Multi-channel MAC Protocol for Wireless Real-Time Communication

Master's Thesis in Computer Systems Engineering

Syed Hasan Yousuf Naqvi

Supervisor: Magnus Jonsson
Multi-channel MAC Protocol for Wireless Real-Time Communication

Master’s thesis in Computer Systems Engineering

School of Information Science, Computer and Electrical Engineering
Halmstad University
Box 823, S-301 18 Halmstad, Sweden

October 2010
Preface

This master’s thesis, entitled Multichannel MAC Protocol for Wireless Communication, is written to fulfill the requirement of master’s degree in computer system engineering at Halmstad University, Sweden. I gained a lot from my supervisor, Magnus Jonsson, who supported me a lot and I benefitted from his experience. I would like to acknowledge his support when describing the protocol. I am very thankful to him for his comments and guidance in this master’s thesis report.

I will like to extend my sincere gratitude to my parents, my sisters, my wife and my friends who always supported and encouraged me during my stay and study in Sweden.

Syed Hasan Yousuf Naqvi
Halmstad University, October 2010
Abstract

Embedded systems have become more complex and it is hard to predict the behavior of network due to its dynamic nature. Several devices can interact to perform highly sophisticated real-time tasks while having the demand of interaction and sharing of resources. The interacting components in these systems have strict timing constraints which require time critical communication.

We have designed the DCC-EDF protocol for wireless communication using the motivation from the research done in optical communication. We have chosen dedicated control channel protocol because it does not require time synchronization. To provide real-time services, we will use the Early Deadline First scheduling algorithm (EDF) because of its optimality and efficiency. The performance for the soft real-time traffic is analyzed through simulation.
Table Contents

PREFACE .................................................................................................................................................................. I

ABSTRACT ............................................................................................................................................................ II

TABLE CONTENTS .................................................................................................................................................. III

LIST OF FIGURES ................................................................................................................................................ IV

1. INTRODUCTION.................................................................................................................................................. 2
   1.1. MOTIVATION .................................................................................................................................................. 3
   1.2. REAL-TIME COMMUNICATION ......................................................................................................................... 4
   1.3. PROBLEM DESCRIPTION.................................................................................................................................. 5
   1.4. RESEARCH GOALS AND APPROACH............................................................................................................. 5

2. REAL-TIME SYSTEMS......................................................................................................................................... 6
   2.1. INTRODUCTION ............................................................................................................................................. 6
   2.2. REAL-TIME SCHEDULING ............................................................................................................................... 7
   2.3. CLASSIFICATION OF REAL-TIME SCHEDULING ........................................................................................... 7
   2.4. EALIEST DEADLINE FIRST SCHEDULING ALGORITHM ................................................................................. 8

3. MULTI-CHANNEL MAC PROTOCOLS FOR WIRELESS NETWORKS ................................................................. 9
   3.1. DEDICATED CONTROL CHANNEL PROTOCOL .............................................................................................. 9
   3.2. COMMON HOPPING ......................................................................................................................................... 9
   3.3. SPLIT PHASE .................................................................................................................................................. 10
   3.4. PARALLEL RENDEZVOUS: ............................................................................................................................ 11

4. MULTI-CHANNEL MAC PROTOCOLS FOR OPTICAL NETWORKS ........................................................................ 14
   4.1. MULTI-CHANNEL MAC PROTOCOL FOR WDM-BASED LANS .................................................................. 14
   4.2. EDF-BASED PROTOCOL WITH PASSIVE STAR AND SEPARATE CONTROL CHANNEL ....................... 15
   4.3. DYNAMIC TIME-DETERMINISTIC TRAFFIC IN A FIBER-OPTIC WDM STAR NETWORK .......................... 16

5. PROTOCOL DESIGN: ....................................................................................................................................... 18
   5.1. KEY FEATURES AND DESIGN ...................................................................................................................... 18
   5.2. LIMITATION OF WORK ................................................................................................................................ 19
   5.3. TERMINOLOGY AND ASSUMPTIONS ........................................................................................................... 19
   5.4. REQUIRED DATA STRUCTURES FOR THE PROTOCOL ............................................................................... 21

6. REAL-TIME ANALYSIS .................................................................................................................................... 23

7. SIMULATION ANALYSIS PARAMETERS, CONFIGURATIONS, AND RESULTS.............................................. 27
   7.1. SIMULATION ANALYSIS PARAMETERS ......................................................................................................... 27
   7.2. CONFIGURATION FOR SIMULATION ............................................................................................................. 28
   7.3. RESULTS ....................................................................................................................................................... 29

8. SUMMARY .......................................................................................................................................................... 38
   8.1. FUTURE WORK .............................................................................................................................................. 38
   8.2. CONCLUSIONS ............................................................................................................................................ 38

9. REFERENCES ...................................................................................................................................................... 39
List of figures

Figure 1: Spectrum of real-time applications ................................................................. 3
Figure 2: Example of a control network ................................................................. 4
Figure 3: Working of multi-channel Dedicated Channel protocol ................................... 9
Figure 4: Working of multi-channel Common Hopping protocol ..................................... 10
Figure 5: Working of multi-channel Split phase protocol ........................................... 11
Figure 6: Working of multi-channel Parallel Rendezvous protocol .............................. 12
Figure 7: Procedure to follow a home sequence or deviate from it ............................... 12
Figure 8: Transmission cycle of multi-channel MAC Protocol for WDM-based LANs .... 15
Figure 9: Topological diagram of EDF-Based protocol with passive star and separate control Channel .................................................................................................................... 16
Figure 10: A Receiver cycle is partitioned into data slots and control slots, where the control slots are received one cycle in advance related to the data slots that they are carrying information about data .............................................................................................................................. 17
Figure 11: DCC EDF protocol architecture ............................................................. 18
Figure 12: Data flow ................................................................................................. 19
Figure 13: Delay at each node .................................................................................. 24
Figure 14: Dependence of average queuing delay on traffic intensity ......................... 26
Figure 15: Graph between No of RT channels & average delay in [us] (case 1) ............. 29
Figure 16: Graph between no of RT channels and Utilization in percentage (case 1) .... 30
Figure 17: Graph between no of RT channels & deadline miss ratio (case 1) .......... 31
Figure 18: Graph between no of RT channels & Average delay [us] (case 2) ............... 32
Figure 19: Graph between no of RT channels and average delay [us] (case 3) .......... 33
Figure 20: Graph between no of RT channels & Utilization [%](case 2) ...................... 34
Figure 21: Graph between no of RT channels and Utilization [%] (case 3) ................. 34
Figure 22: Graph between no of RT channels & deadline miss ratio (case 2) .......... 36
Figure 23: Graph between no of RT channels and deadline miss ratio (case 3) ............ 36
1. Introduction

An enormous number of technological advancements in electric devices and electric based transmission have made it possible for devices to communicate at higher speeds. So, a huge amount of data can be transmitted at high transmission rates over long distances and with low latency and minimum errors.

