

Multi-path Reliable Routing with Pipeline Schedule in Wireless Sensor Networks

Jinbao Li^{1,2}, Li Zhang¹, Longjiang Guo^{1,2(✉)},
Qianqian Ren^{1,2}, and Yahong Guo³

¹ School of Computer Science and Technology,
Heilongjiang University, Harbin, China
longjiangguo@hlju.edu.cn

² Key Laboratory of Database and Parallel Computing, Harbin, Heilongjiang, China

³ School of Information Science and Technology,
Heilongjiang University, Harbin, Heilongjiang, China

Abstract. To solve the problem of data missing caused by communication collision in Wireless Sensor Networks (WSNs), this paper proposes a multi-path reliable routing protocol. This protocol combines the technique of pipeline with multi-path reliable routing, and uses Markov chain to predict the period of using pipeline in order to schedule pipeline dynamically. Therefore, the combined techniques improve the network throughput. In the case that data congestion occurs, and the protocol can keep the data transmission rate unchangeable, and the proposed routing protocol can choose other path to transmit data in order to reduce the number of data missing. Simulation results indicate that the proposed protocol can deal with congestion and improves the network throughput effectively.

1 Introduction

The restriction of resources poses a severe challenge to improve the network performance in WSNs [1–4]. It is important to improve the reliability of data transmission for wireless sensor study under the prerequisite of network throughput.

Compared with the single-path routings, it can improve throughput of the network using multi-path routing. Data transmission have concurrency in multi-path routing, as the data transmitted by more than one path at the same time

L. Guo—This work is supported by the National Natural Science Foundation of China under grant No.61370222, 61300225, Heilongjiang Province Science and Technique Foundation under Grant No. F201324, Program for Group of Science and Technology Innovation of Heilongjiang Educational Committee under grant No.2013TD012, Harbin Municipal Science and Technology Innovation Talent research special funds outstanding academic leaders funded project under Grant No.2015RAXXJ004, Program for New Century Excellent Talents in University under grant No. NCET-11–0955, Programs Foundation of Heilongjiang Educational Committee for New Century Excellent Talents in University under grant No.1252-NCET-011.

[5–8]. Different sending rates will affect the network throughput directly when the source node sends data to multiple neighbors. The high sending rate of the source node will cause data congestion, conversely, the network link can't fully be exploited.

In recent years, there are many researches in control congestion in the wireless network [9–12]. Popa et al. proposed IPS [9] which uses two paths to transmit data simultaneously. When data congested at a node, they can be transmitted to the destination node through a new temporary path, thus reducing the data missing caused by network congestion. Sridevi et al. [10] use the sensor nodes with a variety of perceived parts and assign priorities based on the type of transmitted data to ensure the important data forwarded earlier. Wang et al. [11] use the sequence length to monitor the network congestion, transmit the congested data when the congestion occurs, and thereby control the degree of congestion. Pham [12] use the strategy that allocate rate to different paths at the source node to achieve load balancing and avoid data congestion. The general processing strategies of the above works are adjusting the sending rate of source node in order to reduce data congestion. However, the throughput will be affected by the reduce of sending rate at the source node.

Therefore, it is important to study how to deal with data congestion effectively under the condition of keeping the sending rate of the source node when data congestion occurs. To address the problem, this paper introduce pipeline, and combine it with multi-path routing. this strategy can scheduling data transmission between multiple paths effectively, while keeping the sending rate of the source node. The contribution of this paper is as follows: combining pipeline with multi-path routing based on the data conversion relationship between multiple paths; Proposed a predict method based on Markov chains, predict the appointment time of the multi-path pipeline and schedule it dynamically. Simulation results show that combine pipeline with multi-path can improve network throughput effectively and make the data transmission more reliable.

The remainder of the paper is organized as follows, Sect. 2 proposed the protocol of multi-path reliable routing based on pipeline; Sect. 3 introduced how to select pipeline scheduling time; Sect. 4 is the simulation experiment and result; The final section is conclusion.

