A simple approach for minimal jitter in Ethernet for real-time control communication

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. The Binary Exponential Backoff (BEB) scheme is used. 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 backoff 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, backoff time and network model**

I. INTRODUCTION

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 [1]. Real-time control networks must provide a guarantee of service and consistently operate deterministically and correctly.

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 [2], [3]. 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 application with mixed critically requirement. TTEthernet relies on specific switches to organize the data communication and establish global synchronization. However, there is an issue about latency in TTEthernet which is caused by switch delays.

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 [4]. Therefore, the transmission delay for higher priority messages can be guaranteed. But, compared with the other networks, CAN has a slow data rate of 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, 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 [5], [6], [7], [8], [9], [10], [11].

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 avoidance 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 via 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 analyzes the

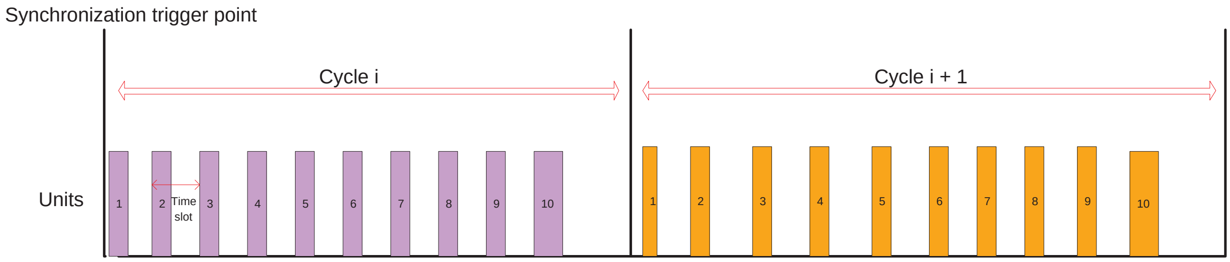

Fig. 1. *Time slot for the communication network approach*

CSMA/CD performance under different minimal back-off interval scenarios. Finally, Section 5 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 [12]. 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 [13]. 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 system particulary, jitter should be measurable 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.

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

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

III. PRINCIPLE OF OUR COMMUNICATION NETWORK APPROACH

We have three 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 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 unit. Figure 1 shows the communication network approach.

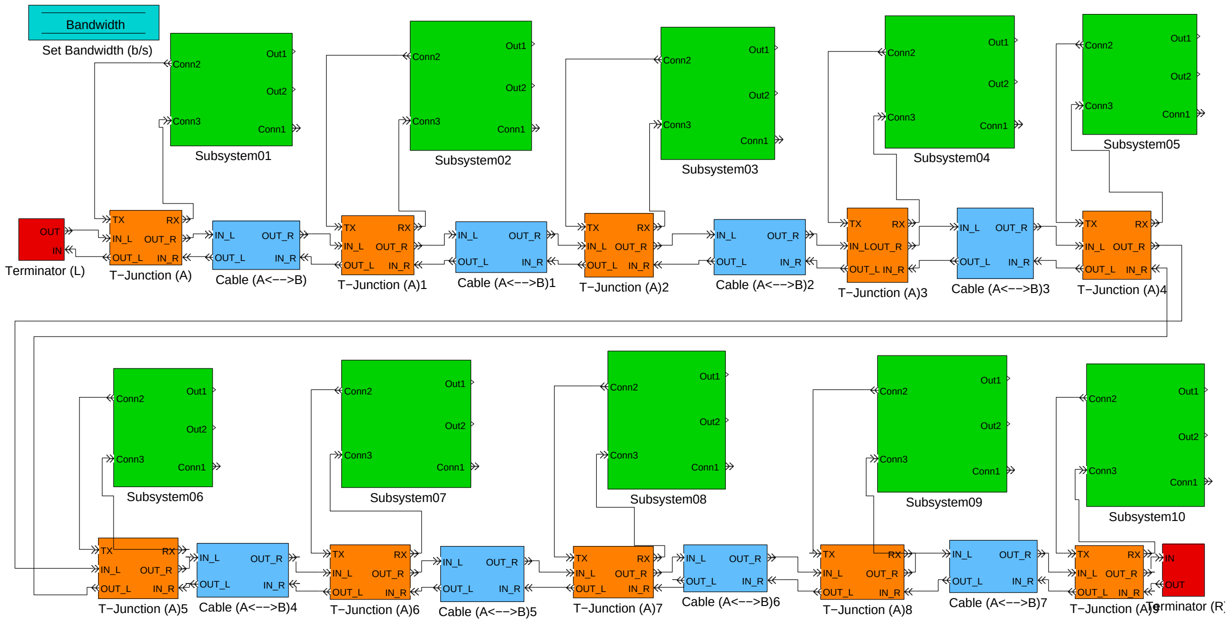

Fig. 2. *Network simulation model*

- **Applying different minimal backoff time** 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.

