Real time Characteristics of Ethernet and its improvement

Zhi Wang, Ye-Qiong Song, Jiming Chen, Youxian Sun

Real Time Characteristics of Ethernet and Its Improvement[1]

Zhi WANG[1*] Ye_qiong SONG[2] Ji_ming CHEN[1] You_xian SUN[1]
1 National Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China
2 LORIA–ENSEM, 2, av. de la Forêt de Haye, 54516 Vandoeuvre-lès-Nancy, France
Author of correspondence: Zhi.Wang@ensem.inpl-nancy.fr or zhiwang_iipc@yahoo.com

Abstract: There is a growing interest of using Ethernet to support time constrained communication raised from process control, factory automation and other real-time applications. This paper first presents a comprehensive analysis of the temporal characteristics of traditional Ethernet and the improvements for them, then summarizes the interesting features offered by a switched Ethernet for supporting time constrained communication. At last, this paper points that it still lacks of general means to handle capability of switched Ethernet after investigating QoS (Quality of Service) of switched Ethernet.
Keyword: Switched Ethernet, Shared Ethernet, Fieldbus, Real-time, QoS, (m, k)-firm

1 INTRODUCTION

Industry applications, such as process control, factory automation and et al., intrinsically need DCS (distributed computer system) because of function of tasks and location of equipment and device being geographically distributed. DCS is characterized by the correctness of its tasks depending on both their logical results and the time at which these results appear. Within a DCS, tasks usually reside on distributed nodes and communicate with one another through message transfer to accomplish a common goal. Therefore, it is difficult to ensure timely results of tasks in a DCS without a network that supports the timely inter-task messages (H.Koptez 1997; C.Bottazzo 1997; Tindell 1997). For meeting real-time communication of industrial applications in the lower layer, fieldbus, a kind of special-purposed industrial network, has been developed and successfully implemented during the last two decades. However the expansion of quantity of intra-plant data and the creation of inter-plant communication needs require real-time networks of higher bandwidth, especially for transmitting a large quantity of data between lower layer and middle layer networks. Whilst most of popular fieldbus, such as WorldFIP, FF, Profibus, P-Net and CAN, only offer low bit rate (31.25Kbps to 5 Mbps).

To resolve this problem, many general-purpose high speed networks like FDDI and ATM were proposed and their capability of supporting both HRT (Hard Real Time) and SRT (Soft Real Time) has been evaluated by Malcolm and Zhao (1995); Song and Simonot (1996); Hansson and Sjödin (1997); Mammeri and Haouam (1997). However these solutions have not met expected success, mainly because of their high costs and high handling complexity. For example, using FDDI for HRT applications requires careful assignment of synchronous bandwidth to each node. Malcolm and Zhao (1995) and later Song and Simonot (1996) have shown that any change (adding or withdrawing nodes) will lead to re-assignment of these bandwidths. As for ATM, real-time applications' time constraints should be carefully mapped to the adequate ATM QoS parameters (Mammeri and Haouam, 1997) and message scheduling in each ATM switch requires special tuning (Hansson and Sjödin, 1997). Another problem of using fieldbuses or specially adjusted FDDI or ATM is that their interconnection is a difficult task since neither standard nor uniformed application layer services and user interface exist. Although IEC in 2000 establishes IEC 61158 International standard for fieldbus, it is nothing than a collection of the eight independent protocols currently on the market: IEC 61158 type 1, ControlNet, Profibus, P-Net, FF, Swift Net, WorldFIP and Interbus-S.

Ethernet is incontestably the most cost-effective solution (very low price, component maturity and hence reliability, stability since widely used by IT industry). Besides its large bandwidths (10Mbps, 100Mbps, 1Gbps, 10Gbps), the fact that it is a *de facto* standard, it also takes advantages of many widely spread the facilities and user protocols (IP/TCP-UDP, HTTP, FTP, SNMP, TELNET,…) implemented on the top of Ethernet. All these makes its use very easy and allows integrating many COTS (Commercial off-the shelf) API like OPC (Open Process Control http://www.opcfoundation.org, 1997), Microsoft DCOM, Corba, Java/RMI, etc.

But classic shared medium is considered as unsuitable for supporting real-time applications because of the uncontrollable collisions of its CSMA/CD MAC protocol, which can lead to an unbounded medium access delay. Nevertheless many tentatives have been made to enable it to support time constrained communication (see section 2) in the past, all these require specific protocol adaptation and the compatibility with Ethernet is not ensured.

Switching technology makes easier all the above mentioned tentatives since switched Ethernet can easily reduce or even completely eliminate the collisions by defining one *collision domain* per switch port. Moreover, *full-duplex* port (with auto-negotiation when to be linked to an half-duplex one or to a different bit rate line), *on-the-fly forwarding* (not IEEE standard), trunk links *aggregation* and inter-switch frame *prioritisation* (IEEE802.1p) considerably improved the performance comparing to the traditional bridge-based micro-segmentation approach. The definition of *VLAN* (IEEE802.1Q) allows for further delimitating the *broadcast domain* by confining broadcast traffic within a same VLAN and enhances thus the security between different VLANs. Redundant paths between stations with deployment of *STP* (Spanning Tree Protocol) improved the reliability. Combined to the VLANs, *traffic optimisation* can further be achieved. The use of *wireless* Ethernet also contributes to improve the deployment flexibility.