2 Protocol Design

This paper make the following assumptions: there exist a ordered path set $Path(s, t) = \{\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_n\}$ from any source node s to destination node t , which contains n disjoint paths and sorted from short to long according to the path length. T_i is the total number of nodes except source node and destination node on path \mathcal{P}_i .

2.1 Establish Conversion Relationship Between Multi-paths

In this paper, we use the bridge chain establishment algorithm which proposed in literature [13] to establish the data conversion relationship between multiple paths.

Algorithm 1. Selection algorithm of multi-path.

Input:1: $Path(s, t) = \{\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_n\}$;**Output:**2: k disjoint paths $OP(s, t) = \{\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_n\}$;

3:

4: select m paths from the input set randomly;5: **if** $m < 2$ **then**

6: Goto step 1;

7: **else**8: **if** $m = 2$ **then**

9: Build-Bridge;

10: **else**11: $i = 1$;12: **while** $i < m$ **do**13: Establish the bridge chain between P_i and $P_{(i+1)}$, $i++$;14: **end while**15: Calculate the reliability of these m paths and goto step 1 until m have selected every data in $\{2, 3, \dots, n\}$. Then select the combination with maximum reliability from multi-path combination;16: **if** the combination is unique **then**17: Output all paths in $OP(s, t) = \{\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_n\}$;18: **else**

19: Select the combination with the minimum numbers of paths;

20: **end if**21: **end if**22: **end if**

If data congestion occurred, data can be transmitted to other path through bridge chain to ensure the reliability of data transmission.

For $\forall P_i \in P(s, t)$, set $t=1$ if data can be transmitted through bridge chain when data congested at node t , otherwise $t=0$. Therefore the transmit rate of path P_i is:

$$Transfer_i = \sum_{k=1}^{T_i} t_k p_k / T_i \quad (1)$$

where $Transfer_i$ is the transmit rate of path P_i . T_i is the total number of nodes except source node and destination node on P_i , p_k is the failure(or data congested) probability of the k -th node on the path. Then we can get the reliability of $P(s, t) = \{\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_m\}$

$$R_{i,j} = (Transfer_1 + \dots + Transfer_m) / m \quad (2)$$

Algorithm 1 is the multi-path selection algorithm. There exist n disjoint paths from source node s to destination node t in the network after s and t have been determined. Algorithm 1 can determine the number of final routing paths $m(m \leq n)$.

2.2 Conversion Between Multi-paths and Pipeline

We use the Algorithm 1 that can select routing paths and regard m disjoint paths as a pipeline which has m pipeline segments with different functions. Assuming that the data congested paths are selected to be the start segment of the pipeline, we can get a dynamic pipeline graph with feed forward and feedback connections through regarding the conversion relationship as processing sequence between pipeline segments. At last, combine the multi-path with pipeline, and guidance the routing schedule by pipeline scheduling scheme.

For example, the network showed in Fig. 1, using the Algorithm 1 to establish bridge chains and determine routing paths. In Fig. 2, the data on \mathcal{P}_1 can be transferred to \mathcal{P}_2 through $b_{\mathcal{P}_1, \mathcal{P}_2}(1, 7), b_{\mathcal{P}_1, \mathcal{P}_2}(3, 9), b_{\mathcal{P}_1, \mathcal{P}_2}(4, 10)$. When data congested on \mathcal{P}_1 , they can be transferred to \mathcal{P}_2 , and then transferred to \mathcal{P}_3 . \mathcal{P}_2 would have output port if the data that transferred from \mathcal{P}_1 to \mathcal{P}_2 can be sent in time by \mathcal{P}_2 , otherwise the data will be transferred to \mathcal{P}_3 .