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 control logic in the communication system.

The system consists of 10 CSMA/CD/MAC units that share the bandwidth on the Ethernet bus. In this particular setup, all the units transmit at the average rate of 100 packets per second with the packet size of 64 bytes. 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.

We can 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. Each data packet is sent at a rate of 100 packets per second, while it is assumed that the data source has a certain jitter: each packet is created with a jitter of $10\mu s$ (standard deviation).

V. SIMULATION ANALYSIS

It is our target to investigate the effect of different minimal backoff intervals for each MAC. The approach is to assign the minimal back-off times either in a random fashion or according to a linear approach to the 10 available MACs. In addition, two comparative cases are considered. Both cases use the same minimal backoff time of 51.2 μs while one of them does not introduce the principle of time slots for data packet sending, i.e all data packages are sent at the time of the synchronization signal. The overall aim is to understand the effect of different minimal back-off times on the communication system.

A. Identical minimal backoff time and identical sending time

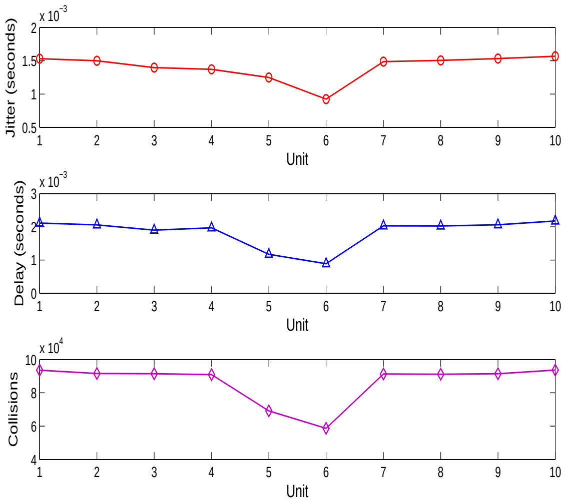

Fig. 3. *Jitter, delay and collisions for minimal backoff time and identical sending time*

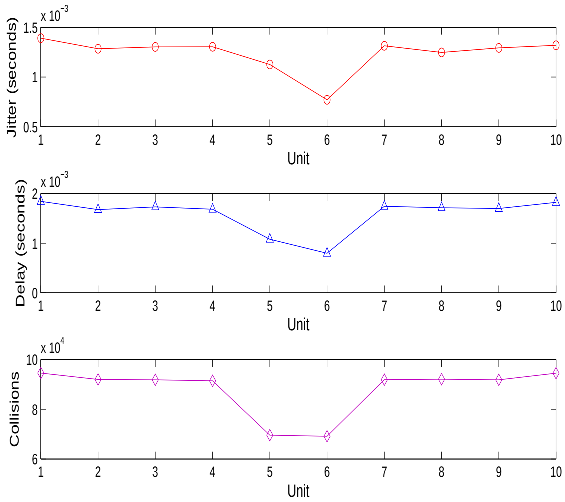

Fig. 4. *Jitter, delay and collisions for identical backoff time*

For this case, all the data packages are sent with identical minimal backoff time of 51.2 μs and an identical sending time is set for each CSMA/CD controller unit. Based on the results shown in Figure 3, the maximum jitter is 1.5683ms while the smallest jitter, and the average jitter are 0.9416ms and 1.4068ms. The average delay is 1.8393ms. Since the data have been sent with the identical sending time, it caused a very high number of collision 94,689.

*B. **Identical minimal backoff times and package sending in time slots***

Under this condition, we observed the backoff characteristic by assigning identical minimal backoff times of 51.2μs to each unit in Ethernet network. Note that the maximum jitter is 1.3903ms and minimal jitter is 0.7701ms for particular MACs. The average jitter has a value of 1.2352ms and the average delay is 1.5769ms.

In fact, the probability of frame retransmission is now slightly lower and shows the maximum number of collisions is 94,523 for one of the MACs. Figure 4 shows the network performance, particularly for jitter, delay and collisions. It will be seen that this is a rather high jitter and a high number of collisions when compared to the next cases.

*C. **Linearly increasing minimal backoff times***

In this case, the minimal back-off times are linearly distributed across the ten MACs. The principal idea is to avoid collision between different packages once a first collision has occurred. By assigning different minimal backoff times to the BEB algorithm 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 minimal backoff time value as in Table 1. Figure 5 illustrates the performance result of the communication in Ethernet networks. In terms of jitter, the minimum and maximum jitter are 0.5731ms and 1.0912ms. The average jitter is 0.9760ms and the average delay is 1.0819ms.

