International Journal of Computational Intelligence Systems, Vol. 4, No. 5 (September, 2011), 910-920

An Adaptive Token Passing Algorithm Applicable to MS/TP Network

Ping REN
School of Information Engineering, Wuhan University of Technology
Wuhan, Hubei, China

School of Engineering, Cardiff University
Cardiff, Wales, UK

Zude ZHOU, Quan LIU
School of Information Engineering, Wuhan University of Technology
Wuhan, Hubei, China
E-mail: rp-2003@163.com

Received 12 October 2010
Accepted 15 June 2011

Abstract

In the MS/TP network, token is passed according to the address of master station. A novel token passing algorithm was proposed based on the characteristics of MS/TP network frame distribution and the requirements of real-time performance. The algorithm adopts distributed control method, which can adjust the token passing order dynamically according to the frame distribution of every master station in the MS/TP network, change the number of frames a master station can send when holding token adaptively and the network bandwidth every master station occupied to reduce the token passing times on the logical ring. OPNET simulation results show that the algorithm has reduced the average service delay, decreased the frame lost, enhanced the network real-time performance and increased the network throughput. It is better than the order token passing algorithm, and has some inspiration for researching other token ring networks.

Keywords: Token passing algorithm, MS/TP network, BACnet standard, Logical token ring.

1. Introduction

BACnet is an open building automation control network data communication standard, which is developed especially for building automation and control system. MS/TP (Master-Slave/Token-Passing) protocol is widely used in BACnet products as an option of the BACnet datalink layer due to its real-time performance, stability and reliability [1, 2].

Token is passed according to the address of master station in the MS/TP network. Light stations and heavy stations occupy the same network bandwidth. In order to guarantee the heavy stations are able to send all frames, token has to be passed for nothing between the light stations, consequently increased the average service delay. Furthermore, the residual bandwidth of light stations can not be used by other heavy stations and result in a waste of bandwidth.

In this paper, a novel token passing algorithm was proposed. A token passing queue is constructed at each maser station to learn the frame distribution of mater stations in MS/TP network, which dynamically adjusts

Published by Atlantis Press

the token passing order on the logical ring, adaptively changes the number of frames a master station can send when holding token and is compatible with the original MS/TP protocol. In this way, the times of token passing for nothing will be reduced to the utmost extent.

This paper is organized as follows: Section 2 introduces the background of frame distribution characteristic of MS/TP network. Besides, the necessity of the proposed algorithm is briefly explained in this section by example network. While a new adaptive token passing algorithm is proposed, and its framework, definition and description are illustrated in section 3. Section 4 is used to evaluate the performance of the new algorithm by OPNET simulation, mathematical model and experiment results. Finally, Section 5 presents discussion, concluding remarks and future work.

2. Background of the Proposed Algorithm

2.1. *Frame Distribution Characteristics*

- Uneven: The length and number of frames every master station need to send are different as the different function every master station completed and the particularity of themselves as fieldbus stations. In addition, the frame sending cycle and interval are also varying from one to another. So the traffic load of each master node is different and the bandwidth it occupied is different too. Consequently, it is very important to change the number of frames a master node can send when holding token.
- Predictable: The master station usually sends frames to monitoring or routing stations periodically, that is to say the master station in the MS/TP network does not need to send frames at all times. The number of frames every master station generated in unit time obeys the Poisson distribution with parameter λ ,

$$P(x=k)=\frac{e^{-\lambda}\lambda^{k}}{k!}(\lambda=np) \quad (1)$$

So the probability of one master station does not have frames to send in two token passing cycle is,

$$P(x_2=0\,|\,x_1=0)=\frac{P(x_1=0,x_2=0)}{P(x_1=0)}=e^{-\lambda} \quad (2)$$

If one master station has frames to send in the later token passing cycle under the condition that it does not have frames to send in the former token passing cycle, the probability is,

$$P(x_2=k\,|\,x_1=0)=\frac{P(x_1=0,x_2=k)}{P(x_1=0)}=\frac{e^{-\lambda}\lambda^{k}}{k!} \quad (3)$$

The ratio of these two probabilities is,

$$\frac{P(x_2=0\,|\,x_1=0)}{P(x_2=k\,|\,x_1=0)}=\frac{k!}{\lambda^{k}}\gg 1(\lambda<1,K>1) \quad (4)$$

From the calculation results can be seen that, if one master station does not have frames to send in the former token passing cycle, then the probability of the master station has frames to send in the later token passing cycle is quite small.

- Adaptive: Due to the uneven and predictable characteristics of MS/TP network frame distribution, not every master node need to send data frame at the same time and the bandwidth every master node occupied is different, so the times of token passing for nothing can be greatly reduced by adjusting the token passing order and the value of $N_{max\_info\_frame}$ according to the proportion of master stations having frames to send in total master stations.

2.2. *Necessity of the Proposed Algorithm*

In the MS/TP network, the order token passing method is simple, but it is easy to result in token passing for nothing when the frame distribution is out of balance, thus will greatly increase the average service delay, decrease the network throughput and reduce the real-time performance.

For example, there are ten master stations in a MS/TP network. At one moment, master station N_9、N_7、N_5、N_3、N_1 has *nine, seven, five, three* and *one BACnet_Data_Not_ Expecting_Reply* frames need to send respectively. But master station N_8、N_6、N_4、N_2、N_0 does not have frames need to send. By comparing the network performance under these two algorithms, whether it is necessary or not to propose a new algorithm to replace the original order token passing algorithm will be apparent. The parameters used in the MS/TP network are shown in Tab. 1.

