http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (21): 6020-6030, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.6020.6030

A New Online Adaptive Segmentation Algorithm in Delay Sensitive Wireless Sensor Networks

[1]Feng Xiao Tang, [1]Ming Zhao, [1]LiPing Liu and [2]Cheng Gao
[1]School of Software, Central South University, Changsha 410083, China
[2]College of Computer Science, Zhejiang University, Hangzhou 310027,China

Abstract: Because the packet delay and energy consumption of wireless sensor network node are effectively reduced by the data flow compression, the data flow compression technique can prolong the network lifetime and is suitable for use in delay sensitive applications. If the time consumption of the compression algorithm is big, then the data flow compression can not be valuable. How to balance time consumption and reduce delay is a big problem. To solve this problem, this paper proposed an adaptive judgment algorithm based on Packet segmentation and compression. In this algorithm, the source data packet was broken into shares and sent to the adjacent nodes separately, the compression strategy are chosen adaptively based on the local information of each individual sensor node. Our extensive experimental results show: this algorithm is highly effective on the reduction of transmission delay.

Ke y words: Wireless sensor networks, data compression, data gathering, packet delay, delay sensitive, packet segmentation, adaptive algorithm

INTRODUCTION

The wireless sensor network is widely used in the military, aviation, explosion-proof, disaster relief, environment, medicine, health care, household, industry, commerce and other fields (Estrin *et al*., 1999; Noury *et al*., 2000). In some fields, the packet delay is a focus besides energy consumption, This kind of wireless sensor network is named as delay sensitive wireless sensor networks (Felemban *et al*., 2006). In delay sensitive wireless sensor networks, the packet delay is the most important parameter (Kim *et al*., 2008; Bisnik and Abouzeid, 2009). Therefore, given a fixed energy consumption ,how to be more effective on the delay reduction and how to improve the performance in the network transmission are of great importance (Zhu *et al*., 2008).

The recent studies on delay reduction in delay sensitive wireless sensor networks focused on the routing algorithm optimizing and the increase in data aggregation efficiency (He *et al*., 2003, 2004). However, D. Xi proposed an Online Adaptive Compression (Adaptive Compression in Delay Sensitive Wireless Sensor Networks) (2012). method. In this method, the local information is analyzed and calculated based on a queuing model. Whether the compression is valuable is judged automatically by the Queue information at each node, so that the network packet delay can be effectively reduced by data compression.

The impact of compression quality to the delay is controlled by the compression states which are automatically adaptive by comparing the results of compression reduced transmission delay and compression increased processing delay of each node, where the date packages production rate and local information are changed at time.

However, the packet generation rate in network is not well-distributed, there is a large number of free nodes at some places when others are busy, this uneven distribution causes the network uneven load which forms delay regional congestion. Thus, this paper proposes a new Adaptive Compression algorithm based on the Packet segmentation technology. The source data are split up to pieces by the Packet segmentation controlling (Theory and Practice) (Stinson *et al*., 1984). By the disjoint identical-hop routing, the data pieces are distributed to the adjacent free nodes and are compressed at those nodes. This new method can effectively reduce the compression processing time and finally reach the purpose of reducing packet delay.

By the analysis of the theory and experiment result, the compression critical value can be reduced by this new algorithm effectively, the packet delay and network load are also reduced.

The contents of paper are as follows: Section 2 reviews related research works, focusing on the Algorithm Strategy of online adaptive compression algorithm, the verification of algorithm optimizing the date delay also

Corresponding Author: Ming Zhao, School of Software, Central South University, Changsha 410083, China

arise . Section 3 introduces the system mode. Section 4 introduces the adaptive compression algorithm based on segmentation technology, including the decision of segmentation quantity, selecting of the adjacent free nodes, transmission routing protocol. We prove and demonstrate the advance of the Adaptive Compression algorithm based on the Packet segmentation by theory and experiment date analysis separately in section 5 and 6. The conclusions are given in section 7.

RELATED WORKS

In recent studies of wireless sensor, compression algorithm is widely used, the algorithm of this study is based on the the existing basis, therefore, I will introduce some relevant compression algorithm at first, the improved adaptive compression algorithm will be explained in detail after that.

In a large number of wireless sensor network researches, compression is often used to reduce the network node energy consumption and improve the network lifetime.

Akyildiz *et al*. (2004) the earliest person who put forward several possible framework in wireless sensor networks, first proved the feasibility of the compression by Studying of the correlation between time and space in WSN (Akyildiz *et al*., 2004). After that, Scaglione and Servetto have studied the source data coding and routing algorithm for data compression effects (Scaglione and Servetto, 2005).Then (Baek *et al*., 2004) have studied the influence of the distributed data collection and data compression in hierarchical network (2004) besides that, they proposed an encoding strategy which according on relevant data and distributed data compression, this strategy have no more need of accurate node information about each other (Pradhan *et al*., 2002). However, these studies did not put forward the strategies of the substance or excessive dependence on the global information network and failed to improve the beneficial efficiency of data compression in the network.

In order to solve these problems and get explicit solutions, a large number of algorithms have been put forward. Such as data aggregation method. According to the research of the statistics (Sharaf *et al*., 2003; Yoon and Shahabi, 2004),the approximate data aggregation strategy based on multiple regression method (Banerjee *et al*., 2005), Approximate line segments dividing method (Schoellhammer *et al*., 2004) and so on. However, these methods still exist some problems such as data reconstruction incomplete and poor accuracy. Therefore, after a period of time,many algorithms such as Coding by Ordering (Petrovic *et al*., 2003), PINCO (Arici *et al*., 2003), LZW (Sadler and Martonosi, 2006), KOM (Reinhardt *et al*., 2009) were successively put forward to solve these problems.

