Reducing delay and jitter for real-time control communication in Ethernet

Mohamad Khairi Ishak*, Guido Herrmann*, Martin Pearson**
*Queens School of Engineering
University Walk, Bristol BS8 1TR, United Kingdom
**Bristol Robotics Laboratory, Bristol, United Kingdom
memki@bristol.ac.uk, g.herrmann@bristol.ac.uk, martin.pearson@brl.ac.uk

***Abstract*—The aim of this paper is to develop an approach for cheap and deterministic control communication using Ethernet. A half-duplex Ethernet network populated with a small/medium number of Media Access Controllers (MACs) is used for timed real-time communication. Data packages are sent at well defined times to avoid collisions. Collisions mainly occur due to jitter of the transmitter system, so that arbitration (similar to CANopen) is necessary. In this paper, simulation models using a Binary Exponential Backoff (BEB) scheme and a Linear Backoff scheme are developed. This paper analyzes and investigates how the backoff time affects the performance of the Carrier Sense Multiple Access protocol with Collision Detection (CSMA/CD) in a basic Media Access Controller (MAC), in terms of data arrival characteristics, i.e jitter and delay. We propose to assign different minimal back-off times for each of the CSMA/CD controller units to minimize packet collisions. Simulated tests show the advantage of our approach over a standard CSMA/CD setting.**

***Index Terms*—CSMA/CD, Ethernet, Binary Exponential Backoff, Linear Backoff, network model**

I. Introduction

Research in networking technology and control engineering are blending uniquely into a challenging area, the development of embedded networked real-time applications. Hence, the applications should timely deliver synchronized data-sets, minimize latency in their response and meet their performance target.

Today, an embedded communication network offers flexibility of the designed system and reduces wiring complexity which leads to use embedded networks for safety critical applications [1], [2]. Most of the applications like robotics, manufacturing, medical, military, and transportation systems, depend on embedded computer systems that interact with the real world [3], [4]. The many fields of applications come with different requirements on such embedded systems [5], [6]. Messages in distributed embedded systems can be sent at a certain time or when an event occurs, known as time-triggered and event-triggered communications.

Manuscript received June 9, 2012. This work was not supported by any organization.

M.K Ishak and G. Herrmann are with Faculty of Engineering, Department of Mechanical Engineering, University of Bristol, BS8 1TR, United Kingdom, e-mail: memki@bristol.ac.uk

M. Pearson is with the Bristol Robotics Laboratory, University of the West of England, Bristol, BS34 8QZ, United Kingdom, email: martin.pearson@brl.ac.uk

Ethernet has been the predominant networking technology for over 20 years. Ethernet is potentially the most practical network solution due to its expendability, robustness and self-configuration capability. It is nonproprietary in contrast to other real-time communication protocols. These are the keys that make it widely used commercially. Our interest is to use Ethernet for real-time control communication. The required key characteristic is deterministic processing in the system [7]. Real-time control networks must provide a guarantee of service and consistently operate deterministically and correctly [8], [9], [10].

Over the last few years, various protocols have been proposed based on a deterministic communication scheme. A number of network architectures solve the communication problems: protocols such as EtherCAT, ControlNet, Interbus, Time Triggered Ethernet (TTEthernet), CANopen, and CAN have been developed specifically for networking embedded real-time systems [11], [12]. EtherCat (Ethernet for Control Automation Technology) is an open real-time Ethernet network, based on standard Ethernet. EtherCat communication employs a master and slave approach where dedicated hardware is used for slave implementation. A frame is composed and periodically sent by the master and transferred to all slave units and finally sent back to the master. TTEthernet combines standard Ethernet network traffic and hard real-time communication for the same infrastructure. It integrates time-triggered and event triggered traffic into a single hardware infrastructure for distributed communication with mixed critical time requirements. TTEthernet relies on specific switches to organize the data communication and establish global synchronization. Custom switches to support the time triggered communication model are only available from TTTech [13].

Basic ideas of our work are lent from CAN. CAN is a deterministic network which is protocol optimized for short messages. The messages have different priority, and higher priority messages always gain access to the network [14], [15], [16]. Therefore, the transmission delay for higher priority messages can be guaranteed. But, compared with the other networks, CAN has a slow data rate of a maximum of 1Mb/s. Variants of CAN, Profibus and CANopen, for example, use a master-slave architecture in which one node controls all communication on the network which carries certain benefits and cost. A CANopen network uses, a synchronization signal,