There are different types of networks based on different mediums of communication, like wired networks, wireless networks and optical networks etc. Depending upon network architecture and the use of the network, the medium of communication is selected. Advancements in communication have made it possible to develop networks having multiple devices which share the data among them. Today, wireless networks are commonly used in the industrial sector and embedded networks are used in a wide range of different applications. Examples of such applications include cluster computing, environment monitoring, radar signal processing, surveillance, streaming video and military use etc.

In future, industrial and embedded systems will constitute many subsystems [1] which require real-time communication among subsystems with reliability and performance. So, the timing requirement will be big factor involved and they will have tight deadlines.

In wireless networks, it is easy to communicate and discover other network devices using the same channel (frequency) because devices have to decide only the time slot to avoid the possibility of collision in data transmission. However, in industrial and embedded networks, where large number of devices exists, it is not easy for multiple devices to communicate using the multiple channels due to factors like collision, data congestion, corruption in data, latency etc. In wireless network, transmission channels can be shared by different devices using the random access method, or scheduling the channel on user requests. The amount of bandwidth per device available is decreased as the numbers of devices are increased. This results in a high demand for bandwidth [3] [10] per device.

Throughput of networks can be enhanced by the use of multiple parallel channels. Several studies have been made using multiple channels. Current studies have proved that multiple channels can be used for communication, and good results can be achieved. Different versions of the multi-channel MAC (Medium Access Control) protocol have emerged to communicate based on their general principles of operation, like dedicated control channel, common hopping, split phase and parallel rendezvous.

In wireless networks, the communication over the medium can be affected by many factors, like weather, electrical signals etc. However in optical networking, a more robust channel is provided. There is great use of WDM (Wavelength Division Multiplexing) and TDM (Time Division Multiplexing) multiplexing techniques in optical networks to achieve higher transmission rates with minimum error rates. In WDM multiplexing, rays of light with different wavelengths are allowed to travel on one channel without generating error in the information being transferred and where TDM multiplexing always considers the time for any certain channel divided into different
slots. To find out which time slot belongs to which channel, different kinds of assignment algorithms are used.

A device, named “passive star”, is used with the WDM multiplexing technique to transport information from one device to another device. It splits all input light signals to all output lines, so that every output line transfers all input lines’ information. It ensures the data travel from all input channels to all output channels without collisions and data loss.

1.1. Motivation
As we have seen a rapid growth of new applications/systems based on real-time embedded systems, such applications have increased the requirement for a reliable network to facilitate these applications to run without any errors. Examples of such systems are automobiles’ engines, avionics and consumer electronics. Such systems consist of many sub systems. So, to complete a single task by these systems, all subsystems have to interact with each other. The output of such systems depends upon the correctness of all their sub systems. In many cases, RT (real-time) systems require sharing of information between subsystems, which should be done in a timely and deterministic manner.

While designing an RT system, it is necessary to apply time constraints on the system to ensure its correct working. Different applications require different timing constraints.

Consider an example of a radar system sending information to a central node; if the message from the radar to the central node is delayed, it can have severe effect because, on the receipt of the message from the radar, an urgent and prompt task has to be initiated and, due to delay in the message, that task cannot be fulfilled. So, a timing constraint (deadline) is important for such systems and these types of systems are called “hard real-time systems”. In other applications like video conferencing or video streaming, where messages are transmitted over the network and timing constraints (deadlines) can influence system performance and quality of service can be retarded at the user end, such systems are called “soft real-time systems”. In some systems, both hard real-time and soft real-time deadlines exist. Other applications, where users download information from the internet, do not have such timing constraints; such systems are called “non real-time systems”. Figure 1 [2] shows the spectrum of real-time applications.

![Figure 1: Spectrum of real-time applications](image)

The work in this thesis is motivated by the need to give efficient support in wireless, real-time systems. It is important to mention that we are only dealing with real-time services. I have drawn all the figures used in the thesis.
1.2. Real-time communication

Basic Concepts

Communication protocols play an important role in a distributed, real-time system where different tasks are generating different signals to other devices in a distributed system via a network. Each different device has its own processor i.e., each task is governed by a processor on a device or other devices. To complete one task, one or many devices may play their roles. However, the important part is the timing constraint. In hard real-time systems, communications delays (delay due to transmission medium and delay while transmitting from source and destination) must be bounded.

The network load and the priorities of other traffic have great impact on the performance of a real-time system which can lead to great delays. In many cases, delays occur randomly, which causes jitter in the system. Delays can cause tasks to miss relative deadlines and increase the time of execution of a task to be completed. Therefore, while designing real-time protocols, it is ensured that the protocol should provide real-time message delivery. So, protocols should have following properties.

- Low jitter
- Low latency
- Easy integration with non real-time traffic
- Adaptability to changing network and traffic conditions
- Good performance
There are number of other properties which should be considered while describing communication protocol, such as network topologies, switching strategies, and traffic characteristics.

1.3. Problem description

The purpose of this project is to come up with a new, multi-channel MAC protocol for wireless communication targeting industrial and embedded networks based on the study of multi-channel MAC protocols, both for wireless and optical networks. The primary protocol in this project will be based on a multi-channel MAC protocol for wireless communication, but enhancement will be carried out by getting inspiration from multi-channel MAC protocols for optical networks.

Different protocols, both for wireless and optical communication, are referred to find out the best suitable protocols for a given network topology, and also to evaluate them for real-time services.

1.4. Research goals and approach

The goals of this thesis work are:

- To study the existing multichannel MAC protocols, both for wireless and optical networks.
- To define a multichannel MAC protocol for wireless networks, targeting real-time performance for industrial and embedded networks.
- To implement a simulator and evaluate the protocol.

The research was started by doing a survey of multi-channel MAC protocols for both wireless and optical communication for real-time systems. For scheduling, we also studied different types of scheduling algorithms and used EDF non-pre-emptive scheduling in our proposed protocol.
2. Real-time Systems

Jonsson and Kunert in [1], presented a framework to use ARQ (Automatic Repeat Request) to enhance the performance of a network in hard real-time systems with the help of real-time worst case scheduling analysis focused on industrial and embedded systems in single hop domain. In this paper, they tried to focus on reducing the message error rate by retransmitting erroneous packets, as long as they remained meaningful with respect to deadlines, and enhancing the reliability of the system at the cost of some utilization penalty. Former work on real-time multi-channel wireless networks is presented in [22] and [23] while other multi-channel networks are presented in chapter 3 (Wireless and optical networks) the work presented in this thesis is influenced by the fore mentioned papers especially [23] but with extra focus on soft real-time support. Below, a general introduction to real-time systems is given.

