Pipelining in Structural Health Monitoring Wireless Sensor Network

Xu Li [a], Siavash Dorvash [b], Liang Cheng[c], and Shamim Pakzad[d]
[abcd]Lehigh Univ.,27 Memorial Drive West, Bethlehem, PA,USA 18015

ABSTRACT

Application of wireless sensor network (WSN) for structural health monitoring (SHM), is becoming widespread due to its implementation ease and economic advantage over traditional sensor networks. Beside advantages that have made wireless network preferable, there are some concerns regarding their performance in some applications. In long-span Bridge monitoring the need to transfer data over long distance causes some challenges in design of WSN platforms. Due to the geometry of bridge structures, using multi-hop data transfer between remote nodes and base station is essential. This paper focuses on the performances of pipelining algorithms. We summarize several prevent pipelining approaches, discuss their performances, and propose a new pipelining algorithm, which gives consideration to both boosting of channel usage and the simplicity in deployment.

Keywords: Pipeline, wireless, sensor, network, schedule

1. INTRODUCTION

According to the Federal Highway Administration (FHWA) report in 2008, over 25% of bridges in the United State are under standard [6]. This report implies the crucial condition of bridge structures and illustrates the need to individual strategy for the maintenance of these structures. A promising solution to this concern is Structural Health Monitoring (SHM). SHM provides the necessary information about the structure which enables the assessment of the bridge performance. This assessment over the lifetime of the bridge gives the opportunity to draw the performance degradation and making the best decision on maintenance and repair strategy of the bridge.

Traditionally, SHM has been developed based on the wired network connection of sensing nodes and centrally depository of data. While, the high costs and the installation difficulties of wired sensor network have limited the large-scale application of SHM, technological advances in wireless sensor network (WSN) have made the SHM more affordable and potentially scalable. In many recent SHM projects WSN is chosen as a promising data acquisition system [3],[4].

One of the challenges in WSN is data transmission protocol. The collected data in any sensing node is supposed to be transmitted to the base station over the network. The routing of this transmission has an effective role in the performance of WSN platform. Depending on the scale and the topology of the network, several algorithms are developed and applied by researchers. In the most simple network geometry, single-hop data transmission is commonly used. WHELAN et al. 2008, WHELAN et al. 2009 used of real-time single-hop data transmission in SHM. A more effective routing for large scale networks is hierarchical network topology, proposed by Gao and Spencer (2008). This transmission scheme divides the sensor network into the several hierarchical communities which each community has a cluster head in charge of collecting and processing the community's data and sending the outputs to the base station. This approach is developed based on the idea of Distributed Computing Strategy (DCS) in WSN (Gao et al. 2008). Another transmission scheme for long network topologies is multi-hop data transmission. In this approach, the data between two nodes which are not in the direct radio range is transferred using intermediary nodes [7]. Multi-hop data transfer in long span bridge monitoring is essentially important since using single-hop data transfer in such geometry requires a long radio range and consequently, a high level of power supply. In the case of scalable network, it is also essential to spatially reuse the network bandwidth which is called data pipelining. In general, data pipelining in WSN means simultaneous transmission of data from several nodes in the same network route. [7] employed data pipelining in the large-scale SHM of a long-span bridge and verified the functionality of this transmission scheme.

Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2010,
edited by Masayoshi Tomizuka, Chung-Bang Yun, Victor Giurgiutiu, Jerome P. Lynch, Proc. of SPIE
Vol. 7647, 76470I · © 2010 SPIE · CCC code: 0277-786X/10/$18 · doi: 10.1117/12.848112

Beside the importance of transmission routing as a part of software platform of WSN, hardware platform has an extensive impact on the performance of SHM. Verification of developed protocols on different hardware platforms is essentially important for widely use of WSN in SHM. Implementation of multi-hop data transfer with data pipelining, using Micaz hardware platform is reported in the literature [7]. This paper discusses the significance of this data transmission protocol for long-span bridge health monitoring using Imote2 hardware platform.

