tubes. Second, a simple threshold-based filter removes small random noises from the raw angular velocity readings. As the capsule flows inside h-tubes, its gyroscope sensor may measure relatively small angular velocity fluctuations due to water turbulence within the tube. Experiments showed that a threshold value of $\pm 100$ deg/sec can effectively filter out angular velocity noises from the gyroscope. Therefore, if the angular velocity is within $\pm 100$ deg/sec, it can simply be ignored. Figure 11(d) shows the resulting filtered angular velocity graph.

The third step integrates the filtered angular velocity graph (Fig. 11(d)) to obtain the rotation rate (Fig. 11(e)). Interestingly, experiment results show that when a PipeProbe capsule makes a 90-degree left h-h-turn, its rotation angle reveals a unique angular velocity pattern – first exhibiting a high positive signal (i.e., a positive value that corresponds to leftward angular velocity) followed by a low negative signal (i.e., a negative value corresponding to rightward angular velocity). This “high-positive-low-negative” angular velocity pattern matches how the PipeProbe capsule makes a right h-h turn due to the tail-like fin design of the final prototype. First, the water flow at the turning joint makes the capsule over-rotate to the right. The capsule then corrects its heading by making a moderate rotation in the reverse-left direction. On the other hand, when the capsule makes a 90-degree right h-h-turn, it exhibits a “high-negative-low-positive” angular velocity pattern. Last but not least, h-h-turn detection selects the peak positive/negative signal from the filtered angular velocity data for left/right turns on the horizontal plane.

The following discussion illustrates how the h-turn detection algorithm detects the green-colored pipe segment in Fig. 11. First, the turning point notation in Section 4.2 is specifies sensor data recorded at two turning points: h-h-turn$a$ and h-h-turn$b$. For example, h-h-turn$a$(12.5 sec, 90°) means that the PipeProbe capsule sensed a turning angle of 90° based on changes in the pressure signals at the time point of 12.5 seconds.

$$h\text{-}h\text{-}turn_a(12.5\text{ sec, }90^\circ) \rightarrow h\text{-}h\text{-}turn_b(17.4\text{ sec, }-90^\circ)$$

Applying Eq. (4) and a water flow velocity of 24 (cm/sec) gives the length of the green h-tube between h-h-turn$a$ and h-h-turn$b$. That is, the 4.9 second time difference (17.4 sec − 12.5 sec) between the two turning points multiplied by the water flow velocity of 24 cm/sec approximates 117.6 cm of horizontal pipe length.

$$h\text{-}h\text{-}turn_a(90^\circ) \rightarrow h\text{-}tube_a(117.6\text{ cm}) \rightarrow h\text{-}h\text{-}turn_b(-90^\circ)$$
Fig. 11. This figure shows how the h-turn detection algorithm works. (a) is an example pipeline structure (only the red part) containing a v-h-turn followed by two h-h-turns, (b) is the corresponding pressure graph collected by the PipeProbe capsule, (c) is the raw angular velocity graph produced by the gyroscope sensor on the PipeProbe capsule, (d) is the noise-filtered angular velocity graph from (c), and (e) is the rotation angle of capsule graph calculated from (d). The water flow velocity was 24 cm/sec.

4.3 Layout Mapping

The turn detection algorithm in Section 4.2 produces the following turn-tube sequence:

\[ \ldots \text{turn}_i(\theta_i) \rightarrow \text{tube}_i(L_i) \rightarrow \text{turn}_{i+1}(\theta_{i+1}) \rightarrow \text{tube}_{i+1}(L_{i+1}) \rightarrow \ldots \]

$L_i$ is the length of the tube. For most vertical and horizontal turns, $\theta_i$ is a known value ($\pm 90^\circ$ vertical angle) determined by sensing positive or negative pressure changes in the v-turn detection algorithm or by sensing positive or negative 90-degree rotation angles in the h-turn detection algorithm. A special turn with an unknown horizontal angle, $\theta_{xy}$, occurs when an h-tube is preceded by a v-h-turn such as the v-h-turn in Fig. 11. That is, $\theta_{xy}$ can be any value in $\{1^\circ \sim 360^\circ\}$ and it will be solved by the constraint satisfaction algorithm described as follows.

The layout mapping step produces a 3D spatial diagram by analyzing the turn-tube. First, the known position $(p_0)$ of the starting point (i.e., the water inlet) and the known position $(p_n)$ of the end point (i.e., the faucet outlet) are inserted at the beginning/end of this tube-tube sequence.

\[ \text{inlet}(p_0) \rightarrow \text{turn}_i(\theta_i) \rightarrow \text{tube}_i(L_i) \rightarrow \ldots \rightarrow \text{outlet}(p_n) \]

Next, layout mapping transforms this turn-tube sequence into a constraint satisfaction problem. The model constraint is that the pipeline network must start from $p_i(x_0, y_0, z_0)$, i.e., the position of the inlet into which the PipeProbe capsule is dropped, and then move through intermediate vertic-
al/horizontal pipe tubes of various lengths to finally reach \( p_n(x_n, y_n, z_n) \), i.e., the position of the outlet where the PipeProbe capsule flows out.

