I.J. Wireless and Microwave Technologies, **2011, 2, 26-32**
Published Online April 2011 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijwmt.2011.02.04

Available online at http://www.mecs-press.net/ijwmt

Research of NCS Real-time Performance with Judging Priority Based on Generalized Stability

Hengrong LU

Marine College, Zhejiang Institute of Communications, Hangzhou, China

Abstract

The conceptions of generalized stability and priority judgment function of real-time node which guaranteed the generalized stability were put forward aiming at the problem of Networked Control System (NCS) real-time performance based on shared Ethernet. From the perspective of control theory and communication theory, generalized stability and priority judgment function were applied into Real-time ETHERnet (RETHER) by using real-time performance improving method of avoiding collision based on protocol in Ethernet communication protocol. Therefore, token was passed to the real-time node with highest priority and real-time data was sent, NCS real-time performance was improved. Finally, simulation results demonstrate the effectiveness of the proposed method.

Index Terms: Networked Control System, network stability, real-time Ethernet, real-time performance

1. Introduction

As the development of field bus and Ethernet technology, NCS has been one of the technology development hotspots in automation area. Comparing with traditional point to point control system, NCS has low cost and convenient installation and maintenance. Furthermore, it can realize resource sharing for remote operation and control [1]. Because of industrial Ethernet great success and many of advantages in information network, industrial Ethernet becomes expecting solution of Omni-directional seamless integration in enterprise network. However, Ethernet is designed for information network. There are many problems on its applications in NCS. Real-time problem led by uncertainty of Ethernet transportation delay is the main obstacle for development of Ethernet applied in field bus layer, which become a hotspot in control area [2].

From the perspective of network, solution was put forward mainly aiming at Ethernet topology and Media Access Control (MAC) protocol which made real-time request satisfied. Furthermore, uncertainty of transportation delay is reduced in NCS application. In improving network topology, uncertainty problem of Ethernet communication network is resolved to some extent and network collision is reduced greatly by the

* Corresponding author.
E-mail address: looger@163.com

appearance of switched Ethernet. In reference [3], the maximum network time delays of switched Ethernet with star topology and tree topology are calculated, and the real-time performance of Ethernet with these structures under different loads of data is analyzed. The results indicate the feasibility of applying Ethernet in real-time industrial control. In improving MAC protocol, Ethernet communication real-time performance is improved by using software strategy aiming at modifying Carrier Sense Multiple Access with Collision Detection (CSMA/CD) protocol. Currently, there are two methods: collision resolving and collision avoiding [2]. When there is no collision, CSMA/CD protocol is adopted; when there is collision, a kind of collision resolving method with certainty is adopted. Reference [4] proposed a new persistent based real-time Ethernet MAC protocol that provides a predictable upper bound for the delivery delay of real-time frames. Moreover, it is compatible with the protocol used by the existing Ethernet controllers for conventional datagram traffic. The paper described the protocol in detail and analyzed the maximum delivery delay for real-time traffic and the efficiency of the channel. Collision avoiding is that a software scheduling layer is added to Ethernet link layer. Collision is avoided by scheduling which sends real-time data and non-real-time data separately [2]. Reference [5] proposed and evaluated a software-based protocol called RETHER that provides real-time performance guarantees to multimedia application without modifying existing Ethernet hardware. Reference [6] introduced standards such as EPA and Power link based on integrated scheduling. Those solutions support scheduling with competing freely and certain scheduling with sending data separately. Network nodes communicate periodically.

However, the researchers did not take impact of plant stability on network node priority in to account. The conception of generalized stability and priority judgment function of real-time node which guaranteed the generalized stability were put forward to guarantee key data occupying channel first. Therefore, to meet plant stable circumstance real-time data was sent with high priority.

2. New Conceptions

A. Generalized Stability

There are two meanings for generalized stability: Lyapunov asymptotical stability and running safely of the whole control system after receiving key data. The conceptions will be illustrated as following. Asymptotical stability: if the equilibrium state of origin is Lyapunov stable, and perturbed movement $\varphi(t;x_0,t_0)$ converges to equilibrium state $x_e = 0$ when t tends to infinity, then equilibrium state of the system is asymptotically stable. In application, it is very necessary to determine the maximum extent of asymptotical stability, which can determine the maximum extent allowable of original perturb x_0 of perturbed movement under asymptotical stability. The function is as following:

$$F = f_{Lyapunov}(t,x)$$

The meaning of guaranteeing control system running safely is that controller sends command on time when receiving important warning data and key parameters. Therefore, control system will run safely and stably. This paper use mission emergent degree $F = f_{safe}(t_{deadline})$ of key data as variable, impact of key data on safety influence of control system $F = f_{safe}(t_{deadline})$ as function value. Therefore, this paper uses the following equation to illustrate the relationship:

$$F_{generalized-stable} = F(f_{Lyapunov}(t,x);\ f_{safe}(t_{deadline})) \tag{1}$$

Plant state is evaluated by integrating the two conceptions comprehensively and with combining node priority.