2. BACKGROUND

In wireless sensor network, there are two scarce resources: battery power and channel capacity. Pipelining is a mechanism that target to save both of them. The basic idea of pipelining is to schedule the behavior of the nodes along the multi-hop communication path, controlling the sending rate, so as to maximize the channel usage and minimize the energy and time consuming packets collisions, just as the example in Figure 1. One can think of a multi-hop channel as a "pipe", and the packets running on it as "water", the goal of pipeling scheduling is to keep this "pipe" full so as to transport the "water" as quick as possible.

As an effective scheduling mechanism, Pipelining is widely adopted when designing a data collection scheme. However, under different assumptions and limitations, there would be different variant of the original pipelining.

Most of the discussion about pipelining so far has a similar simplest scenario: a line network with each node placed with a distance d with its neighbors. In a bridge monitoring case, it could be interpreted as distribute the sensor nodes along the bridge. For simplicity, the communication range of each node is set to r , where d< r < 2d and the interference radius is also between d and 2d. This assumption rules out the interference among pairs of nodes with a distance farer than one hop between them. The node at the left end of the line acts as a base station, which is the source in data distribution and also the sink in data collection. Each of the other nodes has a weight v_i which stands for the amount of traffic generated by this node during the data collection. Finally, each node keeps a queue to buffer all the packets to be sent.

[1] focuses on such a kind of pipeline that all the wireless transceivers have directional antennas, which means sending packet to one neighbor would not affect the neighbor at the other side. The author of [1] develops an algorithm to control the sending rate of base-station, and successfully proved that this algorithm can gain optimal time efficiency in the resulting schedule. Although the analyzing of [1] is correct and helpful, it does make some assumptions that are not true in our application scenario of bridge monitoring. Firstly, the hardware plat form we adopt favors the economy, simplicity and robust, so the directional antenna may not be an available option. Secondly, in [1] the schedule data collection is simply considered as a reverse of data distribution, optimal as it is, it might not be realistic for execution. We will cover this in future part.

Other than theoretical analyzing, there are also instances of pipelining data collection scheme that already have been deployed. [3] described a data collection protocol called Flush, which is used in bridge monitoring. The pipelining issued in Flush defaults that sensor nodes use Omni-directional antennas. To cut down the cost and complexity, Flush adopt a receiver initiative mechanism, meaning that each traffic is activated by a request from the base station, and at any time instance, there is at most one flow in the line. The sending rate is controlled to make sure that there is no intra path interference. Simple as it is, Flush do leave some unused capacity during the process of filling up the pipeline: all the nodes, including the ones that temporally outside the interference range, are required to keep silence due to the one-sender limitation.

3. ALGORITHMS

In the context of bridge monitoring, our purpose here is to provide a pipelining scheme. that can fully utilize the channel capacity as well as the functions of the wireless networks.

As the time synchronization protocol such as TPSN and FTSP have already been widely incorporated in the wireless sensor network applications, we can assume the network is synchronized, that is, given a transmit schedule, all the nodes can execute it precisely. Also we chose Omni-directional antennas for each node to cope with the target hardware [4],[5]. Each node beside the base-station is supposed to know its own depth (the number of hops from base station) i, let t= i mod 3 denote the type of the node, which divides the whole network into three sort of nodes. At each time slot, only one of three types of nodes are sending, which will avoid interference with neighbors on either side. This periodical process would go on and on as well as the node has some packet in its sending queue.

3.1 Optimal Pipeline

First of all, we construct an optimal data collection schedule using the method similar to [1] : As shown in the graph, base station keep pump the data into the pipeline, and each remote node would forward the packets it received with a destination other that itself to its next hop right away. From the graph one can tell that most one-hop transmitting would involve three nodes: the sender, the receiver, and the one that is within the interference range of the sender. So for most of the time, each node would use three time slot to process one packet. Also notice that the sending queue at the basestation should be arranged that the packets for the furthest node are sent earlier.