It is possible to deconstruct a pipeline structure into layers by cutting it from each h-v turn. Hence, each layer begins with a v-h turn followed by one or more h-tubes. The following equations describe the x-, y-, and z-axis movement on one layer:

\[
x = \sum_{i=0}^{m} (L_i^{h-tube}) \cos(\theta_{xyi}) \\
y = \sum_{i=0}^{m} (L_i^{h-tube}) \sin(\theta_{xyi}) \\
z = \pm (L_i^{v-tube})
\]

where \( L_i^{h-tube} \) denotes the i-th horizontal tube in a layer, \( L_i^{v-tube} \) denotes the i-th vertical tube in a layer, and \( m \) is the number of h-h turns in a layer.

Summing up all the x-axis movements from all n layers of pipeline structures connects the inlet’s starting x-position \( (x_0) \) to the outlet’s ending x-position \( (x_n) \). This summary yields the following x-axis constraint satisfaction equation, with similar constraint satisfaction equations derived for y- and z-axes.

\[
x_n = x_0 + \sum_{\text{all layers}} \sum_{i=0}^{m} (L_i^{h-tube}) \cos(\theta_{xyi}) \\
y_n = y_0 + \sum_{\text{all layers}} \sum_{i=0}^{m} (L_i^{h-tube}) \sin(\theta_{xyi}) \\
z_n = z_0 + \sum_{\text{all layers}} (\pm (L_i^{v-tube}))
\]

Another way to understand the above constraint satisfaction equations is as follows. Since h-tubes only lie on the horizontal \((xy)\) plane and v-tubes vary only the elevation \((z\)-axis\) of the pipeline, constraint satisfaction equations can be specified individually on the x-, y-, and z-axes. In other words, chaining and summing all positive and negative x-axis movements from all h-tubes should define the pipeline from the starting inlet’s x-position, i.e., \( x_0 \), to the outlet’s x-position, i.e., \( x_n \). Similarly, chaining and summing all positive and negative y-axis and z-axis movements from h-tubes and v-tubes should define the pipeline from the starting inlet’s y-position and z-position, i.e., \( y_0 \) and \( z_0 \), to the outlet’s y-position and z-position, i.e., \( y_n \) and \( z_n \).

This constraint satisfaction problem is formally defined as follows. Let \( X_1, X_2, \ldots, \) and \( X_n \) be a set of variables, and let \( C_1, C_2, \ldots, \) and \( C_m \) be a set of constraints. Each variable \( X_i \) has a nonempty domain \( D_i \) of possible values. The goal is to find all possible solutions under all constraints. Formulate the variables, values, and constraints as follows:

- **Variable:** \( \{\theta_{xyi} | i = 1 \ldots n\} \), where \( n \) is the number of turns with unknown turning angles
- **Domain:** \( \{1^\circ, 2^\circ, \ldots, 360^\circ\} \)
- **Constraints:** Equations (8) and (9)

With the starting point of the pipeline structure fixed, the ending point must be equal to the location of the water faucet.

The implementation of this constraint satisfaction problem utilizes a 360-ary tree data structure. The tree branches out for each vertical-horizontal turn. Each child node represents a different turn degree and each node of the 360-ary tree tracks the corresponding coordinate after the turn. Thus, at least one of the leaf nodes should arrive at the outlet’s coordinate. Each path from the root to a leaf represents a possible pipeline layout. After constructing the 360-ary tree, simply scan all the leaf nodes to find the closest match(es) to the outlet’s coordinates. The search space can become very large in some complicated pipeline infrastructures. One method to reduce the search space is
to integrate wall layout information as an additional constraint in the constraint satisfaction problem. If the current child node position is not inside the wall, then backtrack to its parent node and find other legal assignments. This method is called backtracking search for constraint satisfaction problem in the AI area. It is also possible to use some other heuristic search algorithms from traditional AI literature.

4.4 Ambiguity Elimination

Solving the constraint satisfaction problem in layout mapping generates multiple possible solutions for some tube-tube sequences. Consider the pipeline structure in Fig. 12(a) and its measured pressure graph in Fig. 12(b). Since there are two identical h-tube length segments, the constraint satisfaction generates 360 possible solutions (i.e., a v-h-turn can be 1 to 360-degrees), all of which satisfy the inlet/outlet positional constraints. Since most water pipelines are hidden on a flat wall plane, all but two solutions are likely. The two possible solutions (Fig. 13) are that the pipeline travels on the left side or the right side of the wall.

The PipeProbe system resolves such ambiguities through additional mapping trips, where listening devices are placed near the ambiguous paths obtained from the previous mapping trip. In Fig. 13, a listener is attached to each wall location closest to the right and left paths. This figure shows that the ideal placements of the two listeners are at mirror locations where they are furthest apart. Then, two listening devices listen in for packets broadcasted from the capsule as it passes by. The listening device on the correct flow path receives many more packets than the listening device on the incorrect flow path. In other words, these two paths are disambiguated by the received packet rates of the two listening devices. Table I shows an experiment result to distinguish the ambiguity.