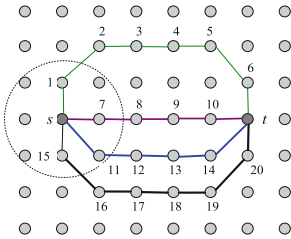


Fig. 1. Network topology

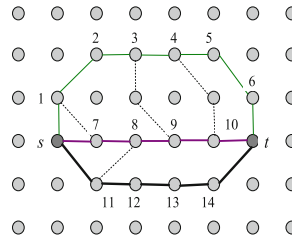


Fig. 2. Routing path

Definition 1. Reservation table is a two-dimensional table which satisfied the following three conditions. (1) Transverse is ordered by time, represented by unit time. (2) Longitudinal represents each path. (3) The cell that row a and column b is set 1 if path b is scheduled at time a , otherwise empty.

When data congestion occurs in the network, we can obtain the reservation table according to the dynamic pipeline graph from the previous section. Monitoring the network throughput at destination node with 10 unit time, and convert the pipeline when the value of the throughput is less than the threshold. As the reservation table is not unique, we can choose different reservation table based on the state of each routing. A row of a reservation table can contains multiple 1 which means a path can be called at different moments. Multiple 1 in a column means that at some point, the data in a path can be transferred to more than one path simultaneously. Figure 4 are several reservation tables without repeated cycle which come from Fig. 3.

Definition 2. The time interval between two starts of pipeline called **waiting Time** w . In waiting time, it will cause resource conflict if start a pipeline segment two or more times so the original congestion path couldn't transfer more congested data to other paths.

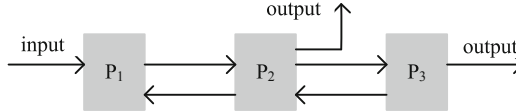


Fig. 3. Pipeline

Definition 3. **banned waiting time** is the waiting time that cause conflict.

Corresponding to banned waiting time is the allowed waiting time which cause no conflict. For a reservation table with t columns, the max banned waiting time is $p \leq t - 1$, allowed waiting time is $1 \leq q \leq p$, and the smaller allowed waiting time, the higher efficiency of the pipeline. Literature [14] points out through checking the distance between any two “1” in a row, we can obtained the banned waiting time. Such as the banned waiting time is 2 and 4 in the third reservation table of Fig. 4, and the allowed waiting time is 1 and 3.

	1	2	3
P ₁	1		
P ₂		1	
P ₃			1

	1	2	3	4	5
P ₁	1				
P ₂		1		1	
P ₃			1		1

	1	2	3	4	5	6
P ₁	1				1	
P ₂		1		1		1
P ₃			1			

Fig. 4. Network topology

In the mechanism of pipeline, we can use the conflict vector to represent the banned waiting time and allowed waiting time simultaneously.

Definition 4. **Conflict vector** is a binary vector with p bits, represented by $C = (C_p, C_{p-1}, \dots, C_2, C_1)$. where $C_p = \{0, 1\}$, $1 \leq p \leq (t - 1)$. Waiting time i is banned waiting time if $C_i = 1$; and allowed waiting time if $C_i = 0$. $C_p = 1$ indicates the highest bit in conflict vector is always equal to 1^[10]. The vector is (1010) in the third reservation table in Fig. 4.

According to the pipeline mechanism, we can use the conflict vector to construct a **state graph** [14] to explain states change, and then get the minimum waiting time of the pipeline. The method to get conflict graph from conflict vector is as follows: The initial conflict vector shift one bit to right, and the vacated one of the extreme left padded with 0. After shifting k times, k is a allowed waiting time if the digital shifted out is 0, then do bitwise OR between the new vector and the original conflict vector. The purpose of bitwise OR is to avoid conflict that start from $k + 1$ and after. Therefore, conflict graph contains all the allowed state changes without conflict. All the waiting time which is larger than the maximum banned waiting time is the allowed waiting time. Although the scheduling conflicts can always be avoid in a sufficiently long period of time,

waiting a long time would lose the meaning of pipeline, no matter the view from industrial demand or network requirements.