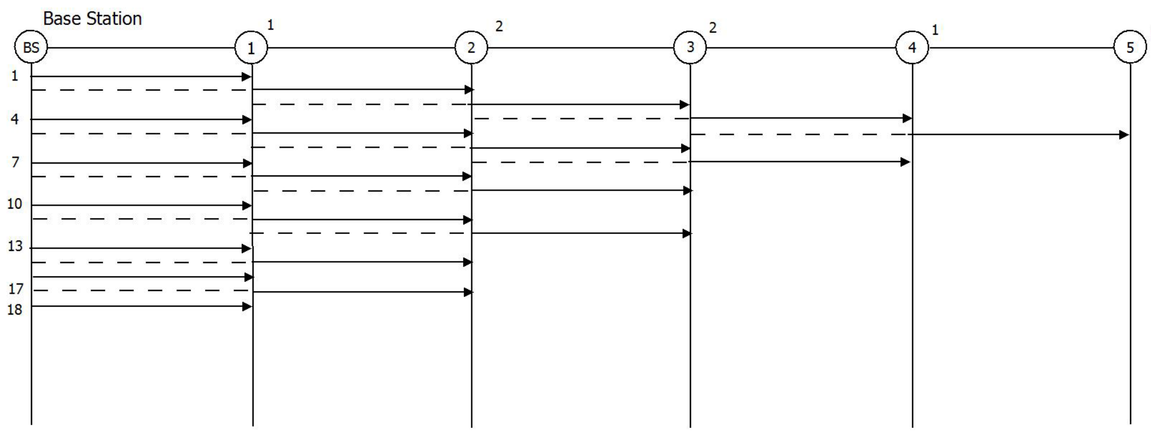

Figure 1 Pipeling used in data distribution in [1]

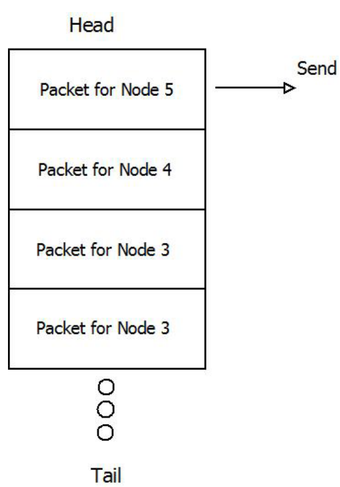

Figure 2 Sending queue for basestation in Figure 1

Let the last busy time slot of node i be denoted by T_i, then in this algorithm, we can calculate the T_i like this:

$$T_1 = 3\sum_{j=3}^{n} v_j + 2v_2 + v_1 \quad (1)$$

$$T_2 = 3\sum_{j=3}^{n} v_j + 2v_2 + 1 \quad (2)$$

$$(i > 2) \quad T_i = 3\sum_{j=i+1}^{n} v_j + 3(v_i - 1) + 1 + (i - 1) \tag{3}$$

Each T_i consists of three parts: the time required to forward the packets for downstream nodes, the time used to receive the packets from upstream nodes and the waiting time for the first packet to arrive (i.e. the pipeline be filled up from empty)

Now we prove this data distribution algorithm to be optimal.

We define the total transmission time as the time interval between the first node in the network become busy and the last node become idle, which can be expressed as $\max\{T_i\}$. Suppose there is a shortest distribution time $T_{optimal}$. Then

$$T_{optimal} \leq \max\{T_i\} \tag{4}$$

To prove the distribution algorithm is actually optimal, we construct a lower bound of $T_{optimal}$:

$$S_i = 3\sum_{j=i+1}^{n} v_j + v_i + (i - 1) \tag{5}$$

S_i consists of the least time a node i has to spend during the distribution, including the time needed for receiving, forwarding and avoiding interference for each packet forwarded (3 slots each), and the time to receive the packets designated to itself (1 slot each), and the time for waiting to be activated $(i - 1)$. By definition, we have

$$\max\{S_i\} \leq T_{optimal} \tag{6}$$

Theorem 1: In the pipelining data collection scheme produced by reverse data distribution,

$$T_{optimal} = \max\{T_i\}$$

This theorem can be verified by showing

$$\max\{S_i\} \geq \max\{T_i\} \tag{7}$$

If $v_i = 0$:

$$S_i = 3\sum_{j=i+1}^{n} v_j + (i - 1)$$

$$T_i = 3\sum_{j=i+1}^{n} v_j + (-3) + 1 + (i - 1) < S_i \tag{8}$$

If $v_i \geq 1$, we have two cases: either $i \leq 2$ or $i > 2$

For $i = 1$:

$$T_1 = 3\sum_{j=3}^{n} v_j + 2v_2 + v_1 \leq 3\sum_{j=3}^{n} v_j + 3v_2 + v_1 = S_1 \tag{9}$$