TABLE I. PARAMETERS USED IN THE MS/TP NETWORK

Name	Mean	Value
L_r	Request message length	*23Bits*
B	Baud rate	*19200bits/s*

T_t	The minimum time after the end of the stop bit of the final octet of a received frame before a station may enable its EIA-485 driver	*20bits time*
T_f	The maximum idle time a sending station may allow to elapse between octets of a frame the station is transmitting	*20bits time*
L_t	Token frame length	*8bits*

N_9 holds token at beginning and we ignore the time of frame processing and that *EIA-485* driver needed to send frames. The impact of different token passing method on network delay is shown as follows:

- When adopting order token passing method, the value of $N_{max\_info\_frame}$ is 1, and then the token passing sequence is $N_9 - N_8 - N_7 - N_6 - N_5 - N_4 - N_3 - N_2 - N_1 - N_0 - N_9$, the token has to be passed for 10 circles on the logical token ring. The time of $N_1 \sim N_9$ needed to send all frames is T_0,

$$T_0 = (25L_r + 100T_t + 100L_t) / B \quad (5)$$

  T_0 = 781.25*ms*;
  Where, the time used for data frame transmission is 239.58ms, so the time efficiency is 30.67%;
  W_0 is the time of the last data frame has to wait in the sending queue of N_9 before being sent out,

$$W_0 = T_0 - L_r / B \quad (6)$$

  W_0 = 771.67*ms*;
  In this case, the token has been passed for 100 times, and the amount of token frame is 800 *bits*, while the total data amount is 5400 *bits*, so the load efficiency in this network is 85%.
- When adopting adaptive token passing method, and the value of $N_{max\_info\_frame}$ changed with the frame distribution dynamically.
  The first token passing sequence is $N_9 - N_8 - N_7 - N_6 - N_5 - N_4 - N_3 - N_2 - N_1 - N_0 - N_9$, at this time, $N_{max\_info\_frame}$ =1;
  The second token passing sequence is $N_9 - N_7 - N_5 - N_3 - N_9$, at this time, $N_{max\_info\_frame}$ =2;
  The third token passing sequence is $N_9 - N_7 - N_5 - N_9$, at this time, $N_{max\_info\_frame}$ =3;
  The forth token passing sequence is $N_9 - N_7 - N_9$, at this time, $N_{max\_info\_frame}$ =4;
  The time of $N_1 \sim N_9$ needed to send all frames is T_1,

$$T_1 = (11T_f + 25L_r + 19T_t + 19L_t) / B \quad (7)$$

  T_1 = 353.96*ms*.
  Where, the time used for data frame transmission is 239.58ms, so the time efficiency is 67.69%;
  W_1 is the time of the last data frame has to wait in the sending queue of N_9 before being sent out,

$$W_1 = T_1 - L_r / B \quad (8)$$

  W_1 = 344.38*ms*;
  In this case, the token has been passed for 19 times, and the amount of token frame is 152 *bits*, while the total data amount is 4752 *bits*, so the load efficiency in this network is 97%.

The performance improvement is shown from seven aspects as follows,

TABLE II. PERFORMANCE IMPROVEMENT

Evaluate Parameter	Order Token Passing	Adaptive Token Passing	Performance Improvement
Network delay	781.25 *ms*	353.96 *ms*	54.69%
Time efficiency	30.67%	67.69%	37.02%
Load efficiency	85%	97%	12%
Maximum delay	771.67*ms*	344.38*ms*	55.37%
Average waiting time	388.23*ms*	174.59*ms*	55.03%
Average queue length	95	61	35.79%
Token passing times	100	19	81%

It can be seen from Tab. 2 that the network performance has been improved greatly when adopting adaptive token passing algorithm, compared with order token passing algorithm. When the frame distribution of master stations is out of balance, token has to be passed for nothing many times to ensure $N_1 \sim N_9$ can send all frames. This will not only increase the network delay, but also decrease the network throughput and reduce the network real-time performance. Therefore, dynamically adjust the token passing order and the value of $N_{max\_info\_frame}$ according to the frame distribution of MS/TP network is very necessary and feasible.

3. Adaptive Token Passing Algorithm

3.1. *Algorithm Definition*

- An element in the token passing queue is described by a three tuple Q_i (M_i, N_i, S_i, $M_i \neq$ TS, $0 \leq i < N - 1$). Among them, M_i is the *MAC* address of master station on the logical token ring; N_i is the number of frames a master station can send when holding token; N is the number of master stations in the MS/TP network; S_i is a flag, which is used to show that the master station M_i whether or not has frames to send in a single token passing cycle,

$$S_i = \begin{cases} 1, N_i \neq 0 \\ 0, N_i = 0 \end{cases} \quad (9)$$

- The token passing queue of master station M_i in the MS/TP network is indicated by Q (Q_1, Q_2……Q_i, $0 \leqslant i < N - 1$) [6-9]. Q_i is in descending order according to N_i but in ascending order according to M_i when N_i is same. It will not only help to ensure these master stations having more frames needed to send can get token in time, but also make for the token can be passed in accordance with the address of master station orderly when the network frame distribution is balanced;
- The frame distribution of master station in the MS/TP network is $D_{message}$, which is the rate of master stations having frames to send in all master stations,

