Building a distributed smart system for waste water networks

Jose Manuel Linares
Lancaster University
InfoLab21
LA1 4WA
UK
j.linares@lancaster.ac.uk

François Taïani
Lancaster University
InfoLab21
LA1 4WA
UK
f.taiani@lancaster.ac.uk

David Singerton
Anglian Water Services Ltd
Innovation
ThorpeWood House
ThorpeWood
Peterborough
PE3 6WT
UK
dsingerton@anglianwater.co.uk

ABSTRACT

The need for water utility companies to save on energy costs on running waste water pumps has become a priority given that there is a forecast that energy prices will be raised in the near future. There is a need for a middleware infrastructure to coordinate smart sewage systems that controls the flow of sewage water, potentially avoiding floods and save energy by reducing pump activations during the day.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Distribution

Keywords

smart waste water systems, wireless sensor nodes, motes, SCADA, WSN, Component Based Software Engineering, fuzzy logic, distributed fuzzy logic

1. INTRODUCTION

Anglian Water is one of the largest water utility companies in the United Kingdom and have been actively researching novel methods of reducing operational costs. The research that was conducted is based in a conceptual sewage system called SEWERNET which is a system that is composed of several sub systems that holistically monitors and controls events and feedback loops such as waste water flow, weather data to predict rainfall and soil saturation, water septicity and receive alarms from sensor input data. One subsystem within SEWERNET is based on the research conducted on Fuzzy Logic to optimise control of the pumps called the *Logic Engine*. The primary goal of Anglian Water is to prevent local flooding in areas whilst activating pumps during cheap energy tariffs which is normally during the night. Ostojin's paper [5], quotes that approximately £60 million pounds was the total energy bill spent by Anglian Water, where £32 million was spent on waste water operations that consist of £23 million on treatment and £9 million on networks.

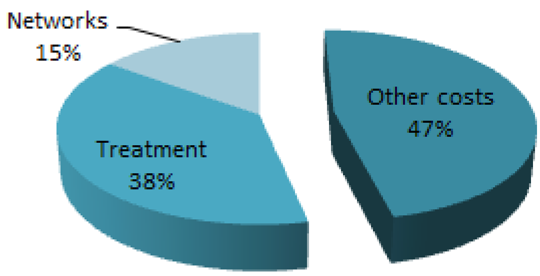

Figure 1: Anglian Water operation energy budget cost of £60m

Water pumps located inside a sewage wet well have two discreet states which are *on* and *off*. This action is controlled by a Programmable Logic Controller (PLC). A PLC will compute when the wet well reaches certain levels of water by data received from ultrasonic sensors and activate the pump depending on the amount of waste water that has flowed in. The main challenge in previous research was to make this switching of states as economical as possible whilst preventing surface flooding due to that energy costs vary during certain times of the day. This has been achieved with success by implementing fuzzy logic techniques. Now there is a need to distribute this intelligence to Wireless Sensor Nodes (WSN) which normally is equipped with more onboard resources such as sensors, CPU power and memory than a PLC has.

This paper explores on other water and oil monitoring systems which are related to this problem as they use smart systems and gain insights on a possible solution developing a smart distributed system middleware for waste water networks.

2. PROBLEM STATEMENT AND RELATED WORK

In this section, we look at other relevant systems that leads us to explore possible solutions that are relevant to the waste water industry in controlling sewage pumps. Some of these solutions have been superceded by other solutions, but does not mean that it is a good fit for our purpose. We then define our problem statement.

2.1 The problem statement

Due to EU and UK regulation in the water industry, there is a need to maintain acceptable standards of water quality and prevention of flooding. If an area becomes flooded, the company may be fined by the regulatory authority (OFWAT). Anglian Water are looking in methods of efficient automation that 1) Prevents floods and 2) Saves energy costs.

The company metioned above are looking at how the current fuzzy rule base system they have developed can be distributed so as to allow as much autonomous control as possible. This does not mean however that the motes will have full control of the system as the current system will need human intervention if alarms are raised. It is possible to have a hybrid system where motes control the activation of the pumps and alongside have human intervention when required. We are highlighting that there is a need to have middleware to solve this problem of distribution and we are looking at different ways and insights from other work done to address this problem.

2.2 Water catchments and problems

An area where water is drained is called a *drainage basin* where rain water is collected by basins which can be a sea or a river. Before this water is pumped into the basin, it has to be pumped out from the city via a series of wet wells. These wet wells have 2 to 3 pumps; a *d*uty pump, an *a*ssist pump and some wet wells have a *s*torm pump (see [5] for more details of pumps). There are several factors that can lead to an area being flooded.