![Fig. 12. An example pipeline structure (a) and its pressure graph (b) produce the two possible pipeline topologies depicted in Fig. 13.](image)

![Fig. 13. Two possible water pipeline topologies satisfy the starting position and ending position constraints and produced by the pressure graph in Fig. 12(b).](image)

Table I. Received packet rate for distinguishing path ambiguity in the pipeline structure shown in Fig. 13. The path receiving the higher received packet rate is the correct path.

<table>
<thead>
<tr>
<th>Path</th>
<th>Received packet # / Total transmission packet #</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path 1</td>
<td>121/352</td>
</tr>
<tr>
<td>Path 2</td>
<td>13/352</td>
</tr>
</tbody>
</table>
5. TESTBED

Figure 14 shows the pipeline testbed used to evaluate the PipeProbe system. Transparent pipes (measuring 5cm in diameter) were used to enable direct observation of how well and consistently the PipeProbe capsule flowed through the pipeline in all five prototype versions in the iterative design-test-analyze process. The testbed measured 18 cm x 140 cm x 345 cm, with 51 transparent pipes and 21 valves (with yellow or red handles) forming a pipeline network with a 3x2 non-uniform grid on one vertical and two different horizontal travel paths. An input water source was attached to the plastic bin at the testbed’s upper right corner. Thus, the upper right corner marks the starting point of all flow paths for our experiment. Figure 15 shows the length of each pipe tube.

Opening and closing different combinations of the 21 valves generated different flow paths with varying lengths and turn points. Figure 17 shows the 12 test scenarios used in the evaluation. Each scenario was tested 6 times, i.e., the PipeProbe capsule made six mapping trips over the same flow path. Figure 15 shows the length of interconnecting pipe tubes and the positions of valves. For example, opening all the top valves and all the left valves generates the simple flow path of test#1 (Fig. 17) with 1 turning point and a traversal length of 320 centimeters. The test best included four possible end points.

Data collection for each of the twelve test scenarios involved the following steps. First, the input water source was turned on to fill the tubes with water. Second, the valves were set to produce a particular flow path. Third, the water faucet was opened to generate a continuous flow at a fixed rate. There are multiple ways to control the water flow rate. A simple method is to calculate the amount of time to consume N liters of water given a fixed input flow rate and pipe diameter. Note that only one faucet was opened at a time to generate a particular flow path. Fourth, the PipeProbe capsule was dropped into the water inlet. The PipeProbe capsule gathered and wirelessly transmitted sensor readings at a rate of 20 Hz while traveling inside the pipeline. Finally, the PipeProbe capsule was retrieved as it flowed out of the water outlet.

Fig. 14. The experimental testbed for evaluating the PipeProbe system. 51 transparent tubes formed a 3-D non-uniform grid testbed. 21 valves with yellow handles were also installed. By opening and closing different valve combinations, different capsule flow paths and test scenarios were generated for evaluating the PipeProbe system.
Fig. 15. A diagram showing the lengths (cm) of 51 pipe tubes and the locations of 21 valves.

Fig. 16. Ground-truth length and position measurements of pipe tubes and turning points.

6. EVALUATION

The main metric used to evaluate the mapping accuracy of the PipeProbe system was defined as positional and length errors. Positional error is the Euclidean distance between the estimated coordinate and the ground-truth coordinate for each turning point on the flow path traversed by the PipeProbe capsule. The positional error of a turning point is accumulative because the positional errors from previous estimation points carry into the error for subsequent estimation points. Figure 16 shows the ground-truth coordinate for a turning point, which is measured as the midpoint of the turn. The length error is the difference between the estimated length and the ground-truth length of each pipeline tube on the flow path traversed by the PipeProbe capsule. Length error is non-accumulative because the length of each pipeline tube is measured relative to its own starting point. Figure 16 shows that the ground-truth length of a pipeline tube is measured from the midpoints of its two connecting pipe tubes.

Fig. 17. Flow paths (marked in red lines) in the 12 test scenarios.
Table II. The size of the collected dataset and the number of measurements categorized into vertical/horizontal tubes and various tube lengths.

<table>
<thead>
<tr>
<th>Actual ground-truth length (cm)</th>
<th>Number of measurements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vertical tubes</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>24</td>
</tr>
<tr>
<td>40</td>
<td>84</td>
</tr>
<tr>
<td>80</td>
<td>48</td>
</tr>
<tr>
<td>100</td>
<td>36</td>
</tr>
<tr>
<td>120</td>
<td>24</td>
</tr>
<tr>
<td>140</td>
<td>12</td>
</tr>
<tr>
<td>Horizontal tubes</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>84</td>
</tr>
<tr>
<td>60</td>
<td>42</td>
</tr>
<tr>
<td>80</td>
<td>54</td>
</tr>
<tr>
<td>100</td>
<td>24</td>
</tr>
<tr>
<td>115</td>
<td>60</td>
</tr>
<tr>
<td>180</td>
<td>18</td>
</tr>
</tbody>
</table>

6.1 Length Errors

Since the methods for deriving vertical tube length and horizontal tube length are different, this study analyzes horizontal and vertical tube length errors separately. Table II shows the number of measurements for each length of tube used in the 12 test scenarios for both horizontal and vertical flows. Each scenario was tested 6 times, i.e., the PipeProbe capsule made six mapping trips over the same flow path. The average tube length in these test scenarios was 76 cm.