But these algorithms mainly aimed at compression algorithms itself or designed according to the network energy consumption, not algorithms for delay sensitive designed wireless sensor network which can effectively reduce the network packet delay .

In order to solve the sorts of problems of traditional compression algorithm, in (Xi and Yang, 2012), the researcher designed an online adaptive compression algorithm specifically for network packet delay. This algorithm utilizes only local information of individual sensor nodes to make online decisions so that the node perform compression only when compression can reduce the overall delay. because of above judgment, the network along with the change of network environment make changes in execution strategy, after that, it can achieve the purpose of reducing the network delay, eventually improving the real-time performance of the network transmission.

The researcher of research by Xi and Yang (2012) puts forward an adaptive algorithm to make online decisions by local information such that compression is only performed when it can benefit the overall performance.

In fact, packet generation rate distribution density in the areas of network is not identical. In some areas, packets density is relatively small, nodes in such areas may still have lots of spare time. This will results in many problems such as uneven pressure distribution, delay regional congestion and energy hole formation based on energy-unbalanced consumption, incomplete network delay optimization and so on (Chen and Zhao, 2005, 2008). Therefore, this paper presents a new adaptive compression algorithms based on packet segmentation. Through segmentation control of data, the algorithm breaks packet into M shares. By the disjoint identical-hop routing, the data pieces are sent to the adjacent free nodes and compressed at those nodes. This new method can exploit the benefit of compression while avoiding the potential hazard of compression and finally reduce packet delay. Next in section3, we will first analyze network system model and do the problem description. In section4, we will detail describe the adaptive compression algorithms based on packet segmentation and show how can it solved those above problems.

SYSTEM MODEL AND PROBLEM STATEMENT

In the above adaptive data compression algorithm, the node according to the local network information and related parameters to independent judge compression state, it adopt a queuing model to approximate the data processing and transmission of nodes (Xi and Yang, 2012). This algorithm also adopt this original queue model and improved, it contains network model and the Medium Access Control (MAC) layer model which will be next described respectively.

For the convenience of the description of the model, we list the (Table 1), has convenient query the meaning of each symbol represents.

Network model:

- The topology of the network is a circle, with the SINK the center and the R the radius, N nodes randomly distributed within this circle, the initial density of nodes deployed no longer move (Cheng and Bajcsy, 2004). After sensed data, all the nodes will transmit them to the SINK

Table 1: Notation

Symbols	Definitions
N	Number of nodes in the network
R	Radius of the circular
r	Node transmission radius
λ_g	Packet generation rate
λ_e	Arrival rate of external packet
λ^i	Arrival rate for nodes in level i
n_i	Number of nodes in level i
P_{kj}	Transition probability from node k to node j
T_{tran}	The minimum time of a successful transmission
l_p	Length of the data packet
l_{ctl}	Sum of length of all control packets in a transmission
n_{sus}	Number of suspensions in backoff stage
T_{mac}	MAC layer service time
c_{mac}	Coefficient of variance (COV) of the MAC layer service time
$\bar{T}$	Average packet waiting time of the queue
λ	Arrival rate of the queue
ρ	Utilization of the queue
c_A	COV of the inter-arrival time
c_B	COV of the service time
γ_c	Average compression ratio
T_p	Average compression processing time
c_p	COV of the processing time
p_c	Ratio of compressed packets in all packets
T_{com}	Average packet waiting time at the compression queue
$\Delta T_{mac}(i)$	Delay reduction in level i
ΔT_{min}	Lower bound of total delay reduction
p_{drop}	Average packet loss rata in transmission
p_{suc}	Packet delivery success rate
E_{mcu}	Data calculation energy consumption
E_{rw}	Data reading and writing energy consumption
E_{com}	Energy consumption of compression
ΔE_l	Reduced energy consumption of transmission by packet compression
ΔE	Saved energy consumption by compression
E_{elec}	Energy consumption of radiating circuit
ε_{fs}, ε_{amp}	Energy for the power amplification of the two kinds of models respectively
l_p	Bit number of data
d_l	Packet number a node needs to undertake
e_l	Energy consumption of one bit data delivery
ΔD_l	Reduced amount of transmission data by compression
e_{com}	Compression energy consumption of one bit data

- In the network node density is large enough, each node has the same transmission radius r and assumes that the network node density is so large that each node can find a adjacent node which is more closer to the SINK within it's transmission range r, assume that it`s neighbor nodes' geographic information is already known
- And in order to describe conveniently, we assume that, within the scope of the r, it can find a node which is just r range more closer to the SINK. So, the data packets deliver from nodes located within the scope between r*i to r*(i+1) to the SINK need i transmissions, those nodes are are considered as level i nodes.

Under the network environment (Fig. 1), if we assume that node transmission radius r = 1, the number of nodes in level i is n_i and the average number of nodes $E[n_i]$ in level i is:

$$E[n_i] = \frac{\pi i^2 - \pi(i-1)^2}{\pi R^2} N = \frac{(2i-1)N}{R^2} \tag{1}$$

Assuming that the arrival rate of the external packet of node is λ_σ (decided by node packets generated rate in our network). So the packet arrival rate of j node in level i is λ^i_j, when I<R, then:

$$\lambda^i_j = \lambda_g + \sum_{k=1}^{n_{i+1}} \lambda^{i+1}_k p_{kj} \tag{2}$$

where P_{kj} is the transition probability from node k at level i+1 to node j at level I.