$$D_{message} = \frac{\sum_{i=0}^{i=N} S_i(N_i \neq 0)}{N} \quad (10)$$

- Master station M_i adjusts the number of frames adaptively when holding token according to the frame distribution $D_{message}$. Its maximum threshold and minimum threshold are D_{Max} and D_{Min}. Their values can be adjusted according to the frame distribution of specific network artificially. By default, $D_{Max} = 0.5$, $D_{Min} = 0.2$;
- In the MS/TP network, the number of frames a master station M_i can send when holding token is $N_{max\_info\_frames}$, the value of which has a segmented function relationship with the network frame distribution as follows,

$$N_{max\_info\_frames} = \begin{cases} N_{max\_info\_frames} + 1, 0 < D_{message} \leq D_{min} \\ N_{max\_info\_frames} + 1/N_{max\_info\_frames}, D_{min} < D_{message} \leq D_{max} \\ 1, D_{max} < D_{message} \leq 1 \end{cases} \quad (11)$$

- In the MS/TP network, when passing token, the maximum times of one master station can pass token to the same but not physically nearby master station is $N_{max\_pass\_times}$. Definition of this parameter is used to prevent token from being passed between parts of master stations to ensure the network real-time performance to the utmost extent. By default, $N_{max\_pass\_times} = 1$, and the value of $N_{max\_pass\_times}$ can be adjusted according to the maximum delay time of MS/TP network;
- MS/TP frames have four different kinds of priorities, according to the descending order, they are P_0, P_1, P_2 and P_4. Their respective proportion in MS/TP network is C_0, C_1, C_2 and C_3;
- Suppose that the MS/TP network frames obey the Poisson distribution with parameter λ, then the different priority frames obey the Poisson distribution with parameter λ_i,

$$\lambda_i = \lambda . C_i \quad (12)$$

- Suppose that the data frames reach in each node is independent and follow Poisson distribution with parameter λ; the time each node needed to send data frame is independent and fellows the same negative exponential distribution with parameter μ; each node is equivalent to service desk and the maximum transmission delay of data frame is K; all the MS/TP network frames will be sent out according to the priority. Such MS/TP node can be modelled as M/M/1/K/∞/PS queuing model, and the performance can be expressed by formula (13)[10-12],

$$\begin{cases} p_0 = \dfrac{1-\rho}{1-\rho^{k+1}} \\ \sum_{n=0}^{\infty} P_n = 1 \\ p_n = \dfrac{1-\rho}{1-\rho^{k+1}} \rho^n \end{cases} \quad (13)$$

In which, $\rho = \dfrac{\lambda}{\mu}$

L is the average number of frames in the network waiting to be sent,

$$L = \frac{\rho}{1-\rho} - \frac{(k+1)\rho^{k+1}}{1-\rho^{k+1}} \quad (14)$$

L_q is the average number of frames in the queue waiting to be sent,

$$L_q = L - (1 - P_0) \quad (15)$$

$W_{12\ldots N}$ is the average time of frames (from priority 1 to N) spent in the network from generation to be sent out,

$$W_{12\ldots N} = \frac{1}{\mu - \sum_{i=1}^{N} \lambda_i} \quad (16)$$

W_N is the average time of one frame spent in the network from generation to be sent out,

$$W_N = \frac{\sum_{i=1}^{N} \lambda_i}{\lambda_N} W_{12\ldots N} - \frac{\sum_{i=1}^{N} \lambda_i W_i}{\lambda_N} \qquad (17)$$

W_{qN} is the average time of one frame spent in the queue waiting to be sent,

$$W_{qN} = W_N - \frac{1}{\mu} \qquad (18)$$

P_k is the probability of MS/TP network running at full capacity,

$$P_k = 1 - \frac{\mu(1-P_0)}{\lambda} \qquad (19)$$

These parameters are important criterions used to evaluate the network performance.

3.2. *Algorithm Description*

When the token holder station *TS* has sent all frames, or sent frames for $N_{max\_info\_frames}$ times, or received response frame within the set time, or waited for response timeout, the token must be passed to the next station *NS* on the logical token ring [3-4]. The proposed token passing algorithm is shown in Fig. 1.

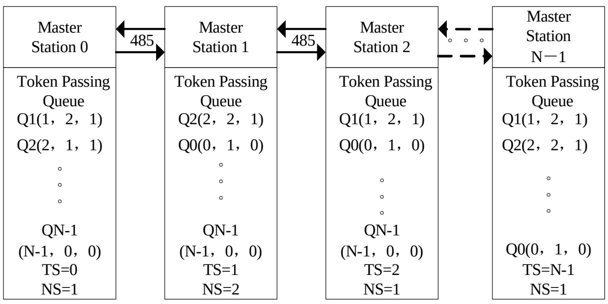

Fig.1 Block diagram of token passing arithmetic.