B. Priority Judging Function

When there are two or more real-time nodes to send data simultaneously, priority of each node is determined by real-time node priority judging function. The impact of variable on real-time node judging function contains generalized stability $F_{generalized-stable}$ and emergent degree of node itself $t_{deadline}$. The equation is showed as following:

$$F_{evaluation} = F(F_{generalized-stable};t_{deadline}) \tag{2}$$

The actual function value should be determined by actual condition of plant.

3. Analysis Of Improved Real-time Performance

C. The RETHER Protocol

The network operates in one of two modes - CSMA-mode and RETHER-mode. The network switches transparently to RETHER-mode when there are real-time (RT) sessions and back to CSMA-mode when all RT sessions terminate. The rationale behind this decision is to minimize the performance impacts on non-real-time (NRT) traffic when there are no RT sessions. Before describing the proposed protocol, we first outline the design constraints –

- It should be implementable on existing hardware, i.e., off-the-shelf Ethernet card.
- It should be implementable as an enhancement to existing communication software platforms.
- When there are no real-time sessions, performance should be at least as good as that of the current Ethernet.

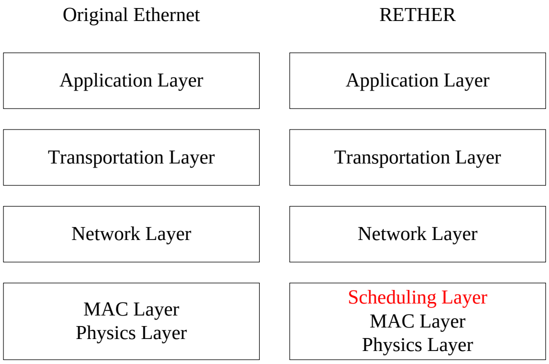

Fig. 1. Network Layers Comparison

In order to satisfy the first constraint and at the same time provide deterministic channel access, we modify the lowest-level software layers to implement a token passing scheme. Its network comparison is showed as figure 1.

The second constraint is satisfied because RETHER can be implemented as an enhancement to the UNIX network subsystem. The third constraint is satisfied because we adhere to the CSMA/CD protocol (CSMA-mode) when there are no RT sessions and use the token bus protocol (RETHER-mode) only when there are RT sessions.

1) CSMA-Mode

Nodes compete for the channel using the original CSMA/CD protocol of the Ethernet. This protocol is implemented in all the Ethernet adapters available in the market. It is fair; its performance is good under light loads and deteriorates only when the network is heavily loaded. In the RETHER protocol, all nodes remain in the CSMA-mode until an RT request arrives. The first RT request initiates to RETHER-mode.

2) Switch to RETHER-Mode

When a node receives an RT request from a local application process and it is in the CSMA-mode, it becomes an initiator by broadcasting a switch-to-RETHER message on the Ethernet. Every node that receives this message responds by setting its protocol mode to RETHER-mode. It then holds off sending any data and awaits completion of transmission of the packet already in the transmission buffer of the network interface. Then it sends an acknowledgement back to the initiator. As soon as the initiator receives all the acknowledgements, it creates a token and begins circulating it. This completes a successful switch to RETHER-mode.

Acknowledgements are crucial to the success of the protocol for two reasons. Firstly, acknowledgements signify the willingness of the nodes to switch to the RETHER-mode. Secondly, the fact that acknowledgements are successfully sent out indicates that the nodes do not have any pending packet in the back off phase of the CSMA/CD protocol. The latter is particularly important because in typical Ethernet cards, software has no control over the data once it has been transferred to the network interface buffer.

3) RETHER-Mode

We adopt a timed token-passing scheme to provide bandwidth guarantees. This scheme is used because it does not require global synchronization and imposes fewer restrictions on the operating environment. We make the following assumptions –

- Ethernet refers to a single Ethernet segment.
- Each RT request specifies the required transmission bandwidth in terms of the amount of data it needs to send during a fixed interval of time, called the Maximum Token Rotation Time (MTRT). The Maximum Token Holding Time (MTHT) for the node is determined from the amount of data it needs to transmit per MTRT.
- An average constant bandwidth is reserved for each session. Variations caused by compression are absorbed by sufficient buffer at each end.