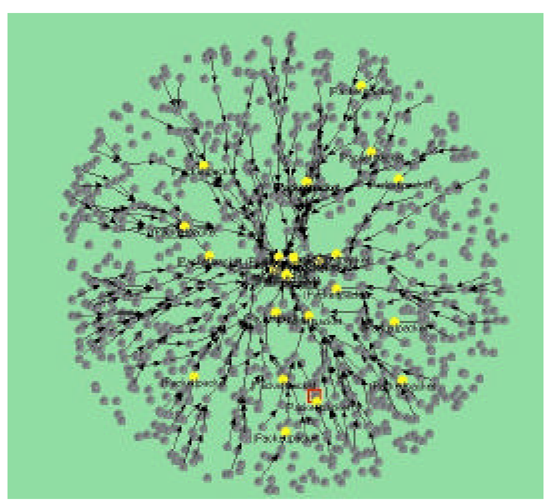

Fig. 1: Network topology

As the nodes of the same level have the same packet generation rate λ, we denote the average arrival rate for nodes in level i as λ^i, then taking expectation on both sides of the equation:

$$E[n_i]\lambda^i = E[n_i]\lambda_g + \lambda^{i+1}E[\sum_{j=1}^{n_i}\sum_{k=1}^{n_{i+1}} p_{kj}] \quad (3)$$

Because of $\sum_{j=1}^{n_i}\sum_{k=1}^{n_{i+1}} p_{kj} = n_{i+1}$,then Eq. 3 is equal to:

$$\lambda^i = \lambda_g + E[\frac{n_{i+1}}{n_i}]\lambda^{i+1} = \lambda_g + \frac{2i+1}{2i-1}\lambda^{i+1} \quad (4)$$

From it,we can obtain the mathematical model of packet arrival rate in each node and network model.

After the packet arrival rate of each nodes are obtained, in order to estimate the queue system efficiency we need to study the average service time of each packet, including packet compression processing time and MAC layer service time.

Packets of data compression processing time, we use LZW compression algorithm (Welch, 1984; Xi and Yang, 2012) is used to similar hardware conditions, the conditions here we employ the STM32F103R8T6 MCU (microcontroller). 32-bit MCU (32-Bit Ultra-Low-Power MCU) 128 k flash memory, 72 MHZ operating frequency, 20 k SRAM.

MAC model: IEEE 802.11 (Hameed *et al*., 2009) used as our MAC layer protocol, it's Distributed Coordination Function(DCF) employs a backoff mechanism to avoid potential contentions for the wireless channel.

In our MAC layer, we use a queue to simulate this kind of compensation Function, in this DCF, we also adopt the RTS/CTS mechanism to reduce transmission collisions. Thus, it requires at least four transmissions to successfully transmit a data packet: the transmissions of RTS, CTS, data and ACK packets.

Let T_{tran} denote the minimum packet transmission time, B_W denote network bandwidth, then, we can have:

$$T_{tran} = \frac{l_{data} + l_{ctl}}{B_W} \quad (5)$$

where, $l_{ctl} = l_{RTS}+L = l_{CTS}+l_{hrd}+l_{ACK}$ and l_{hdr} is the header size of the data packet.

T_{col} denote the average time spent in transmission collisions, T_{sus} denote the average duration of the timer suspension in the backoff stage. Under the assumption of the constant packet length, we can approximately have $T_{sus} \approx T_{tran}$.

The overall MAC layer packet service time can then be calculated as:

$$\begin{aligned} T_{mac} &= n_{sus}T_{sus}+n_{col}T_{col}+T_{tran} \\ &= (n_{sus}+1)\, T_{tran}+n_{col}T_{col} \\ &\approx (n_{sus}+1)T_{tran} \end{aligned} \quad (6)$$

Here, a and b compared to the original packets are small enough, so for the convenience of calculation, can be approximately equal to the last step.

Queue model: In the adaptive compression algorithms, we will transfer the packet process and compression process as two queue model respectively. The departure process of nodes adopting IEEE 802.11 MAC protocol can be approximated as a Poisson process. Thus, we assume that the arrival process of each node is a Poisson process and each node is an M/G/1 (i.e., exponential inter-arrival time distribution/general service time distribution/single server) queue.

Since the compression process can be viewed as a shunt of data arrival process, which can also be regarded as a M/G/1 queue. So in the M/G/1 queue, according to the formula of Khinchin-Pollaczek, average packet queue number N can be obtained as follows:

$$\overline{N}_{M/G/1} = \rho + \frac{\rho^2}{1-\rho}\cdot\frac{1+c_B^2}{2} \quad (7)$$

where ρ is the utilization of the queue and c_b is the Coefficient of Variance (COV) of the service time.