This is why switched Ethernet is more and more considered as an attractive enabling technology for supporting time-

[1] This paper supported by NSFC-60084001 and PRA-S101-04.

constrained communication. On the one hand, IAONA (Industrial Automation Networking Alliance, http://www.iaona-eu.com) and IEA (Industrial Ethernet Alliance, http://ethernet.industrialnetworking- .com) are actively working towards to use Ethernet and TCP/IP as the standard network for the process control and factory automation. On the other hand, to benefit the very good bandwidth/cost ratio, the high reliability of the NICs (Network Interface Cards), the scalability and the flexibility for reconfiguration, many traditional fieldbus protocols now begin to include Ethernet as a lower-layer communication supporting system and add special real-time handling features (corresponding to the fieldbus application layer) to the upper layers (traditional TCP/IP stack does not support real-time handling). For instance, FF HSE (High Speed Ethernet) proposed its upper layer of IEC 61158 over High Speed Ethernet (HSE testing kit and H2 Technology Press Releases 2000, http://www.fieldbus.org/default.htm). Moreover, the willing of using web-based applications (e.g., XML-based device description, device management) and taking advantage of Internet technology has lead to almost all fieldbuses now proposing IP as a unified central point. The unified central point is either directly implemented over lower-layer protocols of fieldbus (FIPWEB of WorldFIP described in Lainé 1999) or at least a gateway allowing access to fieldbus via Internet (LonWorks/IP gateway presented by Soucek 1999). It seems now that both the traditional fieldbus and the newly established industrial Ethernet communities agree to consider Ethernet as an interesting low layer network in industrial region.

Although a lot of work has been done, there exists few fundamental research works on temporal behaviours of the switched Ethernet, which is focus of this paper. The rest of the paper is organized as follows, Section 2 gives a analysis of temporal characteristic of classic shared Ethernet and its improvement method; Section 3 gives a comprehensive analysis of the interesting features offered by a switched Ethernet for supporting time constrained communication; Section 4 proposes QoS of an Ethernet switch; Section 5 concludes this paper and points out some future research directions.

2 SHARED ETHERNET & REAL-TIME

2.1 Communication Procedure of Shared Ethernet

Shared Ethernet uses the CSMA/CD protocol to control the access of all the interconnected stations to the common shared medium, and applies 1-persist BEB algorithm to deal with collision when that occurs.

A station monitors medium when it is ready to sent a data, and wait until the medium is free. This way refers to CSMA. A data is corrupted if collision occurs during the transmission of the data, so immediately stopping transmission of the data not only saves transmission time but also medium bandwidth. The CSMA with monitoring function during transmission of data refers to CSMA/CD.

Collision truly occurs during the transmission of data between two or multiple stations when they sent data simultaneously, therefore in order to deal with the issue proper mechanism must be applied. Normally, according to the mechanism to solve collision, CSMA is divided into three classes, 1-persist CSMA, 0-persist CSMA and p-persist CSMA. Within 1-persist CSMA, a station will first monitor medium when it has data to sent, and then wait until medium is free; it keeps monitoring medium if a collision occurs until medium is free and retransmit. Within 0-persist CSMA, the station will first monitor medium and wait until medium is free too; it delays a random time before next monitoring medium if a collision occurs. The procedure is repeated until the station successfully sent its data. P-persist CSMA is mainly used in slotted medium, wherein a station either sends its data with probability of p or delays its transmission with probability of 1-p if the slotted medium is free. The procedure is repeated until transmission of data is completed or a collision occurs with other stations.

The fact that the more of the selective region of random delay to which stations sent data according, the more of the probability that data is send successfully, motivates the famous BEB algorithm to deal with the collision. Within BEB algorithm, there is a counter, counting number of collisions, in each station. For a station in case of a collision occurring, the station retransmits its data with a transmission delay after medium is free, where the transmission delay is determined according to the number of its counter. When first collision occurs, the station will randomly select a delay time between (0, 1). When second collision occurs, the station will randomly select a delay time among (0, 1, 2, 3). Generally, when the station consecutively suffers i collision, the station will randomly select a delay time among the number of (0, 1, … , 2^i-1). But, the maximum delay time is fixed on 1023 when number of collisions exceeds 10. Further, a failure indication is given after 16 collisions occur and the data is discarded. The structure of data frame in Ethernet is shown in Fig.1.

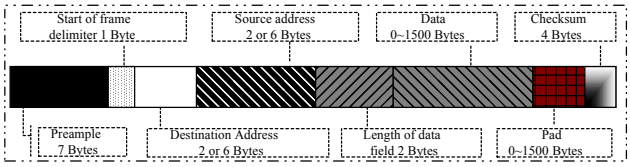

Fig.1 Data frame in Ethernet

The minimum length of data frame is 64 bytes and the length of data component is at least 46 bytes in order to effectively distinguish effective data frame and left data frame. Another reason for requiring minimum length of data frame is to prevent the situation occurs, a collision occurs but it is not found because too short transmission time of small data frame.