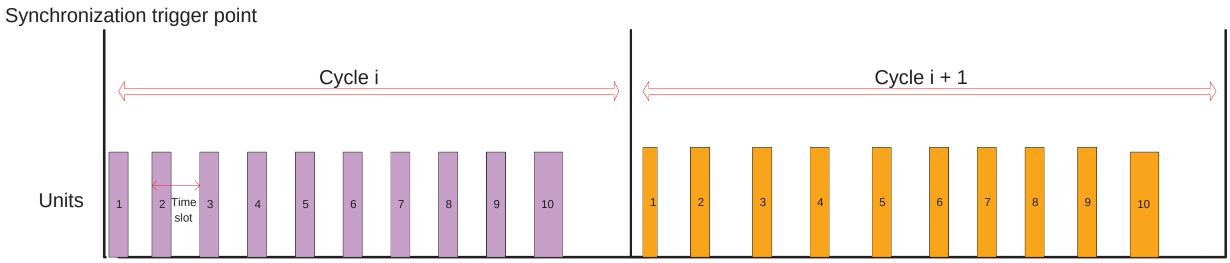

Fig. 1. *Time slot for the communication network approach*

the 'transmit signal', which triggers transmission of different nodes at the same time. The data frames need to be arbitrated. In CAN-networking, prioritisation via the CAN-address is used. As a result, the time-triggered architecture (TTA) offers fault-tolerant and deterministic communication services. For Ethernet, many modifications of CSMA/CD have been extensively studied considering software and hardware based solutions to enhance the operating system and application layer [17], [18], [19], [20], [21], [22], [23].

This paper aims to establish a cheap control solution for communication and deterministic real-time communication in Ethernet using CANopen similar network features. Thus, a synchronized signal causes the sending of short data packages from CSMA/CD controllers, which have a data package scheduled. The main feature of this strategy is that a specific minimal backoff time and a specific slot time for each CSMA/CD controller unit is used for minimal packet collisions. In particular, the assignment of different minimal backoff times to each CSMA/CD controller results in reduction of collisions, as it will be seen later. Thus, this paper uses the existing CSMA/CD protocol and investigates how the back-off interval affects the performance in a LAN (Local Area Network). We explore the benefits of the original CSMA/CD scheme for better performance without significant modification.

The rest of the paper is organized as follows. In Sections 2 and 3, the principles of the communication system and the simulation setup are described. Section 4 describes the simulation setup and Section 5 analyzes the CSMA/CD performance under different minimal back-off interval scenarios. Finally, Section 6 concludes this paper.

II. COMMUNICATION IN ETHERNET NETWORKS

This section revises the media access control (MAC) protocol, based on Carrier Sense Multiple Access/Collision Detection (CSMA/CD) and introduces important network characteristics.

A. *Ethernet and CSMA/CD*

The Binary Exponential Backoff (BEB) algorithm is used [24]. At a time, when a station wants to transmit, it listens to the transmission medium. When a node detects a carrier, its Carrier Sense is turned on and it will defer transmission until the medium is free: if two or more stations simultaneously begin to transmit, a collision occurs. In this case, the BEB algorithm for a random time interval is employed as below:

- When a collision occurs, each CSMA/CD unit chooses to back off for a period of time, determined by the backoff value. The maximal backoff time value at each unit involved in the collision is multiplied by 2 (maximum upper bound of 1024 for the factor). The first or initial backoff time value is termed *'the minimal backoff time'*. Each CSMA/CD unit will choose a random backoff time value which follows an equal distribution with an upper bound given by the maximal backoff value.
- On a successful transmission, the transmitting unit sets its backoff value to zero.
- If a unit has attempted backoff 16 times due to collisions for transmitting the same packet, the BEB algorithm forces that unit to discard that packet. Furthermore, the backoff value of this unit is reset to zero, i.e any new backoff/retransmission attempt will be determined again by the minimal backoff time.

B. *Jitter*

Jitter is one of the critical parameters in high speed data communication channels [25]. In real-time technology, a missed hard deadline can have serious consequences. All real-time systems have a certain level of jitter (a variance on actual timing). In a real-time systems, particulary, jitter should be minimal so system performance can be guaranteed. In high speed communication systems, jitter will generally degrade performance. Jitter is used to express how much individual latencies tend to differ from the mean.

The standard deviation which is related to the message transfer jitter is given by Equation (1), where N is the total number of simulated packets, x_i is the delay of each transferred packet and $\bar{x}$ is the evaluated average packet delay.

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2} \tag{1}$$

In our analysis, we use the standard deviation to measure the jitter.