2.1. Introduction

Real-time systems are the systems which can perform several types of computational tasks, fulfilling the timing constraint to complete the tasks. There are two major factors which make the system real-time. One is the correctness and second one is the timing constraint at which results were drawn from the computations. We can find several examples of real-time systems, like radar control systems, aircraft avionics, flight control systems and space shuttle control systems [13]. Mostly, embedded systems are real-time systems which are designed to perform controlled tasks for a system. If we take the example of fighter jets participating in an air show, this provides an example of real-time system. Pilot of a jet will be controlling the jet as “controller” and jet is controlled by the pilot as “controlled process” where other jets and space is environment. During the air show, pilots have to fly the jets according to the program which restricts them to performing in time to co-ordinate with other jets. During the flight of the jet, the performance of the jet depends upon several factors, like the power of the engine, the weather, and timely actions by the pilots. These include some real-time actions and some non real-time actions. To take off, it is important to move the ailerons under the wings of the jet by the controlled instruction in specific manner to lift the jet up. While flying the jet, the pilot performs several real-time and non real-time tasks. The difference between a real-time task and a non real-time task is that a real-time task always has a deadline to complete the task in time.

A real-time system can be represented by a set of real-time tasks. These tasks can be described in two ways: by the predictability of their arrival and by the consequences of a task not being able to be executed before its deadline.

**Periodic and aperiodic real-time task** A task is said to be periodic if, and only if, it is activated again and again, with a regular interval of time (period), and a task is said to be aperiodic if, and only if, it is unknown when it will be activated again. So, for periodic tasks, we are sure that, after a fixed time, it will be activated again and, for aperiodic task, we remain uncertain about its activation.
**Hard real-time and soft real-time task**  A task is defined as a hard real-time task if it can lead to serious failure of a real-time system when it misses the deadline to complete the task, while for soft real-time systems, it is not important to meet the deadline constraint for a task.

In many applications, both hard and soft real-time tasks are used. Thus, a real-time system should be designed to handle both hard and soft real-time tasks [4] with different strategies. There are other important factors, apart from time, to be considered while designing a real-time system. i.e., precedence constraint and exclusive resource.

- **Precedence:** Precedence constraint specifies that all sub-tasks $\psi_n$ of a task $\psi$ should be executed in a certain order. e.g., if the task $\psi_x$'s output is input for task $\psi_y$ so we can say that task $\psi_y$ is preceded by $\psi_x$.

- **Exclusive resource:** In a processor, all tasks require some time to execute on the processor and it is also possible that the task may share other resources, like bus or memory. There is quite a possibility that a shared resource does not allow simultaneous access, but requires mutual exclusion among participating tasks called exclusive resource.

### 2.2. Real-time Scheduling

We can distribute real-time scheduling in two categories, one is uni-processor real-time scheduling and the other is multi-processor real-time scheduling. In uni-processor (single processor), the processing of tasks is easier, whereas in a multi-processor, it is comparatively difficult to schedule a task and distribute it. In multi-processing real-time scheduling, we keep not only trace of the timing constraint, but also the processor which is handling the task.

Real-time scheduling solves not only the problem of timing constraint, but also specifies the processor in multi-processor domain. Real-time scheduling is the hot topic today and much excellent work has been presented in literature.

In real-time scheduling, scheduling tasks is always considered so that timing constraints for the specific tasks will be fulfilled. This solves the problem of allocating resources. If we consider the uni-processor (single processor), we mean that each task should be allocated a single time slot to fulfil the timing requirement and, of course, there will be only a single processor.

### 2.3. Classification of Real-time Scheduling

Real-time scheduling can be classified in following ways.

- **Static**  
  Static real-time scheduling algorithms are those algorithms which assign the tasks before their activation based on fixed priorities.

- **Dynamic**  
  Dynamic real-time scheduling algorithms are those algorithms which assign the tasks as soon as they get activated, based on dynamic priorities, or it can be said that they assign the tasks during runtime.
- **On-line**
  If the decision of scheduling is made on the arrival of a task to the system by the scheduling algorithm, such a scheduling algorithm is called “on-line real-time scheduling algorithm”.

- **Offline**
  If the scheduling decisions are made in advance before the operation of a system. Such a scheduling algorithm is called an “off-line real-time scheduling algorithm”.

- **Preemptive**
  In preemptive real-time scheduling algorithms, a task can be interrupted by any higher priority task.

- **Non-preemptive**
  In non-preemptive real-time scheduling algorithms, a task can not be interrupted by any higher priority task and it will keep running until completion of the task.

While designing real-time systems, different scheduling algorithms can be used. If we follow time-driven scheduling algorithms, we need to slice the time slots in multiple fixed sized time slots so each time slot can be assigned to the task on the basis of their rate and timing requirement. These kinds of algorithms are not able to handle dynamic situations, for example, they cannot decide after the arrival of a new task. They lack the intelligence to handle such situations.

### 2.4. Earliest Deadline First scheduling algorithm

The EDF scheduling algorithm is a task based scheduling algorithm which means it schedules the task on its absolute deadline. The tasks having shortest deadline will be entertained first and the task which have longest deadline will be entertained at last. EDF can provide feasible schedule on a single processor and its optimality is proved in [17] with a certain condition. i.e., deadline is equal to period. Later, the research showed that EDF can also provide feasible schedule for the tasks having arbitrary deadlines.
3. Multi-channel MAC Protocols for Wireless Networks

For multi-channel MAC protocols, there are two main issues which should be entertained. i.e., channel assignment and medium access [18]. It is important to choose a particular channel for a particular device so that devices share information and then we have to address to contention/collision problem.

3.1. Dedicated Control Channel Protocol

Basic Principle of Working:

In this approach, every device in the network uses two radios. One radio is tuned to control channel and the other is tuned to data channels. All devices use to share control information on the control channel only. So, all devices know about the business of every device and the data channel to be used for communication with the busy time slot. Examples of such protocol can be found in [19] [20].

In figure 3, channel 0 is the dedicated channel while channel 1, 2, 3 are data channels. This protocol works like this; if a device wants to send data to another device, first it sends RTS (request-to-send) on the control channel specifying the availability of lowest numbered channel. In the reply, the receiver sends CTS (clear-to-send) on the control channel and it acknowledges the number of the channel on which communication will take place. There is an exchange of more information, including the duration of transmission for data packets, network allocation vector etc.

![Figure 3: Working of multi-channel Dedicated Channel protocol](image)

Advantages:

The major advantage of dedicated channel is, it does not require time synchronization. We can promote/demote the control channel to be used as a data channel while other nodes are busy.

Disadvantages:

With a lower number of channels, it is harder to perform efficiently. It is a costly protocol that requires dedicated radios and dedicated channels.

3.2. Common Hopping

Basic Principle of Working:
In common hopping approach, devices use only one radio. A pair of devices stops hopping as soon as they have an agreement to share the data on a common hopping channel [19]. As soon as transmission ends, they rejoin the common hopping pattern.

In figure 4, channel 0, 1, 2, 3 are channels for devices to hop and find out their perspective transmitter / receiver. Let us imagine there are many devices hopping on different channels and two of them want to communicate and share some data. The sending device will try to send RTS (request to send) to the receiver on the common channel and then receiver replies CTS on the same channel. So, they will stop hopping and start communicating on the common channel. After completing packet transfer, they start their pattern to hop on different channels.