2.2 The Effect of CSMA/CD and BEB Algorithm

The use of the CSMA/CD protocol to control the access of all the interconnected stations to the common medium results in a non-determinist medium access delay since after every collision, a station has to wait a random delay before to retransmit. That directly leads to unfairness. The unfairness refers to a station will monopolise medium and continuously sends its data while others may have data

(eventually more critical) to transmit. Unfairness can be explained from two aspects, capture effect and starvation effect .

Capture effect refers to the behaviour wherein under high load, one station is able to hold the medium to transmit data consecutively, in spite of contention of other stations for accessing medium. Whetten ever explains what is capture effect with an instance. Consider two stations, station A and station B, interconnected with Ethernet, where one counter per station, counter A and counter B. Assume the initial numbers in the two counters are both 0, and increase 1 when collision occurs. Assume that there are respective data A and data B in the two stations be sent. When data A and data B are simultaneously sent, collision occurs. If within the first collision, station A selects back-off value of 0 (50%) from (0,1) and station B select back-off value of 1 (50%) from (0,1), data A is send successfully. If there is data A1 in station A to be sent after back-off value of 1, then data A1 and data B collide. Station A just uniformly selects back-off value from (0,1) in dealing with this collision, however, station B must do among (0, 1, 2 , 3). It is apparent that station A has more probability for getting a less back-off value and succeed in contending for medium. If there are still more data to be sent in station A, station B will continuously suffer collision until data B is discarded if number of collisions reaches to 16.

Starvation effect is a consequence of the capture effect, i.e., refer to some packets may starve when accessing medium. Ramakrishsan points that starvation effect causes some stations to experience maximum latencies exceed 100 times of the average latencies or being discarded due to its consecutive 16 collisions; it causes part of data to suffer serious delay even under load being as low as 40% and some packets to suffer complete starvation under load being as low as 60%. Iwasaki investigates the relation between maximum delay and load through simulation and concludes that the probability of collision is very low when load is less 70% and delay can be even ignored; but delay increase rapidly when load exceeds 75%, and some data is discarded because complete starvation.

2.3 Improvement on Realtime of Shared Ethernet

Ethernet is incontestably the most cost-effective solution to support communication in industrial application. However, concerning for real-time traffic, the unpredictable delays and unfairness of station to access Ethernet, such as starvation effect and capture effect, will lead to an unpredictable timing behaviour of its supported application, such as manufacturing and process control, and directly confine its application in such region. Therefore, many improvements have been made to enable Ethernet to support time-constrained communication. These solutions are either by modifying the MAC protocol of Ethernet itself or by implementing an additional deterministic sub-layer over MAC protocol.

The most straightforward solution to guarantee a bounded access time is to use a TDMA (Time Division Multiple Access) strategy, where each station has a pre-allocated time intervals to transmit its data. Therefore, a collision-free Ethernet with a predictable timing behaviour is achieved. Its main disadvantage is the inherent non-flexibility, since even if a station has nothing to transmit, it will "use" its share of time, resulting in a non-used time period. In contrast to station-oriented of TDMA, PCSMA (Predictable CSMA) is data-oriented scheduling in allotting medium. PCSMA demands an off-line traffic scheduling, considering that all real-time data are periodic. While this approach can be collision-free and avoiding waste of medium, it has an overhead inherent to the off-line scheduling. P-CSMA (Prioritised CSMA) is a TDMA-like method based on priority of message. Within P-CSMA, media time is divided in n slots and a station may only transmit its data of highest priority in the corresponding slot. Thus no collision between data of different priorities is guaranteed and the fairness of the medium access is improved. Concerning the collision between data of the same priority, the author only mentions that they may be resolved using a random delay (BEB may be considered).

The goal of providing a collision-free environment can also be implemented through MO (mode operation). Wherein MO, CSMA/CD mode and real-time mode are included in Ethernet, and Ethernet is normally in CSMA/CD mode. If a station needs to send a real-time message, it requests a change to the real-time mode and waits for an acknowledgement from all other stations. Inefficiency arises when these receiving stations still have messages in the back-off phase. Examples of this way can be found in two proposals, RETHER and TEMPRA.

All the above approaches to provide a predictable timing behaviour are essentially collision avoidance techniques. Besides, there are some other approaches focusing on the modification of BEB algorithm to achieve the same target, these are called collision resolution technique. Among them, CSMA/DCR (CSMA with Deterministic Collision Resolution) maybe is the most known one. In CSMA/DCR, the probabilistic BEB algorithm is replaced by a deterministic binary searching tree algorithm. Although CSMA/DCR can provide a bounded medium access delay, it still has the following disadvantages: it can not effectively respond realtime requirement of data because of CSMA/DCR being essentially a station address-based policy; only one data can be sent during the period from a collision occurring to its being solved, thus it leads serious medium access delay when multiple stations exist in an Ethernet. Therefore, many variations of CSMA/DCR are proposed. Among them, CSMA/MDCR (CSMA with modified DCR) allow consecutively sending multiple data through dividing its transmission time of data into two parts, data information and data transmission, where data information indicating the needed time for these data. CSMA/LDCR (CSMA with laxity DCR) implements dynamic binary searching tree algorithm through transmission delay technique, where a data is allowed sending only when its laxity exceeds a given limit. CSMA/LDCR can effectively improve realtime capability

Ethernet because of it essentially being dynamic priority-based.