For the third reservation table in Fig. 4, the conflict vector is (1010), and the conflict graph is showed in Fig. 5. Shifting 1 bit to right, then get the new vector (0101). (1111) is the result of (1010)OR(0101), and at the third shift (1010) get to (1011), (1011) go back to itself after the third shift. We can obtain some waiting loops from the state graph, such as (1, 5), (3, 5), (3) and (5), then calculating the average waiting time of them, and guiding the schedule of the pipeline by choose the waiting loop with the minimum waiting time.

3 The Selection of Scheduling Time in Assemble Line

In this paper, we proposed a prediction algorithm based on the Markov chain-MAPA. MAPA divide the network into different states according to the current network throughput, and use Markov chain theory to predict the using time of the current pipeline.

3.1 Pipeline State's Classification

Assuming packet reception ratio R is the ratio of the number of packets successfully received by the destination node to source node sends. Before the congestion occurs, R is R_{high} at the monitoring node and regard R_{high} as the R of the network in a good state. When the congestion occurs, let R_{low} as the R at the destination node without taking any congestion handling strategy. Part of the congested data is lost as the forward isn't timely, so $R_{low} < R_{high}$.

To sum up, the pipeline can be divided into three states. when $R \geq R_{high}$, the pipeline is in the valid state; available state if $R_{low} \leq R < R_{high}$, and invalid state if $R < R_{low}$.

Figure 6 shows an example of the three kinds of states of the pipeline, denoted by A, B, and C. Assuming that the packet reception ratio at the destination node changed from top to down, then the states that the pipeline experienced are A-B-C, that is turn to the available state from the valid state and turn to the invalid state lastly.

For a pipeline, saying that the pipeline is in state i at time t if $X_t = i$, $i = \{1, 2, 3\}$ indicates the pipeline is in the valid state, available state, and invalid state respectively. Assume that there is a probability $P_{i,j}$ make it in the state j at the next time if the pipeline is in state i . Assume for any state i , if $n \geq 0$,

$$P \{X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, \dots, X_1 = i_1\} = P_{i,j} \quad (3)$$

Given the past state and current state, the conditional distribution of the future state is independent of the past state, only depends on the current state.

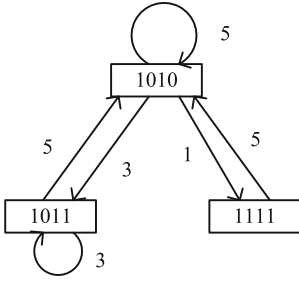


Fig. 5. State graph

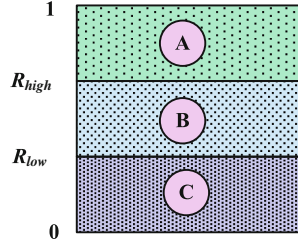


Fig. 6. 3 kinds of states in pipeline

We can regard the converting process of pipeline as a three-state Markov chain process. As Markov chain process must be converted from one state to another state, and the probability is non-negative, so:

$$\sum_{j=1}^3 P_{i,j} = 1, i = 1, 2, 3 \tag{4}$$

M is a one-step transfer probability matrix, like Fig. 7. If we use state 1, state 2 and state 3 denote invalid state, available state and valid state respectively, and assume that the current state of pipeline is in valid state, then $M_{3,1}$ indicates that the probability of the pipeline is in failure state at next time.

$$M = \begin{pmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{pmatrix}$$

Fig. 7. One-step transfer matrix

On the basis of one-step transfer probability, we study n -step transfer probability $P_{i,j}^n$ in this section. $P_{i,j}^n$ is the probability that the current state is i and in state j after $n(n > 0)$ moments, namely:

$$P_{i,j}^n = P \{X_{n+t} = j | X_t = i\} \tag{5}$$

Clearly, $P_{i,j}^1 = P_{ij}$, that is the current state is i and reach to the state j after one-step transfer. According to the approach to calculate n -step transfer probability offered by C-K [15] equation,

