Politecnico di Milano

V Facoltà di Ingegneria

Corso di Laurea Specialistica in Ingegneria delle Telecomunicazioni

Dipartimento di Elettronica e Informazione

PERFORMANCE EVALUATION OF TIME-DRIVEN SWITCHING UNDER STATIC AND DYNAMIC TRAFFIC CONDITIONS

Relatore: Prof. Massimo **TORNATORE**

Correlatore: Ing. Francesco **MUSUMECI**

Autori: Lorena **ZAPATA** Matr. 749033

Luis **HERNANDEZ** Matr. 749088

Anno Accademico 2010/2011

Contents

Abstract **v**

Sommario **vii**

Introduction **ix**

1 WDM Networks **1**

1.1 WDM systems 2

1.2 Network components 3

1.3 Optical Transport Networks 5

1.4 Optical Switching technologies: OCS, OPS and OBS 7

1.4.1 Optical Circuit Switching (OCS) 9

1.4.2 Optical Burst Switching (OBS) 10

1.4.3 Optical Packet Switching (OPS) 11

1.5 TDS as a feasible alternative 11

2 TDS: Time Driven Switching **13**

2.1 Definition of Time Driven Switching 14

2.2 Principles of Operation of the TDS 15

2.2.1 Time Organization in TDS 15

2.2.2 Synchronous Virtual Pipe 16

2.3 TDS Node Structure 18

2.3.1 Time Driven Switch Architecture 18

2.3.2 Alignment Issue 19

2.3.3 Delay Problem 19
2.4 TDS Network Architecture and Deployment 20
2.5 Performance 21
2.6 TDS Forwarding techniques 22
2.6.1 Immediate Forwarding Technique (IF) 22
2.6.2 Non Inmediate Forwarding Technique(NIF) 23
2.6.3 Comparison of the Forwarding Techniques 23
2.7 Wavelength Selection in TDS Networks 23
2.8 Routing scenarios for TDS 24
2.9 TDS Related Works 24

3 Models and heuristics for the design of TDS network 26
3.1 TDS network optimization by ILP 27
3.1.1 Flow Formulation 28
3.1.2 Route Formulation 35
3.2 TDS network design optimization by heuristic algorithms 39
3.2.1 Heuristic Overview 39
3.3 Description of the algorithm 40
3.3.1 Phase 1. Initial greedy RWTA 40
3.3.2 Phase 2. Optimization cycle 41
3.4 Heuristic Complexity 43

4 Static Traffic: Results and Discussion 44
4.1 8-node Network 44
4.1.1 8-node network - IF Flow Formulation vs Route Formulation with variable TF granularity 45
4.1.2 Heuristic approach 49
4.2 24-node US backbone network 52
4.2.1 24-node US backbone network : Analysis of TDS architecture by varying the granularity 57

5 TDS Routing Algorithms **61**
5.1 k-Paths algorithms 61
5.1.1 BellmanFord based Algorithm (BF-based): 63
5.1.2 Yen based algorithm (Yen-based) 68
5.2 Belman-Ford Extended algorithm (BF-Extended): 72
5.3 Routing and resource allocation algorithms in the network: 77
5.3.1 Single Path - Multiple Wavelenght Algorithm (SP-MW-A): . 77
5.3.2 Single Path - Single wavelenght Algorithm (SP-SW-A): . . . 81

6 Dynamic Traffic: Results and Discussion **85**
6.1 Metrics for performance evaluation 85
6.2 USA24 network topology 86
6.3 Analysis of the two k-paths proposed algorithms 88
6.4 SP-MW vs MP: Results Comparison 89
6.5 SP-MW vs SP-SW: Results Comparison 92

Conclusions **94**

Bibliografia **99**

Abstract

The work presented in this thesis analyses the performance of an innovative and new packet switching technique called Time-Driven Switching (TDS) or Fractional Lambda Switching (FLS) and its implementation in optical networks. In order to realize a network where all the operations are accomplished in the optical domain, switching is one of the most studied features and many different techniques have been developed, such as Optical Circuit Switching (OCS), Optical Burst Switching (OBS) and Optical Packet Switching (OPS). OCS is based on wavelength multiplexing and routing where an optical channel, called lightpath, is established on every link between the source and the destination node. Instead, in OPS every packet is converted and its header is processed in the electronic domain: in the meanwhile, the payload remains in a buffer, for example, if it is optical, it will be a Fiber Delay Line (FDL). OBS stands between circuit switching and packet switching: here header and payload are transported in different channels and only header is converted and processed in the electronic domain. However, all these techniques have limitations like, for example, the lack of optimization of resource utilization for OCS or necessity of buffering in OBS and OPS.

Instead, TDS is proposed as a novel switching technique completely developed in optical domain and able to overcome the limitations of the previous techniques(OBS, OPS, OCS). TDS Switches, synchronized through the Global Positioning System (GPS), divide the Common Time Reference (CTR) in Time Frames (TFs), the basic units for switching operation, assembled in Time Cycle (TC). Thus, TDS is able to aggregate and disaggregate fractions of optical chan-

nels (grooming and degrooming), called Synchronous Virtual Pipes (SVPs). An SVP is created reserving resources on nodes and links from source to destination nodes during a pre-determined set of TFs. The forwarding of an SVP implies the research of free TFs in every link composing the path: if there is no possibility to find free TFs, the connection request will be rejected. There are various techniques for creation of an SVP. These are called Forwarding Techniques and are the following:

- *Immediate Forwarding* (IF): data units of a given connection in $TF(i)$ are moved to their output port and sent out in $TF(i+1)$.
- *Non Immediate Forwarding* (NIF): Data units of a given connection c received in $TF(i)$ are moved to their output port and sent out in $TF(i+k_c)$ with $(k_c > 1)$.

In order to analyse the performance of TDS, we considered two different traffic types that may be offered to a WDM network:

- *Static traffic*: for that, we propose novel ILP formulations (considering two approaches Flow and Route formulations) aiming to minimizing the number of wavelengths in the network. Then, we provided a heuristic algorithm with the same objective, since ILP formulations do not scale well, hence, it is not feasible for larger networks.
- *Dynamic traffic*: we develop a decrete-event simulator to evaluate network performance of TDS in terms of SVP-request blocking probability and the average utilization.