By Little's Law, given the arrival rate|Ë, the average packet waiting time, which is the packet delay in the node, can be derived as:

$$\overline{T}_{M/G/1} = \frac{\overline{N}}{\lambda} = \frac{2\rho - \rho^2(1-c_B^2)}{2\lambda(1-\rho)} \quad (8)$$

To derive the average packet waiting time for the transmission queue, we use the diffusion approximation method (Bolch *et al.,* 1998), then, we can get the average waiting time $T_{GI/G/1}$:

$$\overline{T}_{GI/G/1} = \frac{\rho}{\lambda(1-\hat{\rho})} \quad (9)$$

$$\hat{\rho} = \exp(-\frac{2(1-\rho)}{c_A^2\cdot\rho + c_B^2}) \quad (10)$$

ADAPTIVE JUDGMENT ALGORITHM

In section2, we introduced the development course of compression algorithms in WSN network and introduces a kind of new effective adaptive compression algorithm to reduce transmission delay (ACS), but after research, we found that the algorithm has not enough shown its advantages in delay and energy consumption, through a certain improvement, we can also continue to improve the optimization effect of the algorithm .

In this section, we propose an adaptive judgment algorithm based on Packet segmentation and compression. It add a new services structure to MAC layer which is called segmentation Adaptive Compression Service (SACS).This algorithm adds segment parts and new judgment upon the ACS algorithm, so as to further reduce the delay and save energy.

In the following paragraphs, we will expounds this algorithm respectively from four aspects of the algorithm architecture, the queue model, the number of the subcontract decision and the choice of the adjacent nodes.

Algorithm architecture: ACS consists of four functional units: a controller, an segmentation splitter, an compressor, an information collector and a packet buffer, Compared with the ACS algorithm is more an segmentation splitter part. With SACS, the traffic between the MAC layer and the upper layer is now intervened by the controller in SACS. When the controller receives the data, whether the data is compressed is first determined, if already compressed, do not handle directly sent to the buffer zone waiting for transmission. If the data does not compress, according to the collected network and hardware information which from information collector to determine compression state. Compression state is divided into compression and NO-compression and if the node is running in no-compression state, the data is also not handled but sent directly to the buffer zone waiting for transmission; If the node is running in a compressed state, it will first judge whether the data is segmented, if it has been segmented, it will be sent into the buffer zone of compressor waiting for compression. And if the data has not been segmented, it should be send into the segmentation splitter for segment. In the segmentation splitter, according to the same information from information collector, the transmission path is calculated and the number of copies and adjacent node to receive the segmented packet have been obtained. Then, the packet is marked those information and send to MAC layer. on the other hand, all the arrival external packets, only record the arrival time by information collector and then sent to the network layer without delay. The specific process as shown in Fig. 2.

In the above process, the most important include three parts:

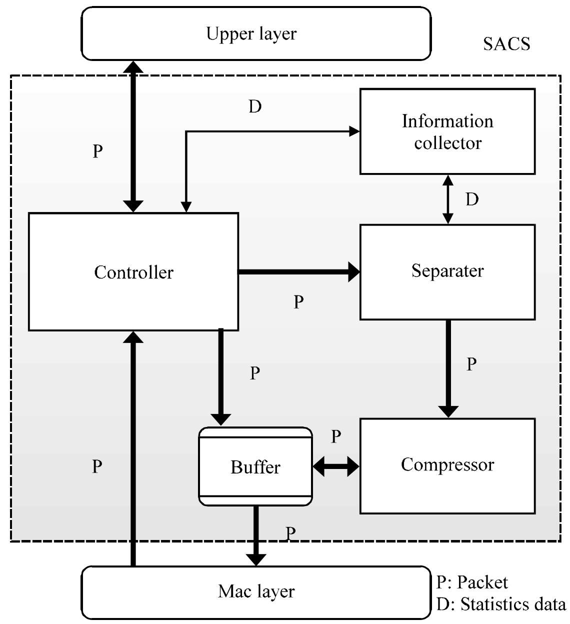

Fig. 2: Algorithm architecture

- Information collection: Packet compression state and segmentation conditions judgment need local information which collected by information collector, the local information includes network conditions and hardware conditions
- The segmentation strategy: Mainly including the idea of number selection of data packets segmentation and idea of the routing path selection
- Compression strategy: Through the analysis of all the information collected from the information collector, the node performing state: wether the compression state or no-compression state can be judged

Next, according to the system model above and the three parts above, the SACS algorithm will be in detail introduced.

Information collection: In the SACS, information collector is mainly responsible for collecting the following three types of data:

Compression statistics: Average compression ratio r_c, the average compression time T_p and COV (variance coefficient) of compression time c_P, Every time nodes perform compression, the compression ratio of compression, compression time, coefficient of variance were recorded. Each average statistics collection and calculation is updated after m times compression, in this experiment, we assume m = 100. Due to this statistic requires an initial value, we assume that all nodes in the network work in compression state to get initial value in the beginning.

Packet arrival rate λ: The arrival rate includes its neighbor nodes transfer data rate λ_e and the packet generation rate of the node itself λ_g. It`s obtaining way of λ is similar to the compression statistics, Arrival rate over a period of time will update for calculation, In this experiment, set T_{slot}. And at the end of each time period, additional statistic the p_c which represent the proportion of compressed packet and all received packets.

Service time at the Mac layer, Mac layer service time in the queue model is a random process, which is calculated through the packet arrival rate.

The segment decision: After the controller have got the network information collected by information collector. Once a packet arrives, the segmentation splitter determines whether the packet have been segmented firstly, if not, it segment the packet. Here we adopt the segmentation algorithm in the reference (Theory and Practice. CRC Press), (Stinson *et al*., 1984), Under the premise of the data security, segment number and the node which to receive segmented packets is decided by the controller.