For $i = 2$:

$$T_1 = 3\sum_{j=3}^{n} v_j + 2v_2 + 1 \leq 3\sum_{j=3}^{n} v_j + 3v_2 + v_1 = S_1 \tag{10}$$

For $i > 2$:

$$T_i = 3\sum_{j=i+1}^{n} v_j + 3(v_i - 1) + 1 + (i - 1)$$

$$= 3\sum_{j=i+1}^{n} v_j + 3v_i + (i-3) \tag{11}$$

$$S_{i-1} = 3\sum_{j=i-1+1}^{n} v_j + v_{i-1} + (i-1-1)$$
$$= 3\sum_{j=i+1}^{n} v_j + 3v_i + v_{i-1} + (i-2) \tag{12}$$

Since $v_i \geq 0$ so $S_{i-1} > T_i$

Accordingly,

$$\max\{S_i\} \geq \max\{T_i\} \tag{13}$$

Intuitively, we can get an optimal data collection scheme by simply reverse the transmission direction in Figure 2 has shown in the graph, this scheme will share the same time cost as the corresponding distribution algorithm.

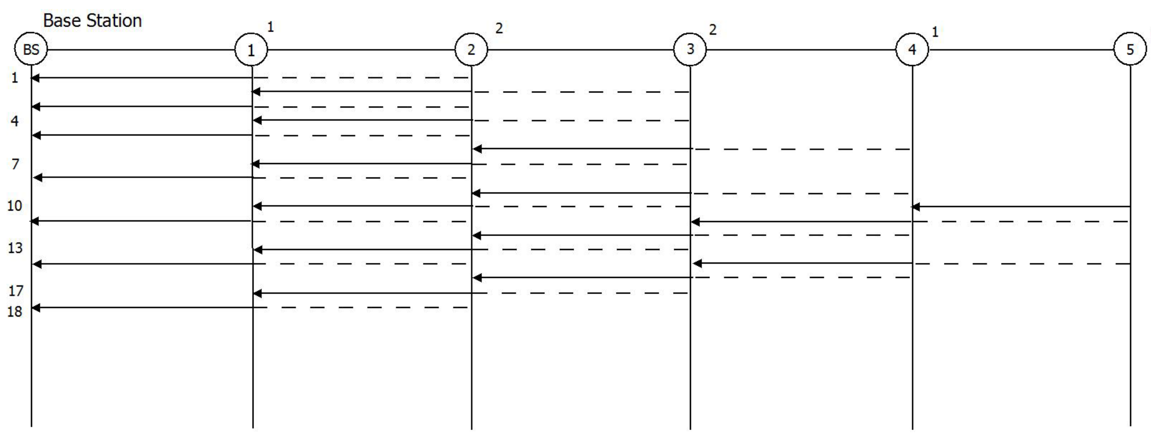

Figure 3 Reverse of the data distribution in Figure 1

3.2 Mod-3 Pipelining Algorithm

Effective as this reverse operation is, it has some potential issues in application. The most important deficit is that it is a centralized algorithm. Take the graph above as an illustration, the timing of the behavior of node 2 depends on v_1, and the timing choice of node 5 depends on the scheme of all 4 nodes ahead of it. Unfortunately this disadvantage is an intrinsic issue of this approach. Since data distribution is in fact a centralized process: all the nodes are following the base-station so is its reverse.

As in real wireless sensor network, it is unrealistic to gain the payload information v_i beforehand. Thus we need to find an alternative decentralized algorithm for data collection. We issue a mod-3 algorithm to address this issue. Before the pipelining start, every node is supposed to know its depth (the number of hops to the base-station) d_i. At time slot t, only nodes $\{i|(d_i \bmod 3) \equiv (t \bmod 3)\}$ can send packets to its parent node. The algorithm is demonstrated in the following graph.

