International Journal of Computer Applications Technology and Research
Volume 3– Issue 1, 20 - 23, 2014

Float-based Pipeline Monitoring Network

Ata Barzegar
Computer Engineering Department
Islamic Azad University, Science and Research Branch, Khouzestan, Iran

Sattar Hashemi
Computer Engineering Department
Shiraz University
Shiraz, Iran

Gholamreza Akbarizadeh
Electrical Engineering Department
Shahid Chamran University of Ahvaz
Ahvaz, Iran

***Abstract*:** Wireless sensor networks are used to monitor water/oil pipelines. The excising schemes for pipeline monitoring are not energy efficient. Network operators have to daily travel across pipelines to change batteries of sensor nodes. In this paper, we propose a novel sensor-based pipeline monitoring protocol called FPMN. FPMN utilizes float nodes to reduce energy consumption in sensor nodes by reducing tasks of sensor nodes. FPMN tries to spread best route information along the pipelines in a way to reduce bandwidth and energy consumption. Our simulation experiments show that FPMN reduces energy consumption by 68.3% without change in packet delay and network traffic.

Keywords: wireless network; pipeline; sensor; forwarding; routing

1. INTRODUCTION

Sensor networks usually contain a large number of sensors distributed in a controlled area and a single sink node that collects sensed data from sensor nodes. Today's wireless sensors are very limited in terms of battery, processing power, and memory capacity except for the sink. For example, MICA2 contains 4 kilobytes memory [1].

During the operation of a wireless sensor network, sensor nodes frequently die because of physical damage or battery discharge. In addition, a packet transmitted between two wireless nodes may contain errors due to noise or interference [2].

In this paper, we propose a novel pipeline monitoring protocol called FPMN. FPMN tries to utilize float nodes instead of fixed nodes. FPMN contains a number of mechanisms to achieve this goal. Our simulation experiments show that FPMN reduces energy consumption in fixed nodes without change in network overhead and delay.

The rest of this paper is organized as follows. We review the related work on wireless sensor networks and pipeline monitoring designs in Section II. We propose FPMN in Section III. Section IV contains our simulation results and section V concludes the paper.

2. RELATED WORK

In this section, we review former researches related to our work.

A. *Wireles Sensor Network*

There are a number of routing protocols such as AODV [3], CTP [4] that discover the network topology to find a route. Then, each node identifies the neighbor to which it has to forward the packet. In this way, packets travel the shortest path from the source to the destination which is the sink in wireless sensor network.

Now we review the existing routing algorithms for wireless sensor networks. Directed Diffusion [5] routing protocol aims at diffusing the data through the sensor nodes by using a naming scheme. Rumor Routing [6] is a variation of Directed Diffusion intended for scenarios where geographic routing is not applicable. LEACH [7] is a cluster-based routing algorithm, but uses single-hop routing and can therefore not be applied to networks deployed in large regions. GPSR [8] uses a greedy forwarding strategy to forward only local information, but it has been designed for mobile ad hoc networks and requires a location service to map locations and node identifiers. GAF [9] is an energy-aware location-based routing algorithm. Its basic idea is to set up a virtual grid based on location information. Spectra uses the cluster-based Ripple routing algorithm. When a CH requires a route to BS, it broadcasts a Route-Request message in the network. If a CH receives a Route-Request and contains a route to BS, then it sends a Route-Reply message to the requesting CH.

SenseCode [10] is a collection protocol for sensor network to employ network coding. SenseCode provides a way to gracefully introduce a configurable amount of redundant information in the network, thereby increasing reliability in the face of packet loss.

B. *Pipeline Monitoring Networks*

PipeNet [11] is a pipeline monitoring system for collecting hydraulic and acoustic/vibration data at high sampling rates as well as algorithms for analyzing this data to detect and locate leaks.

MISE-PIPE is introduced in [12] to provide low-cost and real-time leakage detection and localization for underground pipelines.

SPAMMS [13] is an autonomous sensor-based system that combines robot agent based technologies with sensing technologies for efficiently locating health related events and allows active and corrective monitoring and maintenance of the pipelines.

Authors in [14] present a wireless sensor network system and its reliability assessment model for oil and gas pipelines condition monitoring.

SWATS [15] is a sensor network based system that aims to allow continuous monitoring of the steam flood and water flood systems with low cost, short delay, and fine granularity coverage while providing high accuracy and reliability.

In [16], for sensor nodes that are utilized to monitor oil pipelines, authors study the linear sensor placement problem with the goal of maximizing their lifetime.

In [17], an industrial wireless architecture was designed and simulated to transport data and video for a particular pipeline system.

3. THE FPMN PROTOCOL

In this section, we propose a pipeline monitoring protocol based on wireless sensor network. Our proposed protocol is called FPMN (Float-based Pipeline Monitoring Network).