$$P_{i,j}^{n+m} = \sum_{k=0}^{\infty} P_{i,k}^n P_{k,j}^m \tag{6}$$

Formula (6) denotes the pipeline of the current state which is i , and reach to state k after n times transfer, then it reach to state j after transfer m times. Let $\mathbf{M}^{(n)}$ indicates the transfer probability of $P_{i,j}^n$ after n times transfer, formula (6) can also denoted as follows:

$$\mathbf{M}^{(n+m)} = \mathbf{M}^{(n)} * \mathbf{M}^{(m)} \quad (7)$$

The $*$ in formula (7) indicates matrix multiplication as follows, and formula (8) is determined by induction. $\mathbf{M}^{(3)} = \mathbf{M}^{(2+1)} = \mathbf{M}^{(2)} * \mathbf{M}^{(1)} = \mathbf{M}^{(1+1)} * \mathbf{M}^{(1)} = \mathbf{M}^{(1)} * \mathbf{M}^{(1)} * \mathbf{M}^{(1)} = \mathbf{M} * \mathbf{M} * \mathbf{M} = \mathbf{M}^3$.

$$\mathbf{M}^{(n)} = \mathbf{M}^{(n-1+1)} = \mathbf{M}^{n-1} * \mathbf{M} = \mathbf{M}^n \quad (8)$$

3.2 Prediction Algorithm Based on Markov Chain

This paper uses the pipeline transfer matrix to predict the time slot of the reservation table. The purpose using transfer matrix is to predict the next state of the pipeline according to the current state, and thus predict the state n times later. Suppose the current time is t , stop calculate if the pipeline transfer to invalid state with a high probability at sometime k when calculate the pipeline transfer matrix. Then take the time slot $(k - t)$ as appointment time of current pipeline. In appointment time, Scheduling pipeline in accordance with its current schedule. Convert pipeline if the time reach to k , as current pipeline is in invalid state with a high probability after k .

Now, concrete the one-step transfer matrix in Fig. 7, and then calculate the n -step transfer matrix. Assume that the current packet reception ratio of the pipeline is R , and the biggest change value of packet reception ratio is Δ . For pipeline, this section is concerned on whether the state will eventually reach to the invalid state, convert the pipeline immediately if it happens, so $P_{11} = 1$, $P_{12} = 0$, $P_{13} = 0$.

P_{21} is the probability that the pipeline transfer from available state to invalid state. Such as Fig. 8 shows, the packet reception ratio is R when the pipeline is in available state. At next time, the biggest change value of the packet reception ratio is Δ , so there exist a cycle, R is the center, Δ is the radius, and the packet reception ratio ranging from point a to point b at next moment. P_{21} is the radio of the length of pink line in Fig. 8(a) and 2Δ , $P_{21} = \frac{\Delta - (R - R_{low})}{2\Delta}$.

P_{23} is the probability that the pipeline convert from available state to valid state. Shows as the Fig. 9, P_{23} is the radio of the length of pink line in Fig. 8(b) and 2Δ , $P_{23} = \frac{\Delta + (R - R_{high})}{2\Delta}$. Then $P_{22} = 1 - P_{21} - P_{23} = \frac{R_{high} - R_{low}}{2\Delta}$.

P_{31} is the probability that the pipeline convert from valid state to invalid state. Shows as the Fig. 10, P_{31} is the radio of the length of pink line in Fig. 10 and 2Δ , $P_{31} = \frac{\Delta - (R - R_{low})}{2\Delta}$.

P_{32} is the probability that the pipeline convert from valid state to available state. Shows as the Fig. 11, P_{32} is the radio of the length of pink line in Fig. 11 and 2Δ , $P_{32} = \frac{\Delta - (R - R_{high})}{2\Delta}$. Then $P_{33} = 1 - P_{31} - P_{32} = \frac{2R - R_{high} - R_{low}}{2\Delta}$.