In this mod-3 algorithm, each flow is independent, so the total time can be calculated as

$$T_{total} = \max\{T_i\} \quad (i = 1,2 \ldots n) \tag{14}$$

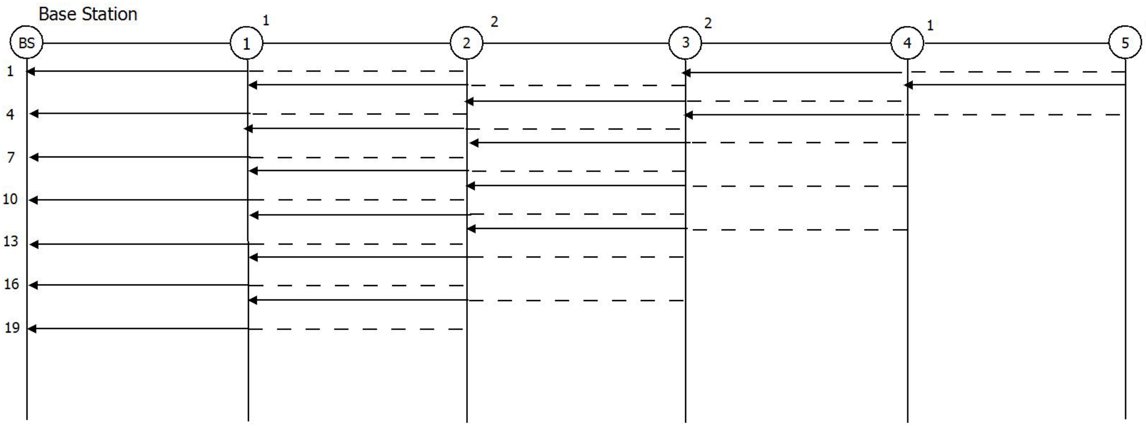

Figure 4 Mod-3 pipeling algorithm

We can see from this example, the mod-3 algorithm do not require knowledge of global scheme, and its speed is only slightly lower than the optimal condition. We will conduct a detailed comparison in next section.

4. PERFORMANCE ANALYSIS

In this part we will compare the speed of three algorithms we covered so far.

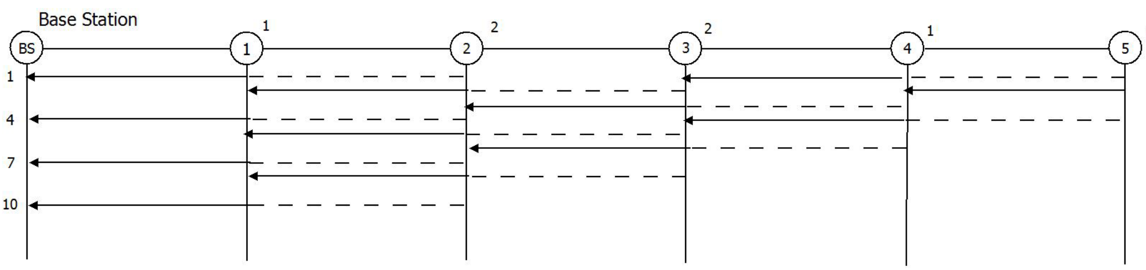

Figure 5 Single Path Pipeling Algorithm as used in Flush

The optimal transmission time gained by the reverse distribution is stated in last part; the single path pipelining algorithm used in Flush protocol fills up and empties the pipeline for each sender, so the total time consumed should be

$$\sum_{i=1}^{n}[3(v_i - 1) + 2i] = 3\sum_{i=1}^{n} v_i - 3n + n(1 + n) \qquad (15)$$

The first part is the time used in transmit the data on node i, and the second part is the time to One may argue that the senders with $v_i = 0$ should not be counted in. However, as the Flush uses a query based mechanism, there is no way for a base station to know in advance whether certain node have something to send or not. So base-station still need to wait at least i-1 time slots to get a report for "nothing to send". In this sense, a node with $v_i = 0$ actually has one packet (null report) to send, thus its weight should be modified to one.