Topography (mountanous or flat surface) of an area which defines how fast water will travel.

Shape contributes the speed at which water will run off to a basin, for example, a long and thin catchment will take more time to drain than a larger catchment.

Size of a catchment determines how much water reaches the basin. The bigger the size of a catchment, the higher the risk is for major flood to occur.

Land use determines how much rain fall will be absorbed by the type of terrain. For example, water falling on concrete pavements will be impermeable, hence will fall within a sewer system or a wet well so that it can be pumped out. Terrain that is *s*andy will absorb most of the water into the ground, but a soil made mostly of *c*lay will be impermeable and has a higher risk of flooding.

2.3 Communication systems

A type of communication system that is being used within the waste water and oil industries is SCADA (Supervisory control and Data Acquisition) [10][8]. SCADA systems are typically centralised systems where sensors (sensing data from actuators) sends all the data to a central head quarters where a human operator monitors the data which presents itself as alarms. Communications within the SCADA network involves using radio, POTS leased lines, cellular (GSM) or sattelite communications. The SCADA system communicates from point to point (i.e from sensor to main headquarters) hence, it cannot be scaled to a fully distributable smart system as sensor nodes will need to communicate to other sensor nodes to make decisions. Unfortunately, SCADA has several shortfalls which is that the equipment is expensive to acquire, it is less scalable, suffers from high latency in sending data, not heterogenous in software use, and unable to be fully distributed as it relies on a point to point protocol with the main server. Therefore, there was a need to develop a new system that can address these issues.. Yoon et al [**?**] have proposed and developed a system which is specifically built to monitor high pressure water, waste water and oil flows though it does offer some advantages over SCADA and offers insights in proposing a system suitable for the wastewater industry and to control flow of wastewater from the wet wells.

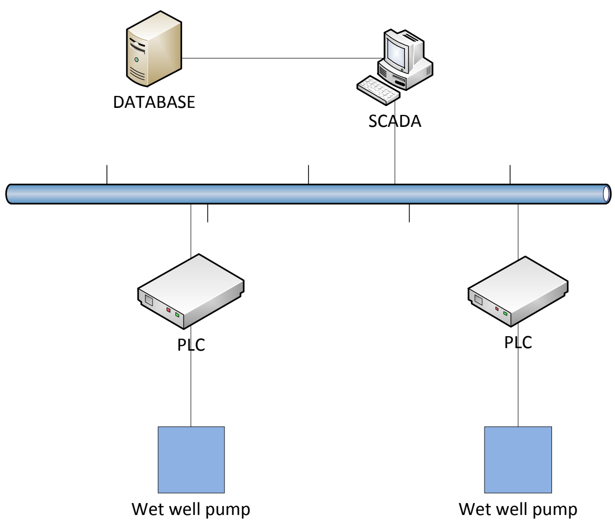

Figure 2: Simple SCADA diagram in the water utility industry

Other research in improving the feasibility of monitoring sewage distributed systems within a similar area is described by Yoon et al [10]. The author of this paper describes the development of a system called SWATS (Steamflood and Water flood Tracking System) and replaced SCADA which had the role of detecting anomalies and identifying where the source of the problem is (e.g. leakage or blockage in pipes). To address the shortfalls in SCADA, Yoon et al [10] have approached this problem by developing SWATS which is low cost, low latency, fine granular coverage, highly accurate and reliable. WSN nodes are cheap to acquire and are able to collaborate with other nodes to improve correctness[4]; this technique known as multi node - multi modal.

This technique makes a comparison with other nodes to detect if there is a leak or blockage within the pipe network by comparing data from other nodes and exploits temporal and spatial correlations from past sensor readings. To determine the causes and the health of a pipeline, a Decision Tree Algorithm (DTA) is used to lower the false alarms that the system sends to a human controller that monitors the system. The type of data that the DTA uses is past data from the central database, the reported events from the other neighbouring nodes and information to the proximity to the equipment in question.

2.4 Past research and development

The monitoring of water systems has several challenges to be overcome. These challenges range from communications, radio signal failures to hardware resource limitations on the actual sensor motes. There have been attempts by researchers to develop heterogeneous WSN systems and middleware software that is able to perform in a prescribed environment. Such environments can be oil fields where sensors takes data samples of the health of an oil field to a water catchment where ultrasonic sensors are used to monitor water levels within a wet well.