Figure 18 shows the cumulative density functions (CDFs) of the length errors for vertical tubes, horizontal tubes, and combined tubes. The flow velocity was 11.7 cm/second and the pressure sampling rate was 20 Hz. The CDF dataset was based on 510 length estimates for the pipeline tubes in the 12 test scenarios. The overall median length error was 2 centimeters, and 90% of the errors were less than 7 centimeters. The median length error for vertical-only tubes was 1 centimeter, and 90% of the errors were less than 4 centimeters. The median length error for horizontal-only tubes was 3 centimeters, and 90% of the errors were less than 7 centimeters. These test results demonstrate that the PipeProbe system achieved centimeter-level positional accuracy. The estimation errors should be considered with respect to the 5-cm diameter, i.e., the error margin, of the pipe tubes through which the PipeProbe capsule flowed.

Figure 19 shows the average (standard deviation) length errors for different pipe tube lengths, separating the vertical from horizontal pipe tubes. The dataset for the length errors was based on 228 length estimates for vertical pipe tubes and 282 length estimates for horizontal pipe tubes. The average (standard deviation) length error for vertical tubes, 1.5 cm (0.86 cm), was smaller than the average (standard deviation) length error for horizontal tubes, 3.6 cm (1.19 cm). This difference is due to the use of different techniques for calculating the lengths of horizontal and vertical tubes. Error is accumulative when measuring horizontal tubes, leading to less accurate results. Figure 19 shows that the average error in calculating length generally increases with the length of the pipeline segment.
Fig. 18. CDF of length errors.

Fig. 19. Average (standard deviation) length errors categorized into horizontal/vertical pipe segments and different pipe segment lengths.

6.2 Positional errors

Figure 20 shows the cumulative density function (CDF) for positional error. The dataset for this CDF was based on 588 positional estimates of pipeline turning points (i.e., v-turns and h-turns) in all twelve test scenarios. The median error was 6.8 centimeters, and 90% of the estimates had an error of less than 15.8 centimeters.

Figure 21 plots the accumulated positional errors of turning points with respect to their traveled distances from the water inlet. The dataset for this plot was based on 588 positional estimates for pipeline turning points. The average traveled distance was 335 cm. The average error was 8.2 centimeters. The effects of error accumulation are evident, as the average positional error increases with the distance traveled. The effect of error cancellation is also present, with occasional drops in average positional error with successive distance increments. Figure 22 illustrates the estimated layouts over the actual traversal paths for all 12 test scenarios.
Fig. 20. CDF of positional errors.

Fig. 21. Positional errors under different distances traveled by the PipeProbe capsule.
6.3 Sampling Rate

The sampling rates of the pressure and gyroscope sensors are system parameters that directly affect positional error. Therefore, pressure and gyroscope sensors were tested separately under different sample rates.

Figure 23 shows the position errors under different sampling rates [0.125Hz, 0.5Hz, 1Hz, 2Hz, 4Hz and 5Hz] for the pressure sensor. The dataset for this plot was based on one flow path from test#1 (Fig. 17). Analytical results show that a higher frequency rate generally decreases the positional error because the increased number of data samples detects the turning points with greater accuracy. Figure 23 suggests that the sample rate should remain above 5Hz for water velocities under 11.74 cm/sec, since the positional error is only 4 centimeters.

Figure 24 shows the rotational angle calculated from angular velocities measured by the gyroscope sensor as the PipeProbe capsule makes a 90-degree right h-turn at different gyroscope sample rates [1.25Hz, 2.5Hz, 5Hz, 10Hz and 20Hz]. These results show that a higher frequency rate produces a rotational angle closer to the actual 90-degree turn. When the sampling rate decreases to 1.25 Hz, the calculated rotational angle becomes 0-degree and completely misses the turn detection. This is because the gyroscope did not collect enough samples at 1.25 Hz during the short amount of time the PipeProbe completed the turn.
Fig. 23. Positional errors at different sample rates for the pressure sensor.

Fig. 24. Rotation angle calculated from angular velocity measured by the gyroscope at different sample rates.

6.4 Data Collection Trips

Figure 25 shows the positional errors of the PipeProbe system using different numbers of data collection trips. For example, six data collection trips mean that the PipeProbe capsule made six mapping trips over the same flow path. The dataset was then gathered from the 12 test scenarios and processed by removing statistical outliers and averaging to remove noise. The dataset for this plot was based on 588 positional estimates of pipeline turning points. Results show that a higher number of mapping trips generally reduces the positional error and its standard deviation. This is likely because an increased number of datasets enable more accurate reconstruction of the spatial topology of the pipeline. After one data collection trip, the PipeProbe system still achieved an average positional error of only 4.9 centimeters and a standard deviation of 5.0 centimeters. After six data collection trips, the PipeProbe system achieved an average positional error of 4.1 centimeters and a small standard deviation of 2.2 centimeters.
7. DETECT VARIABLE-DIAMETER PIPES AND 45-DEGREE BENDS

The experiments described above tested and evaluated PipeProbe under the following two assumptions: (1) uniform pipe diameter, and (2) 90-degree joints/turns. This section describes extensions to the PipeProbe system that relax these two assumptions. The extended PipeProbe system is capable of detecting variable-diameter pipes and sensing 45-degree vertical and horizontal bends.