The concrete steps of this token passing algorithm are described as follows:

- Every master station monitoring every data frame and token frame in the MS/TP network, by analyzing the source address of data frame and the destination address of token frame, and calculates every tuple value of Q_i in the token passing queue. Among them, M_i is the destination address of token frame. If the number of data frames M_i send when holding token is greater than 0 then set $S_i = 1$, or set $S_i = 0$. N_i is equal to the number of data frames M_i can send when holding token;
- In a single token passing cycle, we construct the token passing queue *Q* (Q_1, Q_2……Q_i) for every maser station M_i. Element of the token passing queue *Q* is in descending order by N_i. If N_i is same, *Q* is in ascending order by N_i. It is needed to rearrange the element of *Q* when received token frame every time;
- When *TS* holding token, calculate the value of $D_{message}$ (*The frame distribution of master station*) and the value of $N_{max\_info\_frames}$ according to $D_{message,}$ maximum threshold and minimum threshold. The value of $N_{max\_info\_frames}$ has a segmented function relationship with $D_{message}$, the smaller value of $D_{message}$, the faster $N_{max\_info\_frames}$ growing. After *TS* using token done, then set *NS* to the value of M_1 in element Q_1 of the token passing queue *Q;*
- *TS* pass token to *NS*. *TS* begin to monitor transmission medium when having passed token. When *TS* received events or bytes from the MS/TP network in $T_{usage\_timeout}$ indicate that token has been received by *NS* correctly and being in use, then reset $N_i = 0$；
- If the times of *TS* pass token to the same *NS* continuously over $N_{max\_pass\_times}$ times, then set $N_i = 0$ and reorder the token passing queue *Q*；
- If *TS* does not receive N_{min_octets} number of received events and bytes in $T_{usage\_timeout}$ indicate that the token has not been passed successfully. After retry N_{retry_token} times, if still unsuccessful, which means that the master station *NS* failed or left the MS/TP network, then delete the master station *NS* from the logical token ring and token passing queue, and reset the number of master stations in the token ring $N = N - 1$;
- *S*et *PS* to （（*NS* + 1）modulo（$N_{max\_info\_frames}$ + 1））, and send "*Poll_For_Master_station*" frame to the master station *PS*. If *PS* can response "*Reply_To_Poll_For_Master*" frame correctly, then the master station *PS* is the next logical token passing station. At this time, set *NS* =（*NS* + 1）, and insert master station （*NS* + 1） into the token passing queue *Q*, then set $N = N + 1$，M_N = NS + 1，$N_N = 0$，$S_N = 0$，and pass token to the new station *NS*;
- If the master station *PS* can not response "*Reply_To_Poll_For_Master*" frame correctly in the previous step indicate that the master station *PS* can not be used as the next logical token passing station. Repeat the previous step until find the next logical token passing station, and pass token to the next logical station [5];

- If during the process of finding next logical station in the pervious step, the next logical station may still be the original master station *TS*. At this time, the MS/TP network turns into a master-slave network having a single master station. If *TokenCount* is smaller than N_{poll}, the token will no longer be passed and used continuously. If *TokenCount* is greater than N_{poll}, then carry out polling for master station to determine whether or not has new station joins in the network [5].

4. OPNET Simulations and Performance Analysis

4.1. *Establishment of Simulation Environment*

OPNET is a large-scale communication and computer network simulation software developed by OPNET Technology Company in United States. Modeler provides a wide range of editors to help users to accomplish network modeling and simulation [15-18]. The establishment of MS/TP network simulation environment includes the following modeling components:

- Network Modeling: It is responsible for interconnecting the master stations to a network according to the network topology. MS/TP is a logical token ring network and the network topology is token ring. There are ten master stations in the simulation network, as shown in Fig. 2.

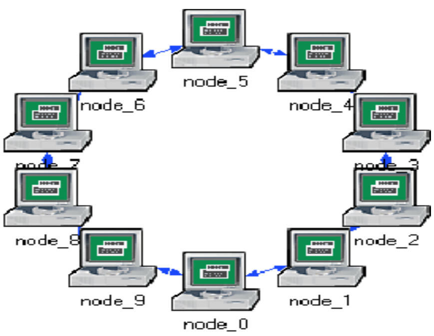

Fig.2 Topology of network.

- Station Modeling: Establish the station modeling which is composed by the appropriate protocol model. Use the node editor to describe the hierarchy of the MS/TP master station, and realize the MS/TP network architecture through the data flow between every function modules.
- Process Modeling: Simulate the action of a single object and use the finite state machine for modeling. Including the MS/TP protocol receive frame state machine and the master station state machine, and write any C/C++ code within each state as well as the library functions designed especially for the MS/TP protocol.

The three-layer model is fully corresponded with the actual MS/TP protocol, which is established by the OPNET three-layer modeling mechanism. Construct the corresponding frame format with the MS/TP protocol using the packet editor.

4.2. *Performance Evaluation Methods*

4.2.1. *Mathematical Model*

Suppose that the data frame reach in each node is independent and follows Poisson distribution with parameter λ; the time each node needed to send data frame is independent and fellows the same negative exponential distribution with parameter μ; each node can be equivalent to service desk and the maximum transmission delay of data frame is *K*. Such networks can be modelled as M/M/S/K queuing model, and the performance can be expressed by formula (12)[13-14]:

The average numbers of frame in the queue wait to be sent is L_q,

$$L_q = \begin{cases} \dfrac{P_0(\lambda/\mu)^S \rho}{S!(1-\rho)^2}[1-\rho^{(K-S)} - (K-S)\rho^{(K-S)}(1-\rho)], \rho \neq 1 \\ \dfrac{P_0(\lambda/\mu)^S (K-S)(K-S+1)}{2S!}, \rho = 1 \end{cases} \quad (20)$$

$$\begin{cases} P_0 = \begin{cases} [1+\sum_{n=1}^{S}\dfrac{(\lambda/\mu)^n}{n!}+\dfrac{(\lambda/\mu)^S}{S!}\sum_{n=S+1}^{K}(\dfrac{\lambda}{S\mu})^{n-S}]^{-1}, & \rho\neq 1 \\ [1+\sum_{n=1}^{S}\dfrac{(\lambda/\mu)^n}{n!}+\dfrac{(\lambda/\mu)^S}{S!}(K-S)]^{-1}, & \rho=1 \end{cases} \\ \sum_{n=0}^{\infty} P_n = 1 \\ P_n = \begin{cases} 0, n>K \\ \dfrac{(\lambda/\mu)^n}{n!}P_0, n=0,1,2,...,S \\ \dfrac{(\lambda/\mu)^n}{S!S^{n-S}}P_0, n=S,S+1,...,K \end{cases} \end{cases} \quad (21)$$