III. PRINCIPLE OF OUR COMMUNICATION NETWORK APPROACH

We have four principles for the suggested real-time communication. The first principle is the synchronization signal, the second is the introduction of a time slot for each

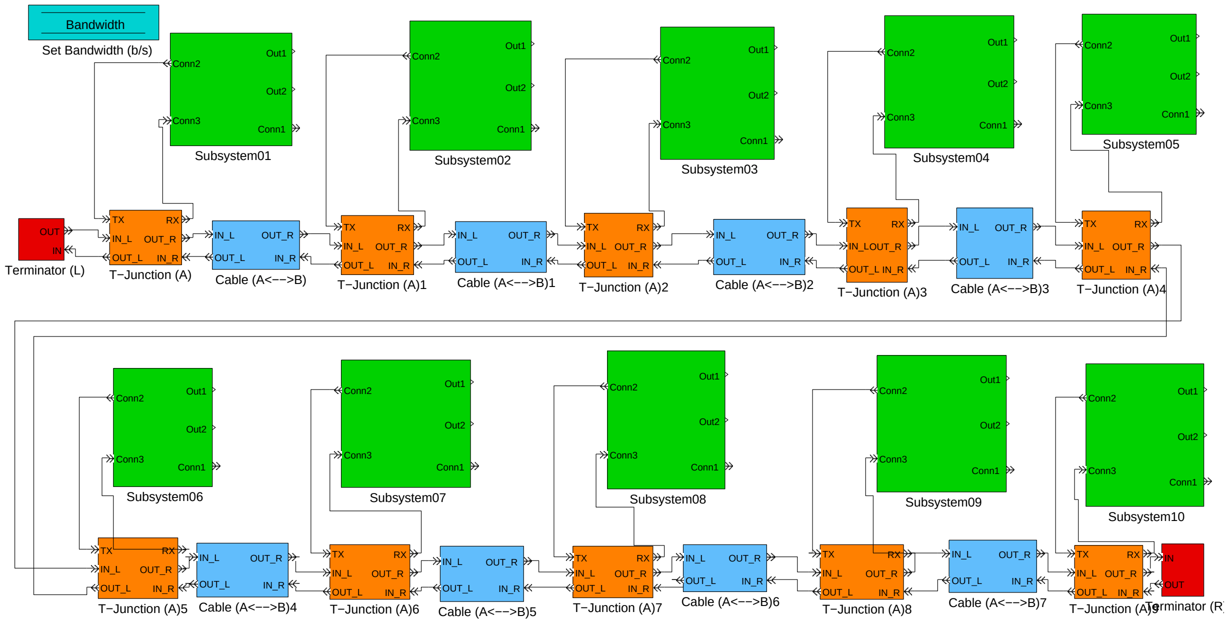

Fig. 2. *Network simulation model [26]*

Ethernet package and finally the application of different minimal backoff times for each MAC. In this paper, the length of one data package is fixed to 64 bytes for a 100Mbps network. Fixed-length and fixed-order transmission has a predictable transmission time and reduce the probability of frame collision.

- **Synchronization signal:** Real-time applications require tight synchronization so that the delivery of control messages can be guaranteed within defined message cycle times. In this paper, we have employed an internal synchronization clock for each CSMA/CD/MAC unit. It implies a precise clock synchronization among the different units so that all nodes are able to agree on their respective transmission slot. Practical implementation would be possible by using the IEEE 1588 clock synchronization approach.
- **Dedicated time slot for each of Ethernet unit:** In this communication network approach, we specify a time slot for each unit to avoid initial collision in the network bus. Messages are sent at time slots assigned to each of the units. Figure 1 shows the communication network approach.
- **Applying different minimal backoff times:** Despite data are sent at well defined time slots, jitter of each CSMA/CD unit may cause that data packages are sent outside their time slot and collisions occur. The BEB scheme comes into play for the half duplex Ethernet network. For, this the minimal backoff is specifically assigned to each of the CSMA/CD units.
- **Linear Backoff Scheme:** In a Linear Backoff scheme, the increase of the maximum backoff window grows linearly on each successive failure.

IV. SIMULATION SETUP

Simevents, part of Simulink for Matlab is used for modeling the communication bus. It uses state flow chart for implementation of the backoff logic in the communication system. The system consists of 10 CSMA/CD/MAC units that share the bandwidth on the Ethernet bus. The physical components of the network are represented by the terminator, T-junction, and cable blocks at the bottom of the simulation model. Apart from that, this model provides the number of collision and delivered packets of each unit [26].

This simulation system enables to evaluate characteristics such as the average latency and jitter of the message transmission. Each of the units consists of the following:

- An Application block that models the consumption of data.

- A MAC controller that governs the Ethernet unit's use of the shared channel.
- A T-junction to connect each of the units to the network model.

This basic setup [26] allows the implementation of our specific problem for synchronized sending of data packages.