Advantages:

Data packets can be transferred using all channels. While data is being transmitted, it is also possible to send data to busy devices using other, non busy channels.

Disadvantages:

In this approach, to make a rendezvous, switching from one channel to the other channel is quite fast, so we cannot neglect channel switching time.

3.3. Split Phase

Basic Principle of Working:

The split phase uses two different kinds of phase session. In figure 5, two types of phases are shown. One phase is only restricted to the control phase where different devices manage to make agreements for data transfer while, in the other phase (data phase), different devices transfer data.

If device ‘A’ wants to transfer data to device B, firstly it has to send control data to device ‘B’ on the control. Then, device B replies to sender A, so they agree on co-operation for data packet transfer in coming phase of data transfer.

In the second phase, i.e. the data transfer phase, they communicate so that data transfer can be accomplished using the idle channel x (x is any lowest ID channel available). However in the meantime, when they get agreed on data transfer [1] [20], they cannot have other commitments for data transfer which can create any conflict with the earlier commitments.
Advantages:

It uses one radio per device.

Disadvantages:

It requires time synchronization among all devices although hopping frequency is even less than common hopping strategy.

3.4. Parallel Rendezvous:

Basic Principle of Working:

To get the basic working of parallel rendezvous protocol, we need to understand the working of SSCH (single seeded channel hopping) and McMAC protocols.

The first approach, which is followed by the SSCH, can be explained as follows.

Parallel rendezvous protocol implements the node’s channel hopping schedule, and schedules the packets within each channel. It keeps informed and updated about the channel hopping schedule to the neighboring nodes and changes in traffic pattern.

SSCH protocol uses the novel techniques [11] optimistic synchronization and partial synchronization. Optimistic synchronization keeps the control channel traffic distributed across all channels, whereas partial synchronization allows the forwarding node to synchronize partially with the source node to destination node.

In SSCH, each device can follow many hopping sequences generated with the help of seed of a pseudorandom generator. Each device can follow multiple sequences (as an example in figure 6, 4 channels are available for device to hop) in a time-multiplexed manner. Let us imagine a device A wants to send some data to device B, so A will wait until B comes to the same channel. If the frequency of sending data by A is very high, then device A adopts one or more of B’s sequences by spending time on the same channel [1].

![Figure 5: Working of multi-channel Split phase protocol](image)
In McMAC, each device hops on all available channels, it should not hop previously hopped channel. This prevents devices from re-aligning repeatedly. The hopping sequence is generated randomly with the help of a seed provided by the device. So, the selected sequence is called the “home hopping sequence”. During the data phase, a device can deviate from its home sequence [12]. In figure 7, the procedure is shown where a device can follow or deviate from its home sequence.

Whenever a sender send packets, each packet contains some information like the seed of the sender, current index of the hopping sequence and the remaining time until the next hop. So, by having this type of information, a receiver can easily guess about the upcoming hopping sequence of the sender. A sender always sends an empty packet if there is no data packets available to transmit for a longer time.

**Advantages:**
There is no control channel bottleneck. The device can get agreement by learning the hopping sequences and using several time slots to transfer data.

**Disadvantages:**
If hopping frequency increases, it is difficult to track down the mutual agreement for data transmission.
In [3], multichannel scheduling for ‘IEEE 802.15.3 networks’ is presented. The scheduling algorithm described uses the ultra wideband frequency to increase the efficiency of data transmission for the wireless devices. UWB satisfies the condition of limitation of the radiated power spectral density to 41 dBm/MHZ, which enables UWB devices to work simultaneously with narrowband systems and, hence, spectrum reusability increases. They studied the scheduling mechanism which uses multiple channels available in UWB based networks and they used discrete event simulation, and the system parameters set by them includes varied number of channels, network traffic and channel bandwidth. The scheduling mechanism they used has the capability to differentiate between different devices by assigning slots to their respective access connection. Their results showed the increase in the throughput by using the multiple channels simultaneously.
4. Multi-channel MAC Protocols for Optical Networks

In [6], the authors presented the analysis of how different properties vary with the design especially where high bandwidth is required is presented. They focused on passive optical star cluster with fixed-wavelength transmitters and tunable receivers. They used the WDM and TDMA to increase the bandwidth utilization & to deal with time-deterministic latency. Analysis is done on MAC protocol and concluded that the applications suggested TD-TWDDMA protocol is well suited to radar signal-processing.

4.1. Multi-Channel MAC Protocol for WDM-based LANs

The basic idea of this protocol is based on its two phases, i.e. the demand notification phase and the packet transmission phase. It uses the scheduling algorithm, i.e. round robin in demand notification phase to collect demands of each node by using a token on each node. Demands notifications are then stored in the local status table of each node. The important feature of this protocol is the usage of tokens circulating between the nodes for the collection of demands of each node only, and not for the contention resolution. For contention resolution, the protocol runs a special heuristic [7] in a distributed manner, which is called the “contention resolution heuristic”.

In figure 8, for a particular channel Z, node ‘a’ has control of the control channel and its tuned transmitter is transmitting its data packets for fixed slot for a specific time. The contention heuristic is run before the node ‘a’ is about to release the control of the particular channel Z. The heuristic provides a facility to other nodes to be able to get control of the specific channel and use the channel for their transmission. The illustration of an example is shown in the diagram below, that heuristic node ‘b’ has won the control for the particular channel Z by using contention resolution heuristic.

The basic idea of the heuristic can be described as follows:
1. Determine the extra time for each transmitter to get tuned to that channel.
2. Determine the transmitter which has the least tuning time to a particular channel.
3. Preference should be given to transmitter if the transmitter is already tuned to a channel and has the same extra tuning time.

In [7], they focused on parameters for the protocol, like the number of nodes, no of channels, no of transmitter per node, propagation delay, channel utilization, real-time traffic, and deadlines.
4.2. EDF-Based protocol with passive star and separate control Channel

This protocol can be described as “reservation based protocol”. It was designed keeping in mind the network MAN (metropolitan area network) which can cover tens of kilometers. For MAN, it is generally considered that the number of nodes is greater than the number of channels available in the network. So, to accommodate the network traffic, it uses a distributed algorithm and keeps re-scheduling the network traffic and sorting the requests by their global priority. They used POS coupler (shown in figure 7), which provides excellent support [9] for WDM, cheap scalability and broadcasting capacity.

This protocol uses early deadline first algorithm and the TWDM (time-wavelength division multiplexing) technique. WDM is an efficient way of using higher bandwidths of light signals, whereas TDM provides/facilitates the avoidance of collisions in the network traffic [8][9]. In this protocol, the number of data channels will be one greater than the nodes because of the control channel as normally resides in reservation based protocols. In general situations, number of data channels will be less than the number of nodes in the network. On the control channel, there will be one fixed transmitter and one receiver tuned to the control channel. On data channels, there will be one tunable transmitter, and one tunable receiver which can be tuned to different wavelengths.