7.1 Variable-Diameter Pipes

Water pipelines often include pipes of different diameters. For example, tap water pipelines often start from large diameter pipes and branch into small diameter pipes. Fluid dynamics dictate that a change in the internal pipe diameter produces a change in the volumetric flow rate and velocity. That is, a large (small) pipe diameter has a large (small) cross-sectional area, resulting in a low (high) PipeProbe flow velocity \( V_{\text{water}} = \text{flow amount} / \text{cross sectional area of the pipe} \). The PipeProbe system estimates the length of a horizontal pipe based on the capsule’s flow velocity (Eq. (4)). Since velocity is no longer constant in variable-diameter pipes, it is necessary to determine the capsule’s variable flow velocity.

A simple method to estimate capsule’s flow velocity is to use the accelerometer in the Eco sensor node. Although the readings from this built-in accelerometer can be used to detect a change in the capsule’s flow velocity as a result of a change in pipe diameter, they contain too much noise to measure flow velocity precisely.

Fortunately, PVC pipes come in standard diameters. For example, North America uses the Nominal Pipe Size (NPS) standard [NPS] to define standard pipe diameters. Therefore, a constant water inflow rate and a set of standard pipe diameters limit the capsule’s flow velocity to a set of possible values. The first pipe segment usually has the largest diameter, and the pipe diameter gradually decreases toward the end of the water distribution network. The pipe diameter is often the smallest at a water outlet. Therefore, the capsule’s flow velocity should increase as it progresses from the starting segment to the ending segment, as the pipe diameter decreases.

The problem of estimating the horizontal pipe length in a variable-diameter pipeline can be described as follows. Given a fixed water inflow rate, find the capsule’s flow velocity within each horizontal pipe segment on its flow path while satisfying two constraints: (1) the horizontal spatial constraint states that the lengths of horizontal pipe segments, computed from the capsule’s velocities, must connect the pipeline’s starting point to its ending point on the horizontal plane; and (2)
the decreasing pipe diameter constraint states that the capsule’s flow velocities must increase or stay the same from one pipe segment to the next (downstream) segment.

The original constraint satisfaction problem without the pipe diameter change, described in Section 4.3, is modified with the additional variables and constraints.

- Additional variable: \( \{ v_j \mid j = 1 \ldots p \} \), where \( p \) is the number of horizontal pipe segments with unknown flow velocity.

- Domain: possible water flow velocities derived from the water inflow rate and the set of standard pipe diameters. For example, given a water inflow rate of \( R \) and standard pipe diameters \( \{ d_1, d_2, \ldots \} \), the domain values are \( \{ R/A_1, R/A_2, \ldots \} \) where \( A \) (the cross-sectional area of a pipe) = \( \frac{\pi d^2}{4} \).

- Rewrite the constraints:
  
  (1) *Horizontal spatial constraint:* the term \( l_{tube}^h \) in Eq. (8) and (9) becomes
  
  \[
  (l_{tube}^h) = \sum_{j=0}^{n} (v_j \cdot t_j)
  \]
  
  where \( v_j \) is the \( j \)-th unknown flow velocity variable and \( t_j \) is its internal flow. Each horizontal segment length consists of \( p \) possible segments.

  (2) *Decreasing pipe diameter constraint:* \( v_1 \leq v_2 \leq \ldots \leq v_p \) states that PipeProbe’s flow velocities must increase or stay the same from one pipe segment to the next.

Pipe diameter changes can also occur in vertical pipe segments. However, since vertical pipe length estimations are based on changes in water pressure readings and independent of the pipe diameter, the flow velocity does not affect length calculations. For vertical pipes, the water pressure at any \( z \)-position remains the same at different flow velocities.

The implementation of this constraint satisfaction problem is similar to the original layout mapping implementation described in Section 4.3. This algorithm involves an exhaustive search by testing all possible combinations of a capsule’s flow velocities for each horizontal pipe segment while satisfying the decreasing pipe diameter constraint. Given \( n \) possible capsule flow velocities (and pipe diameters), the proposed algorithm uses \( n \)-ary tree data structure. The tree branches out for each fixed-diameter horizontal pipe segment. Each child node represents a possible capsule’s flow velocity, and the resulting pipe length and each node of the \( n \)-ary tree tracks the corresponding horizontal coordinate after a change in horizontal pipe diameter. At least one of the leaf nodes should arrive at the outlet’s horizontal coordinate. Each path from the root to a leaf represents one possible horizontal pipeline layout. After constructing the \( n \)-ary tree, scanning all the leaf nodes reveals the closest match(es) to the outlet’s horizontal coordinates.

A water pipeline testbed was constructed with three different pipe diameters to evaluate PipeProbe performance under these conditions (Fig. 26(b)). The large, medium, and small pipes in this testbed had diameters of 9.4 cm, 7.4 cm, and 5.4 cm, respectively. Figure 27(a) shows raw readings from the three-axis accelerometer as the PipeProbe capsule flowed through the three pipe segments in Fig. 26(b). Figure 27(b) presents the summed magnitude of three-axis accelerometer readings. Each change in the horizontal pipe diameter produced a noticeable peak in the acceleration magnitude as the PipeProbe capsule accelerated into a smaller-diameter pipe. Many researchers have used the wavelet-based method to identify peaks from real-world time series data, such as water burst event detection [Srirangarajan et al. 2010]. Thus, this study uses a wavelet analysis method [Pervical et al. 2000] to identify pipe diameter changes in Fig. 27(b).