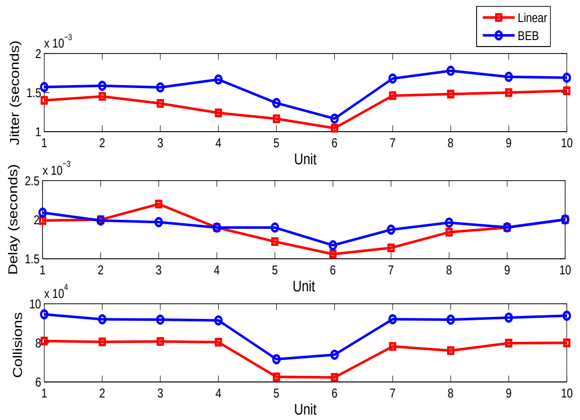

Fig. 3. *Jitter, delay and collisions for Case 1*

TABLE I
CASE 1 : IDENTICAL MINIMAL BACKOFF TIME AND IDENTICAL SENDING TIME.

Scheme	Average Jitter	Average Delay	Maximum Collisions
Binary Exponential Backoff	1.4068ms	1.8393ms	94,689
Linear Backoff	1.3063ms	1.6881ms	80,715

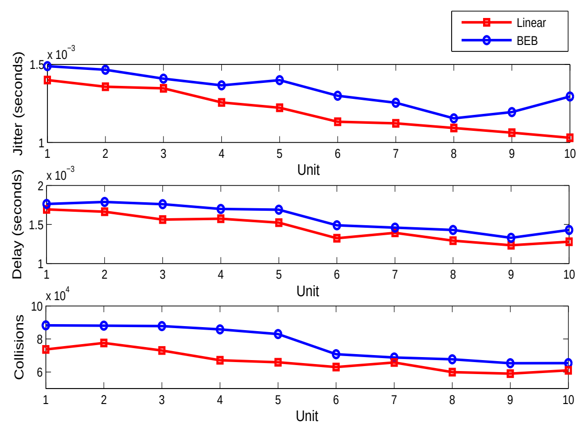

Fig. 4. *Jitter, delay and collisions for Case 2*

For this, we specify the packet generation rate and packet size range at the application blocks, the transmission buffer size of 25 packets at the MAC Controller blocks, and the length of the cable at the Cable blocks. We also implemented additional Simulink logic to measure the latency of the communication network from a transmit point, Tx to receive point

TABLE II
CASE 2 : DIFFERENT SLOT TIME AND IDENTICAL BACKOFF TIME.

Scheme	Slot Time	Average Jitter	Average Delay	Maximum Collisions
Binary Exponential Backoff	(ST1)	1.2761ms	1.6745ms	88,194
Linear Backoff	(ST1)	1.1847ms	1.4913ms	74,670
Binary Exponential Backoff	(ST2)	1.1825ms	1.6211ms	87,637
Linear Backoff	(ST2)	1.1750ms	1.4522ms	72,491

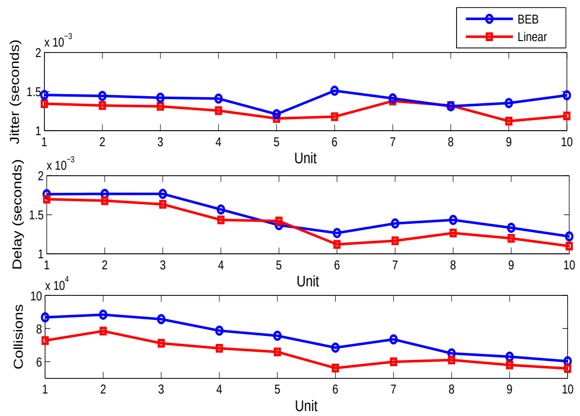

Fig. 5. *Jitter, delay and collisions for Case 3*

for, Rx in the Ethernet MAC control block to measure the sending of a packet. All the units transmit at the average rate of 100 packets per second with the packet size of 64 bytes and the length of the cable determined at the Cable blocks of 100 meter. The synchronization signal has a frequency of 100Hz which is artificially created by a synchronized signal through a frequency generator in the subsystem application block.

It is assumed that the data source has a certain jitter: each packet is created with a jitter of 10μs (standard deviation). The packet rate guarantees that any collision between packets is only due to packets being sent after a synchronization event. Thus, no collision should occur due to overload of the network, but only arbitration between synchronized data packets. This guarantees to understand the true delay and jitter due to our suggested scheme. The statistical evaluation of jitter and packet delay is carried out for 300 seconds. This gives a sufficiently large number of synchronization events for statistical evaluation.