Channel management is done by using TDM. Every N slot makes cycles and N will be the number of nodes in the network. So, in cycle of a node can send a control packet through the control channel. Once a node receives a control packet, it will wait until all the data from current slot has been transferred. To provide interruption, inter node distances must be collected in the matrix which will prioritize the data accordingly.

When a new request comes with a higher priority and with the same destination as the one being transmitted, the previous request will be interrupted and prioritized data will be transmitted and the request which is interrupted will be transmitted later. Interruption is possible so that it can reassemble itself to provide real-time services.
4.3. Dynamic Time-Deterministic Traffic in a Fiber-Optic WDM Star Network

General Features of the protocol:

This protocol uses FT-TR (fixed wavelength transmitters – tunable receivers). Every node in the network has one fixed wavelength transmitter and one tunable Receiver which makes the network single hop network. It has the functionality [5] to reserve time slots according to the demand for messages. It uses a distributed slot-allocation algorithm to avoid collisions.

The working of the protocol is based on three steps.

- Every node transmits a control slot.
- Distributed slot-allocation algorithm is run separately by each node.
- Data slots are transmitted or received by nodes.

Figure 10 shows how cycles are divided into control slots and data slots.

Real-time services:

The protocol has a matrix which can determine that every message which seeks a guaranteed deadline will get the guarantee slots available. If there are no more guaranteed slots available, the owner of the message will be directed to handle the situation so that the required slots can be decremented by the number of slots for that element and, in this way, it provides the facility to the message to reassemble. For best effort messages, the earliest deadline first algorithm is used. A special algorithm [5] is run to transmit control slots. Details on the real-time protocols are found in [6].
Figure 10: A Receiver cycle is partitioned into data slots and control slots, where the control slots are received one cycle in advance related to the data slots that they are carrying information about data.
5. Protocol Design

Here we are presenting a design of a multi-channel medium access control (MAC) protocol focused on embedded and industrial networks to provide real-time services i.e. whenever a packet is required to be transferred in real-time, it should be transferred with top-most priority using EDF scheduling.

5.1. Key Features and design

![Figure 11: DCC EDF protocol architecture](image)

Each entity in the figure 11 is described below:

1. RT channel is described in 5.3.
2. Source nodes who have their local queues sorted using EDF.
3. Global queue is a logical queue where all the nodes share their information and it is also sorted using EDF.
4. Physical channel is a real channel available between the source node and destination node.
5. Destination nodes in the network.
Features:

a. It provides better channel utilization using multiple channels.
b. Our protocol ‘DCC-EDF’ will utilize full bandwidth i.e. in event of failure of any node or any withdrawal of a packet request, the network can call the successor node to utilize the unused bandwidth.
c. DCC-EDF provides a real-time service guarantee by using EDF scheduling algorithm.

5.2. Limitation of work

I would like to acknowledge the limitation of work for simplification.

1. Consider a network architecture in a single hop domain.
2. Consider channels have identical channel capacity.
3. Consider that the network is free from hidden node problems.
4. Channel switching penalty is negligible.

5.3. Terminology and Assumptions

Consider a single hop network having periodic traffic flows (also called logical real-time channels, RT channels) denoted by $\psi_i$. Each traffic flow is defined by the following parameters: sending node $TN_i$, receiving node $RN_i$, period $P_i$, message length $L_i$, and deadline $D_i$. Now we can define each RT channel using the following expression.

$$\psi_i = \{TN_i, RN_i, P_i, L_i, D_i\}$$
A real-time channel $\psi$ can be defined as:

“A RT channel is an abstraction of a traffic flow over a link or a network, where resources have been allocated to guarantee a certain minimum throughput and a bounded end-to-end delay.”

Maximum packet size in terms of time [us] is denoted by $\theta$.

Each node will be equipped with the following radio units:

- Two fixed control radios will be listening or sending RTS/CTS, and both will be tuned to the control channel frequency.
- Two tunable data radios will be listening or sending data channel and they can be tuned to different frequencies.

We are using EDF for network traffic scheduling and we prioritize the data packets by their deadline times.

A message can be a sequence of contiguous packets or a single packet. So, a message can be represented by $P_j(t_s, t_f, i)$, for $i = 1, 2, \ldots, k$, where $t_s$ represents start time of the packet line, $t_f$ is the finish time of the packet series and $i$ represents the transmitting node.

<table>
<thead>
<tr>
<th>Node A</th>
<th>Node B</th>
<th>Node C</th>
<th>Node D</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>d</td>
<td>Ts</td>
<td>c</td>
</tr>
<tr>
<td>70</td>
<td>10</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>70</td>
<td>10</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>70</td>
<td>40</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>70</td>
<td>40</td>
<td>11</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 1: EDF Scheduling

Abbreviations:
- d: deadline
- p: period
- c: capacity
- Ts: transfer sequence
- R: receiving node

In Table 1, some data is mentioned for a scenario to give an overview of working of the protocol. Suppose, there are some RT channels which are feeding the four sources with some messages. In this example, we are considering all messages have same length and equal to two. So, each message in this example has two packets. Each node has a transmitter and a receiver which means, it has the capacity to transfer one packet and receive one packet. It cannot receive two or more packets at the same time and visa versa. Transmission time for a packet in this example is assumed to be 1 [us]. Generation time for all messages is also assumed to be 0 [us].
It is also assumed to use two physical channels for the transmission. P1 for node A will be first packet to be transferred with transfer sequence id “1” on physical channel 1 because the packet P1 for node A has the shortest deadline see table 2 for transmitting sequence. Now node A is busy and it cannot send 2nd packet on second physical channel. Node B is also busy as it is receiving the packet from node A. Node A has another packet P2 with same deadline and it cannot transmitted because there are conflicts as it has the same sender and receiver. So node B is invited to transmit its first packet with transfer sequence id “2”. So, P1 for node B will be transmitted because it has the shortest deadline among all nodes.

Now there will be contest among all packets and selection will be made for that packet which has the shortest deadline. So, P2 of node A will be selected and transmitted using physical channel one with transfer sequence id “3”. After this, P2 of node B will be selected and transmitted using physical channel two with transfer sequence id “4”.

Now the nodes C and D have the packets with shortest deadline. Only P1 of node C can be transmitted on physical channel 1 and at the same time physical channel 2 will remain vacant as node C is have transmitting conflict and node D cannot transmit its packet because node A is having receiving conflict. So, P1 of node C, P1 of node D and P2 of node C will be transmitted one by one using only one physical channel. But with P2 of node D and P3 of node A will use both channel as the same time and so on.

<table>
<thead>
<tr>
<th>Time Line</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical Channel 1</td>
<td>A</td>
<td>A</td>
<td>C</td>
<td>D</td>
<td>C</td>
<td>D</td>
<td>D</td>
<td>B</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>Physical Channel 2</td>
<td>B</td>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td>A</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Nodes transmitting sequence

Table 2 describes only the transmitting sequence and it does not show any delay due to conflicts.

5.4. Required data structures for the protocol

- **Table of Request**
  In each node, there will be a request table which keeps track of the information regarding the requests waiting or being processed in the network by a node.