Sommario

Il lavoro presentato in questa tesi ha come obiettivo lo studio di una nuova tecnica di commutazione, il Time-Driven Switching, e la sua applicazione nelle reti ottiche. Con l'intento di realizzare architetture dove i flussi informativi vengano trattati principalmente nel dominio ottico, la commutazione ottica ata studiata e sono state sviluppate varie tecniche, presentate nella parte iniziale della tesi, come OCS, Optical Circuit Switching, OBS, Optical Burst Switching e OPS, Optical Packet Switching. LOCS si basa sullinstradamento di unintera lunghezza donda λ dove un canale ottico, o lightpath, viene instaurato riservando una lunghezza donda su ogni link dal nodo sorgente al nodo destinazione. In OPS, invece, lheader di ogni pacchetto viene convertito in elettronico e processato: nel frattempo il payload permane in un buffer, ad esempio se ottico in una Fiber Delay Line (FDL). OBS a tecnica che sta a meta la commutazione a circuito e la commutazione a pacchetto: qui vengono separati su due canali trasmissivi diversi il payload e lheader e solo questultimo viene processato elettronicamente. Tutte queste tecniche hanno peritazioni quali, ad esempio, la mancata ottimizzazione nell'utilizzo delle risorse per OCS o la necessit buffer e processamento elettronico in OBS e OPS.

Il TDS si propone come una tecnica di commutazione implementabile completamente in ottico e in grado di superare le limitazioni delle tecniche precedenti. TDS utilizza commutatori, chiamati TDS Switch che, sincronizzati tramite Global Positioning System (GPS), suddividono il tempo comune (Common Time Reference CTR) in Time Frames (TFs), le unitse che ogni switch grado di commutare ed instradare, i quali sono a loro volta raggruppati in Time Cycle (TC). Con TDS

si indi in grado di aggregare e separare frazioni di canali ottici (grooming e degrooming), chiamati Synchronous Virtual Pipes (SVPs). Un SVP viene creato riservando capacit commutazione e trasmissione lungo il percorso da sorgente a destinazione durante un certo set di TF, che si ripresenta con una certa periodicitutto questo richiede la ricerca di una sequenza di TF non occupati, durante la quale verrservata capacitngo i link del percorso. Se non vi ssibilit trovare una sequenza completa di TF non occupati da altri SVP gi atto, la richiesta viene bloccata. Esistono varie tecniche attraverso le quali vengono instaurati gli SVP, anche chiamate Forwarding Techniques:

- *Immediate Forwarding* (IF): ad un TF in ingresso al commutatore corrisponder TF successivo in uscita.
- *Non Immediate Forwarding* (NIF): ad un TF in ingresso al commutatore corrisponder primo TF libero nel TC.

Proprio da questa riflessione ta la volont capire quanto, energeticamente parlando, convenga implementare TDS. Il nostro lavoro si particolare focalizzato anche sulla capacit aggregare e disaggregare il traffico e quanto sia dispendioso per una rete rendere questa cosa possibile, a seconda della tecnologia usata.

Al fine di analizzare le prestazioni di TDS, abbiamo considerato due diversi tipi di traffico che potrebbero essere offerti ad una rete WDM:

- *Traffico statico*: Per questo, proponiamo nuove formulazioni ILP (prendendo in considerazione due approcci: formulazioni per flusso e per cammino) puntando a minimizare il numero di lunghezze d'onda nella rete. Poi, abbiamo fornito un algoritmo euristico con lo stesso obiettivo, poich formulazioni ILP non scalano bene, quindi, non ttibile per reti di grandi dimensioni.
- *Traffico dinamico*: abbiamo sviluppato un simulatore a eventi discreti per valutare le prestazioni della rete TDS in termini di probabilit blocco di una richiesta SVP e l'utilizzo medio della rete.

Introduction

Internet is changing and evolving in these years. Number of demands, as well as their volume, grows at a rapid rate and brings new challenges with some problems to face. Recent studies show that Internet demands for bandwidth are exponentially increasing, and so does its energy consumption[1]. As described in [2], based on data from year 2009, ICT (Information and Communication Technology) consumes about 8% of total electricity all over the world. According to [3] in 2007, the footprint of ICT (complete life cycle) was about 4% of the overall primary energy consumption, and, without considerable counters, this percentage is expected to grow about 8% in 2020.

Since our limited global energy resources and the serious environmental problems facing the world are gaining concerns these days, most of the existing investigations in a large area of scientific disciplines are focused on new technological solutions for energy conservation. Thus, power consumption is a major issue for the future Internet, that is widely recognized to have to be “green”.

Despite ICT’s environmentally friendly image, there is external pressure to adopt sustainable policies (i.e. institutions implementing sustainable procurement policies) towards this area and to pursue energy minimization in the next years. For the latter, it is necessary to design new network paradigms so that ICT will maintain the same level of performance, while consuming a lower quantity of energy [1]. Energy efficient solutions in optical networks will contribute to save the energy consumed by ICT and, further, reduce the energy consumption of the whole society and partly protect the environment. Disregarding some negligible values, the power distribution in the network is shown in Figure 1. It is clear that only three operations are strongly involved in the power usage, having the following distribution: processing (22%), switching/routing (34%) and electrical regeneration (27%). Based on the previous analysis, current communication networks require 50 to 100 W to transfer a Gbps, having an efficiency within the 50-100 W/Gbps range. Therefore, minimizing the total power per network throughput is the main architectural goal, focusing on the three most contributing operations. The fulfilment of this goal in part depends on the ability to transfer the bits in the optical domain with minimal convertion into the electrical domain. The ability to keep the signal in the optical domain, as it is transmitted from the origin point in a network, until the sink point that it leaves, brings a clear benefit from the power density viewpoint. The so-called all-optical network (AON) is going to be an enabler in developing a "greener" network, with the additional benefit of being able to cope effectively with the fully expected network growth and cost reduction requirements.

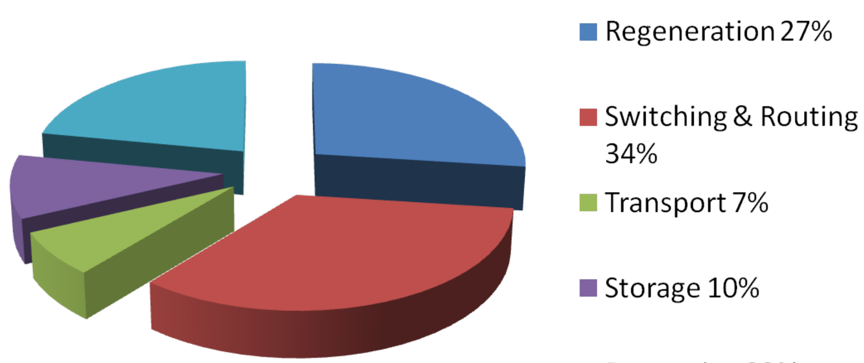

Figure 1: *Network Power Distribution*

Minimizing energy consumption of optical networks can be generically addressed at four levels, i.e. component, transmission, application and network. We can considered that a telecommunication network can be subdivided in three main network domains:

- *Access network domain*, as a lowest sublevel, is a part of telecommunication network which enables end-users (business and residential customers) to connect to the rest of the network infrastructure and spans a distance of a few kilometers;

- *Metro network domain*, as the middle part of the structure, comprises the intermediate links between the core network, or backbone, of the network and the small sub networks at the edge of the entire hierarchical network;

- *Core network domain*, often called backbone network, as the central part of a telecommunication network provides nationwide or continental coverage.