- Firstly, the judgment of the segmentation number is principally, In order to ensure a certain packet delivery success rate p_{suc}, we are here to make a concept of packet loss ratio, Average packet loss rata in transmission is p_{drop}, in this experiment we set $p_{drop} = 1\%$. If we break the packet into n shares, so the probability of loss one share is $1-(1-p_{drop})^n$. In order to ensure the success rate p_{suc}, we should have:

$$1-(1-p_{drop})^n > p_{suc} \tag{11}$$

At the same time, because of the limitation of network intensity, nodes can only select their own peer node within the transmission range of one jump, so the segmentation number can`t be larger than the adjacent node number M.

So, the n can be obtained by:

$$\begin{cases} n < \log_{1-p_{drop}}(1-p_{suc}) \\ n \le M \\ n \in N^+ \end{cases} \tag{12}$$

According to the formula to calculate the largest value of n, According to the segmentation number to determine necessary adjacent node number, then select appropriate nodes to transmit.

- The selection of the intermediate nodes: we use "disjoint identical-hop routing": In the disjoint identical-hop routing phase, The M shares are transmitted to other sensor nodes that are dispersively distributed in the network with disjoint routes, where all the sensor nodes are more idle and along the same routing path have equal hops to the sink node after disjoint identical-hop routing, the nodes who received segmented packets, called intermediate nodes

Its associated transmission protocol is as follows: Each node know they hop distance to the sink, also know the neighbor node`s hop distance to the sink. Then the intermediate nodes nodes selection rule is as follows:

- Selected the node which is within the scope of one jump from the source node S
- Select the nodes of the same level
- Select the nodes of shortest queue length

Its implementation method is as follows: first of all, the candidate node requested by the RTS packet and replies CTS packet which includes it's queuing information. Data collector collected the CTS packets of adjacent nodes. By comparing the queuing information of each node, the controller selects the optimal node and names it as intermediate nodes.

Compression strategy: After finished the segmentation of packet, We should first get all the information by information collector, then summary and calculate it. Though the calculated value, Judging which state of node to perform-compression or no-compression.

Compression state judgment: After the the segmentation judgment have been done, We need to determine whether to compress received data packets. After a period of time T, the Controller will make a judgment according to the collected information and decides this node to perform which state during next time period.

Doing the judgment of compression state is to effectively reduce the packet delay. So, we will separately calculate the increase delay of compress implementation and reduce of data transmission delay by compression.

From the above introduced queuing model we can know the formula of increased and decreased delay:

Increased delay:

$$T_{com} = \frac{2T_p - \lambda_c T_p^2(1-c_p^2)}{2(1-\lambda_c T_p)} \tag{13}$$

Table 2:

```
Algorithm 1:
 1.Initialize node status.
 2.All nodes work in compression state with a period of time T.
 3.Judge node state:(compression/no-compression)
 Nodes in level i:
  if state = No-Compression then
  read statistics from the information collector
  compute Tcom and ΔT_min  if T_com≤ΔT_min then
  Set state to Compression
  Else
  Set i to the node`s level number
  Set ΔT_mac = 0
  While i>0
  Calculate λ
  Compute reduction ΔT_mac(i)
 ΔT_mac(i) = ΔT_mac(i)+λ^iΔT_mac(i)
  i = i-1
  End while
  If ΔT_mac(i)≤ ΔT_mac(i)  then
  Set state to Compression
  End if
  End if
 Else as the same as above.
Do segment
 Nodes in level i:
  if state=No-Compression then
 If Packet state = No-Compressed then
  read statistics from the information collector
  calculate n_max
  set n_dec = 0,
```

Decreased delay:

$$T'_{mac} = T_{mac} \cdot \frac{l_p / r_c + l_{ctl}}{l_p - p_c(l_p - l_p / r_c) + l_{ctl}} \tag{14}$$

$$\frac{n^k_{sus}}{n^i_{sus}} = \frac{\rho^k}{\rho^i} = \frac{\lambda^k}{\lambda^i} \cdot \frac{T^k_{mac}}{T^i_{mac}} = \frac{\lambda^k}{\lambda^i} \cdot \frac{(n^k_{sus}+1)T^k_{tran}}{(n^i_{sus}+1)T^i_{tran}} \tag{15}$$

Where:

$$n^k_{sus} = \frac{c \cdot n^i_{sus}}{n^i_{sus} - c \cdot n^i_{sus} + 1}, c = \frac{\lambda^k T^k_{tran}}{\lambda^i T^i_{tran}}$$

With n_{sus} and T_{mac}, the service time can be obtained and the normalized delay reduction can be calculated as:

$$\Delta T_{mac} = \sum_{j=1}^{i} \lambda^j \Delta T_{mac}(j) \tag{16}$$

Which can calculate the reduced total transmission time by compression. After having calculated the increase delay of compress implementation and reduction of packet transmission delay by compression, we can compare these two values to judge what state a node in the next period should be in.

The scheme: In the scheme, the first step is make sure whether the nodes are in compression process or not.

- If the nodes is not in compressing

At first, according to the local information collected by information collector, the node compares the relationship between T_{com} and T_{min}, the specific process is as follows (Table 2):

- If the nodes are in compression, we perform the analogical scheme of above, After that we can decide what state the node should to be in next step

THEORY ANALYSIS

According to the Queuing Theory and Eq. 7, 8, 14, we can get the main parameters ρ (Queue Utilization), C_B (COV of the queen)and λ (arrival rate in the M/G/1 queen)of the queue.