- **Table of state of nodes**
  The state of a node will determine whether a node is sending any information or not. If a node is active and sending, this means it is sending information to some other node, so it will have information about the node to which it is communicating.

- **Table of free channels**
  This table keeps track of available data channels for data transmission channel DC (data channel) in any specific time slot.
d. **Table of channel owners**
   The channel owner is the list of requests which occupy each channel for data transmission for a given time i.e., every request in the network needs a channel to be fulfilled.

e. **Table of channel demanders**
   This table contains the entries of those requests which are still waiting for the channel acquisition to be processed.
6. Real-time analysis

In this chapter, we will define the some common parameters used by real-time systems to establish real-time channels and then we will conduct delay analysis. To accomplish delay analysis, we assume that all real-time channels have the highest priority and then the network efficiency will be analyzed to guarantee the real-time services.

Real-time parameters

In real-time traffic, we guarantee a certain number of packets to be delivered within the specific time limit. To find out whether the timing constraint is going to be fulfilled or not, we need to understand the real-time traffic flow which can be described by the following characteristics [1].

Traffic parameter 1: Source address (Sa)
Traffic parameter 2: Destination address (Da)
Traffic parameter 3: Period (Pt)
   It can be defined as the time interval between the message releases.
Traffic parameter 4: Deadline (Dt)
   It is the maximum time required to deliver the packet.
Traffic parameter 5: Traffic volume or message length (Cm)
   It shows the total number of bits for all packets in a message including header.

Definition: Periodic logical real-time channel (Ti)
Periodic logical real-time channels are the virtual channels which generate messages regularly with constant intervals of time. i.e., source node release messages with fixed intervals of time. Transmission of these messages depends upon certain factors. i.e., period, message length (message can contain several small message within its own entity) and end-to-end relative deadline.

Real-time communication services

In real-time communication, where applications are given importance due to their services, different applications require different type of services, depending upon their usability. For packet switched embedded networks [16], we can classify the types of services provided by applications in three classes:

- Deterministic
- Probabilistic
- Predicted services

If a service provided is deterministic then, for an application, it is necessary to serve the packets to achieve real-time performance, even in worst case. Thus, it can be concluded that a deterministic service guarantees zero loss.
Delay analysis:

Consider an example of a simple packet, in packet switched network, which is allowed to travel from host (source) to destination. On the way, it may pass through several nodes or routers to reach the destination in multi-hop networks. In our case, we have a single hop network and a packet also suffer from different types of delays. The common delays are described below.

![Figure 13: Delay at each node](image)

- **Processing delay**

  Processing delay, \( D_{\text{proc}} \) is defined as the time consumed in recognizing the packet header and locating the direction in which the packet will be transferred. Processing delays are typically in the order of microseconds or less.

- **Queuing delay**

  Queuing delay, \( D_{\text{queue}} \) is defined as the time consumed in releasing the packet from the source node towards the destination node. e.g., if a node has some packets to be transmitted from the source node towards the destination node, then the originating packet will suffer a queuing delay and, if there is heavy traffic, then it has to wait longer. On the other hand, if there is no packet waiting to be transmitted, then the originating packet will not suffer any queuing delay and this will result in zero queuing delay. Queuing delays are typically of the order of microseconds or milliseconds.

- **Transmission delay**

  In our case, we are dealing with EDF (early deadline first), so packets will be transmitted using the early deadline first scheduling algorithm. Assume the length of the message
from source node to destination node is \( L \), and the rate by which the source node is generating messages to be transferred to destination is \( R \). So, the transmission delay will be \( \frac{L}{R} \). The transmission delay, \( D_{\text{trans}} \) is normally of the order of microseconds to milliseconds.

- **Propagation delay**

  Once a packet is pushed onto the channel, it requires some time to propagate from source node to destination node. The time required to propagate from source to destination is called “propagation delay”. So, the propagation speed depends upon the physical medium of the link (medium can be air, optical fiber, twisted pair cable etc). Propagation delay, \( D_{\text{prop}} \) can be calculated by dividing the distance between two communicating devices by propagation speed. So, if ‘\( d \)’ is the distance and ‘\( s \)’ is the speed, propagation delay is written as \( \frac{d}{s} \).

**Difference between Transmission and Propagation Delay**

It is important to understand the difference between transmission delay and propagation delay. Transmission delay is the time required for a node to push out a packet. This delay deals with packet length and transmission rate and is not concerned with the distance between the nodes. So, we can say that transmission delay is the function of packet length and transmission rate. On the other hand, propagation delay depends upon the distance, so it can be said that it is the function of distance between the two communicating nodes.

**Delay by a node**

If \( D_{\text{proc}}, D_{\text{trans}}, D_{\text{queue}} \) and \( D_{\text{prop}} \) is the processing delay, transmission delay, queuing delay and propagation delay, respectively. Then, the total delay can be expressed by the mathematical equation as follows:

\[
D_{\text{node}} = D_{\text{proc}} + D_{\text{trans}} + D_{\text{queue}} + D_{\text{prop}}
\]

In our case, \( D_{\text{proc}}, D_{\text{trans}} \) and \( D_{\text{prop}} \) will be almost negligible because we are dealing with embedded networks in single hop domain. However we will have some queuing delay because we are considering a small network having nodes with limited resources. So, the queue will be finite. Queuing delay is affected by the packet length. It can vary from packet to packet. For example, if 20 packets arrive at queue when the queue is empty, for the first packet there will be no queuing delay, but further coming packets will have increasingly longer queuing delays. The last packet will have a relatively large queuing delay. Therefore, we use statistical methods (like average queuing delay, variance of queuing delay, probability) to represent queuing delay.

**Traffic Intensity:**

Traffic intensity is expressed as follows

\[
\text{Traffic intensity} = \frac{\text{Avrate} \times L_{\text{packet}}}{R}
\]
Where ‘Av\text{rate}’ is the average rate at which traffic reaches the queue and ‘L\text{packet}’ is the number of packets in a message and ‘R’ is the transmission rate at which packets are pushed out from the queue. Traffic intensity plays an important role in estimating and diagnosing the scope of the queuing delay.

It is found that, in some networks, the arriving of packets to queue is quite random. We are restricting our analysis to only periodic data.

The average queuing delay becomes infinite if traffic intensity is greater than 1. That means intensity of incoming traffic is more towards queue compared to outgoing traffic away from the queue.

\[
\text{Avrate} \times \frac{L\text{packet}}{R} \leq 1
\]

If traffic intensity approaches 1, this shows the queuing delay will grow exponentially large, so the small amount of increase in the traffic will result in a much larger queuing delay percentage-wise, whereas, if traffic intensity approaches to 0, it shows the average queuing delay is small see in figure 13.