The following discussion illustrates how the extended PipeProbe system detects changes in pipe diameters in Fig. 26(b). First, the water inflow (or outflow) rate was set at 3.4L/sec using a measurement bucket to record the amount of water outflow over a known time interval. Second, three possible internal diameters: {9.4 cm, 7.4 cm, 5.4 cm} were used to construct the water pipeline depicted in Fig. 26. The set of possible flow velocities was {49 cm/sec, 79 cm/sec, 148
cm/sec}. Third, pre-measuring the horizontal positions of the starting inlet and the ending outlet provided the horizontal pipe length $L = 445$ cm, and the horizontal spatial constraint: $v_1 * t_1 + v_2 * t_2 + v_3 * t_3 = 445$ cm. Note that $t_1$, $t_2$, and $t_3$ are the flow time intervals over these three horizontal pipe segments (Fig. 26(b)). The peaks in Fig. 27(b) at 4.15 seconds and 5.35 seconds indicate the occurrences of two changes in pipe diameters. Given the starting/ending timestamps of 0 second and 6.15 seconds, $t_1$, $t_2$, and $t_3$ were calculated as 4.15, 1.2, and 0.8 second(s). The horizontal spatial constraint was $v_1 * 4.15$ (sec) + $v_2 * 1.2$ (sec) + $v_3 * 0.8$ (sec) = 445 cm.

Based on the horizontal spatial constraint derived from the previous paragraph, i.e., $v_1 * 4.15$ (sec) + $v_2 * 1.2$ (sec) + $v_3 * 0.8$ (sec) = 445 cm, the proposed algorithm enumerates the search space shown in Fig. 28. Ten possible lengths were computed based on two pipe diameter changes and three flow velocities. Since the horizontal pipe length $L$ is 445 cm, the 5-th pipe length (421 cm) would be the correct pipe length due to its smaller error distance than other possible lengths.

Each test scenario in Fig. 26 was tested ten times repeatedly. The overall median length error for the testbed scenario in Fig. 26(a) was 14 cm, and its standard deviation was 20 cm. The overall median length for the testbed scenario in Fig. 26(b) error was 24 cm, and its standard deviation was 18 cm.

Fig. 26. Two test scenarios in the variable-diameter pipeline testbed. Scenario (a) connects two horizontal pipe segments of diameters ($L$-$S$). Scenario (b) connects three horizontal pipe segments with diameters ($L$-$M$-$S$). The internal pipe diameters of $L$, $M$, and $S$ pipes were 9.4 cm, 7.4 cm, and 5.4 cm, respectively.
Fig. 27. (a) 3-axis acceleration graph recorded by the PipeProbe capsule traversing the flow path depicted in Fig. 26(b). The acceleration bias was removed. (b) Transformed 3-axis acceleration magnitude graph, where the peaks indicate changes in pipe diameter.
7.2 Detecting 45-Degree Bends

Although most pipelines use 90-degree joints for pipe connections, there are some special cases where 45-degree bends connect pipes. This section discusses how the PipeProbe system detects 45-degree vertical and horizontal bends.

7.2.1 Detecting 45-Degree Vertical Bends

PipeProbe detects vertical turns based on differences in water pressure. For 90-degree turns, the water pressure difference increases/decreases at a faster rate than a 45-degree turn, since PipeProbe flows downwards/upwards directly. On the other hand, for 45-degree turns, the water pressure difference increases/decreases at a slower rate because PipeProbe flows obliquely. By analyzing the speed of water pressure change, i.e., the slope of the pressure graph, it is possible to differentiate between a 90-degree and a 45-degree vertical turn.

To perform this experiment, we built a testbed including three test scenarios (Fig. 29). These three scenarios include (1) a 90-degree vertical turn, (2) a 90-degree turn followed by a 45-degree turn, and (3) a 45-degree turn. Figure 30 shows the data collected from this testbed, in which the pressure change graphs exhibit distinguishable slopes between 45- and 90-degree vertical turns. The 90-degree line increases at 78(mbar)/sec, which is 1.6 times faster than the 45-degree line at 48(mbar)/sec.
Fig. 29. Testbed layout for 45- and 90-degree vertical turns. The first test path has a 90-degree turn. The second test path has a 90-degree turn followed by a 45-degree turn. The third test path has a 90-degree turn.

Fig. 30. Water pressure graph (after applying the median filter from Section 4.1) for Fig. 29.

The experiments above were performed using pipes with a uniform diameter. If the pipe diameter changes during PipeProbe's vertical flow, it can also generate different speeds of water pressure change. The accelerometer described in Section 7.1 can differentiate between a vertical 45-degree bend and pipe diameter change using a water pressure graph.