V. SIMULATION ANALYSIS

The performance of the proposed strategy is compared to the traditional setting of the backoff scheme in Ethernet. It is our target to investigate the effect of different minimal backoff intervals and dedicated sending time slots for each MAC unit. The approach is to assign the minimal back-off times and sending time slots either in a random fashion or

TABLE III
CASE 3 : DIFFERENT MINIMAL BACKOFF TIMES AND IDENTICAL PACKAGE SENDING IN TIME SLOTS.

Scheme	Backoff Time	Average Jitter	Average Delay	Maximum Collisions
Binary Exponential Backoff	(BT1)	1.2187ms	1.5766ms	85,698
Linear Backoff	(BT2)	1.1531ms	1.4563ms	75,945
Binary Exponential Backoff	(BT2)	1.1471ms	1.3271ms	81,760
Linear Backoff	(BT2)	1.1092ms	1.2116ms	72,512

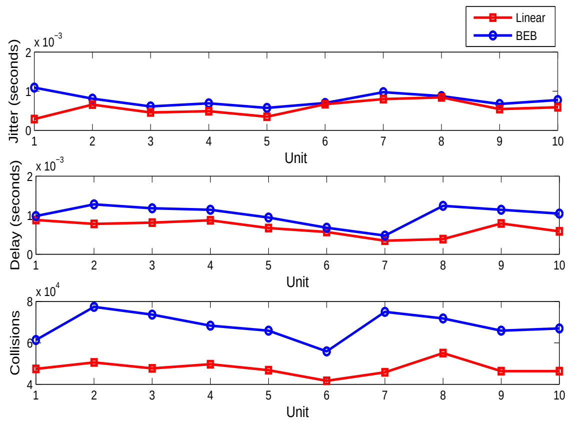

Fig. 6. *Jitter, delay and collisions for Case 4*

according to a linear approach to the 10 available MACs.

Case 1: Identical minimal backoff time and identical sending time.

For this case, all the data packages are sent with identical minimal backoff time of 51.2 μs and an identical sending time triggered by the synchronization signal, is set for each CSMA/CD controller unit. Table 1 summarizes the performance of both schemes and Figure 3 and Table 1 illustrate the values of jitter, delay and number of collisions for each MAC unit controller. As illustrated, the results show for this case that the Linear Backoff setting is the better strategy to reduce jitter, delay and collisions. Clearly, the Linear backoff scheme appears to be more deterministic than the BEB scheme as jitter in general about is 9.2% smaller.

Case 2: Identical minimal backoff times and different package sending in time slots.

We investigated the backoff characteristic by assigning different time slots for the sending of each data package (see Table 5, ST1 and ST2). The minimal backoff times for each MAC unit is identical at the value of 51.2 μs for both the BEB and Linear Backoff scheme. The assignment of different sending slots to data packages will reduce the need

TABLE IV
CASE 4 : RESULTS OF DIFFERENT MINIMAL BACKOFF TIMES AND DIFFERENT SLOT TIMES.

Scheme	Average Jitter	Average Delay	Maximum Collisions
Binary Exponential Backoff	0.9760ms	1.0819ms	71,392
Linear Backoff	0.7642ms	1.0431ms	54,885

for arbitration, although collisions are still expected due to jitter in the transmission units/times. Again, based on Table 2 and Figure 4, the Linear Backoff scheme shows an advantage over the BEB scheme. Nevertheless, a general decrease in jitter is also evident from Case 2 in relation to Case 1. This is clearly due to the different time slots for each MAC-unit. However, note that for the ST1 scheme, some MAC-unit use similar slot times in this case. This contrasts scheme ST2, where each MAC unit has different time slots. Thus, less arbitration, collisions and lower sending jitter, delay are observed for ST2 in relation to ST1.

Case 3: Different minimal backoff times and identical package sending in time slots.

We observed the backoff characteristic by assigning different minimal backoff times (as in Table 5 (BT1)), multiples of 51.2μs, to each of the MAC controller in the Ethernet network with identical package sending time slots. Table 3 and Figure 5 depict the network performance for both schemes where the random different backoff times still affect the results of jitter and delay in comparison to Case 2. In this case, the Linear Backoff scheme has again achieved a better level of performance over the BEB scheme. It is interesting to note that there is an advantage of the Linear Backoff scheme in Case 3 over Case 2 results. Thus, the backoff algorithm and the minimal backoff times have significant effect on jitter.

Case 4: Linearly increasing minimal backoff times and different package sending in time slots.