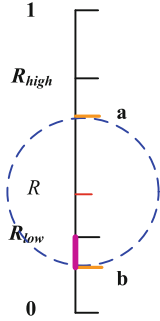


Fig. 8. Available state to invalid state

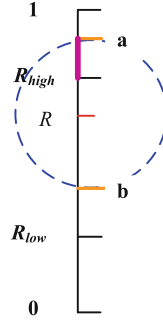


Fig. 9. Available state to valid state

We can calculate the n -step transfer matrix after get the one-step transfer matrix. If $M_{3,1}^{(n)}$ is larger than a certain threshold, stop the calculation and record the matrix order number n as a prediction time of the reservation table. During that period, select an pipeline schedule to guide the congested data transfer, convert the pipeline after n units of time. Should be noted, After $P_{21}, P_{22}, P_{23}, P_{31}, P_{32}$, and P_{33} have been determined, Learned by observation and calculation, there is a limiting probabilities making $M_{3,1}^{(n)}$ values are no longer change. The proposed prediction algorithm based on Markov chain can be calculated in the finite number of steps.

Based on the above thoughts, this section presents a prediction algorithm based on Markov chain (MPRR). The basic idea is as follows: monitoring packet reception ratio R at the destination node if congestion occurs. According to the probability of the packet reception ratio decreases, construct the one-step transfer equation, and then construct the n-step transfer equation. If the probability that the pipeline convert to invalid state is larger than threshold Q or reach to the limiting probabilities of $M_{3,1}^{(n)}$, stop calculate. Output the order of the matrix, and regard it as the length of the appointment time of the pipeline reservation table, then obtained the data transmit between the multiple paths through pipeline.

4 Experiment and Analysis

This section verifies the performance of the proposed method through simulation experiments. 1000 sensor nodes are evenly deployed in a 100 m*100 m network, we use the grid structure network in order to make experiment more convenient. Assume that the transmission period of each node is 50 ms, and the length of each packet is 32 Bytes, the running time is 60 s, a pipeline segment converts its state with the 7th and 33th seconds which are unable to transmit data in time. Experiments investigate four aspects: the sampling period affected the network throughput, the sampling period affected the loss of the transferred data, the

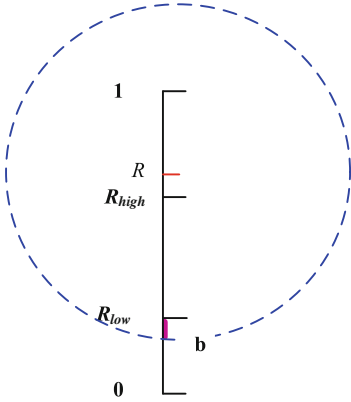


Fig. 10. Valid state to invalid state

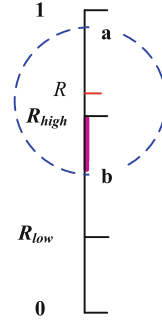


Fig. 11. Valid state to available state

number of pipeline segments affected the throughput and the number of pipeline segments affected the loss of the transferred data.

4.1 The Impact of Sampling Period

Figure 12 is the impact of execution time on the throughput when the shortest path is 5 hops. F is the sampling period and we set the time of real-time monitoring is one second. From Fig. 12, we can see that with the increase in execution time, the number of packets received by the destination node is on the rise no matter how the sampling period changes. However, from the Fig. 12 we can clearly observe, the network throughput was reduced with the increase of the sampling period. This is because the changes of the pipeline can be found by real-time monitoring in a timely manner, when a pipeline is no longer suitable to continue to use for some reasons, real-time monitoring can convert the pipeline in time to ensure that the network has a larger throughput.