At any time, there is only one RT request per node. This could be generalized to multiple RT requests per node by treating these requests as virtual nodes sharing one physical node.

4) Basic Algorithm

The control token circulates among two sets of nodes with different rotation frequencies – the real-time (RT) set and the non-real-time (NRT) set. Only nodes that have made a bandwidth reservation belong to the RT-set while all nodes belong to the NRT-set. When the token visits the nodes in one of these sets, the node can send the corresponding type of message. The MTHT for the nodes is given by –

$$MTHT_{RT} = \frac{Data\ per\ MTRT}{Eth.bandwidth} + S / w\,Overhead \tag{3}$$

$$MTHT_{NRT} = MTU\ transmission\ time \tag{4}$$

where MTU stands for Maximum Transmission Unit or the maximum packet size on the network. It is possible to tune to $MTHT_{NRT}$ get better performance.

In each MTRT, the token visits all the nodes in the RT-set in order. When a node in the RT-SET receives the token, it sends out one unit of RT data and passes the token to its neighbor in the RT-set. The last node in the RT-set passes the token to the NRT-set, tagging it with a $TimeToDeadline$ field where –

$$TimeToDeadline = MTRT - \sum_{i \in RTset} MTHT_i \tag{5}$$

This represents the residual bandwidth available for the NRT-set. When an NRT node receives the tagged token, it determines if there is the packet, updates the $TimeToDeadline$ field in the token and passes it on to its neighbor in the NRT-set. If not, it informs the last node in the RT-set that it should be the first NRT node to receive the token in the next MTRT. It then passes the token rotation cycle. Therefore, the token could visit the nodes in the RT-set more frequently than the ones in the NRT-set, thus giving RT traffic priority over NRT traffic.

Figure 2 shows a generic Ethernet segment.

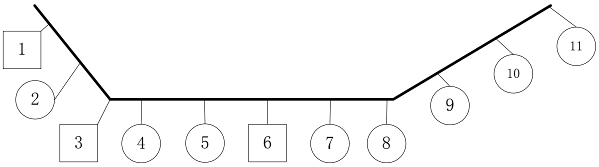

Fig. 2. Sample Network Configuration

Here, italicized nodes 1, 3 and 6 belong to the RT-set and nodes 1, 2, 3, … , 11 belong to the NRT-set. A possible sequence in two MTRTs would be –

$$\mathit{1},\mathit{3},\mathit{6},1,2,3,4,5,6,7,\mathit{1},\mathit{3},\mathit{6},7,8,9,10,11,1,2\ldots$$

5) Admission Control

Every new RT request goes through an admission control system which determines if the request can be satisfied without affecting the existing schedule. An important feature of the RETHER protocol is that admission control decisions are made “locally” at each node. A new connection with a reservation $MTHT_{new}$ can be admitted if and only if

$$\sum_{i\in RTset} MTHT_i + MTHT_{new} + TB_{NRT} \le MTRT \tag{6}$$

where TB_{NRT} is the Total Bandwidth reserved for the NRT-set to prevent starvation. This is also fixes an upper bound on the time between token-visits to an NRT node. TB_{NRT} can be tuned in order to preserve the timeout values associated with other operating system functions that depend on the network.

Admission decision is postponed until the node receives the token. In other words, a process’ request to initiate a real-time connection will not return until the associated node receives the token. The reason is that the token carries the most up-to-date information about the RT-set and bandwidth reservations. Alternatively one could maintain the current RT-set information at each node. However, this scheme may lead to incorrect admission decisions when two nodes receive RT requests simultaneously and both nodes admit the request without the knowledge of admission decisions made at other nodes. Given the number of nodes on an Ethernet and the number of RT nodes supportable with 10 Mb/sec, including the RT-set information in the 64-bytes token seems reasonable.

When an RT node wants to terminate its real-time connection, it merely removes itself from the RT-set information on the token.

6) Switch to CSMA-Mode

When the last RT session terminates, the corresponding node broadcasts a Switch-to-CSMA message. On receiving this message, all nodes switch back to the CSMA/CD protocol. The last RT node also destroys the token.

It is possible that the broadcast message is not received by some of the nodes. To address this problem, the RETHER protocol enforces a maximum interval between successive receipts of the token by an NRT node by adjusting in TB_{NRT} equation (5) appropriately. A node will automatically switch to CSMA-mode when this timer expires. Consequently, it is not necessary to acknowledge the Switch-to-CSMA message because all the nodes will eventually converge to the CSMA-mode.

