A New Scheduling Algorithm Based on Traffic Classification Using Imprecise Computation

Farzad Abtahi, Sahar Khanmohamadi, and Bahram Sadeghi Bigham

Abstract—Wireless channels are characterized by more serious bursty and location-dependent errors. Many packet scheduling algorithms have been proposed for wireless networks to guarantee fairness and delay bounds. However, most existing schemes do not consider the difference of traffic natures among packet flows. This will cause the delay-weight coupling problem. In particular, serious queuing delays may be incurred for real-time flows. In this paper, it is proposed a scheduling algorithm that takes traffic types of flows into consideration when scheduling packets and also it is provided scheduling flexibility by trading off video quality to meet the playback deadline.

Keywords—Data communication, Real-time, Scheduling, Video transport.

I. INTRODUCTION

A real-time system is one whose correctness involves both the logical correctness of the outputs and their timeliness. There are many characterizations for ''real-time system'' and the term is often used ambiguously because real-time systems have such differing time constraints. For example, some real-time systems only need the application to satisfy average time deadlines, with small variability, when processing external events. This might be the case when controlling mixing or heating processes or in the case of display processes. Critical real-time systems, on the other hand, have very strict time deadlines that must be met every time, for example, if a controlled dose of radiation must be delivered to some sample. The application for a critical real-time system must have sufficient time to process an external stimulus, called the response time, within a predetermined value under all possible circumstances [13].

What distinguishes real-time systems are classified as hard, firm, or soft systems. In hard real-time systems, the most critical type, failure to meet its time constraints will lead to system failure (imagine a nuclear reactor control system). In firm real-time systems, the time constraints must be met in most cases, but can tolerate missing a low number of deadlines (consider a food processing plant control system). In soft real-time systems, the performance is degraded when time constraints are not met but the system will not catastrophically fail (this is often the case with real-time animation of visual displays).

The real-time traffic is sensitive to delay. When some real-time data is lost during transmission, the lost data will not be retransmitted [11]. Beside real-time systems behave under predefined time constraints, interact with its environments. The development cycle include specification, verification and testing phases. Specification phase define the expected functionalities of the systems. Verification and testing aim checking the system properties in order to make sure of its correctness. Verification works on the system model or specification while testing is faced to the system implementation. Verification analyses the model on general and it can prove some properties on it. Testing is based on the observation of an implementation actually working. Both testing and verification are fundamental steps of communication system design since they contribute to the development of reliable and good quality systems. Standardization issues constitute another important aspect of system design. In order to facilitate the interoperability and the interchangeability between the different parties involved in the telecommunication industry, reference standards as well as methodology frameworks are defined by international institutions [7].

The engineering of time-critical systems poses significant challenges to their correct specification, design and development. In such systems, the time, at which each input is processed or output is produced, is critical. It has been recognized that the use of formal methods, in the development of such systems, is fundamental if ''correctness'' is to be assured (Typical applications can be found in avionics, robotics, process control, and healthcare). The use of formal approaches increases our understanding of a system by revealing inconsistencies, ambiguities, and incompleteness that might otherwise go undetected. An important aspect in the development of a time-critical system is how to cope with its ''evolution''. The evolution of a software system could be due to changes in the original requirements, adopting a different hardware platform or to improve its efficiency [10].

The recent increase in the number of soft real-time applications has necessitated the development of protocols which provide real-time services. The large bandwidth requirement of these applications has given rise to the use of the optical fiber with Wavelength Division Multiplexing (WDM) supporting a number of channels (wavelengths) in

Farzad Abtahi is with Department of Information Technology, Advanced Basic Sciences University, Zanjan, Iran (corresponding author to provide phone: +98 9123411288; fax: +98 241 4152182; e-mail: f_abtahi@iasbs.ac.ir).

Sahar Khanmohamadi is with Department of Information Technology, Advanced Basic Sciences University, Zanjan, Iran (e-mail: s_khanmohamadi@iasbs.ac.ir).