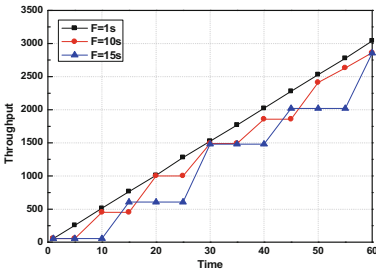


Fig. 12. Sampling period vs throughput

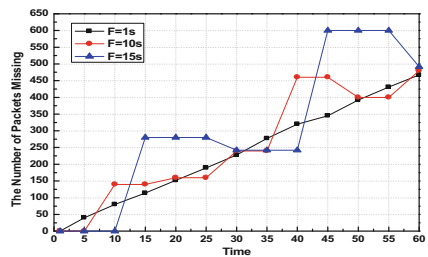


Fig. 13. Sampling period vs the number of packets missing

Figure 13 is the impact of the sampling period on the transfer of data missing when the shortest path is 5 hops. From Fig. 13 we can observe that with the increase of the sampling period, the loss of transferred data is also increased. When $F = 10\text{ s}$ or $F = 15\text{ s}$, transferred data missing remain unchanged in some moments for the periodic sampling, more data loss in some moments for the state of pipeline changes, and once this phenomenon was discovered by periodic monitoring, it convert the pipeline to declined the loss of transferred data.

4.2 The Impact of the Number of Pipeline Segment

When the shortest path length is 5 hops, the sampling period is 10 s, the number of pipeline segments affected the network throughput was shown in Fig. 14. ALN is the number of pipeline segment, as the number of pipeline segment is the number of routing path, so when increase the former, the number of paths in the network can be used in parallel is increased, thus the network throughput is also increased. When increasing the number of pipeline segment, the transferred data can be transmitted to other more paths, which led to the growth of the transfer path, so as the number of data missing.

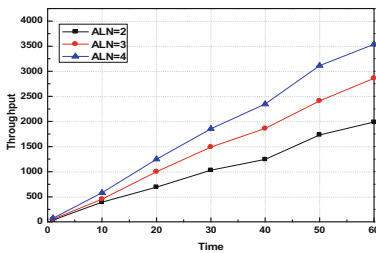


Fig. 14. Numbers of pipeline segment vs throughput

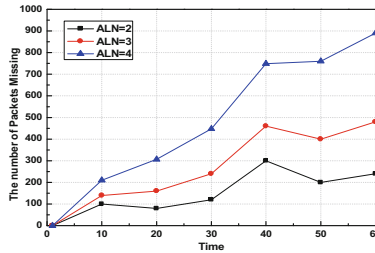


Fig. 15. Numbers of pipeline segment vs data missing

Figure 15 shows the number of pipeline segment affected the transferred data missing when the shortest path length is 5 hops and the sampling period is 10s. The loss of transferred data increases with the time increase. For periodic sampling, data loss can't be found timely at the destination node, just be found in the periodic time. What's more, as the increase of ALN, data missing get seriously for the elongate of transfer path.

4.3 The Impact of Predicting Time

Assuming when the program is running, the threshold is set to 0.735, and the others don't change. F is the fixed sampling period, $F = 1$ is real-time monitoring. PF is the prediction period, more flexible than fixed period. From Fig. 16 we can see that the throughput of periodic monitoring is close to the throughput of real-time monitoring which explain the feasible of the periodic monitoring approach.

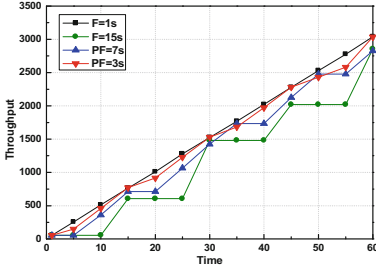


Fig. 16. Prediction time affected the network throughput

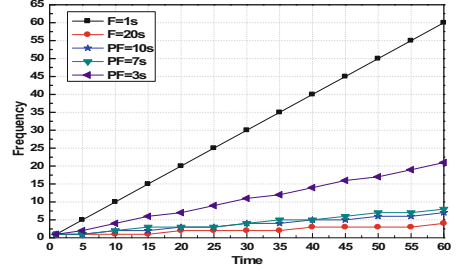


Fig. 17. Prediction time affected the frequency of monitoring

When the shortest path length is 5 hops, Fig. 17 illustrates the impact of the prediction time on the frequency of monitoring. In sensor networks, there is a direct relationship between frequency of monitoring and energy consumption, monitoring the more, the greater energy consume. Therefore, ensuring the network throughput and reducing the frequency of monitoring is of great significance for extending the network life cycle.