The latter will be the focus of our attention during this study. This part of the network is characterized by smaller number of nodes and larger dimensions of links both in terms of length and capacity. It is usually a mesh interconnecting network that provides increased protection flexibility and network-utilization efficiency. It carries huge amount of traffic collected through the peripheral areas of the network. Thus it needs to be equipped by interfaces towards metro and access networks which are in charge to distribute and collect traffic. There are two very important operations in a backbone networks, i.e.:

- *Aggregation*: it grooms the incoming flows deriving from the same terminal nodes;

- *Switching*: it routes and switches traffic data flow, depending on its source and destination.

In core networks, optical technologies are widely used to support the two aforementioned. Since the current grooming operation of data is only possible in the electronic domain, a significant amount of energy is consumed in network routers and transponders, which perform the O/E/O (Optical/Electronic/Optical) conversion. Therefore, one of the main concerns in these days is the creation of a network where every operation is performed in the optical domain, in order to minimize energy consumption and to obtain better performance despite limitations of the electrical domain. Various energy-eficient approaches using optical switching technologies allow to significantly reduce the quantity of high power-requiring O/E/O conversions and electronic processing operations at the core network layer. A further improvement can be obtained with Time Driven Switching (TDS), a technique which allows to switch fractions of wavelengths *directly* in the optical domain exploiting the time-coordination of all network components. The work presented here focuses on performance analysis of TDS networks. The fulfilment of this objective is achieved by setting up a software tool able to perform the following functions:

- *Design*: Optimization of TDS-based networks starting from a given IP traffic forecast and a physical topology (nodes and links). The final goal can be the minimization of the number of wavelengths employed, which is directly related to the cost due to transport traffic flows, or the number of transponders used, which corresponds to the minimization of network equipment. Hence, to obtain results, Integer Linear Programming (ILP) formulations are presented.

- *Simulation*: Analysis of the behaviour of the TDS-based network under dynamic-traffic scenario. Statistical properties of traffic will be devised to emulate demands for traffic flows coming from the IP layer. A discrete-event simulator is used to evaluate network performance in terms of blocking probability.

The thesis is divided into seven chapters and organized as follows:

- In **Chapter 1** the network infrastructure that underlies this work is introduced, highlighting the potentialities provided by the Wavelength Division Multiplexing (WDM) technique. Moreover, we review the advantage and problems of the three optical switching technologies - Optical Circuit Switching (OCS), Optical Burst Switching (OBS) and Optical Packet Switching (OPS).

- In **Chapter 2** the Time Driven Switching (TDS) is introduced as a novel switching technology.
- In **Chapter 3** Integer Linear Programming (ILP) formulations are discussed as a possible approach for a network design. In particular, two basilar and well-known approaches are used: Flow Formulation (FF) and Route Formulation (RF) for different TDS scenarios. Besides, we describe a heuristic approach for minimizing the number of wavelengths in a realistic network.
- In **Chapter 4**, We present results regarding static traffic using ILP formulations and heuristic.
- **Chapter 5** introduces the new algorithms used in the simulator to perform the routing of connections applying different techniques of TDS (i.e. immediate and non immediate forwarding) in dynamic-traffic conditions.
- **Chapter 6** focuses on the simulation results under dynamic condition: we will analyze the blocking probability for a range of scenarios and their components in order to understand the factors that influence the TDS performance.

Finally, we draw the conclusion of the thesis, summarizing the obtained results and addressing possible future works.

Chapter 1

WDM Networks

A telecommunications network consists of a series of nodes, where signal processing is done (such as grooming and degrooming of the signals, multiplexing, etc.), and a series of links between nodes. Figure XXX shows a simplified model of the telecommunication network, that can be subdivided in three main network domains:

- *Access network domain*
- *Metro network domain*
- *Core network domain*

End-users use the access network domain to reach the elements of traffic concentration of the metro network domain (Central Offices: COs) via dial-up modem, DSL or directly via optical fiber, typically along a tree topology which is not necessarily bi-connected and therefore without protected. The metro network domain is generally composed of rings based on SDH/SONET covering lengths of tens and hundreds of miles. This domain carries out the function of concentrating and transporting the traffic to the core network domain. The higher cost of the metro network is due to network equipment (OXCs and ADMs) that are placed on all nodes. Since the links are relatively short in length, nodes will be particularly numerous, and because of their large number, there is a greater effect on the cost of the network.

All outgoing and incoming traffic of the metro network domains pass through the Points of Presence (POPs), which must then construct the access nodes to the core network domain (often called backbone network) that connects all the metro networks of different regions until arriving at the national and international scale. The backbone network, covering great distances, has costs that are the most part due to the equipment (fibers, amplifiers and regenerators are used to enable transmission over long distances). The main function of transport networks is to provide both a large transmission capacity and high reliability.

This thesis work is based on backbone networks that form the transport network. A transport network has the task of transferring safely a large amounts of information of all kinds. The topologies traditionally used for the backbone domain were based on ring networks as the topology of the access network, but this network configuration have been proved to be inefficient and not scalable, so we opted for a move towards meshed topologies.

1.1 WDM systems

Nowadays, optical technology is the key element for developing links especially in the transport network that uses Wavelength Division Multiplexing (WDM) technology at the transmission layer. The optical transmission systems include the needed equipment to transmit client traffic over optical fibers. They can be divided into two major categories: single-channel and multi-channel systems. The latter (generally called WDM), exploits several channels on different wavelengths, using a bidirectional fiber link (constituted by two optical fibers in the two directions). In WDM systems the transmitting terminals, whose outputs used different wavelengths, are followed by an optical multiplexer which aggregates all channels (wavelengths) to be transmitted on the same fiber. An optical demultiplexer with complementary functions to those of the multiplexer, is placed at the receiving terminals. The typical length of amplification sections (i.e. the maximum length beyond which the amplification is required), is 80 km. The electric regeneration performed in the receiving terminal removes the effects of any degradation (introduced by switches, EDFAs, propagation in optical fibers) along the link. Depending on the specific application, the spectral spacing between channels in terms of wavelength (or, frequency) is different. From this belief, WDM systems are divided into two main categories: DWDM systems (Dense WDM) with channel spacing in the order of 100 GHz (about 0.8 nm) and CWDM systems (Coarse WDM) with channel spacing of 1600 GHz (about 20mn).

1.2 Network components

In this section, we describe the main components used in WDM network. Some of them become part of the TDS architecture which is studied in more detail in the following chapters:

- *Router*: A router is a networking device whose software and hardware (in combination) are customized to the tasks of routing and forwarding information among networks. It makes intelligent decisions on the best routing paths for the data transmission in the network.