We assume we have a number of crossing pipelines where wireless sensor nodes are placed along the pipelines (such as Fig. 1) such that the resulted wireless network is connected. A base station (sink) is placed at each end of every pipeline. Sensor nodes sometimes send monitoring data packets to the pipeline control room. All sinks are connected using a separate high speed network (such as the Internet) in a way that receiving a packet by the pipeline control room is equivalent to receiving that packet by any sink.

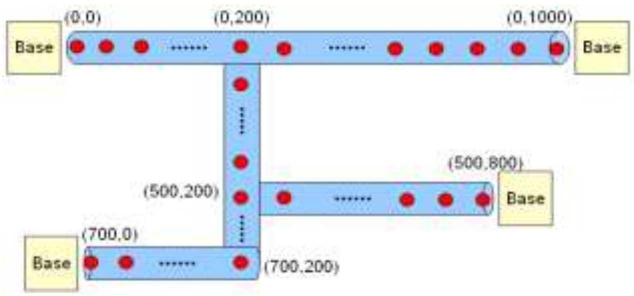


Figure 1 An pipeline network used in our simulation; Red points indicate sensor nodes; (x,y) indicates the physical cordinate of a point in the network.

The routing algorithm can be any routing algorithm that discovers the next hop to reach the sink. Each node has a fixed buffer capacity to temporarily save packets. Packet contains its generation time and expiration time.

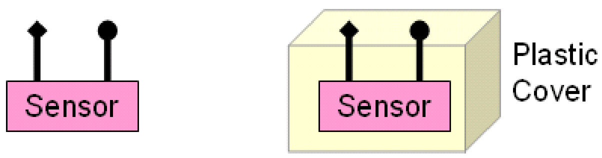
A. *System Design*

We use two kinds of nodes:

- Fixed Node: It does both sensing, routing, and forwarding.
- Float Node: It does only packet forwarding.

The simple way for making a float node is use of a fixed node inside a water-proof plastic cover (Fig. 2). Float nodes move by the power of the streaming liquid inside the pipe. Thus, float node requires a small battery only for data processing and packet forwarding.

Figure 2 Network nodes



(a) Fixed Node (b) Float Node

Workers are assigned at each pipeline head to do the followings.

- They periodically insert float nodes inside pipelines.
- They collect outgoing float nodes from pipelines and repair them and/or change their batteries.

Fixed sensor nodes monitor the pipeline and send the sensed data toward the sink. Both fixed and float nodes are able to forward data packets. The aim is to utilize float nodes as many as possible.

Using the neighbour identification mechanism, each node is able to find out which fixed/float nodes are its neighbours in what directions.

When a node is going to forward a packet, it checks whether there is a multipipe node inside its neighbourhood. Then, it select the next hop according to the following priorities.

1. Multipipe node (highest priority)
2. Float node
3. Single-pipe node (lowest priority)

We give the highest priority to multipipe nodes, because a multipipe node determines the best route for packets. A float node is not fixed and has no information about pipe routes.

B. *Buffer Management*

When the node receives or generates a new data packet, it tries to insert it in its buffer. If the buffer has no room, the node drops the packet.

Each node periodically checks its buffer. When the expiration time of a packet is passed, it is removed from node's buffer. This mechanism avoids delayed packet from staying in the network for a too long time.

C. *Node Behavior*

A sensor node gets one of the following roles when forwarding a packet in FPMN:

- Source Node: A sensor node that generates a data packet.
- Single-Pipe Node: A sensor node that has only two directions of a single pipe toward the sinks. (such as node *s* in Fig. 3)
- Multi-Pipe Node: A sensor node that has more than two directions toward the sinks due to residing on the junction of multiple pipes. (such as node *t* in Fig. 3)

The node acts differently when it gets one of the roles as below.

- Fixed Source Node: Fig. 4.
- Float Node or Fixed Single-Pipe Node: When the node receives a packet from side of the pipe, it simply forwards the packet toward the other side of the pipe. If the node is fixed, it increases the packet's FixedNodeCount by one before forwarding.
- Fixed Multi-Pipe Node: Fig. 5.

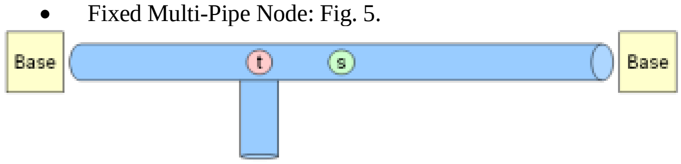


Figure 3 An example of pipelines

Algorithm. SourceSteps
Assumption: A node generated a packet *pkt* to send toward the sink.
I. Set FixedNodeCount(*pkt*) = 0
II. Mark *pkt* as a data packet.
III. Do the steps of a multi-pipe node.