Bahram Sadeghi Bigham is with Department of Information Technology, Advanced Basic Sciences University, Zanjan, Iran (e-mail: b_sadeghi_b@iasbs.ac.ir).

World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:9, 2008

parallel at each node in the network as opposed to the traditional copper wire which provides only a single channel for transmission. Each node in this physical configuration has a limited number of tunable transmitters. Over the last few years, the tuning speed of the transmitters has decreased from millisecond tuning times to microsecond and in fact, even nanosecond tuning times [12].

Real-time transport of either live or stored video is an important component of many real-time multimedia systems. It differs from the traditional network applications (e.g. emails, web, etc.) in terms of reliability, bandwidth, and time requirements. A real-time video transport system has a minimum bandwidth requirement, and tight constraint on the end-to-end delay (i.e. time-sensitive), and is tolerant to certain amount of packet loss. If the amount of bandwidth is not available, the transport system needs to either reduce data sending rate or give up since receiving half of the needed bandwidth is of no use to such a bandwidth-sensitive application. Every data packet must arrive at the receiver's end system in time in order to be decoded and displayed because the video must be played out continuously and a packet that arrives beyond the time constraint is useless and considered lost [16].

Real-time communication protocols are an important component in DRES for industrial applications in order to ensure a safe and timely operation. Fiedlbus protocols such as Profibus, Foundation Fieldbus, DeviceNet and CAN widely adopted and well established at field/shop floor level. Ongoing efforts in extensions to these industrial communication protocols have shifted from low level aspects (physical and data link layer standards) to the definition of higher-level automation objects, such as Profinet mechatronic objects or CIP application layer objects. At the same time, Ethernet has also been considered for use in real-time applications, either in the industrial domain or in large embedded systems. Attractive factors include wide availability, high bandwidth and low cost. The use of Ethernet-based communication protocols should also enable an easy integration to realize the access to data in various layers of an enterprise information system [1].

The rest of this paper is organized as follows. Related work is discussed in Section II. In Section III, two scheduling algorithms, their benefits and limitations have been explained. Then, it has been tried to improve TD-FQ algorithm by partitioning streaming media into two subtasks.

II. RELATED WORKS

Traditional scheduling algorithms and analysis methods have focused on transactions with hard deadlines, and they are based on the knowledge of the activation period and worst case execution time (WCET). Once this information is provided, some classical analysis methods are available [9].

Many research results exist on smoothing. Salehi et al. proposed a work ahead smoothing technique which achieves the greatest possible reduction in rate variability when transmitting stored video from a server to a client across a network. Grossglauser et al. argued that a renegotiated service best addresses the presence of burstness over multiple time-scales and design the Renegotiated Constant Bit Rate (RCBR)

service for carrying compressed video traffic. However, rate smoothing cannot be used for VBR video over the best-effort network with bandwidth variation. Layered video multicast has been discussed to accommodate the heterogeneity of receivers. Due to the delivery deadline constraint and the limitation of the available network, not all layers can be transmitted. The number of layers transmitted is dynamically varying according to the available network bandwidth. The streaming server will send the data to client through network as many layers as possible when the available bandwidth is large, and will truncate some layers when the available bandwidth is small. However, frequent adding and dropping of layers can incur significant quality variability which is annoying to human eyes [4]. Gao et al. proposed real-time scheduling algorithm but did not take into according smoothing scheme.

Multimedia applications, such as teleconferencing and video on demand service, require large bandwidth for periodic media transmission. However, a common network based on TCP/IP such as the Internet cannot guarantee the requested bandwidth which is necessary for real-time periodic media transmission. One approach to deliver multimedia traffic in a dynamic network environment is via buffer handling. Many researches have been done in this area and the current buffer management techniques are generally classified into two groups: dynamic quality adjustment and static smoothing. The dynamic quality adjustment (it is called DQA hereafter) technique is to adapt the changes in the network load by scaling media quality to fit the available bandwidth. The static smoothing (it is called SS hereafter) technique is to smooth the media's bandwidth requirement by preloading data into client buffer adequately [3].