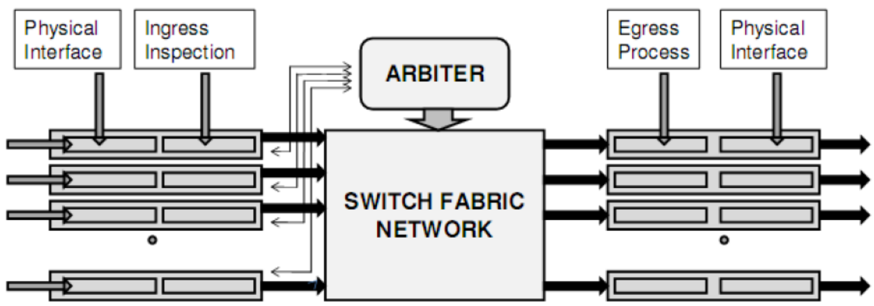

Figure 1.1: Block Diagram of a Network Router

As shown in Figure 1.1, a network router consists of three parts: the ingress packet process unit, the egress packet process unit, the arbitration unit and the switch fabrics. The ingress packet unit transforms the serial dataflow on the transmission line into bus dataflow, and inspects the header and the content of the incoming packets. The egress process unit re-assembles the processed packets and delivers the packets to their destination ports. The arbitration unit determines when and where a packet should be routed from the ingress ports to the egress ports [4]. A switch fabric circuit is an interconnect network that connects the ingress ports to the egress ports. Different switch fabrics have different impacts on the router performance.

- *WDM Transponders* Most of the existing transport networks are composed of a set of point-to-point transmission WDM systems, enabling the optical signals to reach distances of thousands of kilometers. In these networks, an optical signal is converted into an electronic signal and then again is retransmitted as an optical signal, in every intermediate node (including source and destination) before it reaches its destination.
  In our case, WDM transponder are used in source and destination nodes, operating at 10Gbit/s.

- *OXC-Optical Cross Connector*: When the optical network topology is a mesh (i.e. nodes are interconnected by bers to form an arbitrary graph), an additional fiber interconnection device is needed to route the signals from an input to the desired output port. These devices are called optical cross connects (OXCs) and have the task of the wavelength switching in any input fiber to any output fiber. In our study the OXC's switching fabric is implemented using SOA (Semiconductor Optical Amplifier).

- *Buffers*: The optical buffers within the core nodes are indispensable because each packet is treated separately. Logically, the node itself has a more complex structure because it has to be ready to perform processing operations for each of the incoming data payloads. The payloads latter must wait in buffers and be forwarded later to the next node. Buffers here are implemented as fiber delay lines. The major FDL's disadvantage is that we do not know how long they have to be, thus how much attenuation we have to deal with. In addition, if we want to overcome the noise due to transmission in the FDLs, we will need to add a certain number of the EDFAs (Erbium Doped Fiber Amplifiers), which will impact (better say, increase) the total cost of the switching system.

- *Amplifiers*: an optical amplifier is a device that amplifies an optical signal directly without having to convert it into an electrical signal as it is in a classic signal repeater or regenerator. An optical amplifier may be thought of as a laser without an optical cavity, or one in which feedback from the cavity is suppressed. Stimulated emission in the amplifier's gain medium causes amplification of incoming light. The erbium doped fiber amplifier (EDFA) is the most widely used fiber amplifier, since its amplification window coincides with the third transmission window of silica optical fibers. Two bands have developed in the third transmission window the Conventional, or C-band, from approximately 1525 nm 1565 nm, and the Long, or L-band, from approximately 1570 nm to 1610 nm. Both of these bands can be amplified by EDFAs, but it is normal to use two different amplifiers, each optimized for one of the bands. The principal difference between C- and L-band amplifiers is that a longer length of doped fiber is used in L-band amplifiers. The longer length of fiber allows a lower inversion level to be used, thereby giving at longer wavelengths (due to the band-structure of Erbium in silica) while still providing a useful amount of gain.

1.3 Optical Transport Networks

The diffusion of the Internet in recent years has had a major impact on the evolution of telecommunications networks, especially for what concerns the demand for bandwidth. The causes are to be found mainly in the increase in the number of users who have broadband access via ADSL or optical fiber (FTTH - Fiber To The Home), and in the interest in new multimedia services and new applications that are just born with Internet. Services such as video streaming, high-definition IPTV, file sharing using peer-to-peer networks, remote data backup, or applications such as distributed computing resources, require significant transmission and their future development is difficult to predict . For this reason, future telecommunications networks should be able to support traffic demands in a scalable, cost efficient way and more automated with management techniques. In particular, optical networks allow to satisfy these needs by providing a scalability and offering available bandwidth on the order of Tb/s, hence, they are primarily used as a long-range transport infrastructure.

The massive scalability of optical networks is due to the exploitation of WDM technology at the physical layer which, as already mentioned above, can efficiently use the available bandwidth of optical fibers offering a high number of usable channels. The current network model more suited to support large volumes of traffic is the so-called IP-over-WDM (different from many networks still operating that involves the transport of IP over SONET/SDH or ATM).
For many years, with the aim of creating an interaction between electronic devices and the WDM layer, several combinations of protocols have been implemented , including IP-over-SDH/SONET and IP-over-ATM. Recently, in order to achieve greater integration between IP and WDM layer, it is preferable to adopt an alternative solution that eliminates redundancy features due to the overhead introduced by intermediate layers of protocol. This solution is made possible as a result of technological achievements, not only in transmission but also in the optical switching: in particular, the introduction of new switching equipments has allowed to completely overcome the bottleneck represented by the conversion of signals from the optical domain to the electronic one and vice versa.

Optical networks using integrated WDM are currently potential candidates for the next generation transport networks, and this is mainly due to their huge spectrum (50 THz) and scalability that other technologies cannot offer. Among all the architectures of optical networks, IP-over-WDM is considered to be the future trend of the Internet backbone network, as IP plays a dominant role in networking technology. In an IP-over-WDM network, IP packets are directly transported over WDM channels, avoiding the intermediate layer of ATM or SDH. This approach leads to a more efficiently data transport network and to save significant overhead. Moreover, in this type of networks, source and destination nodes can be connected by a wavelength channel, the so-called lightpath. A lightpath is an end-to-end connection between any two nodes which are not necessarily neighbors. One of them is defined as source and the other one as destination. The signal along a lightpath remains in the optical domain even while traversing the intermediate nodes, all way down from source to destination. Once two nodes are connected by a lightpath, they become virtually neighbours, regardless of the physical connectivity between them [5]. Such logical topologies (e.g. IP/MPLS layer) are implemented directly on top of WDM physical layer, thereby forming two-layer network architectures as shown in Figure 1.2. In the IP layer, a core IP router is connected to an Optical Cross Connect (OXC) through short-reach interfaces and aggregates data traffic coming from low-end access routers. The optical layer provides the ability to communicate between IP routers. Optical switching nodes are interconnected with fiber links and each link can contain more fibers. Each fiber is associated with a pair of multiplexer/demultiplexer and such wavelength is transmitted/received by a pair of transponders for transmission of data. With the O/E/O processing capacity of each transponder, switch nodes guarantee the opportunity to have full wavelength conversion (i.e. an incoming traffic at the node, may be scheduled onto multiple wavelengths at the desired output port).