In this case, the minimal back-off times are linearly distributed across the ten MACs. The principal idea is to reduce collision between different packages once a first collision has occurred. By assigning different minimal backoff times and different slot time (see Table 5 (ST1 and BT1) to the Linear and BEB algorithms of each MAC unit, it is hoped that the MACs will have a lowered probability of collision. Hence, in this case, each unit in the network model has been specified by the different minimal backoff and slot time value as in Table 1. Figure 6 and Table 4 illustrate the performance result of the communication in Ethernet networks.

Case 5: Random minimal backoff times and package sending in time slots.

For further investigation, the minimal back-off times are randomly distributed across the ten MACs (see Table 3 (ST2 and BT2)). Table 6 and Figure 7 illustrate the performance

TABLE V
SIMULATION : DISTRIBUTED SLOT TIMES WITH RANDOM BACKOFF TIMES

Unit	Different slot time(ST1)	Different minimal backoff time(BT1)	Different slot time(ST2)	Different minimal backoff time (BT2)
1	0.0512 ms	0.0512 ms	0.2048 ms	0.1024 ms
2	0.0512 ms	0.1024 ms	0.1024 ms	0.2560 ms
3	0.0512 ms	0.1536 ms	0.0512 ms	0.0512 ms
4	0.1024 ms	0.2048 ms	0.4096 ms	0.1536 ms
5	0.1024 ms	0.2560 ms	0.2560 ms	0.4096 ms
6	0.1024 ms	0.3072 ms	0.4608 ms	0.5120 ms
7	0.2048 ms	0.3584 ms	0.3072 ms	0.2048 ms
8	0.2048 ms	0.4096 ms	0.1536 ms	0.1024 ms
9	0.2048 ms	0.4608 ms	0.3584 ms	0.3072 ms
10	0.4096 ms	0.5120 ms	0.5120 ms	0.3584 ms

of the different test for the combination of ST2 and BT2. It is evident that the Linear Backoff scheme provides an advantage over the BEB case, as the smallest average jitter is provided by the Linear Backoff scheme. Moreover, a too close set of sending and backoff times increase the chance for collisions, which is evident when comparing Case 4 with Case 5. Case 5 shows better results since each unit has now indeed a dedicated time slot (and minimal backoff time).

Comparative analysis

After simulating all the communication scenarios as above, it has been found that both numbers of successful transmissions and collisions are affected by the back-off times for both schemes, Linear and BEB. When the minimal back-off time is not long enough the number of collisions detection scheme can be very significant. The different minimal back off time and time slot method are able to avoid the stations from repetitively entering the back off state, minimizing the collision possibility. This is particularly obvious for the Case 3 in relation to Case 2.

It is shown that Linear Backoff is significantly lower than BEB in the results of jitter, delay and collisions. In Case 5, up to 54.8% of jitter and delay can be reduced using the Linear Backoff compared to the standard setting of Ethernet, BEB scheme. One of the reasons of this scenario is that linear increments give enough backoff time to enhance the network performance by reducing the number of transmission failures. The number of collisions for both schemes particularly MAC-unit 10 has indeed significantly decreased more than 55.6%. Hence, Linear Backoff introduces a higher level of fairness. Case 5 is in fact providing the best overall result for the investigated cases here. This is also true when considering the combination of sending time slots ST2 together with minimal backoff times as in BT1 (See Table 3 for ST2 and BT1 and Table 7 for results)

In future work, longer minimal backoff times will be investigated. It is expected that a long random backoff time cause ineffective channel utilization, but a short one may suffer from high collision rate. The backoff time relates directly to the ability to access to the channel in Ethernet network communication, lower backoff time means more chance to succeed in channel contention. Thus, it can be seen that selection of the backoff algorithm is very significant for the network performance.

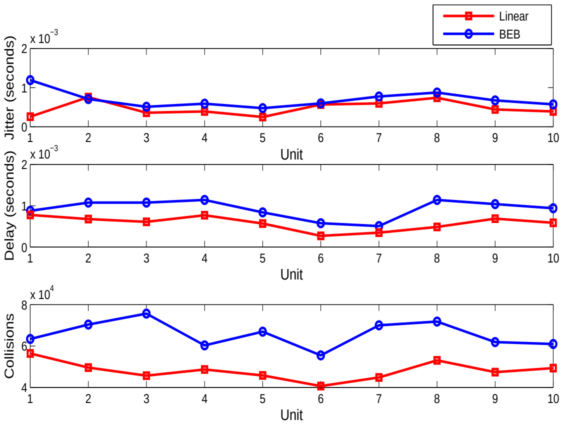

Fig. 7. *Jitter, delay and collisions for Case 5*

TABLE VI
CASE 5 : RESULTS OF RANDOM MINIMAL BACKOFF TIMES AND SLOT TIMES.