Although the above approaches can effectively support hard realtime application or soft realtime application, the goal is implemented at the cost of changing basic structure of Ethernet. Besides, adding an additional deterministic sub-layer over MAC protocol of Ethernet and applying proper data scheduling policies can get the same target. Among them, (Virtual Time Protocols), WP (Window Protocol) and TS (Traffic Smoothing) are well known ones. VTP implements data release delay mechanism, which is function of some relevant parameters, usually deadline, laxity or priority. The function is maybe liner or non-liner, and proportional parameter is key factor when liner is adopted. The disadvantage of VTP is lacking of history of data transmission, and that can be improved in some extend by WP. Within WP, a station is allowed sending its data at an instant only the instant is in a time window. The size of time window is dynamically changed depending on the previous medium state: idle, busy or collision. TS statistically bounds the medium access time by limiting the packet arrival rate at the MAC layer (smoothing non real-time traffic bursts).

When medium access delay is only best effort, that means to support soft real-time applications, collision counter technique can also be used, such as CABEB and BLAM.

3 SWITCHED ETHERNET & REAL-TIME

It is known from previous section that a collision exists only if more than two stations send data simultaneously in an Ethernet. It is apparent that the less the collision domain is, the less the collision probability is. One traditional way to decrease collision domain is forming the micro-segments separated by bridges, which have been more and more replaced by switches today. Although functionally a switch can be considered as a multi-ports bridge, in practice a switch is much more powerful than a traditional bridge mainly due to its ASIC based hardware architecture and its ultra rapid simultaneous multiple access memory. Moreover, a switch can have an IP address and as many as MAC addresses as the ports it has, facilitating thus its configuration (remember that a bridge has only one MAC address and is plug & play).

3.1 Switch Architecture

The number of LAN switches continues to proliferate. Since there is no standard for Ethernet switch implementation, there exist now many different kinds of switch internal architectures called switch fabrics. The three main architectures of today are matrix, bus and shared-memory.

Matrix based switch fabric (crossbars) was issued from the telecommunication switches. The interest of this architecture is its great number of ports that neither bus nor shared-memory can currently achieve. But this architecture is more problematic when broadcast or multicast and unicast occur simultaneously. In fact, no unicast can be transmitted when a broadcast or a multicast is taking place.

Bus architecture has a very high-speed core bus (collapsed backbone) shared by modules (i.e., input/output ports). The bus access control is often based on TDMA. An advantage of the bus comparing to the matrix architecture is that the bus architecture naturally supports broadcast traffic. One of the problems of this architecture is the output buffer overflow when many inputs should be forwarded to a same output port.

The third widely used architecture is based on an ultra rapid simultaneous multiple access memory shared by all ports. A data entering to the switch is firstly stored in memory. The data forwarding is performed by ASIC engine which looks up the destination MAC address in the forwarding table, finds it, then sends it to the appropriate output port. Output buffering is used instead of input buffering to avoid the HOL (head-of line) blocking. Output buffer overflow can be minimised by using shared-memory queuing since the buffer size is dynamically adjusted. In fact, all output buffers share a same global memory reducing thus the buffer overflow comparing to the per-port queuing.

All recent Ethernet switches are announced operating with wire-speed and non-blocking. Wire-speed means that all ports of a switch can simultaneously transmit or receive at their full bit rates. This requires that the switch fabric can operate at a bit rate equalling to the aggregate speeds of all the ports. For example, a 24 full-duplex ports fast Ethernet switch needs a fabric forwarding at 4.8 Gbps (24x2x100Mbps). A switch is a non-blocking when it can forward a message to the destination port as long as that port is free, while a blocking one may be not able to forward a message although the destination port is free because of internal conflict in the switch fabric (one example is the HOL blocking in input buffering switches). Switches with output buffering are non-blocking.

Buffering and buffering delay exist in a switch whatever the switch is with full wire-speed or not. In fact message buffering occurs whenever the output port cannot forward all input messages at time. This corresponds to the burst traffic arrival. The analysis of the buffering delay depends on knowledge on the input traffic pattern. The only true no buffering switch is that with each output port bit rate always higher than or equal to the sum of all possible input ports' bit rate! But considering that communication is essentially bi-directional, such a switch should not exist.

3.2 Performance Enhancement

As explained above an Ethernet switch provides one collision domain per port (dedicated bandwidth segment), which allows completely eliminating the collisions if the port is in full-duplex (IEEE802.3x) and only one station is connected to it. In fact in such configuration, CSMA/CD protocol is kept only for ensuring the compatibility with the classic shared Ethernet since no collision is possible.