1.4 Optical Switching technologies: OCS, OPS and OBS

Optical networking research is moving from circuit-oriented networks to networks capable of performing packet switching. Several approaches exploiting optical switching have been proposed to improve transport networks. One of such approach is Optical Circuit Switching (OCS), based on whole wavelength switching. Here, a lightpath needs to be established using a dedicated wavelength on each link from source to destination. Once the connection is set up, data remains in the optical domain throughout the lightpath. An alternative solution to OCS is Optical

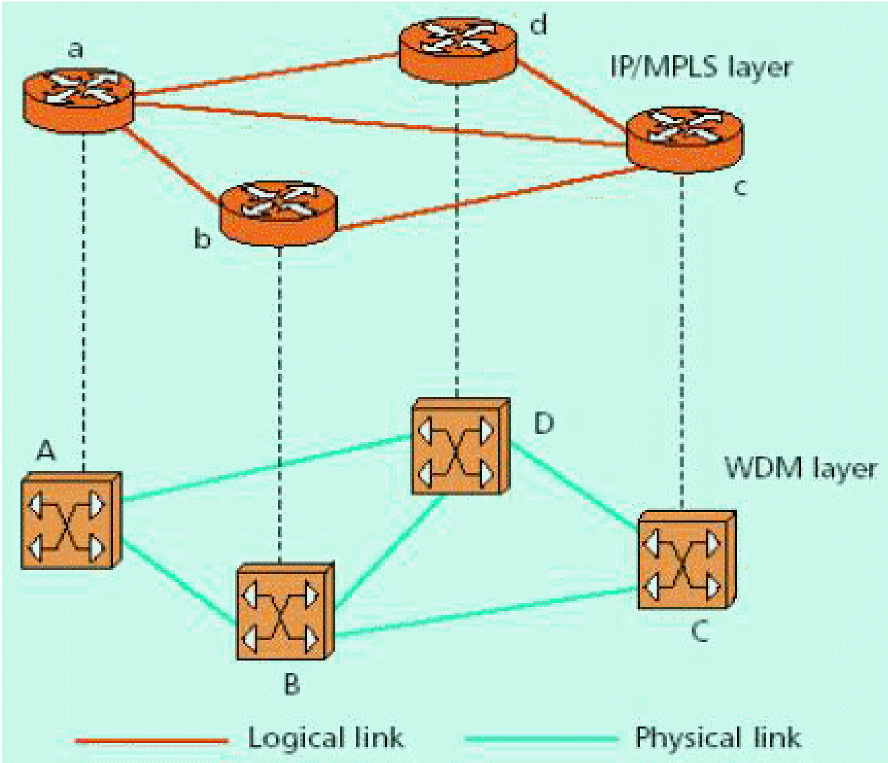

Figure 1.2: *IP-over-WDM Network Architecture*

Packet Switching (OPS). In OPS, while the packet header is being processed either all-optically or electronically (after an O/E conversion at each intermediate node), the data payload must be buffered in the fiber delay lines before being forwarded to the next node. However, nowadays fiber delay lines are still impractical from a technology point of view. In order to provide optical switching for next generation Internet traffic in a flexible yet feasible way, a new switching paradigm, called Optical Burst Switching (OBS), can be considered. In the OBS paradigm, only a few control channels (e.g., one per fiber) go through O/E/O conversions. Given that data is switched all-optically at burst level, data transparency and statistical multiplexing can be achieved concurrently. Recently, another proposed switching technique called Time-Driven Switching (TDS), which uses a UTC (Universal Coordinated Time) for implementing pipeline forwarding inside the network [6]. TDS is a promising candidate to overcome current limitations of OCS, OBS and OPS.

While TDS will be the main focus on this thesis, in this section we provide an overview of pros and cons of OCS, OPS and OBS.

1.4.1 Optical Circuit Switching (OCS)

In the Optical Circuit Switching, once a lightpath is setup, all data sent by a source node will go to a specific destination node. Thus, a new circuit is created and immediately reserved for the specific source-destination pair of nodes. Due to that, we have to face the problem of the number of different lightpaths needed to connect all nodes: this problem is known as the N^2 problem.

N^2 Problem

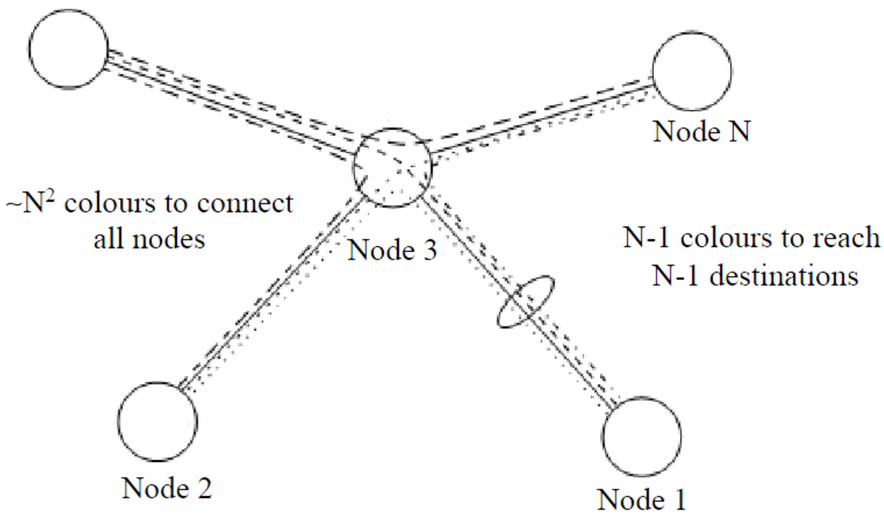

Figure 1.3: N^2 *Problem*

The Figure 1.3 illustrates an N -node network: every node has to use a different wavelength for each (N-1) possible destinations. If every node pair used a different wavelength, it would be necessary to have N-1 wavelength for the connections between the first node and the rest of the possible destinations, (N-2) for the second, and so on. Thus, the total number of the wavelengths needed is $\sum_{i=1}^{N-1} i = \frac{N(N-1)}{2}$.