Scheme	Average Jitter	Average Delay	Maximum Collisions
Binary Exponential Backoff	0.9322ms	1.0566ms	69,795
Linear Backoff	0.7167ms	1.0163ms	52,655

Overall, the backoff mechanism dramatically affects the performance of the MAC protocol, and hence the overall network performance. The backoff period is directly related to the nodes' idle times. As a result, the standard exponential back-off scheme has been shown on 5 cases to result in high packet delays and jitter.

VI. CONCLUSION

In this paper, we constructed and simulated the original CSMA/CD protocol through Simevents-Matlab block. We

TABLE VII
RESULTS OF DIFFERENT MINIMAL BACKOFF TIMES AND DIFFERENT SLOT TIMES (BT1 & ST2)

Scheme	Average Jitter	Average Delay	Maximum Collisions
Binary Exponential Backoff	$0.9773ms$	$1.0733ms$	72,655
Linear Backoff	$0.8421ms$	$1.0345ms$	54,828

created a time synchronized bus communication by taking inspiration from CANopen, i.e packets are sent at given time slots while any collision is resolved through the BEB and Linear Backoff schemes of Ethernet in a random approach for the minimal backoff time. We analyzed the effect of the backoff time on the system performance, in terms of jitter and delay. The simulated test results have shown that the delay jitter can be reduced by choosing the correct backoff time to be implemented in the MAC controller. The special assignment of the minimal backoff times to each MAC unit allowed to minimize the packet transmission time jitter by up to 55%. The key results are that a Linear Backoff scheme exhibits lower jitter and access delay than a BEB scheme. Linear Backoff appears to be more deterministic. Our approach of an Ethernet network based communication system improves determinism at low cost.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the contribution of the Ministry of Higher Education Malaysia (MOHE) and the reviewers' comments.