3.2.1 Bit Rate Improvement

A switch can support different bit rate ports: 10Mbps, 100Mbps, 1Gbps and even 10Gbps and can adapt the bit rate of their ports according to that of the connected

equipment through auto-negotiation function (see http://www.10gea.org and http://www.gigabit-ethernet.org). Nevertheless, when the network operates in half-duplex mode, for maintaining a 200-meter diameter at high speed, one should be aware of the modifications in IEEE802.3u or 100Mbps fast Ethernet:

- The collision window size is reduced from 51.2µs to 5.12µs (and consequently the covered distance is reduced)
- IFS (Inter Frame Spacing) is reduced from 9.6µs to 0.96µs

Furthermore, IEEE802.3z and IEEE802.3ab made the following modification:

- Ethernet slot time is extended to 512 octets instead of 64 octets to keep the collision window size to 5.12µs (which can be a big problem for process control applications in which the most of messages are very small). Note that the minimum frame length of 64 bytes has not been affected since frames smaller than 512 bytes have been automatically augmented with a new carrier extension field following the CRC field.
- Packet bursting (grouping several small frames into on frame) is possible

It is noted that full-duplex devices are not subject to the carrier extension, slot time extension and packet bursting changes. They continue to use the regular Ethernet 96-bit IFS and 64-byte minimum frame size.

3.2.2 Reducing the Forwarding Delay

Besides the data store & forward as like as a bridge does, a switch reduces the data transmission delay by forwarding a frame before it is completely received by the switch (on-the-fly forwarding) as illustrated by the Fig.2.

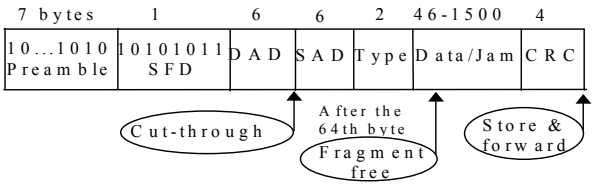

Fig. 2 Forwarding modes in a switch

« Store & forward » mode:

- Waiting until the complete reception of the frame, looking for its destination address DAD, eventually storing it (buffering), then forwarding it to an output port
- Advantage: error detection by CRC calculation
- Drawback: long frame transmission latency, which is function of the data length.

« Cut-through » mode:

- Waiting until the reception of the DAD, then forwarding it to an output port
- Advantage: very short frame-forwarding latency, which is independent of the frame length.
- Drawbacks: risk to transmit erroneous frames for CRC not calculated before forwarding, risk to transmit the small fragments call runt for collisions occurred after the beginning of the forwarding would not be detected.

« Fragment-free » mode :

- Equivalent of the « Cut-through » mode but waiting until the 64th byte before its forwarding
- Advantage: eliminating the small fragments (runts) caused by collisions

« Auto-select » mode (capable of adapting to the transmission quality of the network):

- Beginning by the « store & forward » mode
- If no errors (i.e., transmission is reliable), shifts to « fragment-free » mode
- If always no errors, shifts to « cut-through »
- If the small fragments (runts) appear, shifts to « fragment-free » mode

It is worth noting that on-the-fly forwarding only improves the transmission delay but not the buffering delay when frames have to be buffered in an output buffer. It is not available for broadcast or when the destination is not already memorised in the switch address table or still when the line speed of the input port is different from that of output port (e.g. input port with 10Mbps and output port with 100Mbps).

3.2.3 Congestion Control

Congestion can occur in two cases. One is when several ports forward their traffic to a same output port while the total input traffic is greater than the output port bandwidth, and in this case the sources bit rate should be slowed down to avoid buffer overflow. Another one is when the output port is connected to a shared segment and the available bandwidth left by the traffic of the segment is smaller than the input traffic.

Congestion control can be achieved by a switch to prevent buffer overflow. Two techniques can be used for the two above cases:

- IEEE802.3x “Pause command”: a “Pause command” can be send by the switch to the sender if a port in full-duplex mode receives more traffic than it can handle.
- Back pressure: the switch can simulate a collision by sending to the segment of the output port a jam frame when this port is in half-duplex mode.

The congestion control mechanism, although contributes to enhancement of the performance and the robustness of Ethernet, should be used with precaution since they can make non-deterministic the message response times. Moreover, how they will influence TCP RTT (Round Trip Time) estimation remains to be studied.

3.2.4 Priority Handling

Priority handling is fundamental for achieving real-time communication feature. IEEE802.1p defined a priority field, which is kept within a tag field in IEEE802.1Q in the extended Ethernet frame (Fig.3).

With the 3 bits priority field, eight priority levels can be defined allowing to handling real-time traffic. Unfortunately there is no standard regarding to the number of queues per port a switch should implement. Current

practice is to implement 2 to 4 queues of different priorities with either weighted round-robin scheduling or non pre-emptive fixed priority scheduling (Phipps 1999; Holmeide 2001). Another problem of using the IEEE802.1p priority is that most of stations' NICs (Network Interface Cards) up to now do not support priority.

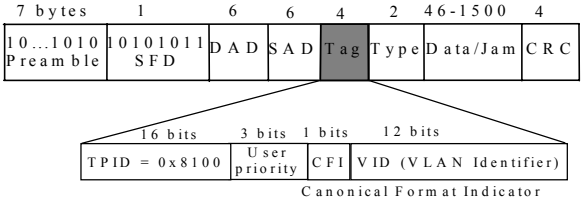

Fig.3 Extended Ethernet frame format with tag (IEEE802.1Q) field

Fortunately many configurable switches can accept the priority based on **MAC address** and/or **source port**. This enables almost all switched Ethernet to handle prioritised traffic even if IEEE802.1p is not deployed. The use of priority mechanism in layer 2, 3 and 4 to achieve real-time performance is given by Holmeide (2001).