In which, $\rho = \dfrac{\lambda}{S\mu}$

L is the average number of frames in the network waiting to be sent,

$$L = L_q + S + P_0 \sum_{n=0}^{S-1} \frac{(n-S)(\lambda/\mu)^n}{n!} \qquad (22)$$

W is the average time of one frame spent in the network from generation to being sent out,

$$w = \frac{L}{\lambda \sum_{n=0}^{K-1} P_n} \qquad (23)$$

W_q is the average time of one frame spent in the queue waiting to be sent,

$$w_q = \frac{L_q}{\lambda \sum_{n=0}^{K-1} P_n} \qquad (24)$$

P_w is the probability of network running at full capacity,

$$P_w = 1 - \sum_{n=0}^{S-1} P_n \qquad (25)$$

These parameters are important criterions used to evaluate the network performance.

4.2.2. *Experimental Model*

In the MS/TP network, the performance of adaptive token passing algorithm is evaluated from five aspects: average service delay, frame lost, network throughput, token access delay and time efficiency.

The average service delay is the time from the request frame arrived at the send queue of client to the response frame arrived at the receive queue of client sending from server, which is reflected by the statistic of OPNET: Node Statistics/Token Ring/Delay (s). The frame lost refers to the frame lost number. Network throughput refers to the amount of data received and sent by the MS/TP network in unit time, which is calculated by two statistics: Node Statistics/Token Ring/Load (bits/s) and Node Statistics/Token Ring/Traffic Received (bits/s). Token access delay is the average time one node needs to get token. Time efficiency shows the proportion of time used for token passing in total time.

In this paper, by comparing the above five parameters in the same network load condition to determine that the token passing algorithm whether or not has an improvement on the network performance, assume that the network load is G,

$$G = \frac{1}{B} \sum_{i=1}^{N} \frac{L_i}{T_i} \qquad (26)$$

Among them, the value of G is between 0 and 1 and increases with the network load; L_i indicates the average length of frame generated by the master station M_i; B is the data transfer rate (bits/s); N shows the number of MS/TP master stations in the simulation model; The time interval of every master station generates frames is expressed by T_i.

4.2.3. *Simulation Results Analysis*

TABLE III. EXPERIMENTAL PARAMETERS

Name	Value	Name	Value
G	0.2~0.5	L_t	8 *bit time*
Simulation time	20 *min*	T_{frame_abort}	60 *bit time*
Node number	10	T_{reply_delay}	250 *ms*
Frame length	23 *Bits*	$T_{reply\_timeout}$	255 *ms*
B	19200 *bits/s*	$T_{usage\_timeout}$	20 *ms*
T_t	20 *bit time*	N_{min_octets}	4
T_f	20 *bit time*	T_{no_token}	500 *ms*

Use the simulation model to accomplish a total of five experiments, by analyzing the experimental data we can obtain the following results:

(1) Improvement on average service delay

First group of experiments shows the improvement of adaptive token passing algorithm on the average service delay, the experimental results shown in Fig. 3.

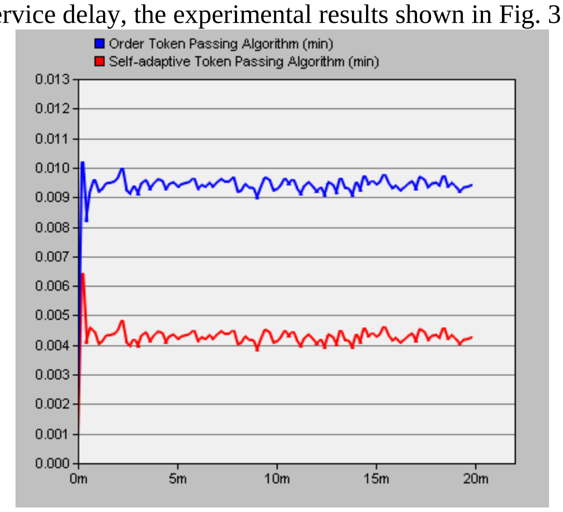

Fig.3 Improvement on average service delay.

From Fig. 3 can be seen that, the average service delay has reduced by about 52.6% when adopting adaptive token passing algorithm and it can be explained from two aspects: (1) By adjusting the token passing order, the times of token passing for nothing has been greatly reduced, as well as the token access time when having frames to send; (2) By adjusting the number of frames the master node can send when holding token, the time interval is greatly reduced between multiple data frames. So the time of frame has to wait in the sending queue before it can be sent out is less than before, and the performance on the average service time is improved.