Hojung Cha et al. proposed a bandwidth adaptive smoothing (it is called BAS hereafter) technique [3] which is adaptive to the dynamic networks as in DQA and also supports the network bandwidth smoothing as in SS.

Gürbüz and Owen proposed a dynamic scheduling algorithm for radio resources, which supported the real-time traffic for QoS provisioning, with the help of power assignment and code hopping. A resource scheduler was implemented in the base station. It collects requests from all mobile users. The real-time traffic in a radio access network have priority serving mechanism and have a higher bit rate than can non-real-time traffic [2].

As far as we are aware, this problem of ordering the download of segments of scalably-coded coded videos in P2P networks to maximize the viewer's real-time viewing experience has not been addressed before. The work that most closely relates to our work is by Ross et al. where the authors study strategies for dividing available bandwidth between the base-layer and enhancement-layer of a two-layered stream. They conclude that heuristics which take into account the amount of data remaining in the pre-fetch buffer outperform static divisions of bandwidth. They also conclude that maximizing just the overall amount of data streamed causes undesirable fluctuations in quality, and provide additional heuristics that produce a smoother stream while minimally reducing the overall quality.

World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:9, 2008

Some on-line algorithms have developed and evaluated in [8] that coordinate the pre-fetching of scalably-coded variable bit-rate video. These algorithms are ideal for P2P environments in that they require no knowledge of the future variability or availability of bandwidth, yet produce a playback whose average rate and variability are comparable to the best off-line pre-fetching algorithms that have total future knowledge.

The introduction of the wireless LAN standard IEEE 802.11 stimulated the development of many prioritized carrier sense multiple access (CSMA) protocols. Some of these protocols changed parameters in the IEEE 802.11 standard to be a function of deadlines. These techniques have two drawbacks: they only approximate priority scheduling; it may happen that a high-priority message has to wait for one or many lower-priority messages; and collisions can occur, hence causing deadline misses. Other prioritization protocols based on IEEE 802.11 use black-bursts. Black-bursts work as follows. If the channel is idle, then a node transmits a message immediately. Otherwise, the node waits until the channel becomes idle and transmits a black-burst (a jamming signal) for a time duration that is proportional to the priority. When a node finishes transmitting its jamming signal, the node listens to find out whether or not other nodes transmit a jamming signal. If so, the node did not have the highest priority, and so, it waits until the channel is idle again. All these black-burst schemes have the drawback that the maximum length of the black-burst is proportional to the number of priority levels. Therefore, only a small number of priority levels can be supported. Another technique, not based on IEEE 802.11, is to implement prioritization using two separate narrowband busy-tones to communicate that a node is backlogged with a high-priority message. This technique has the drawback of requiring specialized hardware, which requires extra bandwidth (for the narrow band signals), and it supports only two priority levels. The IEEE 802.11 standard also defined another MAC protocol where a base station polls a node and gives it the right to transmit in a time interval. Naturally such an approach is inefficient to schedule sporadic messages. Recently, the IEEE 802.11e profile was introduced with the intention of offering better support for quality-of-service. The previous proposed approaches of choosing back-off times as a function of priorities were adopted, and the polling scheme in IEEE 802.11 was refined with traffic classes. MAC protocols have also been proposed from the real-time systems community with the goal of meeting deadlines. Some protocols use tables [sometimes called time-division multiple-access (TDMA) templates] with explicit start times of message transmission. It is also conceivable to use a TDMA template designed before runtime and use it to schedule wireless traffic. However, all these timetable approaches have the drawback of requiring that sporadic message streams are dealt with using polling, which, as previously stated, is inefficient. Another approach, implicit EDF, is based on the assumption that all nodes know the traffic on the other nodes that compete for the medium, and all these nodes execute the EDF scheduling algorithm. Unfortunately, this algorithm is based on the assumption that a node knows the arrival time of messages on other nodes, and this implies that polling must be used to deal with sporadic message streams. The conclusion of this section is that several prioritization protocols and real-time scheduling algorithms exist, but they do not efficiently solve the problem of sporadic scheduling in wireless networks [6].