After employing the algorithm of the segmentation, the above three parameters are mainly changed, then we changed the parameters through the theoretical analysis and observe the change of the compression queen.

According to the theory of segmentation, Suppose that a packet is break into n shares (λ = nλ), compression time become:

$$1/n(T_{p1} = \frac{1}{n} T_p)$$

Coefficient of variation C_B is the ratio of the standard deviation T_p and average of service time,so it will not change by segmentation ($c_{p1} = c_p$). we can have:

$$\begin{cases} \rho_1 = \lambda_1 \cdot T_{p1} = n \cdot \lambda \cdot \frac{1}{n} \cdot T_p = \lambda \cdot T_p = \rho \\ \lambda_1 = n\lambda \end{cases} \tag{17}$$

Coefficient of variation C_p does not change, according to the formula in Eq. 14, the average wait time of compression queue T_{com} will become:

$$T_{com1} = \frac{2T_{p1} - \lambda_1 T^2_{p1}(1-c^2_{p1})}{2(1-\lambda_1 T_{p1})} = \frac{\frac{2}{n}T_p - \frac{1}{n}\lambda T^2_p(1-c^2_p)}{2(1-n\lambda\frac{1}{n}T_p)} = \frac{1}{n}T_{com} \tag{18}$$

It's meaning that the average queue waiting time become 1/n of before.

But the length of the queue (the packet number waiting in the queen) become $T_{com1}\lambda_{c1} = n\lambda_c.1/nT_{com}$, as the same as before. But after segmentation, the length of each packet is 1/n smaller than original packets and the Compression delay is 1/n smaller than before. Because of this, the delay of Compression reduced obviously.

The change of the transmission queue:

By the formula of Eq. 14 ,we get T'_{mac}. By the formula of Eq. 15,we get the relationship of parameter n_{sus} whin each nodes .After this, by the formula of Eq. 16, we can have the total reduction of packet transmission delay.

Then combining with Eq. 14-16,as the same as compression queue.

According to these formulas and when the network is stable, we can get delay influences: (1) Average compression ratio r_c, (2) MAC layer service time and (3) the parameter c in the formula.

When using segmentation and compression, compression ratio r_c and service time T_{mac} are not changed. The parameter:

$$c = \frac{\lambda^k T^k_{tran}}{\lambda^i T^i_{tran}}$$

because of segmentation, $\lambda_{g1} = n\lambda_g$, so ,with Eq. 13, we can get the Arrival rate of each nodes is $\lambda^{k1} = n\lambda^k$. In (5),

$$T^{k1}_{tran} = \frac{1}{n} T^k_{tran}$$

though those formulas,we can known formula $c_1 = c$ which is not change.

Above of all, it's show that the reduction of total transmission delay ΔT_{mac} after segmentation and compression is not change.

So under the packet generation rate of the same, to compare with original adaptive algorithm, the compression time of this segmentation algorithm is decrease, transmission time is not change ,and total delay is reduced.

At the same time, according to the judgment strategy of adaptive compression algorithms, the node is compared with T_{com} and ΔT_{mac} to determine whether to execute compression. So, because of the smaller T_{com} after segmentation and no-changed Δt_{mac}, the critical point becomes smaller when the two values turn equal. With the same packet generation rate, the nodes will have so more possibility to be compressed that total delay is significantly reduced.

After the theoretical analysis of delay optimization, we will research the optimization of node energy consumption and network lifetime by employing this algorithm.

Based on the theoretical conclusion, then we will verify the degree of delay optimization by experiment by applying our algorithm .

EXPERIMENTAL RESULTS

After used the adaptive compression algorithm of segmentation, we break the packets which are only compressed in one node into shares which should be sent to multiple nodes for compression. It greatly enhances the efficiency of compression queue. Then, we will use the experiment to verify the performance improvement by employing the algorithm of the network.

Here we use two parameters namely delay and energy for reference to compare the algorithm improved performance of the system respectively. Experiment using omnet++ 4.0 simulator, the relevant network parameters reference shown in section 3.

We compare the the effect of SACS algorithm (D-S-MA), no-compression algorithm(NO-MA), full-compression algorithm (MA) and adaptive compression algorithm (S-MA) respectively. Set up the network environment as shown in the section3, nodes of the network distributed like a circular with 7 levels, the transmission range of the node is 1.5 times of the distance between neighboring nodes, original packet length is 512B, other parameters are the same as before.

Packet generation rate growth from the original point 0.5 to the point 6, it totally recorded of 12 times.

We set the every simulation Lasts 1000 seconds and the interval collection time of information T is 10 sec. The experimental result every time T recorded once, it total taken of 100 times and averaged.

Figure 3 show the result of the experiment, from it we can see that the overall compression efficiency is quite close to that of adaptive compression algorithm and it is obviously improved than that by no-compression or whole compression algorithms.

Compared with adaptive compression algorithms, when the packet generation rate is less than a certain critical point S_1 and larger than S_2, the delay between this and adaptive compression algorithm (S-MA) are basically the same. But when the packet generation rate is between S_1 and S_2, the delay will be significantly reduced, the maximum delay reduced by 48% when the generation rate is equal to 2.2.