3.3 VLAN (Virtual LAN)

The concept of VLAN has been proposed mainly for forming the subnets in a logical LAN by using switches, just as like as one can do with traditional routers and corresponding net masks. In that way a LAN configured in terms of VLAN can:

- Reduce the broadcast traffic by confining it within one VLAN (broadcast domain)
- Enhance the security between different VLANs
- Allow the mobility of stations: an user can always remain in the same VLAN when his physical location is changed

For real-time communication, the first point is the most important since this can further limit the network load. The inconvenience of VLAN is the necessity of using a router for communication inter-VLAN. But a router can seriously decrease the real-time performance unless a layer 3 switch is used.

It is well known that the performance of legacy routers, even if with faster interfaces, is fundamentally bounded by centralised CPU, software-based architectures. However, Ethernet layer 3 switch is a hardware-based router (ASIC: application specific integrated circuits), which provides all functions of a traditional router:

- Determining forwarding path based on layer 3 information (RIP, OSPF, BGB)
- Validating the layer 3 header's checksum
- Verifying packet TTL and its update
- Processing and responding to any option information
- Updating forwarding statistics in the MIB (Management Information Base)
- Applying security controls if required (Access-list, …)

Although hardware-based, a layer 3 switch provides the ultimate flexibility not only in performance, but also in parallel processing, which makes it suitable for implementing new policies (QoS for time-critical applications for example), managing security, load balancing, protocol option processing (Ciampa). A first discussion of using layer 3 switch (especially the ToS field of the IP header defined by Diffserv) for real-time communication is discussed in Holmeide (2001) and will not be detailed in this paper.

The use of the STP (Spanning Tree Protocol) should be avoided as long as possible in a real-time Ethernet because of the additional delays during the STP configuration (switch state moves from blocking to listening, learning then forwarding). Nevertheless for a network requiring more reliability, redundant inter switch links can exist and the STP should be deployed in order to avoid forming loops. In that way, if a forwarding link is broken, STP automatically reconfigures the corresponding switches' blocking ports to find a new spanning tree. STP combined with VLAN can optimise the whole network traffic, improving thus the performance.

3.4 WLAN (Wireless Ethernet)

WLAN consists in an interesting extension for gaining more flexibility. Recent technologic breakthrough now made very easy to integrate wireless Ethernet to a wired Ethernet backbone although the WLAN uses other protocols like IEEE802.11. In fact, what is important for integrating a WLAN to Ethernet is that it must be compliant to Ethernet. This is the case for many commercially available WLAN NICs.

Real-time communication can be extended to the wireless part if there is only one hope between a base station and the mobile ones. Otherwise the WLAN part forms a so-called ad hoc mobile network and the real-time features will be difficulty preserved because of the dynamic routing protocols in ad hoc network. A first discussion on the use of the wireless LAN for factory communication can be found in Cutler (2001).

4 QUALITY OF SERVICE IN SWITCHED ETHERNET

4.1 Frame Transmission Model in Switched Ethernet

One can distinguish a fully switched Ethernet from those including both switches and hubs. In a fully switched Ethernet there is only one equipment (station or switch) per switch port. In case that wire-speed full-duplex switches are used, the end-to-end delay can be minimised by decreasing at maximum the message buffering. A frame travelling through the switches in its path without experienced any buffering has the minimum delay. The total delay introduced by a switch is composed of:

- The *switching latency* (traffic classification according to IEEE802.1p mapping table, DAD look-up and switch fabric set-up time),

- The *frame forwarding latency* which depends on the forwarding mode and eventually on the frame length if the "store & forward" mode is running,
- The *buffering delay* when the frame is queued.

The switching latency is a fixed value, which depends on the switch performance and often provided by the switch vendor (typically about 10 μs). The frame forwarding delay can be obtained knowing in which mode the switch is running (refer to section 2.2). The analysis of the buffering delay depends on the knowledge on the input traffic pattern.

Generally a communication system with n station connected by a full-duplex Ethernet switch of N ports is shown in Fig.4, and correspondingly the frame transmission in the Ethernet switch with output buffering could be modelled as shown in Fig. 5.

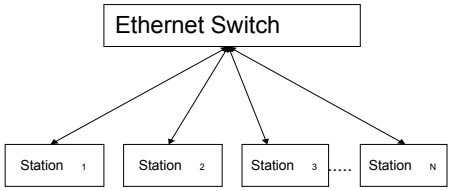

Fig. 4 Stations connected with an Ethernet switch

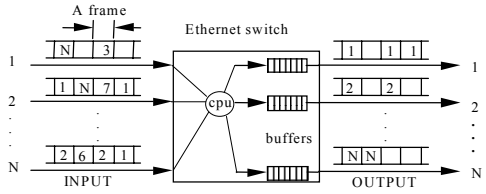

Fig. 5 Model of Frame transmission in an Ethernet switch

4.2 QoS in Switched Ethernet

Applying switched Ethernet in industry applications mainly meet needs of quantity of intra-plant data and inter-plant communication, where real-time network of higher bandwidth is imperative, especially for transmitting a large quantity of data between lower layer and middle layer networks. Normally, in the context of real-time theory, real-time just means guarantee temporal requirement (normally in term of deadline) of a task (frame in network), and the time of a no real-time task is not considered at all. Consequently, two main policies of schedule, guaranteed/best-effort, is adopted. However, in the context of practical engineering, occasional loss of some deadline can be usually tolerated (even in hard real-time systems) even for hard realtime task.