In addition, the average service delay increased with the network load *G*. The greater value of network load *G*, the more frames wait to be sent, the longer time needed to send all frames. Under the same network load condition, the average service delay of the adaptive token passing algorithm is smaller than the order token passing algorithm evidently. When the value of *G* is between 0.2 and 0.5, the improvement of adaptive token passing algorithm on the average service delay is most obvious. This is because, when the value of *G* is small, the value of frame distribution becomes smaller correspondingly and the token only need to be passed between master stations having frames need to send. At the same time, the value of $N_{max\_pass\_times}$ increased gradually, the more frames the master stations need to send, the greater network bandwidth they occupied, thereby the times of token passing for nothing on the logical ring reduced effectively. When the network load *G* is greater than 0.5, the value of frame distribution increases correspondingly, then the value of $N_{max\_pass\_times}$ reset to 1 at this time and the network performance approaches the order token passing algorithm.

(2) Improvement on frame lost

Second group of experiments shows the improvement of this token passing algorithm on the frame lost, the experimental results shown in Fig. 4.

Because the MS/TP network is a control network, if the time of one frame waited in the sending queue exceed the maximum allowed delay time, and then this frame can be treated as lost. From Fig. 4 can be seen that, the times and amount of frame lost has been greatly reduced when adopting adaptive token passing algorithm, this is because: (1) The token is mainly passed between the nodes having frames need to send, so the probability of more frames can be sent in maximum delay time will be improved greatly; (2) when one master node holding token, the number of frames it can send is adjusted according to the frame distribution of the network, so the heave node can occupy more bandwidth than light node. As a result, the improvement on frame lost is obvious when adopting adaptive token passing algorithm especially under heave traffic load.

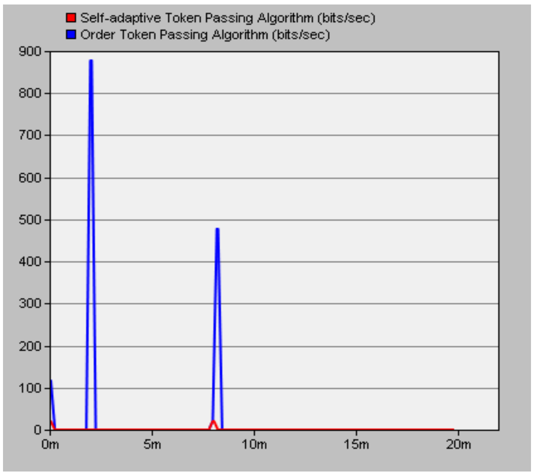

Fig.4 Improvement on frame lost.

In addition, the frame lost increases steadily with network load *G* when using order token passing algorithm. When using adaptive token passing algorithm, the frame lost speed up significantly when the value of *G is* greater than 0.5. This is because, when the value of *G is* less than 0.5, MS/TP network load is relatively small, the number of master stations need to send frames is also relatively small, while the adaptive algorithm can adjust the token passing order dynamically, change the value of $N_{max\_pass\_times}$ adaptively and enable more frames can be sent out within the specified time to reduce the frame lost. When the value of *G* is greater than 0.5, the token is passed orderly on the logical ring and the frame lost amount and times approach the order token passing algorithm gradually.

(3) Improvement on network throughput

The third group of experiments shows the improvement of adaptive token passing algorithm on the network throughput, the experimental results shown in Fig. 5.

From Fig. 5 can be seen that, the throughput has doubled when adopting adaptive token passing algorithm comparing with order token passing algorithm. The improvement is achieved for the flowing two reasons: (1) The token just be passed between the master nodes having frames need to send, so the time of token passing between the master nodes not having frames need to send has been greatly reduced especially when the frame distribution is small, and the heave master node has more opportunities to hold token and send frames; (2) master node can send more frames when holding token, thus reduced the token passing time and more frames will be sent in unit time.

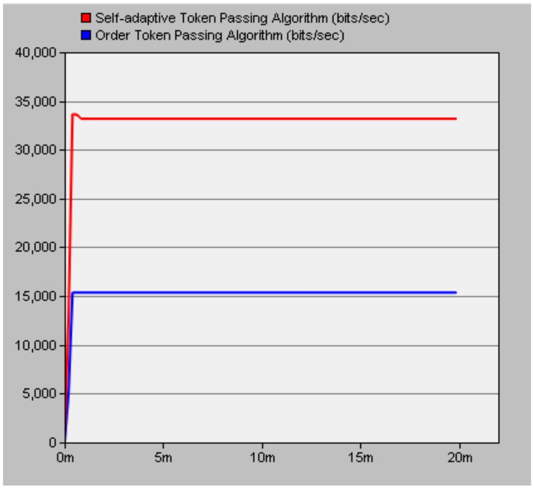

Fig.5 Improvement on network throughput.

In addition, the network throughput has a linear relationship with network load *G* when the value of *G* is lesser than 0.5. When the value of *G* is greater than 0.5, the growth rate of network throughput begin to slow down gradually. When using adaptive token passing algorithm, the network throughput slow down lesser than the speed of order token passing algorithm significantly. This is because, with the increase of network load *G*, frame loss rate is gradually increased, when the frame loss rate is greater than the speed of network load *G*, then the network throughput begins to reduce. As the improvement of adaptive algorithm on the frame loss rate is greater than the order token passing algorithm, so that the reduction speed of network throughput is slower than that of order token passing algorithm significantly when the use of adaptive algorithm.

(4) Improvement on token access delay

The fourth group of experiments shows the improvement of adaptive token passing algorithm on the token access delay, the experimental results shown in Fig. 6.