2.4.1 PIPENET

PIPENET [9] is a WSN network that was tested within a 22 month trial period in Boston, US, that monitor pipelines that detects, localises and quantifies bursts and leaks in water transmission pipes and the monitoring of water levels in sewer collectors and combined sewage outflows. The system also reports anomalies in pipelines such as blockages and malfunctioning control valves and also monitors levels in sewer collectors. The deployment that consists of Intel Mote platform [1] is a centralised network where data is sent from the nodes to the backend servers and require substantial computing power. Although PipeNet has support for high data rate time synchronized data collection, it can not support high levels of local data processing as the motes are specifically designed for this application. The motes [9] consist of an ARM7 core, 64kB RAM, 512kB Flash memory and a bluetooth radio that are deployed on the first tier which is the sewage pipe system, the 2nd tier is the gateway which is essentially the head of a cluster of motes and sends data from the motes to the backend via GPRS or EDGE. The third tier is the backend where the data is recieved from the gateway and data is stored for offline analysis.

2.4.2 Fault tolerance in sensor data

Mihai Marin-Perianu et al [4] highlights that a single sensor will not guarantee reliable data received from sensor inputs. SWATS addresses tihs problem using a Decision Tree Algorithm to make comparisons over which alarm to trigger and hence avoid false alarms. D-FLER [4] is a distributed, general purpose reasoning engine for WSN. This approach uses fuzzy logic to make comparisons with observations with neighbourhood nodes for detecting events andl produces a more reliable and accurate result. D-FLER was confined within a fire prevention network simulation which had promising results where initiated false fire alarms was less than 2%.

2.4.3 Other Middleware

In terms of middleware, Iqbal et al [3] proposes a middleware that is designed to monitor large scale water distribution systems. The system itself is event driven, specialised and reconfigurable. The author Rocha, et al [6] proposed a semantic middleware for an Autonomic Wireless Sensor Network (AWSN) with the vision of being applied to Structural Health Monitoring applications (SHM). Although it is used in SHM areas, it can be used for other areas such as ambient intelligence, habitat monitoring and fire detection. It uses the concept of fuzzy logic to create fuzzy rules in order to compose a knowledge base for the domain.

2.5 Fuzzy logic control

Fuzzy logic is an area of control theory which is currently used in our case study to optimise the energy efficiency of the pumps as a second priority to the stakeholders whilst maintaining its first priority which is to avoid an area being flooded. The steps taken to get from a crisp input to an output using the Mamndani inference engine is as follows:

Input A crisp input is one that has a defined member set within a range.

Fuzzification The crisp input is *fuzzified* by a process of evaluating fuzzy rules in parallel. These rules are expressed as IF (antecedent)... THEN .... (consequence), where the consequent part is the result for this particular rule depending on the degree of membership.

Defuzzification ***(optional)*** The results from the fuzzified rules are summed together. Using either logical AND (max), probabilistic OR, or summation which is the sum of each rule's output set.

Output The result is a non-fuzzy number by using either centroid (winner takes all), bisector, middle of maximum, largest of maximum or smallest of maximum.

In order to control the execution of the pumps, Ostojin et al [5], used the Mamdani inference system to control the execution of the pumps. The execution of the pumps was more predominant during the night as energy tarrifs are much lower than energy tariffs during the day. There are five membership functions that are used to control the activation of the pumps. The function member inputs are as follows:

Qin Quantity of water going in to the wet well.

Qh Rate of change of water.

Tariff Expressed in time/price.

Dimension Wet well dimensions.

Backup The minimum level of the wet well.

The output is expressed as dU which is the rate of control; i.e. start or stop 2 pumps, start or stop 1 pump, or do nothing.

An example of an output in this context would be using the following antecedents which have been expressed linguistacally as examples:

Qin A lot

Qh A lot

Tariff low

Dimension small

Backup small

The example output would be to activate 2 pumps to prevent flooding as the rate of flow is high and the dimensions of the wet well is small and the backup level is small.

These variables are expressed using triangular sets to express variance within each of the sets to obtain a fuzzified result on the inputs which are then defuzzified to obtain a crisp output; i.e. to switch on or off the wet well pump.

The tests conducted by Ostojin et al [5] concluded that using a fuzzy logic system to optimise pump control has proved an energy saving of up to 2.5% which is approximately between £1million and £2million per year. The result also indicates that because there have been less pump switching, it also improves the longevity of the pumps which leads to a decrease in maintenance costs (although not quantified in the study), and the procuring of new pumps. This is future work that is currently taking place.