Nevertheless, the term *occasional* is so ambiguous that it has no practical meaning for a specification. The extent to which a system may tolerate missed deadlines has to be stated precisely. The tolerance to missed deadlines cannot be adequately specified by a single parameter, for example with the percentage of deadlines to be met or missed (although it is common practice to do so). To capture this situation another parameter describing the window of time within which the number of deadlines must hold should be specified. For the present, there are mainly five kind of specification for QoS of tasks:

- Hard Real-time with response time< deadline
- skip-over model with skip 1 task of s tasks
- Statistical real-time with P[response time < deadline] > p
- (m,k)-firm with guarantee the deadline of m messages among k consecutive messages
- Windowed lost-rate with constrained lost rate over a finite range of consecutive messages

Fortunately, the above specifications can be described by (m,k)-firm through properly changing parameters of m and k. After specifying the QoS of all frames, the key problem is properly managing these frames according to their states to meet their respective requirement of (m,k)-firm. It is apparent that a frame may experience different states, which can be indicated as a instance model in Fig.6. A frame may experience either failure state or success state based on whether at least m_i packets out of its k_i consecutive packets violates deadline.

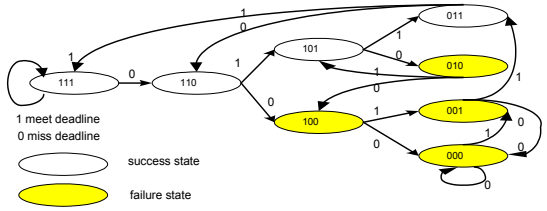

Fig 6 State transition diagram of task with (2, 3)-firm constraint

Considering QoS of (m, k)-firm for frame in switch Ethernet, the frame transmission in an Ethernet switch could be modelled as indicated Fig.7.

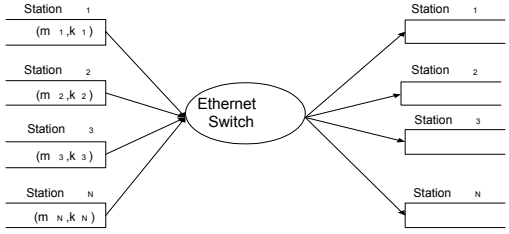

Fig. 7 Model of Frame with (m, k)-firm in an Ethernet switch

4.3 Scheduling Algorithms QoS in Switched Ethernet

From the above section we know that there are different temporal requirement for frames in term of (m, k)-firm, therefore we need scheduling mechanisms to guarantee QoS of to these frames.

Among scheduling mechanisms, DBP (Distance Based Priority) is the most simple, that is motivated by the fact that the more the closer of a task to its failure state, the more the easier the task suffers failure. Within DBP, different tasks are serviced according to their priority, and a task is assigned a priority based on its position of m^{th} meet (position of m deadline meet occurs) among its last previous k instances.

DBP schedule essentially is dynamic priority schedule and belongs best-effort, that means DBP can't provide any deadline guarantee at all. In fact, the goal of any schedule is to effectively manage tasks and distribute proper resource for these tasks. (m, k)-firm constraints is just to abandon some instance of a task if the task can tolerant some deadline miss under system overload. From this sense, IC

(Imprecise Computation), that divides each instance of a task into a mandatory and an optional part, where the latter is rejected when system overload, is same with (m, k)-firm. Combining the ideas from the IC approach and RM schedule, other schedule approach, ERM (Enhanced Rate Monotonic Schedule), is proposed. That is the computation time of each instance of task is divided into a mandatory and an optional part, and the mandatory part of each instance is scheduled according to RM policy, and the optional part each instance is assign the lowest priority and is scheduled according to First Come First Service.

The above schedule algorithms mainly solve the scheduling problem under the situation that only real time tasks with (m, k)-firm exist. In fact, in many real time systems, hard real time task and soft real time task co-exist. Therefore, effectively scheduling soft real time task while guarantee the behavior of hard real time task is becoming a problem, and many techniques has been proposed to the problem. Among these techniques, DPS (dual priority schedule) is an intuitively simple method and lower overhead. The key of DPS is the promotion of priority for hard real time task, combining task (m, k)-firm further complex the problem.

DWCS (Dynamic Window Constrained Schedule) is designed to maximize network bandwidth usage in the presence of multiple packets each with their own delay constraints and loss-tolerance. DWCS schedules packets for transmission based on the current values of their loss-tolerance and deadlines, where precedence is given to the packet at the head of the stream with the lowest loss-tolerance.

4.4 Problem of Application of (m, k)-firm in Switch Ethernet

It is obvious from the above algorithms that there are the common characteristics:

- How to assign priority for the instances selected as mandatory.
- How to select partition algorithm among k consecutive instances of a task.