One of the possible approaches in solving the coarse granularity problem is the exploitation of traffic grooming [7]. A straightforward approach for traffic grooming is to treat wavelength channels as network links, and directly multiplex sub-wavelength client connections onto wavelength channels. However, in WDM optical networks, the optical nodes cannot switch traffic at the subwavelength granularity without the O/E/O conversion. The main reason for using O/E/O onversion is to increase efficiency in bandwidth utilization. On other hand, the electronic switches are known to suffer from problems such as limited capacity, huge power/space consumption and heat dissipation in addition to requiring expensive O/E/O conversions [7].
In spite of the obvious disadvantages in terms of resource utilization, there are a couple of surpassing advantages such as: guaranteed service quality, fixed latency in the network, no jitter[1], fixed signaling overhead, and, the most important, low blocking probability[8].

1.4.2 Optical Burst Switching (OBS)

In order to provide optical switching for next generation Internet traffic in a flexible yet feasible way, another switching paradigm called optical burst switching (OBS) was proposed.

The most significant advantages of the OBS are flexibility and efficiency compared to OCS network; it is more scalable and cost effective compared to O/E/O approaches; its data units (bursts) have smaller overhead and are more practical than packet switching approaches because the data bursts and control packets are transmitted on separate planes.

There are certain disadvantages to be mentioned to give the more complete description of the technique: the node structure complexity and the blocking probability are higher than in OCS, due to O/E/O conversion and collisions, respectively. The contention resolution problem can be resolved in three ways: using deflection, which can be applied in wavelength, space and/or time domains; dropping and

[1]Jitter denotes the time variation of a periodic signal in electronics and telecommunications, often in relation to a reference clock source.

preemption policies.

1.4.3 Optical Packet Switching (OPS)

Optical Packet Networks are based on a store-and-forward paradigm. It consists of sending IP packets directly over an all-optical backbone: while the packet header is being processed electronically after O/E conversion at each intermediate node, the data payload must be buffered in the fiber delay lines and forwarded to the next node. OPS is dicult to implement because of its need for a large number of O/E/O conversion devices (one set for each wavelength), header extraction/insertion mechanisms as well as FDLs (Fiber Delay Lines) and packet synchronizers. Moreover, packets that are bounced to different paths may cause congestion in other wavelengths or other parts of the network, spreading local congestion across larger areas of the network. In addition, packets do not follow the same path any longer, and so they may arrive out of order, which may be interpreted by TCP as losses due to congestion, and TCP may thus throttle back its rate. Packet reordering within a TCP session also causes unnecessary retransmissions, prevents the congestion window from growing properly and degrades the quality of the RTT estimator in TCP. Thus, we can conclude that OPS does not have much room to solve contention. Another problem that is perceived with OPS is that IP packet sizes are very short for some optical crossconnects to be rescheduled. Unfortunately, due to the immature technology, OPS network still cannot present its real advantages, which are optimized resource utilization and flexibility in adapting to the traffic requests, unlike OCS and OBS.

1.5 TDS as a feasible alternative

Analysing the pros and cons of the previous switching techniques, in this thesis we would like to investigate TDS architecture under static and dynamic conditions on WDM networks that demonstrastes the success to inherit all the good aspects (as the bufferless OCS with the lowest blocking probability, i.e. drop-out rate, the simplest node architecture, the lowest power consumption and almost neglectable values of delay and no jitter at all since we were dealing with circuits). Moreover, TDS architecture should avoid characteristics as granularity levels of OBS and especially, OPS, but to improve the resource utilization of OCS, which affect a high quality performance. A detailed description of TDS technique will be provided in the following chapters.

Chapter 2

TDS: Time Driven Switching

Modern telecommunication networks use light to transport information; in WDM systems multiple wavelengths travel on optical fibers increasing the total bandwidth. All optical networks are being studied to reduce conversion from optical to electric signal every time a switch is crossed, in order to forward it towards the proper destination.
however optical switches can only divert entire wavelengths: all the data on a colour should be switched and forwarded from the same source to the same destination. This leads to some constraints in the network itself, that is:

- a source needs a different colour for each destination it addresses; if the network has N access points and they have to be all connected to each other, the number of wavelengths needed can grow up to N^2.
- no aggregation/separation of multiple flows on/from a single wavelength can be operated: the wavelength travels unchanged switch by switch.

All these problems limit the extension of an all-optical network core, because of wavelengths growth and bandwidth mismatch between sub-networks. Therefore it is advisable to directly provide the availability of fractions of the wavelength capacity, so as to support "sub-lambda" end-to-end connections. Time Driven Switching (TDS) provides a solution to make available a capacity equal to a fraction of that transported by a wavelength in order to permit the above mentioned operations;

for this reason, a switch operating according to TDS is called Fractional Lambda Switch.

2.1 Definition of Time Driven Switching

TDS is an optical switching technique which uses a global Common Time Reference (CTR[1]) for implementing pipeline forwarding [9] inside the network: so the necessary condition for PF is having the same clock reference in all switching elements (as shown in Figure 2.1). In other words, Time-Driven Switching uses switch, called Fractional Lambda Switch (FLS), that, if synchronized with the CTR (i.e., the Global Positioning System (GPS)), can divide the CTR into Time Frames (TFs).

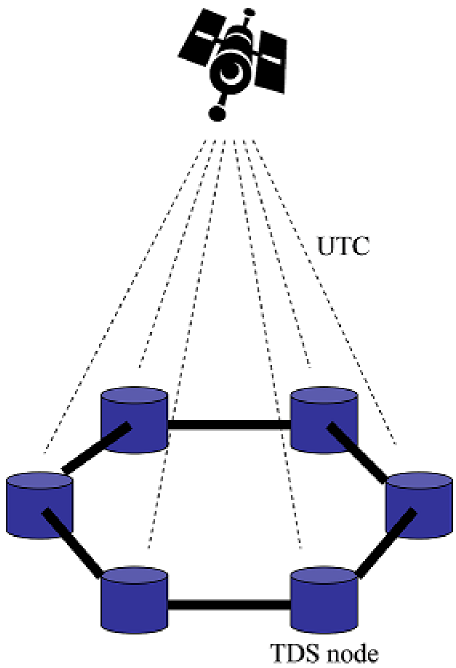

Figure 2.1: *TDS Synchronization with UTC*

Time Frames can be considered as data units containers that every switch can

[1] A global CTR is realized with the Coordinated Universal Time (UTC) standard. The UTC is globally available via GPS in the USA or via GALILEO in the EU.

switch and route, thus aggregating or separating fractions of the optical channel (grooming e degrooming). These fractions are called Synchronous Virtual Pipes(SVP) of variable capacity. The FLS transport data packets through SVPs with a completely deterministic quality of service, without performing any header processing and with constant end-to-end delay.