Suppose N packets arrive concurrently. When the first packet will reach the queue there will be as such no queuing delay but, on the arrival of second packet, there will be delay of \(\frac{L\text{packet}}{R}\) and if we try to calculate for n-1 packet. Queuing delay will be \(( (n-1) \frac{L\text{packet}}{R} ) \) [us]. So, the average queuing delay can be expressed as follows:

\[
\frac{L\text{packet}}{R} + 2 \frac{L\text{packet}}{R} + 3 \frac{L\text{packet}}{R} + \ldots + (n-1) \frac{L\text{packet}}{R} = (n-1) \frac{L\text{packet}}{R}
\]

A deeper theoretical delay analysis is not given in this thesis. Instead, a simulation study of soft real-time performance is given in the next chapter.
7. Simulation analysis parameters, configurations, and results

In this chapter, first I describe the analysis parameters and why those parameters were selected to analyze the described protocol. The configurations are important to justify the simulator and then I will describe the results taken.

7.1. Simulation analysis parameters

The following parameters are used to analyze simulation results. By keeping an eye on these parameters, one can make a guess regarding the different changes occurring in the network performance and reliability. i.e., if average delay is high so we expect that there will be higher deadline miss ratio etc.

Average delay:

Average delay is the common parameter to analyze the network behaviour. If the average delay is big, this means that there will be large amount of packets waiting in the waiting queue. If there is rapid increase of average delay there will be rapid increase in the size of waiting queue. Average delay presented here is the average of delays for the number of simulations runs. To compute delay for a packet, I used the formula written below.

\[ \text{Delay [us]} = \text{simulation-time at reception} - \text{generation-time} \]

Utilization:

Utilization is the ratio between the numbers of packet received who meet their deadlines and total number of possible packets received.

\[ \text{Utilization [%]} = \left( \frac{\text{packet duration} \times \text{successful received packets}}{\text{simulation length} \times \text{no of physical channels}} \right) \times 100 \]

Deadline Miss Ratio:

Deadline miss ratio is the ratio between the number of packets who missed their deadlines and the total number of packets received. Deadline miss ratio is extremely important for real-time systems to check their reliability and accuracy. It becomes more critical, if we talk about hard real-time systems rather than soft real-time systems.

\[ \text{Deadline miss ratio} = \frac{\text{number of packets that miss their deadline}}{\text{total number of packets received}} \]
7.2. Configuration for simulation

Configuration of my simulator is given below. The variation of case 1 configuration, case 2 configuration and case 3 is differentiated by *. The case 1 configuration is having maximum size of a message which is 4 packets and deadline is supposed to be equal to period.

**Case 1 configuration**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>No of Nodes</td>
<td>15</td>
</tr>
<tr>
<td>RT channels</td>
<td>20</td>
</tr>
<tr>
<td>Physical Channels</td>
<td>1 – 5 physical channels</td>
</tr>
<tr>
<td>Period</td>
<td>3000 us</td>
</tr>
<tr>
<td>Simulation length</td>
<td>150000 [us]</td>
</tr>
<tr>
<td>* Deadline</td>
<td>3000 [us]</td>
</tr>
<tr>
<td>* Message size in terms of packets</td>
<td>4 packets per node</td>
</tr>
<tr>
<td>Message size in [us]</td>
<td>125 [us] * 4 = 500 [us]</td>
</tr>
<tr>
<td>Packet duration</td>
<td>125 us</td>
</tr>
<tr>
<td>Averaging</td>
<td>50 times</td>
</tr>
</tbody>
</table>

Table 3 of case 1 configurations for simulation

**Case 2 configuration**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>No of Nodes</td>
<td>15</td>
</tr>
<tr>
<td>RT channels</td>
<td>20</td>
</tr>
<tr>
<td>Physical Channels</td>
<td>1 – 5 physical channels</td>
</tr>
<tr>
<td>Period</td>
<td>3000 us</td>
</tr>
<tr>
<td>Simulation length</td>
<td>150000 us</td>
</tr>
<tr>
<td>* Deadline</td>
<td>375 [us] - 875 [us]</td>
</tr>
<tr>
<td>* Message size in terms of packets</td>
<td>1 – 4 packets per node</td>
</tr>
<tr>
<td>Message size in [us]</td>
<td>125 [us] - 500 [us]</td>
</tr>
<tr>
<td>Packet duration</td>
<td>125 us</td>
</tr>
<tr>
<td>Averaging</td>
<td>50 times</td>
</tr>
</tbody>
</table>

Table 4 of case 2 configurations for simulation

**Case 3 configuration**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>No of Nodes</td>
<td>15</td>
</tr>
<tr>
<td>RT channels</td>
<td>20</td>
</tr>
<tr>
<td>Physical Channels</td>
<td>1 – 5 physical channels</td>
</tr>
<tr>
<td>Period</td>
<td>3000 us</td>
</tr>
<tr>
<td>Simulation length</td>
<td>150000 us</td>
</tr>
<tr>
<td>* Deadline</td>
<td>1500 [us] - 2000 [us]</td>
</tr>
<tr>
<td>* Message size in terms of packets</td>
<td>1 – 4 packets per node</td>
</tr>
<tr>
<td>Message size in [us]</td>
<td>125 [us] - 500 [us]</td>
</tr>
<tr>
<td>Packet duration</td>
<td>125 us</td>
</tr>
<tr>
<td>Averaging</td>
<td>50 times</td>
</tr>
</tbody>
</table>

Table 5 of case 3 configuration for simulation
7.3. Results

To analysis the network performance and stability, I have plotted three types of graph with the variation of different network parameters (number of packets generated by a node on its period), deadline of a packet, physical channels and RT channels). i.e., average delay, network utilization and deadline miss ratio.

To justify my simulation, I have a base case (case 1) which will verify the simulator. In the case 1, deadline of a packet is set to be equal to the period and all nodes are generating a message of fixed length i.e., a message contains 4 packets for the base case simulation.

**Average delay:**

![Graph between No of RT channels & average delay in [us] (case 1)](image)

It can be seen from the graph and average delay is increasing very slowly with 1 physical channel until RT channel six is exposed to the network. This shows that size of the waiting queue is very small. As soon as, the network is exposed to seventh RT channel, there is sudden rise in the average delay. So, at seventh channel, network gets saturated, size of the waiting queue rise rapidly.

But when I expose two physical channels, we can see the shift in the saturation point of network and it starts accepting until RT channel seven.
Network Utilization:

![Graph between no of RT channels & utilization [%]](image)

By using case 1 configurations, we can see the network utilization is reached 100% with six RT channels and one physical channel. We can prove it mathematically.