The MAC layer scheduling for delay-sensitive variable bit rate (VBR) streaming video in the IEEE 802.15.3 WPANs has been considered in [14]. Because devices are battery powered, energy efficiency is a critical concern in designing the MAC scheduling. Furthermore, video quality should be as good as possible.

Since the erroneous flows are compensated in a first-come-first-served manner, real-time lagging flows may still suffer from long queuing delay. Effort-limited fair (ELF) suggests to adjust each flow's weight in response to the error rate of that flow, up to a maximum defined by that flow's power factor. However, since the scheduler does not have immediate knowledge about the error rates of a flow, there could be some delay in adjusting its weight to respond to its channel and queue condition. Besides, when a real-time flow just exits from errors, it is emergent to deliver packets for the flow, or these packets may be dropped. Unfortunately, adjusting weights cannot guarantee higher priorities for such flows [15].

An imprecise computation workload model is adopted and a real-time scheduling algorithm for scalable streaming media is proposed in [5]. The scheduling task of a stream is partitioned into two subtasks: the mandatory subtask for the base layer and the optional subtask for the enhancement layers. The workload model and scheduling algorithm improve the utility of the bandwidth and smooth the playback quality reconstructed in client by determining how to select and transmit the packets subject to a given time.

III. IMPROVEMENT ON TD-FQ ALGORITHM USING IMPRECISE COMPUTATION

In this section, two real-time scheduling algorithms is explained, and their benefits and limitations. The first scheduling algorithm is TD-FQ algorithm that takes traffic types of flows into consideration when scheduling packets. The second scheduling algorithm is real-time scheduling based on imprecise computation that provides scheduling flexibility by trading off video quality reconstructed in client to meet the playback deadline.

At the end, it is tried to improve TD-FQ algorithm by partitioning streaming media into two subtasks: the mandatory subtask for the base layer and the optional subtask for the enhancement layers, as used in real-time scheduling based on imprecise computation.

In TD-FQ, real-time and non real-time flows can be scheduled through one scheduler called TD-FQ scheduler. Below, firstly the system model and basic operations of TD-FQ have been introduced.
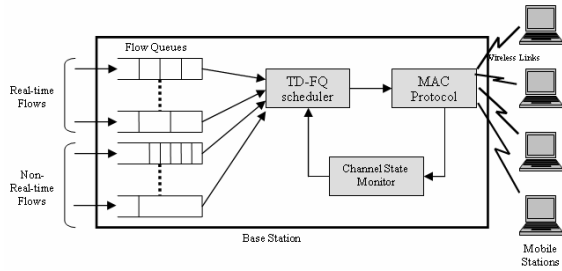
World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:9, 2008

Fig. 1 The system architecture of TD-FQ [15]

A packet-cellular network is considered as in Fig. 1. Traffics arriving at a base station (BS) are mixed with real-time and non-real-time flows. TD-FQ develops mechanisms to reduce queuing delays of real-time flows by giving them higher priorities. Nevertheless, TD-FQ guarantees that the special treatment of real-time flows will not starve non-real-time flows. Thus, it still maintains fairness and bounded delays for all flows.

Packets arriving at a BS are classified into real-time traffic and non-real-time traffic and dispatched into different queues depending on their destination mobile stations. These traffic flows are sent to the TD-FQ scheduler, which is responsible for scheduling flows and transmitting their head-of-line (HOL) packets via the MAC protocol. The Channel state monitor provides information about the channel state of each mobile. For simplicity, it is assumed that BS has immediate and accurate knowledge of each channel's state.

In TD-FQ, mobile stations may suffer from bursty and location-dependent channel errors. However, error periods are assumed to be sporadic and short relative to the whole lifetime of flows so that long-term unfairness would not happen.

The second scheduling algorithm called scalable video streaming system, proposed a real-time scheduling algorithm based on imprecise computation workload model for delivery of scalable streaming media, which can be adapted to network status and QoS requirement over the best-effort Internet.