2.2 Principles of Operation of the TDS

2.2.1 Time Organization in TDS

TDS switches and networks are operated based on the standard UTC[2] second delivered by GPS/Galileo and other terrestrial systems. The UTC second is partitioned into time frames (TFs). TFs are fundamental information and switching unit, i.e. the basis for scheduling data unit forwarding throughout the network. Time is organized as follows: k contiguous TFs of duration d are grouped into a time cycle and l contiguous time cycles are grouped together into a super cycle. The TFs in a cycle are numbered from 0 to $k-1$ and all arithmetic expressions involving TF numbers are modulo k; for example, if i is a TF number, then $(i+1)$ means $(i+1) mod\, k$. A typical duration of a super cycle is equal to one UTC second. A 1 pps (pulse per second) signal aligned to UTC with a 10-20 ns accuracy can be obtained from the GPS at a low cost ($100-200). One of the possible problems of TDS is the correct mapping of the incoming TFs to the TFs that are directly derived from UTC [9].

[2]Coordinated universal time,(a.k.a. Greenwich mean time or GMT) second is defined by counting 9,192,631,770 oscillations of the cesium atom. UTC is available everywhere around the globe from several distribution systems, such as, GPS (USA satellites system), GLONASS (Russian Federation satellites system), and in the future by Galileo (European Union and Japanese satellites system). There are other means for distribution of UTC, such as, radio stations, CDMA cellular telephone system and TWTFT(Two-Way Satellite Time and Frequency Transfer) technique based on communications satellites [9]

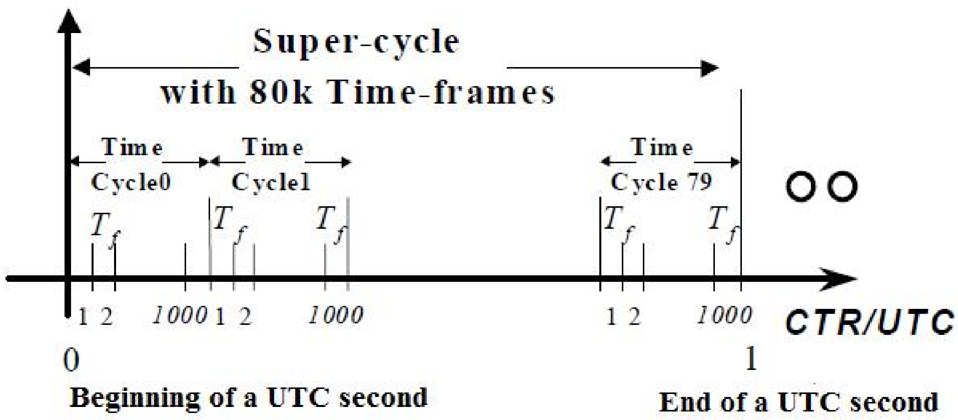

Figure 2.2: *The Common Time Reference (CTR) with* $d = 12.5\mu s$

2.2.2 Synchronous Virtual Pipe

A Synchronous Virtual Pipe (SVP), is created by reserving switching and transmission capacity along its path during a set of TFs, which reoccurs with a predefined periodicity. If there are no feasible schedules while there is available capacity, the SVP is considered blocked. In pipeline forwarding (PF) data units are forwarded in SVPs composed of an integer number of TFs within a TC: in other words, data is transmitted through TDS networks one hop every predefined integer number of TFs,(i.e. every TC. SVPs), as fractions of a λ, can be allocated with the proper size to satisfy the specific needs of the access networks to which a λ fraction is connected. The SVP is equivalent to a leased line in circuit switching. It provides switching and transport for kinds of data units from both the packet switching world (e.g., IP packets, ATM cells, Ethernet frames) and the circuit switching world (e.g., SONET/SDH frames). Data units of a different kind can coexist on the same communications channel during different TFs (i.e., carried over different SVPs).

As already mentioned above, today's switching elements are able to optically switch only whole wavelengths without intervening on single packets. For example, in an N-node network every node has to use a different wavelength for each of the $N-1$ possible destinations. Thus since each node pair requires their own wavelength, it would be necessary to dispose/offer $N-1$ wavelengths for the first node, $N-2$ for the second node, and so on. The total number of wavelengths required hence would be:

$$\sum_{i=1}^{N-1} i = \frac{N}{2}(N-1)$$

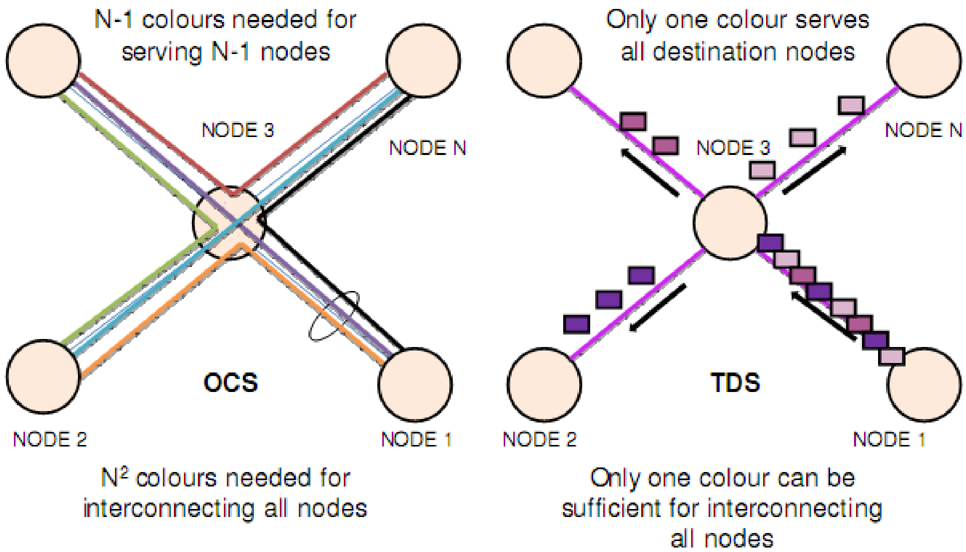

Figure 2.3: *The OCS N^2 Problem is solved in TDS networks*

Switching of whole wavelengths is inefficient and costly. Consequently, such optical core will be relatively small. In TDS it is theoretically possible to interconnect all nodes with only one colour. Instead of using whole lambda, each node uses a proper number of time frames to stablish the connection with the other nodes. Moreover, it enables operations such as multiplexing and demultiplexing: time frames of the same colour can exit on different outlets or enter the same switching port.

2.3 TDS Node Structure

2.3.1 Time Driven Switch Architecture

Here we present a possible architecture for a time driven switch with 16 ports, each featuring 16 colours, show in Figure 2.4.