Figure 4. Steps when a source generates a data packet

Algorithm. MultiPipeSteps
Assumption: The node knows the shortest path SP to the sink. The node has to forward a packet *pkt*. The packet is either generated in this node or received from another node.
I. Set FixedNodeCount(*pkt*) = FixedNodeCount(*pkt*) + 1
II. If all the branches have fixed node estimation, then
a. Find branch B that has the least number of fixed nodes among all the branches.
b. Send *pkt* on B.
III. Otherwise, if SP has fixed node estimation, then
a. Find branch B that has the least number of fixed nodes among the branches with fixed node estimation.
b. Send *pkt* on B.
IV. Otherwise, send *pkt* on SP.
V. End

Figure 5 Steps when a multi-pipe node has to forward a packet

TABLE I. SIMULATION SCENARIO I AND II

Parameter	**Scenario I**	**Scenario II**	**Scenario III**
Number of fixed nodes	125 and 250	125	125
Network Traffic	A Pareto-On/Off flow from every fixed node to the sink		
Average Bit Rate per traffic flow	1 Kbps		
The idle_time period per traffic flow	2 seconds		
Packet Expiration Duration	3 seconds		
Node's Buffer Capacity	20 packets		
Number of float nodes	125		
T_H	1.5 s	1.5 s	multiple cases
Float node speed	0.2 m/s	multiple cases	0.2 m/s
Simulation Duration	1 hour		

4. SIMULATION

We implemented the FPMN protocol in the NS2 [18] network simulator. In this section, we evaluate the performance and the overhead of FPMN. To evaluate FPMN, we define the two simulation scenarios presented in Table I. Fig. 1 depicts the pipeline network topology.

Every node sends traffic to the sink. We use a Pareto-On/Off model (one of the traffic generators in NS2 [18]) for each traffic flow with idle_time equal to 2 seconds. In this way, every node sends variable-bit-rate data and all the nodes do not synchronously send data to the sink.

In Scenario I, we use different numbers of nodes in different executions. In Scenario II, we use different values for float node speed in different executions whereas the other parameters are fixed. In Scenario III, we use different Hello intervals.

We simulated the following protocols:

- FPMN
- No-FPMN: Simplified FPMN without float nodes in which each source sends generated packets to the sink along the shortest path.

A. *Simulation Results*

End-to-end packet delay depends on node delay and path length. For packet delay, the main difference between FPMN and No-FPMN is that FPMN uses float nodes instead of some fixed nodes in data path. However, FPMN uses the least fixed node path instead of the shortest path sometimes. The least fixed node path is a short path and probably the shortest path. Therefore, there seems no delay difference between FPMN and No-FPMN. Fig. 6 shows end-to-end delay in the simulated protocols.

FPMN tries to utilize float nodes instead of fixed nodes as much as possible. Fig. 7 shows the energy consumed in fixed nodes. Energy consumption is proportional to number of nodes. In this experiment, FPMN averagely requires 68.3 percent less energy in fixed nodes compared to No-FPMN.

We define traffic overhead as follows.

Traffic Overhead = (Control Traffic) x 100 / (Data Traffic + Control Traffic)

Fig. 8 compares traffic overhead in the two simulated protocols. In this experiment, traffic overhead is averagely 20.8% in FPMN and 19.2 in No-FPMN.