Therefore, we need to know the formal description of the problem of (m, k)-firm and whether there is an *optimal* partition between mandatory and optional instances?

Further in many actual applications of switch Ethernet, the distributed function of realtime tasks often communicate among multiple switch, and require realtime guarantee, that means end-to-end realtime guarantee. To properly utilise switch Ethernet resource and guarantee end-to-end requirement of tasks, the following problems have to solved:

- How to design end-to-end deadline and(m, k)-firm of a frame along the switches over which the frame transfer
- How to automatically adapt local deadline and (m, k)-firm for a frame according to its history state and current situation ?

5 CONCLUSION AND FUTURE WORK

The main contributions of this paper resides in on the one hand having given a comprehensive analysis of the real-time features offered by a shared and switch Ethernets, and on the other hand in the QoS and related problems for switch Ethernet. The analytics indicates switch Ethernet can provide realtime guarantee for realtime applications in industry region, however many works still need be done to effectively utilise switch Ethernet by making use of temporal requirements of tasks (end-to-end response time guarantees both HRT and SRT in term of (m, k)-firm).

6 REFERENCES

1) A. Sahoo, B. Devalla, Y. Guan and W. Zhao, Adaptive Connection Management for Mission Critical Applications over ATM Networks, *International Journal of Parallel and Distributed Systems and Networks*, Vol.3(2), pp.51–63, 2000

2) C.C. Chou, K.G Shin, Statistical Real-Time Channels on Multi-access Bus Networks, *IEEE Transaction on Parallel and Distributed Systems,* Vol.7(8), 769-780, 1997

3) D.W. Pritty, J.R. Malone, S.K. Banerjee and N.L. Lawrie, A real-time upgrade for Ethernet based factory networking, *IECON'95*, pp1631-163, 1995.

4) H. Kopez, Real Time System Design Principles for Distributed Embedded Application, *Lower Academic Publishers*, 1997.

5) Holmeide, O. and T. Skeie, VoIP drives realtime Ethernet *The industrial Ethernet book*, Issue 5, pp26-29, Spring 2001.

6) H. Hansson and M. Sjödin, Response time guarantees for ATM-networked control systems, *WFCS'97*, pp213-222, Barcelona, 1997

7) G. Bernat, A. Burns, Weakly Hard Real-Time Systems, *IEEE Trans. on Computer,* Vol.50(4), pp.308-330, 2001

8) J. Turiel, J. Marinero, J. González, CSMA/PDCR: A Random Access Protocol Without Priority Inversion, *IEEE Conference on Industrial Electronics Society*, pp.910-915, 1996

9) K.G. Shin, C.C. Chou, Design and Evaluation of Real-Time Communication for Fieldbus-Based Manufacturing Systems, *IEEE Transaction on Robots and Automation*, pp.357-367,1996

10) L.C. Miguel and J.P. Thomesse, Fieldbuses and Real-Time MAC Protocols, *IFAC International Symposium on Intelligent Components and instruments for Control Applications,* 2000.

11) M. Hamdaoui and P. Ramanathan, A Dynamic Priority Assignment Techniques for Stream with (m, k) Firm Guarantees, *IEEE Trans. on Computer*, Vol.44(12), pp.1443-1451, 1995

12) N. Malcolm and W. Zhao, Hard Realtime Communication in Multiple-Access Networks, *Journal of Real Time Systems*, Vol.9(1), pp.75-107, 1995

13) N. Audslay, A. Burns, and A.J. Welling, Applying New Scheduling Theory to Static Priority Pre-emptive Scheduling, *Software Engineer Journal,* Vol.8(5), pp284-292, 1993

14) Song, Y.Q. and F. Simonot, Messages scheduling in FDDI for real-time communication", *RTS&ES'96*, pp287-301, France, 10-12, 1996.

15) S.K Kweon, K. G. Shin and G. Workman, Ethernet-based Real-time Control Networks for Manufacturing Automation Systems, *International Symposium on Manufacturing with Applications,* 2000

16) W.lindsay and P.Ramanathan, DBP-M, A Technique for Meeting end-to-end (m, k)-firm Guarantee Requirements in Point-to-Point Networks, *IEEE Conference on Local networks,* pp.294-303*,* 1999.

17) W. Zhao, J. Stankovic, and K. Ramamritham, A Window Protocol for Transmission of Time Constrained Messages, *IEEE Transactions on Computers*, Vol.39(9), pp.1186-1203, 1990

18) Z. Wang, Y.Q Song, J.M. Chen, Y.X Sun, Worst-Case Response Time of Aperiodic Message in WorldFIP Fieldbus, 15th IFAC World Congress, Barcelona, Spain, 2002

19) Z. Wang, Y. Zhou, Y.X Sun and T.R. Wang, How to Improve Critical Realtime Traffic of Aperiodic Message in FF Fieldbus, *IFAC conference on New Technology for Computer Control*，2001

20) Z. Mammeri, and K.Haouam, Connection allocation schemes for guaranteeing hard real-time communications with ATM network, *WFCS'97*, pp.203-212, Barcelona, 1997

21) Venkatramani, C. and T. Chiueh, Supporting real-time traffic on Ethernet, *IEEE RealTime Systems Symposium*, pp282-286, 1994 .