7.2.2 Detecting 45-Degree Horizontal Bends
PipeProbe detects horizontal turn angles using a gyroscope. To differentiate between 90- and 45-degree horizontal turns, simply look at the turning degree recorded by the gyroscope. Figure 31 depicts a testbed that includes a 45-degree left turn followed by a 90-degree left turn. Figure 32 shows the corresponding experimental results. Figure 32 shows that the gyroscope first produces a 54-degree left turn followed by a 82-degree left turn.
Fig. 32. Real gyroscope data from the testbed in Fig. 31. (a) Raw angular velocity from the gyroscope. (b) Angular velocity after applying a threshold-based filter. (c) Rotation angle of the Pipe-Probe
8. DISCUSSION AND LIMITATIONS

The PipeProbe system assumes that the positions of water inlet and outlet are known. For 3D position measurements, there are many cheap and handy tools that architects use on a daily basis. For example, a barometer can measure building height. Laser rangefinders can measure width and length. Higher-end meters generally provide more accurate measurements. To access the inlet location to insert the PipeProbe, a water inlet can often be found next to a water heater, near the water tank at a building’s rooftop, or near a water meter. For instance, in Taipei city, water inlets can often be found on the roof of a building.

The PipeProbe capsule is currently too large for some small diameter pipes. However, there are several ways to reduce the size of the PipeProbe such that it can fit into most pipes. For example, the current PipeProbe has a loose packaging and does not fully utilize all its internal space. The largest component in the PipeProbe is the Eco mote, whose size is 13 mm (L) × 11 mm (W) × 7 mm (H). Therefore, custom-making a spherical shell at the mm scale would make it possible to shrink the PipeProbe capsule. A custom printed circuit board would also eliminate most of the wiring, which takes up space. We are currently working on the next version of PipeProbe, which reduces the diameter from 4-cm to 2-cm. Further reduction of the PipeProbe size, i.e., below the centimeter scale, is possible with current SoC (System-on-a-chip) technology. This technology integrates all sensing, processing, and communication hardware components on a single chip. For example, [Chen et al. 2010] reported a millimeter-scale wireless sensor node using SoC.

Poiseuille’s law [CENGEL et al. 2005] states that the pressure will drop in a fluid when flowing through a long cylindrical pipe. This is important because a water pressure drop affects the PipeProbe’s positional accuracy. Although we did not observe significant pressure drop in our experiments (e.g., Figure 6 shows a relatively stable water pressure on a horizontal plane), this was mainly due to PipeProbe traveling inside large diameter pipes. Recognizing that the Poiseuille’s law has an increasing effect in decreasing pipe diameter, the PipeProbe system would need to consider and model the pressure drop effect of Poiseuille’s law in order to improve the estimation of PipeProbe’s positional accuracy when smaller PipeProbe prototype is developed.

Other sensors were considered for use in the PipeProbe design phase, but were not chosen due to various limitations. For example, many navigation systems use a digital compass [Eisenman et al. 2010] [Constandache et al. 2010], as readings from a digital compass can be used to calculate the unknown v-h-turn {1º, ..., 360º} described in Section 4.3. However, these readings contain too much noise [Jekeli et al. 2000] for calculating a precise v-h-turn angle. Moreover, metal pipes contribute to large noises in a digital compass. Thus, a layout mapping algorithm with water inlet/outlet position is required to generate an accurate layout mapping result, as described in Section 4.3. Another possible sensor is a proximity sensor, which measures the distance between nearby objects without physical contact. We considered mounting a few proximity sensors on the PipeProbe capsule to sense the pipe’s internal diameter. Unfortunately, a proximity sensor requires
a minimum distance (e.g., typically a few centimeters) between itself and its sensing object, rendering it unsuitable for small-diameter pipes. Therefore, the final capsule design adopted an accelerometer for sensing variable-diameter pipes.

PipeProbe assumes that the pipeline consists of straight pipes for layout mapping. This is true for most pipeline structures, but in reality, there are moldable pipes (e.g., PEX pipes) and plumbers may bend pipes slightly. This causes curved PipeProbe movement, which invalidates system assumptions and produces errors in the layout mapping algorithm. A possible solution to this problem is to carefully calibrate inertial sensor readings to calculate directions and account for curved pipe segments.

9. RELATED WORK

Recent projects involving WSN technologies for monitoring water pipelines include the PipeNet project [Stoianov et al. 2007], the NAWMS project [Kim et al. 2008], and the HydroSense project [Froehlich et al. 2009]. The PipeNet project detects, localizes, and quantifies leaks and bursts in water pipelines. PipeNet attaches a variety of acoustic, vibration, pressure, and flow sensors to wireless sensor nodes mounted on the pipeline externals and internals to sense these faults and anomalies. The signals received from these sensor nodes are analyzed to identify and locate leaks. For example, a pipe rupture produces noises. By performing time delay estimation on the rupture event sensed (heard) by nearby sensor nodes, the system can infer the pipe rupture location. The NAWMS project [Kim et al. 2008] detects the water outflow rate for each individual pipe and outlet. This method attaches a vibration (accelerometer) sensor to each individual pipe to separate the water outflow rate on an individual pipe from a master household water meter. By calibrating and deriving the relationship between the standard deviation of pipe vibration and the mean flow rate in the pipe [Evans et al. 2004], the NAWNS system can infer a pipe’s water flow rate from the sensed vibration level produced by each flowing pipe. HydroSense [Froehlich et al. 2009] proposes a novel, single-point sensing technique that separates water usage activities from each individual water outlet using a single pressure sensor installed at one point within a building. The HydroSense system senses and recognizes the unique “water hammer” pressure fingerprint produced by each water fixture. By training and recognizing individual fixture’s fingerprint, the system can accurately infer which water fixture is turned on or off. A HydroSense extension [Froehlich et al. 2011] develops classifiers that detect compound and collision events, i.e., water usage activities that occur simultaneously at multiple water outlets. Unlike these projects, which are based on either multi-point or single-point sensing (i.e., fixing multiple/single sensor(s) to point(s) on the water pipelines), the PipeProbe system adopts a mobile sensing approach in which a mobile sensor travels and observes pipelines from within the pipes.