4.Simulation

The simulation was programmed by C#.NET and run under Visual Studio 2008. Calculated data of simulation was imported into Excel 2007 to draw graphics. There are some rational assumptions for simulation –

- After sending real-time data, NCS switched to CSMA-mode directly.
- There is no time cost on calculating for each node.
- Real-time node gets ready with one MTU time cost.
- Real-time node frame is sent with one MTU time cost under RETHER-mode.
- Ignoring frames gap.
- Ignoring sending time of non-real-time node.

Simulation picked out 8 nodes: 3 real-time nodes and 5 non-real-time nodes. They competed for the network channel simultaneously. The simulation adopted 10M Ethernet net card, MTU cost $51.2\mu s$. The simulation was run 5 times under original CSMA/CD-mode. The result was 5 maximum time delay for each node. Simulation result was displayed as following:

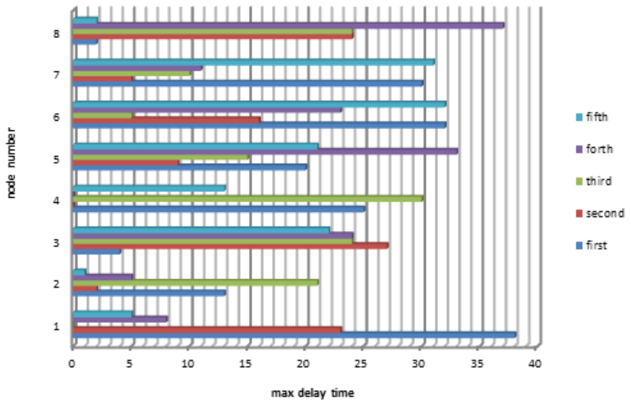

Fig. 3. CSMA/CD Mode

The simulation was run 5 times under RETHER-mode. The result was 5 maximum time delay for each node. The simulation result was showed as following:

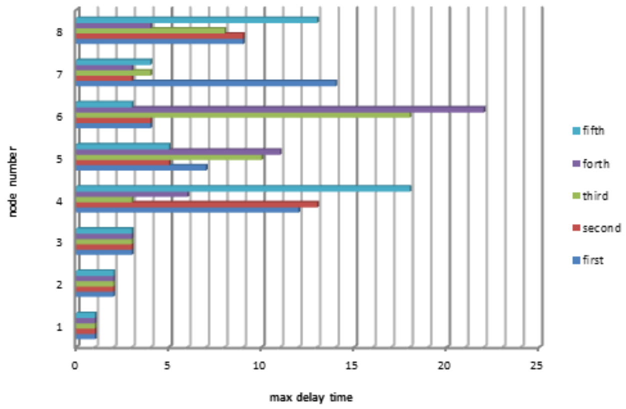

Fig. 4. RETHER Mode

From the simulation result, RETHER Ethernet based on generalized stability priority judging function was compared with traditional Ethernet, the former guaranteed that real-time data transported according to priority on time greatly. Average maximum time delay of real-time nodes and average maximum time delay of all nodes were reduced obviously. Real-time performance of the NCS was improved.

5. Conclusion

This paper based on shared Ethernet NCS put forward the conception of generalized stability and priority judgment function of real-time node which guaranteed the generalized stability according to uncertainty of traditional Ethernet i.e. real-time problem of transporting data. Real-time performance of NCS was improved by being importing RETHER protocol. From the perspective of communication theory and control theory the problem of real-time node priority was resolved. The effectiveness of the proposed method is demonstrated by simulation results.

References

[1] Q. Zong, “Class of networked control systems design under medium access constraints,” *System Engineering and Electronics*, Vol. 29, No. 5, pp. 778-781, May 2007.

[2] Y. Y. Chen, “Research and application of Ethernet in networked control system,” M. S. thesis, Power Electronic and Power Drive, Xihua Unv., Sichuan, China, 2007.

[3] J. Liu, W. Wei, “Study on maximum network delay of switched Ethernet with different network structure,” *Process Automation Instrumentation*, Vol. 27, No. 10, pp. 19-22, October 2006.

[4] G. Shen, X. M. Xu, X. He, “A new industrial Ethernet media access control protocol,” *Journal of Shanghai Jiaotong University*, Vol. 20, No. 4, April 2002.

[5] C. Venkatramani, T. Chiueh, “Supporting real-time traffic on Ethernet,” *Proc of Real-time System Symposium*, 1994.

[6] X. Q. Miao, “Exposition of six type of communication protocols of real-time ethernet,” *Process Automation Instrumentation*, Vol. 26, No. 4, pp. 1-6, April 2005.

[7] L. Chen, D. Q. Feng, J. X. Jin, J. Chu, “Study on real-time characteristics of Ethernet in industrial application” *Journal of Zhejiang University*, Vol.38, No. 6, pp. 670-675, 2004.