3. CHALLENGES IN WASTEWATER DISTRIBUTED SYSTEMS

There are several key challenges in distributed systems that range from the lowest communication physical level to the application level where a software system requires to be heterogenous with hardware motes.

3.1 Real time control

Schutze, et al [8] relates that with today's available technology and methodologies, it is possible to allow real time control (RTC) of urban wastewater systems. RTC is a desirable property in wastewater systems as it allows a better control when problems happen. For example, if a wet well is flooded, the central control would like to know of this situation on the moment it occurs and not 1 hour later which by then might have flooded an area. The author also emphasises that there is a need for clear terminologies in RTC so that scientists and experts from other domains can collaborate when developing an RTC system.

3.2 Sensor accuracy and placement

Research conducted by Yoon, et al [10] describes some of the challenges that are inherent in distributed systems in monitoring either oil or water flows. Sensors that monitor water flow are inherently inaccurate and unreliable when monitoring flows thus giving to a rise to a series of false alarms which is also a problem in the context of our case study with Anglian Water. A single sensor cannot guarantee accurate readings due to that the sensor is malfunctioning or running low on power (if battery powered). Hence, Yoon et al [10] has developed SWATS which will address this problem and offers some insights to how this approach can be used in our case study. Once a sensor has been deployed, it is difficult to physically access the sensors as these sensors could be put inside a sewer system, a wet well, a remote area or areas which present a hazard to humans. Some reasons to physically access a sensor might include to replace batteries, check the health of the sensor or replace it, hence it is desirable that the system deployed requires as little human intervention as possible as in our case study, the sensors will be deployed within hazardous and remote areas.

3.3 Distributing intelligence across sensor motes

Ostojin's [5] results are promising for a small scale catchment area. However there is no mention of how such a system could scale up to a catchment that has more than five pumps in future work. There is concern that because Mamdani rules have to be created manually, it can add extra burden on the domain expert. Furthermore, the rules given are computationally expensive which means that a small WSN mote will not have the CPU capability to process the rules, hence there is a possibility that either a hybrid approach in distributing intelligence is required to process instense calcualtions, or research another fuzzy inference system which currently exists and will be mentioned later on.

4. DISTRIBUTED CONTROL FOR WATER UTILITIES

Some novel techniques and possible solutions to the problem at Anglian Water are given here. We discuss the possibility of building a middleware framework as a solution to distribute the intelligence to the nodes.

4.1 Distributing intelligence by developing a Fuzzy Rule Base framework

The author Ostojin et al [5], mentions that the Mamdani type Fuzzy Rule Base (FRB) system to model input sets just like an expert would do, expressed in the form of triangles, trapezoids, gaussian, etc. The current approach may be ideal for a set number of pumps but might be difficult to scale up when additional pumps are added to the system.

4.2 Using Self evolving Parameter-free Rule-based Controller

AnYa (Angelov - Yager) or Granular Decomposition with Vector Membership, is a novel FRB developed and has shown promise in certain applications such as robotics, image processing and control of water temperature. The strength of utilizing AnYa, lies on its antecendents; it is expressed by using a non-parametric approach where memberships are not required in an explicit form and its approach relies on the relative distribution of all the real data forming data clouds.

4.2.1 Basics of AnYa

FRB's such as Mamdani, require at least two input parameters so that it can execute online. In basic terms, the AnYa controller [7] *evolves* hence its structure and parameters change over time. AnYa requires no parameters to be stated as these parameters are collected as the controller is running online. As data is collected by the controller, these are then associated within a *data cloud* (or ith cloud) which represents the local density vector (or granlulation). Euclidean distance equation or Cosine distance can be used to calcualte the distance between two samples. For clarification, a *data cloud* is a boundless collection of data points that has no centre point whilst *data clusters* are defined by boundaries and represents a prototype which is usually the centre (or mean). After the local density for a data sample is computed, its control signal is generated which is the weighted average by using either Mamdani or TSK consequents. The global density is the same as the local density, but defines a centre and a radius for each data cloud formed for learning purposes. After gathering data points recursively by adding more data points, the consequents are updated, the data points are associated with the *cloud* that has a higher density. The cloud is then defined as a new rule after it meets two conditions which:

1 The sample must have a good generalisation and summarization capabilities.

2 Check that the other clouds are far away from the current data sample.

If the conditions are not met, then the associated cloud is updated. The flow starts by updating its consequents again. For readers who are interested in learning the details of the AnYA FRB system, read the paper written by Sadeghi-Tehran [7].