It is interesting to note that the number of collisions (Figure 5) has indeed significantly decreased (71,392 for MAC-unit 10) while others have increased to 110,338 collisions (MAC-unit 1), in comparison to the first considered cases (Sections 5A and 5B).

TABLE I
EQUAL DITRIBUTED BACKOFF TIME

Unit	Backoff selection times
1	0.0512 ms
2	0.0512 ms
3	0.0512 ms
4	0.1024 ms
5	0.1024 ms
6	0.1024 ms
7	0.2048 ms
8	0.2048 ms
9	0.2048 ms
10	0.4096 ms

*D. **Randomized distributed minimal backoff times***

In this simulation test, we choose to use minimal back-off times generated randomly. They are multiples of 51.2μs. The aim is again to guarantee for each unit a different back-off time which may again minimize jitter. We have selected the random value interval backoff time in the CSMA/CD protocol as in Table 2.

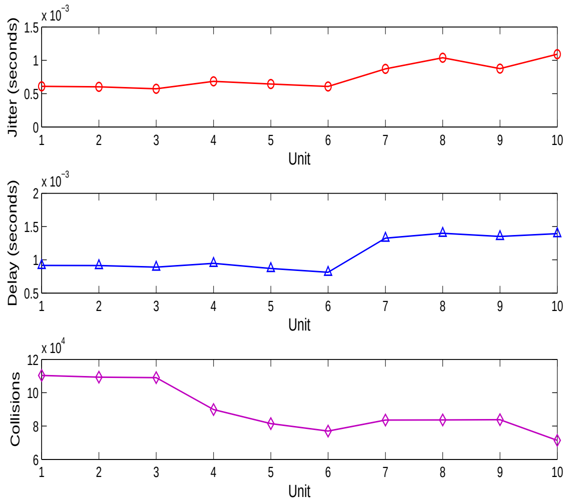

Fig. 5. *Jitter, delay and collisions for equal distributed backoff time*

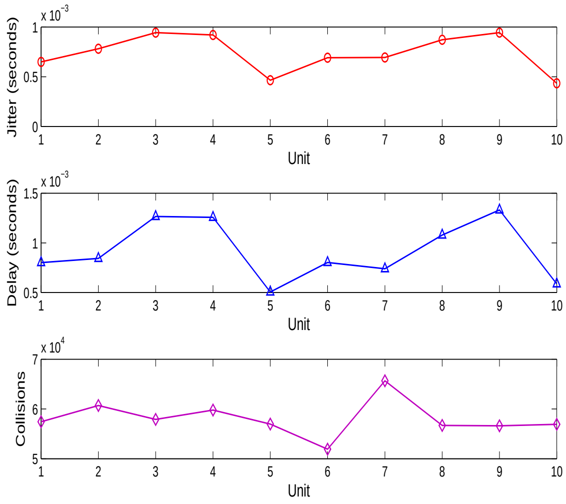

Fig. 6. *Jitter, delay and collisions for random distributed backoff time*

TABLE II
RANDOM DISTRIBUTED BACKOFF TIME

Unit	Backoff time selection times
1	0.2560 ms
2	0.2048 ms
3	0.4608 ms
4	0.4096 ms
5	0.1024 ms
6	0.3072 ms
7	0.1536 ms
8	0.3584 ms
9	0.5120 ms
10	0.0512 ms

In this case (see Figure 6), the minimum jitter and the average jitter are 0.4342ms and 0.7396ms. The maximum jitter is 0.9438ms and the average delay is 0.9210ms. A specific MAC has high jitter when the unit has larger minimal backoff time. With different minimal backoff times, this random back-off time approach is able to avoid the stations from repetitively entering the backoff state and minimizes the collision possibility. In fact, the average collision number random minimal backoff time is 58,056 collision.

E. *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 time. In addition, when the minimal back-off time is not long enough the number of collisions detection scheme can be very significant. This is particularly obvious for the cases of Section 5A, 5B and 5C. Small and similar minimum backoff times cause a higher probability.

The random minimal back off time method is able to avoid the stations from repetitively entering the back off state and minimizes the collision possibility thus improves the performance significantly. Using random minimal backoff time can improve the overall network performance, since it rearranges the traffic pattern more randomly than the usual BEB scheme. Thus, it takes full advantage of the network, and has a good control to the network congestion.

VI. CONCLUSION

In this paper, we constructed and simulated the original CSMA/CD protocol through Simevents-Matlab block. We created a time-synchronizd bus communication by taking inspiration from CANopen, i.e packets are sent at given time slots while any collision is resolved through the BEB scheme 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 back-off times to each MAC unit allowed to minimize the packet transmission time jitter by more than 30%. 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] 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.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.