Fig. 2 shows streams that are decomposed into two parts by the FGS encoder: the base layer and the enhance layers. The scheduling task of a streaming media is partitioned into two subtasks: the mandatory subtask M for the base layer and the optional subtask O for the enhancement layers. The mandatory subtask M is required for an acceptable QoS and must be scheduled before the deadline. The optional subtask O refines the result. It can be left unfinished and terminated at its deadline, if necessary, lessening the playback quality at the client end. Furthermore, the optional subtask O is dependent on the mandatory subtask M; the mandatory subtask M must execute before the optional subtask O.
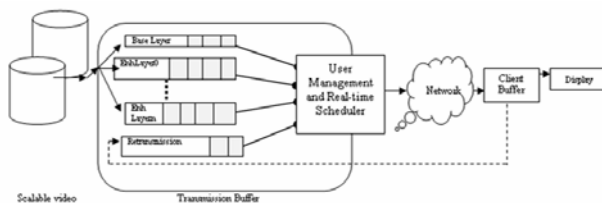
The imprecise computation scheduling algorithm combines importance of different layer to reconstruct the playback quality and real-time scheduling algorithm. Thus, it is guaranteed that the most important data (i.e. the base layer sub stream) can be transmitted before the deadline. The scheduling algorithm not only improves the utility of the bandwidth but also smoothes the playback quality.

It is assumed that in TD-FQ which mentioned first long-term unfairness would not happen. It is attempted to improve the TD-FQ to work better in bad channel status. In cases the data sent is non real-time, the receiver can send NACK to request the packet retransmission. In real-time streams, if data not received or received with error, it is impossible to send it again. So receiving real-time data in good time interval is more important than non real-time services. For reliable transmitting real-time data in TD-FQ every time, the new TD-FQ algorithm is proposed that use the base layer and the enhance layers as mentioned in the imprecise computation scheduling algorithm for each flow queue in TD-FQ, as shown in Fig. 3. In this case, the reliability of real-time data transmission has been improved in TD-FQ in bad channel status.
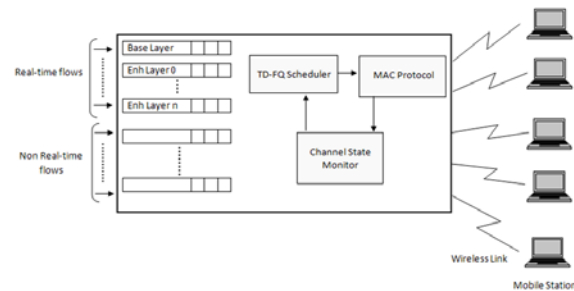

Fig. 3 A new TD-FQ scheduling algorithm based on imprecise computation

The new TD-FQ algorithm in compared with the former algorithm have improved in terms of reliability, utility of bandwidth, confidentiality in most important data transmission and playback quality smoothness.

- **Confidentiality**. As the new TD-FQ algorithm can transmit the base layer data in a bad channel status, it is expected that the most important data (in base layer) is transmitted.
- **Reliability**. In the new TD-FQ algorithm it is attempted to increase the probability of the most important data transmission, the reliability of data transmission have improved.
- **Utility of bandwidth**. By dividing each real-time flow into two subtasks and send them to a base layer and enhance layers queues, a partial amount of bandwidth that remains useless in the former algorithm can be utilized.
- **Playback quality smoothness**. No rapid degradation in data transmission led to playback quality smoothness.

Fig. 2 The architecture of FGS video streaming system [5]

World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:9, 2008

IV. CONCLUSION

In TD-FQ, the mobile stations may suffer from bursty and location-dependent channel errors. However, error periods are assumed to be sporadic and short relative to the whole lifetime of flows, so that long-term unfairness would not happen. In this paper, we proposed a new scheduling algorithm that improves reliability in real-time transmission. In cases that the sent data is non real-time, the receiver can send NACK to request retransmission of the packet. On the other hand in the real-time transmission, if the real-time data is not received or is received with errors, we cannot send it again. So receiving real-time data in a good time interval is more important than non real-time services. In order to always transmit real-time data in TD-FQ reliably, we can use the base layer and the enhance layers as mentioned in the imprecise computation scheduling algorithm for each flow queue in TD-FQ.