5 Conclusion

Aiming at data missing caused by data congestion in WSNs, the paper proposes a pipeline based multi-path reliable routing protocol. The protocol establishes the conversion relationship in multiple paths to reduce data missing and improve the reliability of data transmission. Combining multi-path routing with pipeline based on the conversion relationship to improve network throughput. Proposed predict method based on Markov chains for the lack of real-time monitoring and periodic monitoring at the destination node, predict the appointment time dynamically, and convert pipeline in time to ensure the network throughput. Experiments show that the proposed routing can handle data congestion effectively, improve the network throughput, and reduce the number of monitoring.

References

1. Cai, Z., Lin, G., Xue, G.: Improved approximation algorithms for the capacitated multicast routing problem. In: Wang, L. (ed.) COCOON 2005. LNCS, vol. 3595, pp. 136–145. Springer, Heidelberg (2005)
2. Cai, Z., Chen, Z., Lin, G.: A 3.4713-approximation algorithm for the capacitated multicast tree routing problem. *Theor. Comput. Sci.* Elsevier, Amsterdam (2009)
3. Cai, Z., Goebel, R., Lin, G.: Size-constrained tree partitioning: a story on approximating the multicast k-tree routing problem. *Theor. Comput. Sci.* **412**, 240–245 (2011). Elsevier, Amsterdam
4. Cheng, S., Cai, Z., Li, J.: Curve query processing in wireless sensor networks. *IEEE Trans. Veh. Technol.* **64**, 5198–5209 (2015). IEEE Press, New York

5. Gao, D., Yang, O., Zhang, H., Chao, H.C.: Multi-Path routing protocol with unavailable areas identification in wireless sensor networks. *J. Wireless Pers. Commun.* **60**, 443–462 (2011). Springer, Berlin
6. Lee, S.W., Choi, J.Y., Lim, K.W.: A reliable and hybrid multi-path routing protocol for multi-interface tactical ad hoc networks. In: *Military communications Conference*, pp. 2237–2242. IEEE Press, New York (2010)
7. Teo, J.Y., Ha, Y., Tham, C.K.: Interference-minimized multipath routing with congestion control in wireless sensor network for high-rate streaming. *IEEE Trans. Mobile Comput.* **7**, 1124–1137 (2008). IEEE Press, New York
8. Ren, F., He, T.: Traffic-Aware dynamic routing to alleviate congestion in wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **22**, 1585–1599 (2011). IEEE Press, New York
9. Popa, L., Raiciu, C.: Reducing congestion effects in wireless networks by multipath routing. In: *International Conference on Network Protocols*, pp. 96–105. IEEE Press, New York (2006)
10. Sridevi, S., Usha, M.: Priority based congestion control for heterogeneous traffic in multipath wireless sensor networks. In: *Computer Communication and Informatics*, pp. 1–5. IEEE Press, New York (2012)
11. Wang, C.G., Sohraby, K.: Priority-based congestion control in wireless sensor networks. In: *IEEE International Conference on Sensor Networks*, pp. 22–31. IEEE Press, New York (2006)
12. Pham, P.P., Perreau, S.: Performance analysis of reactive shortest path and multipath routing mechanism with load balance. In: *International Conference on Computer Communications*, pp. 251–259. IEEE Press, New York (2003)
13. Zhang, L., Li, J.B.: A path-transfer based multi-path reliable routing in wireless sensor networks. *J. Integr. Plant Biol.* **58**, 171–175 (2011). Wiley, New York
14. Wang, K.: *Advanced Computer Architecture*(The front page). Tsinghua University, Beijing (1995)
15. Ross, S.M.: *Applied Stochastic Processes*(The ninth edition). Posts Telecom press, Beijing (2007)