The mod-3 Algorithm takes advantages in the phase of filling up the pipeline, it only takes one period, that is, 3 time slots to fill up the whole line. Here we again consider the weight of nodes without generating any local packets as one, as they will produce a "hole" in the pipeline that take exactly the size of one packet. This lead to the result of total transmitting time to be:

$$3(\sum_{i=1}^{n} v_i - 1) + 1 = 3\sum_{i=1}^{n} v_i - 2 \qquad (16)$$

Obviously, it will be much faster than the single path pipeline.

The figure below is the illustration of comparison among different kinds of pipeling algorithms. The horizontal axes are the mean value of packets number and the number of hops along the path. The vertical number is the time slots needed to deliver all the packets. We assume that the number of packets waiting to be sent at each node is a random number with uniform distribution. From the top to bottom, the plotting are for: Single path pipelining, mod-3 pipelining and the optimal condition.

From the graph we can see that the performance of mod-3 algorithm is very close to the optimal case, and constantly superior to the single path algorithm.

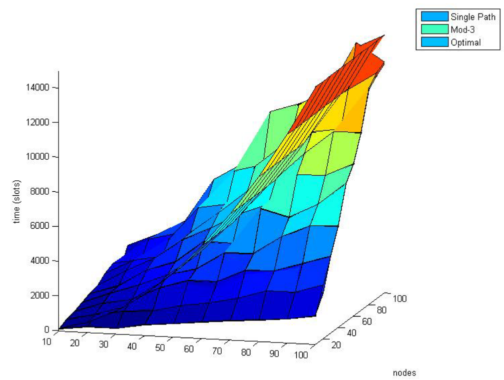

Figure 6. Performance of three pipelining algorithms

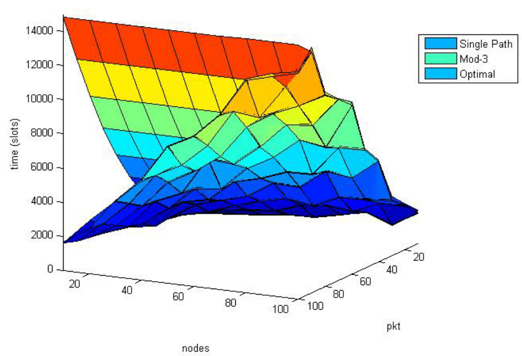

Figure 7. Performance of three pipelining algorithms (another angle)

5. CONCLUSIONS AND FUTURE WORK

Pipelining is proved to be an effective way to fully utilize the bandwidth as well as avoid packets collisions. Similar as the underlying logic, different pipeline strategy should be utilized basing on the payload size and network scale. Specifically, when the distance is long (hop number is high) enough, the deficiency of single path pipelining is enlarged so much that may even overweight its advantage in system simplicity.

As another major advantage of pipelining, the energy efficiency is not fully covered in this paper (although the avoidance of data collision and retransmission should be relevant). Energy analyzing of multiple pipelining protocol should be our focus in the future.

ACKNOWLEDGMENTS

The research described in this paper is supported by the National Science Foundation through Grant No. CMMI-0926898 by Sensors and Sensing Systems program. The authors thank Dr. Shih-Chi Liu for his support and encouragement.

REFERENCES

[1] Florens, C. Franceschetti, M. and McEliece, R.J. "Lower Bounds on Data Collection Time in Sensory Networks", Selected Areas in Communications, IEEE Journal on 22, pp 1110-1120, (2004)

[2] Choi,H. Wang,J. and Hughes,E.."Scheduling for information gathering on sensor network", Wireless Networks, 15, pp 1022-0038,(2007)

[3] Kim,S. "Wireless Sensor Networks for High Fidelity Sampling". Technical Report No. UCB/EECS,91 (2007)

[4] Illinois Structural Health Monitoring Project (ISHMP)http://shm.cs.uiuc.edu/

[5] Crossbow http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICAz_Datasheet.pdf

[6] Rice, J. and B. F. Spencer, Jr., "Structural Health Monitoring sensor development for Imote2 Platform", Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems. San Diego, California, USA, 6932, 693234, (2008)

[7] Pakzad, S.N., Fenves, G.L., Kim, S., and Culler, D.E. "Design and Implementation of Scalable Wireless Sensor Network for Structural Monitoring", ASCE Journal of Infrastructure Engineering, 14(1):89-101.(2008)