REFERENCES

[1] Carlos Eduardo Pereira, Luigi Carro, 2007, "Distributed real-time embedded systems: Recent advances, future trends and their impact on manufacturing plant control", Elsevier, Annual Reviews in Control 31 (2007), Universidade Federal do Rio Grande do Sul (UFRGS), Brazil, pp. 81-92.

[2] Chiang-Shiang Wan, Wei-Kuan Shih, Ruei-Chuan Chang, 2003, "Real-time packet scheduling in next generation radio access system", Elsevier, Computer Communications 26 (2003), Hsinchu, Taiwan, pp. 1931-1943.

[3] Hojung Cha, Jaewook Kim, Rhan Ha, 2003, "Bandwidth constrained smoothing for multimedia streaming with scheduling support", Elsevier, Journal of Systems Architecture 48, Seoul, South Korea, pp. 353-366.

[4] Kui Gao, Wen Gao, Simin He, Yuan Zhang, 2005, "Real-time smoothing for network adaptive video streaming", Elsevier, J. Vis. Commun. Image R. 16, Beijing, China, pp. 512-526.

[5] Kui Gao, Wen Gao, Simin He, Yuan Zhang, 2004, "Real-time scheduling based on imprecise computation for scalable streaming media system over the Internet", Elsevier, Real-Time Imaging 10, Beijing, China, pp. 307-313.

[6] Nuno Pereira, Björn Andersson, Eduardo Tovar, 2007, "WiDom: A Dominance Protocol for Wireless Medium Access", IEEE Transactions on Industrial Informatics, Vol. 3, No. 2, pp. 120-130.

[7] O.Kone, 2002, "Conformance testing to real-time communications systems", Elsevier, Computer Communication 25, Nancy, France, pp.32-45.

[8] Raj Kumar Rajendran, Dan Rubenstein, 2006, "Optimizing the quality of scalable video streams on P2P networks", Elsevier, Computer Networks 50, New York, United States, pp. 2641-2658.

[9] Samy Semghouni, Laurent Amanton, Bruno Sadeg, Alexandre Berred, 2007, "On new scheduling policy for he improvement of firm RTDBSs performances", Elsevier, Data & Knowledge Engineering 63, Le Havre Cedex, France, pp.414-432.

[10] Shikun Zhou a, Hussein Zedan b, Antonio Cau, 2005, "Run-time analysis of time-critical systems", Elsevier, Journal of Systems Architecture 51, Portsmouth, United Kingdom, pp. 331-345.

[11] Show-Shiow Tzeng, 2006, "Resource allocation for real-time and non-real-time traffic in wireless networks", Elsevier, Computer Communications 29, Kaohsiung, Taiwan, pp. 1722-1729.

[12] S.J. Kowshik, S. Babu, G. Manimaran, C.S.R. Murthy, 2001, "A distributed real-time MAC protocol for DM-based LANs", Elsevier, Computer Communications 24, Urbana, USA, pp. 654-666.

[13] Walter Cedeno, Phillip A. Laplante, 2007, "An Overview of Real-Time Operating Systems", JALA February2007, pp. 40-45.

[14] Xi Chen, Yang Xiaob, Yu Cai, Jianhua Lu, Zucheng Zhou, 2006, "An energy diffserv and application-aware MAC scheduling for VBR streaming video in the IEEE 802.15.3 high-rate wireless personal area networks", Elsevier, Computer Communications 29, Beijing, China, pp. 3516-3526.

[15] You-Chiun Wang, Shiang-Rung Ye, Yu-Chee Tseng, 2005, "A fair scheduling algorithm with traffic classification for wireless networks", Elsevier, Computer Communications 28, Hsin-Chu, Taiwan, pp. 1225-1239.

[16] Zefeng Ni, Zhenzhong Chen, King Ngi Ngan, 2005, "A real-time video transport system for the best-effort Internet", Elsevier, Signal Processing: Image Communication 20, Singapore, Singapore, pp. 277-293.