5. HETEROGENOUS MIDDLEWARE FOR NETWORKS IN THE WASTEWATER INDUSTRY

SCADA systems are centralised systems which has several shortfalls mentioned in this paper, hence the reason why Pipenet and SWATS have been developed. Pipenet offers a hybrid tree communication platform between the motes and the backend server whilst SWATS is fully decentralised and its motives are due to Pipenet's inability to communicate with the motes to provide data fault tolerance.

From observing Stoianov paper [9], the motes adopt a self healing and self configuring form to communicate with each other which provides a degree of fault tolerance. This can be included in the framework as *microprotocols* that can be implemented to provide desirable properties for a specific environment. As an example, a microprotocol that supports middleware level encryption to provide secure transmission between motes is a desirable property. In Stoianov's work [9], the system is built on specific hardware and software for the purposes of battery efficiency and data processing efficiency (water pH and min/max average water pressure).

In Yoon's paper [**?**], SWATS is a fully deployed distributed network that is capable of processing data at a local level which Pipenet lacked. This level of reasoning is necessary so that nodes can collaborate with each other to determine whether the data is reliable .

It is our intention to explore and develop a middleware framework system that is future proof so that it can accomodate 3rd party Fuzzy Rule Base systems and be utilized within a smart heterogenous WSN for waste water utilities. The methodology used to build the proposed middleware will be CBSE (Component Based Software Engineering). We are aware that projects using CBSE has worked in engineering middleware [2] and it is our intention of applying it within this project as its focus will be in looking at modularizing components and interact solely on its interfaces. The intention is not to build a *black box* middleware that cannot be altered in a future date, but to engineer a *white box* middleware as it will provide other potential users to tailor and fit on to their own system.

6. CONCLUSIONS

We have seen that SCADA systems has its shortfalls and there is a need to move on to other distributed systems so that smarter systems can be achieved. SWATS is an example of a smart system in montoring leakages and blockages though we are more concerned in the control aspect of wastewater flow. Research has been accomplished in fuzzy logic in the application of wastewater systems in order to save energy costs though its shortfall is in its distribution to the PLC. A smart middleware framework was proposed earlier on to address the distribution problem with the use of modern WSN nodes which have more capabilities than a simple PLC.

7. ACKNOWLEDGMENTS

This section is optional; it is a location for you to acknowledge grants, funding, editing assistance and what have you. In the present case, for example, the authors would like to thank Gerald Murray of ACM for his help in codifying this *Author's Guide* and the **.cls** and **.tex** files that it describes.

8. REFERENCES

[1] ARM7 processor family http://arm.com/products/processors/classic/arm7/.

[2] G. Coulson, G. Blair, P. Grace, F. Taiani, A. Joolia, K. Lee, J. Ueyama, and T. Sivaharan. A generic component model for building systems software. *ACM Transactions on Computer Systems*, 26(1):1–42, Feb. 2008.

[3] M. Iqbal and H. Lim. Poster Abstract : A Cyber-physical Middleware Framework for Continuous Monitoring of Water Distribution Systems. pages 401–402.

[4] M. Marin-Perianu. D-FLER âĂŞ A Distributed Fuzzy Logic Engine for Rule-Based Wireless Sensor Networks. *Ubiquitous Computing Systems*, (1):86–101, 2007.

[5] S. Ostojin, S. Mounce, and J. Boxall. An artificial intelligence approach for optimizing pumping in sewer systems. *Journal of hydroinformatics*, 13(3):295–306, 2011.

[6] A. R. Rocha. A Semantic Middleware for Autonomic Wireless Sensor Networks. *Architecture*, pages 19–25.

[7] P. Sadeghi-tehran, A. Bel, and P. Angelov. Self-evolving Parameter-free Rule-based Controller. *Update*, 2001.

[8] M. Schütze, A. Campisano, H. Colas, W. Schilling, and P. a. Vanrolleghem. Real time control of urban wastewater systemsâĂŤwhere do we stand today? *Journal of Hydrology*, 299(3-4):335–348, Dec. 2004.

[9] I. Stoianov, L. Nachman, and S. Madden. PIPENET: A wireless sensor network for pipeline monitoring. *in Sensor Networks*, pages 264–273, Apr. 2007.

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