REFERENCES

[1] K. C. Lee and S. Lee, "Performance evaluation of switched ethernet for real-time industrial communications," *Computer Standards and Interfaces*, vol. 24, no. 5, pp. 411 – 423, 2002.
[2] C. Venkatramani and T.-c. Chiueh, "Design, implementation, and evaluation of a software-based real-time ethernet protocol," *SIGCOMM Comput. Commun. Rev.*, vol. 25, pp. 27–37, October 1995.
[3] "Embedded network," in *http://techon.nikkeibp.co.jp/article*, 1 November. 2009.
[4] G. Kopetz, H. Griinsteidl, "Ttp-a protocol for fault-tolerant real-time systems," in *IEEE Computer*, pp. 14–23, January 1994.
[5] D. Wiklund and D. Liu, "Socbus: switched network on chip for hard real time embedded systems," in *Parallel and Distributed Processing Symposium, 2003. Proceedings. International*, p. 8 pp., april 2003.
[6] M. Felser, "Real-time ethernet - industry prospective," *Proceedings of the IEEE*, vol. 93, pp. 1118 –1129, june 2005.
[7] V. Pallares-Lopez, A. Moreno-Munoz, M. Gonzalez-Redondo, R. Real-Calvo, I. Garcia, A. de Castro, F. Perez, and J. de la Rosa, "Deterministic ethernet synchronism with ptp-base system for synchrophasor in smart grid," in *Compatibility and Power Electronics (CPE), 2011 7th International Conference-Workshop*, pp. 22 –27, june 2011.
[8] R. Focardi, R. Gorrieri, and F. Martinelli, "Real-time information flow analysis," *Selected Areas in Communications, IEEE Journal on*, vol. 21, pp. 20 – 35, jan 2003.
[9] R. Metcalfe, "Computer/network interface design: lessons from arpanet and ethernet," *Selected Areas in Communications, IEEE Journal on*, vol. 11, pp. 173 –180, feb 1993.
[10] F. Tobagi and I. Dalgic, "Performance evaluation of 10base-t and 100base-t ethernets carrying multimedia traffic," *Selected Areas in Communications, IEEE Journal on*, vol. 14, pp. 1436 –1454, sep 1996.
[11] S.-K. Kweon and K. Shin, "Statistical real-time communication over ethernet," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 14, pp. 322 – 335, march 2003.
[12] F. Carreiro, R. Moraes, J. Fonseca, and F. Vasques, "Real-time communication in unconstrained shared ethernet networks: the virtual token-passing approach," in *Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE Conference on*, vol. 1, p. 8 pp.432, sept. 2005.
[13] F. Recacha, J. Melus, X. Simon, M. Soriano, and J. Forne, "Secure data transmission in extended ethernet environments," *Selected Areas in Communications, IEEE Journal on*, vol. 11, pp. 794 –803, jun 1993.
[14] J.-L. Scharbarg, M. Boyer, and C. Fraboul, "Can-ethernet architectures for real-time applications," in *Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE Conference on*, vol. 2, pp. 8 pp. –252, sept. 2005.
[15] J. Cloud, L. Zeger, and M. Medard, "Mac centered cooperation; synergistic design of network coding, multi-packet reception, and improved fairness to increase network throughput," *Selected Areas in Communications, IEEE Journal on*, vol. 30, pp. 341 –349, february 2012.
[16] B. Wiese and K. Jacobsen, "Use of the reference noise method bounds the performance loss due to upstream power backoff," *Selected Areas in Communications, IEEE Journal on*, vol. 20, pp. 1075 –1084, jun 2002.
[17] T. Skeie, S. Johannessen, and O. Holmeide, "Timeliness of real-time ip communication in switched industrial ethernet networks," *Industrial Informatics, IEEE Transactions on*, vol. 2, pp. 25 – 39, feb. 2006.
[18] L. Seno, S. Vitturi, and C. Zunino, "Real time ethernet networks evaluation using performance indicators," in *Emerging Technologies Factory Automation, 2009. ETFA 2009. IEEE Conference on*, pp. 1 –8, sept. 2009.
[19] L. Tao, T. Jiang, and Z. Xiangli, "A solution for ethernet-based real-time communication network of distributed numerical control system," in *Technology and Innovation Conference 2009 (ITIC 2009), International*, pp. 1 –6, oct. 2009.
[20] L. Lo Bello, G. Kaczynski, and O. Mirabella, "Improving the real-time behavior of ethernet networks using traffic smoothing," *Industrial Informatics, IEEE Transactions on*, vol. 1, pp. 151 – 161, aug. 2005.
[21] P. Ferrari, A. Flammini, D. Marioli, and A. Taroni, "A distributed instrument for performance analysis of real-time ethernet networks," *Industrial Informatics, IEEE Transactions on*, vol. 4, pp. 16 –25, feb. 2008.
[22] G. Chockler, D. Malkhi, and M. Reiter, "Backoff protocols for distributed mutual exclusion and ordering," in *Distributed Computing Systems, 2001. 21st International Conference on.*, pp. 11 –20, apr 2001.
[23] O. Kleineberg, P. Frohlich, and D. Heffernan, "Fault-tolerant ethernet networks with audio and video bridging," in *Emerging Technologies Factory Automation (ETFA), 2011 IEEE 16th Conference on*, pp. 1 –8, sept. 2011.
[24] K. Wang, C. Zhang, X. Ding, S. Ji, and T. Hu, "A new real-time ethernet for numeric control," in *Intelligent Control and Automation (WCICA), 2010 8th World Congress on*, pp. 4137 –4141, july 2010.
[25] A. Mejia, J. Flich, J. Duato, S.-A. Reinemo, and T. Skeie, "Boosting ethernet performance by segment-based routing," in *Parallel, Distributed and Network-Based Processing, 2007. PDP '07. 15th EUROMICRO International Conference on*, pp. 55 –62, feb. 2007.
[26] "Simevents," in *http://www.mathworks.co.uk/products/simevents/demos.html*, 15 November. 2010.

Mohamad Khairi Ishak received a B.Eng. (Hons.) in Communication Engineering from the International Islamic University Malaysia. In 2008, he received an MSc. in Embedded System from the University of Essex, United Kingdom. Currently, he is a final year PhD student at the University of Bristol. His PhD research focuses on real-time control communication system particularly in Ethernet.

Dr. Guido Herrmann received the German degree Diplom-Ingenieur der Elektrotechnik (with highest honours) from the Technische Universitt zu Berlin. In 2001, he received a Ph.D. from the University of Leicester. From 2001 to 2003, he worked in the mechatronics and micro-systems group of the A*Star Data Storage Institute (Singapore) doing research and consultancy for the data storage industry. From 2003 until February 2007, he was a Research Associate, Research Fellow and a Lecturer in the Department of Engineering at Leicester University. In March 2007, he took up a permanent lecturing position at the Department of Mechanical Engineering of the University of Bristol. In August 2009, he was promoted to Senior Lecturer. Since August 2012, he has been a Reader in Control and Dynamics.

Martin Pearson received a B.Eng. (Hons.) in Electrical and Electronic Engineering from the University of Manchester in 1996, an MSc. in Advanced Technologies in Electronics from the University of the West of England in 2002 and a PhD. in Computational Neuroethology and Robotics in 2008. He is currently a Research Fellow and theme leader of the Biomimetics and NeuroRobotics group at the Bristol Robotics Laboratory, United Kingdom.