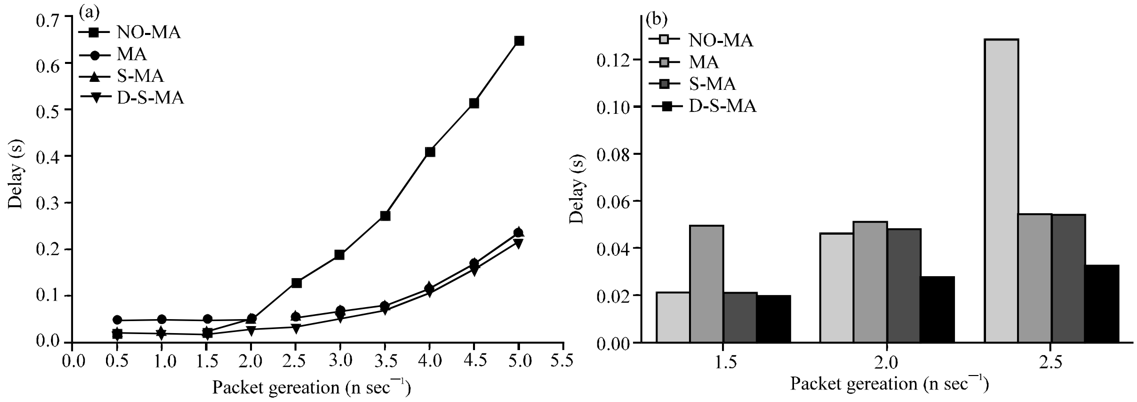

Fig. 3: The change of network delay along with the packet generation rate

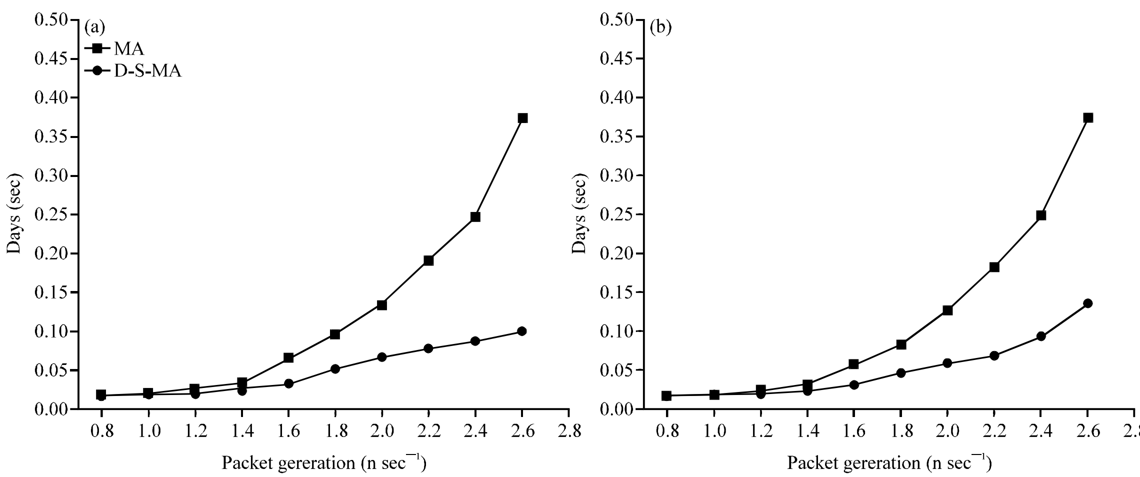

Fig. 4: The change of data delay in Level1 and level 7

Because network nodes` execution state of each level is different, we respectively make the statistics of nodes in different level (Fig. 4), here we only compares the differences of this and that of no-compression state, the result show that the trend in each level is as same as the trend of the whole network.

CONCLUSION

In this section, we propose an Adaptive Compression of routing scheme named Adaptive Algorithm Based On Packet Segmentation And Compression.

Adaptive algorithm based on Packet segmentation and Compression have a greater performance than the traditional algorithm from both sides of data real-time ability and network lifetime. This algorithm relies on local information of nodes to make mathematical models. Through analysis of the mathematical model, comparison of the relations between increased processing delay by compression and reduced transmission delay by compression, we employ a queue model to simulate the situation of packets in line. By this, the source data packet is broken into M shares and sent to the adjacent nodes separately, the compression strategy were chosen adaptively based on the local information of individual sensor nodes.

Our analysis and experiment show that our algorithm is less delay than the other existing scheme.

ACKNOWLEDGMENTS

This research work was supported in part by the Hunan Provincial Natural Science Foundation of China (13JJ3007) and National Natural Science Foundation of China (61379110).

REFERENCES

Akyildiz, I.F., M.C. Vuran and O.B. Akan, 2004. On exploiting spatial and temporal correlation in wireless sensor networks. Proceedings of the Modelling and Optimization in Mobile, Ad Hoc and Wireless Networks, March 24-26, 2004, Cambridge, UK.

Arici, T., B. Gedik, Y. Altunbasak and L. Liu, 2003. PINCO: A pipelined in-network compression scheme for data collection in wireless sensor networks. Proceedings of the 12th International Conference Computer Communications and Networks, October 22, 2003, Annapolis, MD., USA.

Baek, S.J., G. de Veciana and X. Su, 2004. Minimizing energy consumption in large-scale sensor networks through distributed data compression and hierarchical aggregation. IEEE J. Selected Areas Comm., 22: 1130-1140.

Banerjee, T., K. Chowdhury and D.P. Agrawal, 2005. Tree based data aggregation in sensor networks using polynomial regression. Proceedings of the International Conference Information Fusion, July 25-28, 2005, Cincinnati Univ., OH., USA.