Message size (in terms of packets) = 4 packets

Message size [us] = packet duration * number of packets

= 125 [us] * 4 = 500 [us]

Deadline = 3000 [us] for each packet

<table>
<thead>
<tr>
<th>No of RT ch</th>
<th>No of Physical ch</th>
<th>No of Packets</th>
<th>Time lapsed [us]</th>
<th>Utilization %</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>4</td>
<td>125 * 4 = 500</td>
<td>(500 / 3000) * 100 = 16.6</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>8</td>
<td>125 * 8 = 1000</td>
<td>(1000 / 3000) * 100 = 33.3</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>12</td>
<td>125 * 12 = 1500</td>
<td>(1500 / 3000) * 100 = 50</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>16</td>
<td>125 * 16 = 2000</td>
<td>(2000 / 3000) * 100 = 66.6</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>20</td>
<td>125 * 20 = 2500</td>
<td>(2500 / 3000) * 100 = 83.3</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>24</td>
<td>125 * 24 = 3000</td>
<td>(3000 / 3000) * 100 = 100</td>
</tr>
</tbody>
</table>

Table 6 Network utilization

Deadline = period and message size is equal to four packets

Table 6, verifies my simulator that it can achieve 100 % utilization with certain conditions and other graphs are also related with these conditions. But in my actual simulation, the deadline and message size is varying dynamically (randomly).
Deadline miss ratio:

For a network having single physical and six RT channels, it can be seen that no packet is missing its deadline. But there is a sudden change when number of RT channel is more than 6. Packets start missing their deadline and with seven RT channels 93% of the packets can not meet their deadlines.

With two physical channels, it can be seen that one more RT channel is accepted and no packet is missing its deadline. With one more RT channel i.e., with eight RT channels with two physical channels, there is an increase in the uplift of the graph compared to the graph having single physical channel. Now with eight RT channel and two physical channels almost 82% of packets will miss their deadlines and so on.

For hard real-time systems, where deadlines of packets are very critical, this type of networks can be used with two physical channels and seven RT channels so none of packet will miss its deadline.

For software real-time systems, where deadlines of packets can be compromised, this type of networks can use one more RT channel as compared to hard real-time system example. i.e., five physical channel and eight RT channels, 40% of packets will miss their deadlines.

Compared to above case 1, we have two variations in simulation i.e., deadline is variable and message size is also variable. i.e., deadline between 375 [us] - 875 [us] and message size (1 – 4 packets)
Average delay:

For a network having condition of case 1, it is visible that saturation point for the network comes at 7th RT channel with 1 physical channel. The traffic is markedly confined to between 221 [us] and 8022 [us] with one physical channel. For the network, with two physical channels saturation point comes with 10th RT channel and the traffic is confined between 163 [us] and 5818 [us] which is less 38 % less than average delay with one physical channel so there is visible improvement in terms of average delay. For 5 physical channels, saturation point comes at 13th RT channel and traffic is confined to between 200 and 5216 [us]. So, we can conclude that with the increase the number of physical channels, it is accepting more RT channels and by increasing physical channels, we can move the saturation point for the network significantly.

Figure 18: Graph between no of RT channels & Average delay [us] (case 2)

In figure 18 below, the graph between no of RT channels and average delay is presented for a network having packets deadline varying between 1500 [us] and 2000 [us]. If we compare both graphs in figure 17 and 18, it is seen that there is not as such big difference in both graphs related to saturation point but there is difference in the maximum average delay attained at 20th RT channel using two and five physical channels. The delay is increased with one physical channel and twenty RT channels using condition of case 3.
Figure 19: Graph between no of RT channels and average delay [us] (case 3)
Network utilization:

By looking at the graph of network utilization, I found that the network could use 21.5 % of network bandwidth for the case of three RT channels and one physical channel. With increase of one more physical channel, the saturation point goes further and it accepted one more RT channel and less utilization which also prove that utilization is decreased and it can accept more traffic.

Figure 20: Graph between no of RT channels & Utilization [%](case 2)

Figure 21: Graph between no of RT channels and Utilization [%] (case 3)
If we compare figure 19 and figure 20, network utilization is increased as I change deadline for the packets from 375[us] – 875[us] to 1500[us] – 2000[us]. With one physical channel, network was able to achieve 62.5 % and 43.29 % utilization and with one and two physical channels respectively.

The reason of low utilization is described below with an example.

As I am mentioned in previous chapter that I am using EDF algorithm for the transmitting of the packets which means no packet having higher deadline can be transmitted if there is any packet available with the shorter deadline exists in the global queue.

<table>
<thead>
<tr>
<th>Packet id</th>
<th>Deadline</th>
<th>Generation time</th>
<th>Source node</th>
<th>Destination node</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>375</td>
<td>0</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>375</td>
<td>0</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>375</td>
<td>0</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>375</td>
<td>0</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>500</td>
<td>0</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>6</td>
<td>500</td>
<td>0</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>7</td>
<td>500</td>
<td>0</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>500</td>
<td>0</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>625</td>
<td>0</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>625</td>
<td>0</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>11</td>
<td>625</td>
<td>0</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>12</td>
<td>625</td>
<td>0</td>
<td>3</td>
<td>6</td>
</tr>
</tbody>
</table>

Table 7 Blocking effect example

Suppose the case when the network has 3 physical channels at any time instant and the packets available in the global queue are mentioned in Table 7. Packet with packet id 1 will be transmitted using physical channel 1 only because packets with id’s 2, 3 and 4 have the source and destination conflict but I can not transmit other packets in the global queue because they have higher deadline. So, these packets will be in waiting queue and at this time instant only one physical channel will be used. So, it can said that if there is large variation in the deadline of packets so there will be less utilization of the physical channels.
Deadline miss ratio:

It is seen in the graph below, there is a rapid inclination in the deadline miss ratio with one RT channel. There is not as such big different in terms of deadline miss ratio for the networks with one physical channel and three physical channels until with two RT channels. The difference can be seen after two RT channels. The deadline miss ratio is significantly decreased compared to the network with single physical channel and so on.

![Graph between no of RT channels & deadline miss ratio](image1)

**Figure 22:** Graph between no of RT channels & deadline miss ratio (case 2)

![Graph between no of RT channels and deadline miss ratio](image2)

**Figure 23:** Graph between no of RT channels and deadline miss ratio (case 3)
If we compare figure 21 and 22, it is seen that there is considerable change in deadline miss ratio. Network showed less deadline miss ratio with the network having longer deadlines i.e., case 3 (1500 [us] – 2000 [us]). It shows that with one physical channel there is no packet which misses its deadline until RT channel 4 and achieved almost 47 % of deadline miss ratio compared to network having shorter deadlines.
8. Summary

8.1. Future work
In our approximate simulation, we assumed that control channel information is shared by nodes in negligible time. But simulation can be improved by adding control channel and its delays to make it more effective.

Our protocol can be enhanced by using class relationship to the messages. So, specific class messages can be transferred using specific channels.

8.2. Conclusions

A study of existing multichannel MAC protocols, both for wireless and optical networks was carried out. Later, an approximate simulation is also carried out to find out the facts and figures, we studied.

As we know that WDM multiplexing is a source of increasing higher bandwidth utilization in WDM networks in optical communication. By using multiple physical channels, we have also seen that network performance increased in terms of delay, deadline miss ratio and network utilization. The reason of bad utilization can be the limitation of one transmitter and one receiver per node. We are strictly following the EDF scheduling so if there is any delay with any node in the network can jeopardize other traffic.

Depending on which QoS we aim for, we could have different performances of our network by adjusting capacity of a node and deadline of packets. We can only guarantee at least one packet to be transferred in worst case scenario.
9. References