From Fig. 6 can be seen that, the token access delay is almost same under these two algorithms. When using the adaptive token passing algorithm, the volatility of token access delay is greater than that of order token passing algorithm, this is because that when using the order token passing algorithm, the token access delay has a relationship with whether or not the master node needs to send data and the data length to be sent, whereas the amount of data each time need to send is a certain value, so the token access delay volatile in a range and can be expected; when using adaptive token passing algorithm, the token access delay has a close relationship with frame distribution, so the token access delay is more volatile. Some nodes may need to send multiple frames, and resulting in token access delay will be greater than the order token passing algorithm at some times.

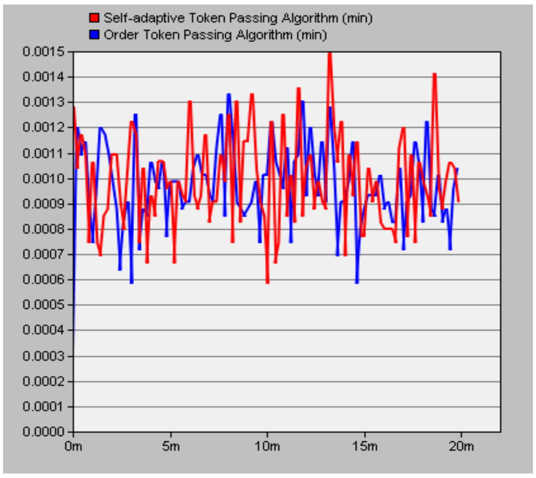

Fig.6 Improvement on token access delay.

In addition, adaptive token passing algorithm has a closer relationship with network load *G*. When *G* is less than 0.5, then there are only some of the nodes need to send frames, the token passing order and the number of frames a master node can send when holding token will be adjusted dynamically, and then the token access time is volatile according to the frame distribution. When *G* is greater than 0.5, and then the token access delay under adaptive token passing algorithm is almost equal to order token passing algorithm, because the value of $N_{max\_pass\_times}$ will be set to 1 at this time and the token will be passed orderly on the logic ring according to the address of master node.

(5) Improvement on time efficiency

The fifth group of experiments shows the improvement of adaptive token passing algorithm on the time efficiency, the experimental results shown in Fig. 7.

Time efficiency is the proportion of time used to send data frame in the total time. From Fig. 7 can be seen that, when using adaptive token passing algorithm, especially if the traffic load is less than 0.5, then the token passing times on the logic ring will be reduced greatly and the throughput will be increased remarkably, so the time efficiency has been improved for about 60%. Because the token passing order and the value of $N_{max\_pass\_times}$ will be adjusted dynamically according to the frame distribution, so the time efficiency is not as steady as order token passing algorithm. When the traffic load *G* is greater than 0.5, the time efficiency is almost equal to order token passing algorithm, because the value of $N_{max\_pass\_times}$ will be reset to 1 and the token will be passed orderly.

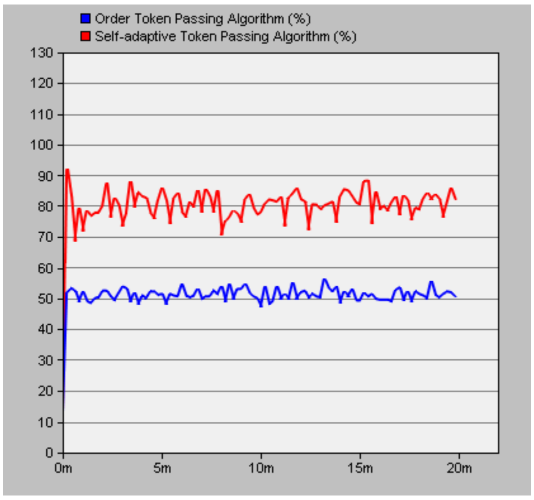

Fig.7 Improvement on time efficiency.

From analysis of the simulation results we can see that this token passing algorithm has an obvious improvement on network delay, frame lost, network throughput, token access delay and time efficiency especially when the value of network load G between 0.2 to 0.5. In addition, it can be seen from the simulation that by adjusting minimum threshold and maximum threshold of the network frame distribution can further improve the network performance, when the values of these two parameters are equal to the minimum and maximum value of network load, the network performance will achieve optimal value. Therefore, the algorithm plays a very important role to the performance improvement of MS/TP network.

5. Conclusions and Future Work

5.1. *Conclusions*

The characteristics of frame distribution in the MS/TP network provide the feasibility to propose an adaptive token passing algorithm, while the performance analysis of the example network shows the necessity of the proposed algorithm. As a control network, the fames have been divided into different priorities, the high priority frames will be sent earlier to guarantee reliability and real time performance. The mathematical models of single node and entire network provide the theory evidence when designing the input queue of master node, and present a method to calculate the performance of MS/TP network. Finally, the performance of the algorithm has been evaluated from five aspects by experiments.

The adaptive token passing algorithm in the MS/TP network is a novel token passing algorithm, which can adaptively change the token passing order on the logical ring, solve the problem of token can only be passed orderly in the original MS/TP network and make the token passing order become more rational. OPNET simulation results show that this algorithm has reduced the average service delay, decreased the frame loss rate, increased the network throughput, and enhanced the network real-time performance.

The proposed token passing algorithm is very suitable for the MS/TP network whose traffic load is less than 0.5, and at this time the improvement on network performance is most obvious. When the traffic load is greater than 0.5, the performance is almost same with the order token passing algorithm. The number of frames a master node can send when holding token just has relationship with frame distribution, if it can be adjusted with the length of sending queue, that is to say it has a functional relationship with parameter L_q, the performance will be better. In addition, the value of $N_{max\_pass\_times}$ may be greater than 1, so the master node will occupy more time to send data frame, thus result in other nodes have to wait for longer time, the real-time performance will be declined at some times.