Bisnik, N. and A.A. Abouzeid, 2009. Queuing network models for delay analysis of multihop wireless ad hoc networks. Ad Hoc Networks, 7: 79-97.

Bolch, G., S. Greiner, H. de Meer and K.S. Trivedi, 1998. Queuing Networks and Markov Chains. John Wiley and Sons, New York.

Chen, Y. and Q. Zhao, 2005. On the lifetime of wireless sensor networks. IEEE Commun. Lett., 9: 976-978.

Chen, Y., P.N. Chuah and P.Z. Jan, 2008. Network configuration for optimal utilization efficiency of wireless sensor networks. Ad Hoc Networks, 6: 92-107.

Cheng, T.E. and R. Bajcsy, 2004. Congestion control and fairness for many-to-one routing in sensor networks. Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, November 3-5, 2004, Baltimore, MD., USA., pp: 148-161.

Estrin, D., R. Govindan, J. Heidemann and S. Kumar, 1999. Next century challenges: Scalable coordinate in sensor network. Proceedings of the 5th ACM/IEEE International Conference on Mobile Computing and Networking, August 15-19, 1999, Seattle, WA., USA., pp: 263-270.

Felemban, E., C.G. Lee and E. Ekici, 2006. MMSPEED: Multipath multi-SPEED protocol for Qos guarantee of reliability and timeliness in wireless sensor networks. IEEE Trans. Mobile Comput., 5: 738-754.

Hameed, S.A., E.M. Shaaban, H.M. Faheem and M.S. Ghoniemy, 2009. Mobility-aware MAC protocol for delay-sensitivewireless sensor networks. Proceedings of the International Conference on Ultra Modern Telecommunications and Workshops, October 12-14, 2009, St. Petersburg, pp: 1-8.

He, T., B. Blum, Q. Cao, J. Stankovic, S. Son and T. Abdelzaher, 2003. SPEED: A stateless protocol for real-time communication in sensor networks. Proceedings of the IEEE Conference on Distributed Computing Systems, May 2003, Providence, RI., USA., pp: 46-55.

He, T., B.M. Blum, J.A. Stankovic and T. Abdelzaher, 2004. AIDA: Adaptive application independent data aggregation in wireless sensor networks. ACM Trans. Embedded Comput. Syst., 3: 426-457.

Kim, J., X. Lin, N.B. Shroff and P. Sinha, 2008. On maximizing the lifetime of delay-sensitive wireless sensor networks with anycast. Proceedings of the 27th Conference on Computer Communications, April 13-18, 2008, Phoenix, Arizona.

Noury, N., T. Herve, V. Rialle, G. Virone and E. Mercier *et al*., 2000. Monitoring behavior in home using a smart fall sensor. Proceedings of the IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine and Biology, October 12-14, 2000, Lyon, pp: 607-610.

Petrovic, D., R.C. Shah, K. Ramchandran and J. Rabaey, 2003. Data funneling: Routing with aggregation and compression for wireless sensor networks. Proceedings of the IEEE 1st International Workshop on Sensor Network Protocols and Applications, May 11, 2003, Berkeley, CA, USA., pp: 156-162.

Pradhan, S.S., J. Kusuma and K. Ramchandran, 2002. Distributed compression in a dense microsensor network. IEEE Signal Process. Magazine, 19: 51-60.

Reinhardt, A., M. Hollick and R. Steinmetz, 2009. Stream-oriented lossless packet compression in wireless sensor networks. Proceedings of the IEEE 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, June 22-26, 2009, Rome, pp: 1-9.

Sadler, C.M. and M. Martonosi, 2006. Data compression algorithms for energy-constrained devices in delay tolerant networks. Proceedings of the 4th International Conference Embedded Networked Sensor Systems, November 1-3, 2006, Boulder, Colorado, USA.

Caglione, A. and S. Servetto, 2005. On the interdependence of routing and data compression in multi-hop sensor networks. Wireless Networks, 11: 149-160.

Schoellhammer, T., B. Greenstein, B. Osterwiel, M. Wimbrow and D. Estrin, 2004. Lightweight temporal compression of microclimate datasets wireless sensor networks. Proceedings of the IEEE 29th Annual International Conference Local Computer Networks, November 16-18, 2004, Los Angeles, CA., USA., pp: 516-524.

Sharaf, M.A., J. Beaver, A. Labrinidis and P.K. Chrysanthis, 2003. TiNA: A scheme for temporal coherency-aware in-network aggregation. Proceedings of the 3rd ACM International Workshop on Data Engineering for Wireless and Mobile Access, September 19, 2003, San Diego, California, USA.

Stinson, D.R., 1984. Cryptography, Theory and Practice. CRC Press, New York.

Welch, T.A., 1984. A technique for high-performance data compression. Computer, 17: 8-19.

Xi, D. and Y.Y. Yang, 2012. Adaptive compression in delay sensitive wireless sensor networks. IEEE Trans. Comput., 61: 1429-1442.

Yoon, S. and C. Shahabi, 2004. An energy conserving clustered aggregation technique leveraging spatial correlation. Proceedings of the Sensor and Ad-Hoc Communications and Networks, October 4-7, 2004, Santa Clara, CA.

Zhu, S., W. Wang and C.V. Ravishankar, 2008. PERT: A new power-efficient real-time packet delivery scheme for sensor networks. Int. J. Sensor Networks, 3: 237-251.