Several projects adopt the mobile sensing approach. SewerSnort [Kim et al. 2009] is a WSN system for monitoring sewer gas pipelines. A drifter node prototyped on a MicaZ mote and connected to an electrochemical sensor flows inside sewer pipelines. As the drifter node floats downstream, it measures in-sewer gas concentrations, while a set of beacon nodes pre-installed beneath manholes determines their geographic location. Alvarado et al. [Alvarado and Youssef-Toumi 2005] developed a swimming robotic fish under a foot long that closely mimics a real fish’s natural swimming motion. This robofish is equipped with sensors to detect environmental pollutants. However, it is considerably larger than the PipeProbe capsule, and requires a 2.5-5W external power source to operate.

Several methods leverage the special properties of water pipelines for activity recognition and energy harvesting. Fogarty [Fogarty et al. 2006] proposed an activity inference system, targeted at the elderly, that senses water usage activities in a home plumbing system. Microphone-based sensors strategically placed on pipes throughout home listen to the pipe’s water flow activities and different activity-related water outlets. For examples, clothes washing activity is inferred when water flows to a washing machine, dishwashing activity is inferred from water flowing to a dishwasher, and toilet activity is inferred from water flowing to a toilet, etc. WATTR [Campbell et al. 2010] proposes a novel energy harvesting technique to power a sensor node used to monitor water pipelines. A change in the water pressure inside the pipe drives a tiny power generator con-
nected to a sensor node. This design harvests enough energy to allow the sensor node to perform sensing and data transmission.

There are different methods to locate and map hidden pipelines. For example, [RADIODETECTION] uses a Radiodetection method to map hidden pipelines. This method works by inducing a radio signal into the metal pipe (by either directly contacting the pipe with a transmitter or by indirect induction method via a broadcast coil mounted on surface). Once the signal is induced into the pipe, one can use a hand-held receiving antenna to map the pipeline by walking around the building surface. However, this only works for metal pipe. For non-metal pipes, it is possible to place a small radio-transmitting probe into the pipe and follow along topside with the receiver as the probe is pulled, pushed, or flushed down the pipeline. However, PipeProbe does not need to know the pipe material or require human effort to walk around the building. Another method of locating pipelines is to use a thermal camera. If a hot water pipeline is behind the wall and the hot water temperature is conducted to the surface of the wall, a thermal camera can be used to inspect the location of hot water pipeline by recording temperature differences. However, this method does not work for cold-water pipelines, since the temperature of the wall surface will not change. PipeProbe can work with cold and hot water systems.

10. CONCLUSION & FUTURE WORK

The proposed PipeProbe system is a novel mobile sensor system for identifying the spatial topology of hidden water pipelines. Experimental results achieved a median length error of 2 centimeters, and 90% of the tests had a length error of 7 centimeters or less when estimating the lengths of pipe tubes. The proposed method achieved a median positional error of 6.8 centimeters, and 90% of the tests had a positional error of 15.8 centimeters or less when estimating the pipe’s turning points. The PipeProbe system enables accurate mapping by using a tiny capsule to sense pressure readings as it traverses pipelines. This study also presents sensing techniques to distinguish variable-diameter pipes and differentiate 90-degree pipe turns from 45-degree pipe bends.

For the future work, we are interested in developing a self-deployment WSN for pipeline monitoring and applications. The idea is to have a single release point of sensor nodes in the pipeline infrastructure that incrementally releases sensor droplets. These sensor droplets will flow inside pipelines and automatically place themselves at specific locations to form a connected WSN. A node placement algorithm will optimize detection area while maintaining radio connectivity. A sensor droplet will include mechanical actuators that act as spines of porcupine fish, i.e., they stretch out to latch and fix the sensor droplet within pipes. A base station will collect sensor data through an in-network multi-hop protocol. Self-deployment has great potential in significantly reducing the amount of human efforts needed in installing a large-scale sensor infrastructure. Once this self-deployment mechanism is implemented, we aim to develop applications for this pipeline monitoring system. These applications might include leakage detection, water contamination monitoring, or water usage monitoring.

11. REFERENCES


AMERICAN SOCIETY OF HOME INSPECTORS.
http://www.ashi.org/media/press/.


INTERSEMA. 2010. MS5541C miniature 14 bar module.


LISY300AL The STMicroelectronics LISY300AL gyroscope chip


RADIODETECTION
http://www.radiodetection.com