5.2. *Future Work*

Due to the limitation of the proposed token passing algorithm, a possible extension could be the incorporation of average sending queue length to the existing framework that will increase the flexibility and broaden the application scope. Furthermore, how to dispose the high priority frame in the light node or the heave node will permit the algorithm to deal with emergency and meet the requirements of real-time performance. An algorithm that incorporates adaptive bandwidth allocation method based on traffic load and priority promotion mechanism under emergency circumstance is the future direction of our work.

Acknowledgements

This paper was supported by National Natural Science Foundation of China (Grant No.50775167). We hereby express our thanks to Hubei Key Laboratory of Broadband Wireless Communication and Sensor Networks for help in conducting the experiments.

References

1. ISO 16484-5-03, Building automation and control systems-part 5 Data communication protocol. Geneva: ISO, 2003.

2. Quan LIU, Ping REN. The Realization of MS/TP Module Based on uC/OS-Ⅱ[J]. Journal of Wuhan University of Technology, 2007, 29(11): 51-53.
3. Tae-Jin Park, Young-Chan Kwon, Seung-Ho Hong. Performance evaluation of BACnet MS/TP protocol using experimental model[C]. ICIT 2005, Hongkong, 2005.
4. W. S. Song, S. H. Hong, S. T. Bushby, "A Simulation Analysis of the BACnet LANs", NISTIR 7038, National Institute of Standards and Technology, October, 2003.
5. S. H. Hong, W. S. Song, Y. C. Kwon, T. J. Park, "Implementation of a Bandwidth Allocation Scheme in the MS/TP Protocol", IEEE/ISIE'05 Proceedings - Industrial Infomatics, June 2005, pp.1393 - 1398.
6. SUN Xian-pu, ZHANG Yan-ling. A novel algorithm of token transfer [J]. Journal of Xi’an University of Post and Telecommunications, 2005, 10(2): 122-125.
7. S. H. Hong, Y.C. Kim "Implementation of a Bandwidth Allocation Scheme in a Token-Passing Fieldbus Network", IEEE Trans. on Instrument ans Measurement, Vol.51, No 2, pp.246-251, 2002. 4.
8. SUN Xian-pu, ZHANG Yan-ling, SONG Bin. Research on Multiple Access Control Protocol with Dynamic Token Algorithm for MANET [J]. Acta Electronica Sinica, 2006, 34(1): 118-122.
9. S. Lee, K. C. Lee, M. H. Lee, and F. Harashima, "Integration of mobile vehicles for automated material handling using profibus and IEEE 802.11 networks", IEEE Trans. Ind. Electron., vol. 49, no. 3, pp. 2002.
10. A. Willig, "Polling-based MAC protocols for improving real-time performance in a wireless PROFIBUS", IEEE Trans. Ind. Electron., vol. 50, no. 4, pp. 2003.
11. S. Lee, S. H. Lee, K. C. Lee, M. H. Lee, and F. Harashima, "Intelligent performance management of networks for advanced manufacturing systems", IEEE Trans. Ind. Electron., vol. 48, no. 4, pp. 2001.
12. G. C. Walsh and Y. Hong, "Scheduling of networked control systems", IEEE Control Syst. Mag., vol. 21, no. 1, pp. 2001.
13. Y. Tipsuwan and M. Y. Chow, "Gain scheduler middleware: A methodology to enable existing controllers for networked control and teleoperation\— Part I: Networked control", IEEE Trans. Ind. Electron., vol. 51, no. 6, pp. 2004.
14. S. H. Hong and Y. C. Kim, "Implementation of a bandwidth allocation scheme in a token-passing fieldbus network", IEEE Trans. Instrum. Meas., vol. 51, no. 2, pp. 2002.
15. H. J. Li, and X. K. Lin, "An OPNET-based 3-tier network simulation architecture," IEEE International Symposium on Communications and Information Technology '05, pp. 793-796, October 2005.
16. Hasan M. S., Harding C., Yu H., Griffiths A (2005) "Modelling delay and packet drop in networked control systems using network simulator NS2". International Journal of Automation and Computing, vol 2. 2005.
17. CHEN Chang-xing, GAO Xiao-guang, ZHANG Jing-wei, ZHANG Jun. Network Simulation Based on OPNET [J]. Journal of Air Force Engineering University (Natural Science Edition), 2007, 8(2): 86-88.
18. Min CHEN. OPNET Network Simulation[M]. Beijing: Tsinghua University Press, 2004. 12-16.

Ping REN, who was born in Sichuan province, China, in September 1982. He earned bachelor and master degree of engineering from Wuhan University of Technology, Wuhan, China in 2005 and 2008 respectively, and now study in Wuhan university of Technology as a PHD candidate and in Cardiff University, UK as a Joint-PHD student. His major field of study is Wireless Industrial Network.

Zude ZHOU, who was born in Hunan province, China, in April 1946. He graduated from Huazhong University of Science and Technology, Wuhan, China in 1970. Professor ZHOU is the director of Hubei Key Laboratory of Digital Manufacturing. His major field of study is Mechantronics and Digital Manufacturing.

Quan LIU, who was born in Yunnan province, China, in September 1963. She graduated from Huazhong University of Science and Technology, Wuhan, China in 1985. Professor LIU is the Dean of School of Information Engineering, Wuhan University of Technology. Her major field of study is Signal Processing and Embedded Technology.