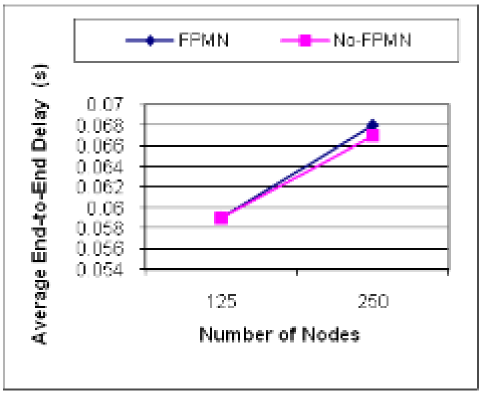


Figure 6 Average end-to-end packet delay versus number of nodes

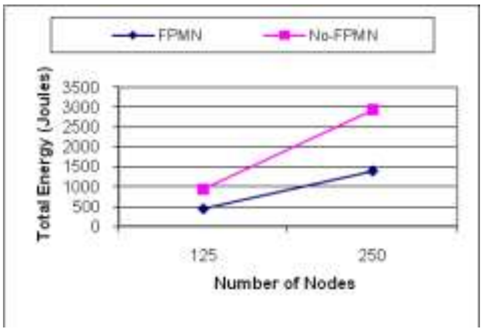


Figure 7 Total energy consumption in fixed nodes versus number of nodes

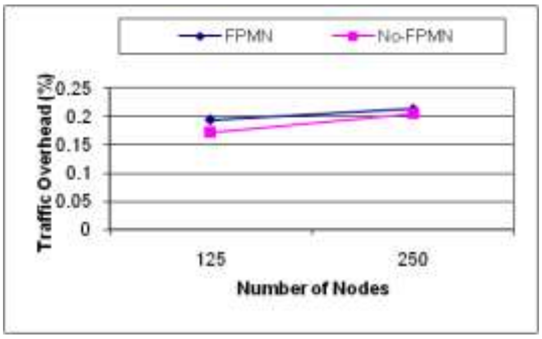


Figure 8 Traffic overhead in float and fixed nodes versus number of nodes

5. CONCLUSIONS

In this paper, we propose a novel pipeline monitoring protocol called FPMN. FPMN tries to efficiently spread packets along the pipelines by choosing the best path to send a packet. This path is either the shortest path or the least fixed node path. FPMN finds the least fixed node path using the fixed-node-count field in data packets. Our simulation experiments show that FPMN reduces energy consumption with acceptable overhead.

6. ACKNOWLEDGEMENT

The work described in this paper was supported by the Shahid Chamran University of Ahvaz (SCU) as a M.Sc. research thesis. It was financial supported with Grant treaty in 2013 with number 874095.The authors would like to thank from the Shahid Chamran University of Ahvaz (SCU) for financial support.

7. REFERENCES

[1] Crossbow technology inc. URL: http://www.xbow.com

[2] H. Karl, and A. Willig, Protocols and Architectures for Wireless Sensor Networks, John Wiley & Sons, 2005, ISBN: 0-470-09510-5.

[3] C. E. Perkins, "Ad hoc On Demand Distance Vector (AODV) Routing", Internet Draft, draft-ietfmanet-aodv-01.txt, 1998.

[4] "The Collection Tree Protocol (CTP)", http://www.tinyos.net/tinyos-2.x/doc/html/tep123.html.

[5] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks", In Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking, August 2000, Boston, USA.

[6] D. Braginsky, and D. Estrin, "Rumor routing algorithm for sensor networks", In Proceedings of the First Workshop on Sensor Networks and Applications (WSNA), October 2002, Atlanta, GA.

[7] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks", In Proceedings of the 33rd Hawaii International Conference on System Sciences, January 2000, Hawaii. IEEE, pp. 1–10.

[8] B. Karp, and H. Kung, "GPSR: Greedy perimeter stateless routing for wireless sensor networks", In Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking, August 2000, Boston, MA.

[9] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing", In Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking, July 2001, Rome, Italy.

[10] L. Keller, E. Atsan, K. Argyraki, and C. Fragouli, "SenseCode: Network coding for reliable sensor networks", EPFL Technical Report, October 2009.

[11] I. Stoianov, L. Nachman, and S. Madden, "PIPENET: A Wireless Sensor Network for Pipeline Monitoring", Int'l. Conf. Info. Process. Sensor Net., Apr. 2007.

[12] Z. Sun, P. Wang, M. C. Vuran, M. Al-Rodhaan, A. Al-Dhelaan, and I. F. Akyildiz, "MISE-PIPE: Magnetic induction-based wireless sensor networks for underground pipeline monitoring", Ad Hoc Networks 9(3), pp.218-227, 2011.

[13] J. Kim, G. Sharma, N. Boudriga, and S. Iyengar, "SPAMMS: A Sensor-based Pipeline Autonomous Monitoring and Maintenance System", Proceedings of The 2nd International Conference on Communication Systems and Networks (COMSNETS), Bangalore, India, pp.118-127, IEEE, January 2010.

[14] K. EL-Darymli, F. Khan, M. H. Ahmed, "Reliability Modeling of Wireless Sensor Network for Oil and Gas Pipelines Monitoring", Sensors & Transducers Journal, Vol. 106, Issue 7, pp. 6-26, July 2009.

[15] S. Yoon, W. Ye, J. S. Heidemann, B. Littlefield, and C. Shahabi, "SWATS: Wireless sensor networks for steamflood and waterflood pipeline monitoring", IEEE Network 25(1), pp.50-56, 2011.

[16] Y. Guo, F. Kong, D. Zhu, A. S. Tosun, and Q. Deng, "Sensor Placement for Lifetime Maximization in Monitoring Oil Pipelines", in the Proc. of the ACM/IEEE First International Conference on Cyber-Physical Systems (ICCPS), Stockholm, Sweden, Apr. 2010.

[17] R. Beatriz, and A. Meza, "Wireless sensors network for the cerro corona tailings pipeline system", in proc. of the Ibero-American Conference on Trends in Engineering Education and Collaboration, 2009.

[18] Information Sciences Institute, NS-2 network simulator, Software Packet, 2012, http://www.isi.edu/nsnam/ns/.