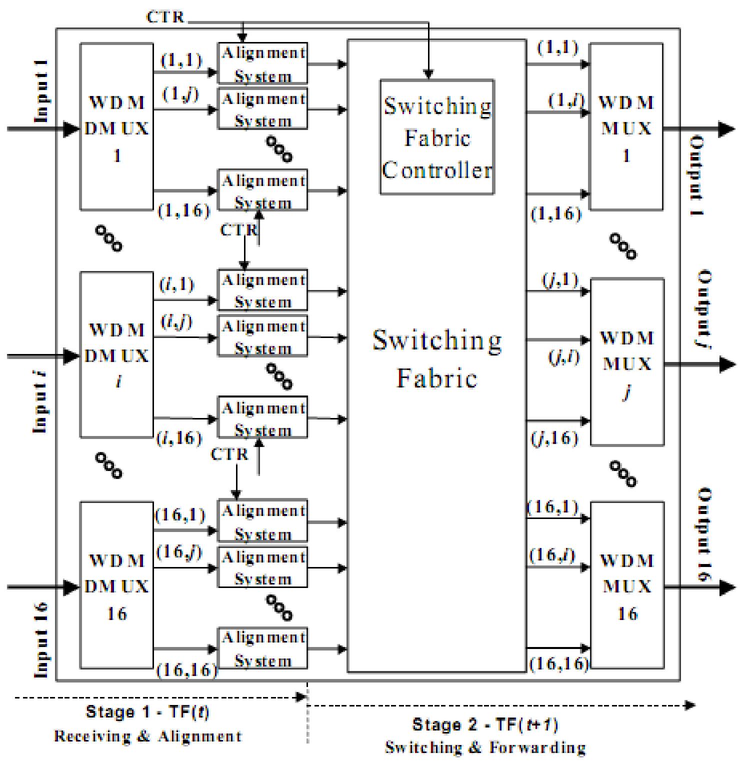

Figure 2.4: *TDS Switch Architecture and Operating Stages*

The switch operates in 2 stages, each having the duration of 1 TF:

- In stage 1, data units belonging to a TF are received on each wavelength, separated by a WDM de-multiplexer (DEMUX), and are aligned to the CTR by an alignment system, which also provides input buffering of data units until they can be switched.
- In stage 2 all the data units are transferred through the switching fabric to

their corresponding output port. At the output port data units are transmitted on their selected λ through a WDM MUX.

2.3.2 Alignment Issue

Simpler switching control and higher switching fabric utilization (that is independent of data unit format and switching technology) can be obtained by having aligned fixed size switching units - where transfer to the output ports of switching units starts and ends concurrently from all the input ports. TFs, the switching unit used in SVP, have a fixed size and, as explained in the foregoing, TFs on all the switches' output links are aligned to UTC. However, TFs at the input ports of a time driven switch are usually not aligned: TFs are aligned to UTC at the transmitting end of all links, but the propagation delay across those links will most likely not be an integer multiple of the TF duration. Hence, TFs at the input ports are aligned to the unique time reference (UTR) of the link. As shown in the Time-Driven switch architecture depicted in Figure 2.4, data units received on the input links need to be aligned to the CTR before being transferred through the switching fabric. During each TF, a TF's carrying data units is switched from each input port to the proper output port, the transfer starting and ending concurrently for all the input/output port pairs [9].

2.3.3 Delay Problem

The delay experienced by data units of a given connection is predefined in a deterministic manner by imposing that the delay between an input port of one node and the input port of the next node is a predefined integer of TFs. Thus we define delay as a sum of propagation time and switching time. The maximum variation of the delay (i.e the jitter), experienced by data units through a TDS network, is one TF due to alignment: as explained in [9], the actual UTC accuracy requirement is $\pm\frac{1}{2}TF$.

2.4 TDS Network Architecture and Deployment

The wide deployment of the current Internet based on asynchronous packet switching makes any changes to its working principles(and consequently to network devices implementing them), impractical for the TDS network environment. Moreover, the current asynchronous Internet is well suited to traditional data traffic, e.g., e-mail messages, file transfers, web browsing, etc.

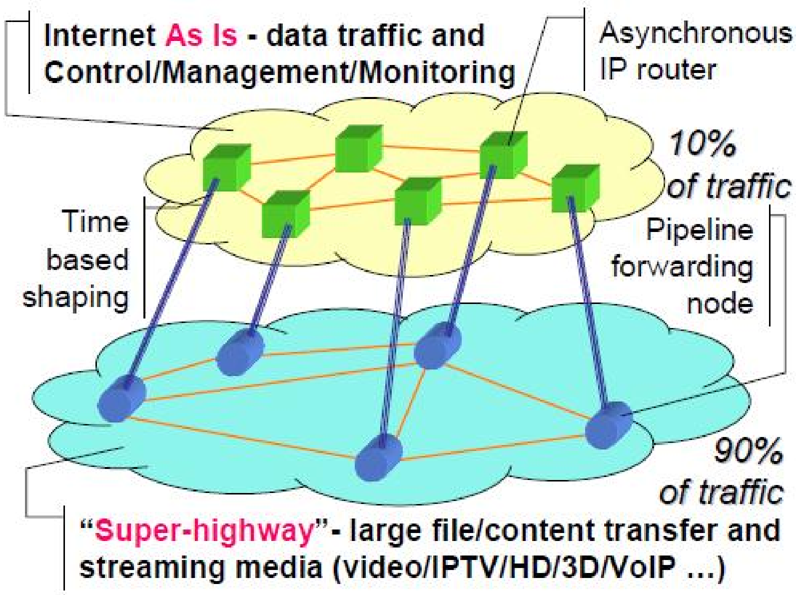

Figure 2.5: *Parallel Networks on the same Fiber Infrastructure with TDS*

One of the possible solutions discussed in [10] is the creation of a parallel network (on the same fiber infrastructure with WDM) based on pipeline forwarding that coexists with asynchronous IP technology. Besides carrying typically "best-effort" traffic, such as e-mails, low priority web browsing and file transfers, traditional (asynchronous) IP routers are used to transport the signaling required to setup Synchronous Virtual Pipes in the pipeline forwarding parallel network. The latter carries traffic requiring a deterministic service, such as phone calls, video on demand, videoconferencing, and distributed gaming. Given the large bandwidth required by most of such video-based services, and especially the ones still to come in the near future, such as 3D video and virtual reality, they are expected to account for more than 90% of future Internet traffic. The current IP infrastructure or its evolution, notwithstanding its limited scalability, is able to support the remaining traffic as it will be only a small fraction (10% of the total) and will not require a service with deterministic quality, i.e., it will basically require the same service (i.e., "best-effort" or differentiated) and capacity currently provided by the Internet.

2.5 Performance

The key issue in TDS networks is scheduling for each TF in the TC, such that each TF has the proper resources at each switch along the path. Scheduling is a fundamental problem since if a TF arriving to the input cannot be scheduled for transmission from the output, then for the flow using that TF admission must be denied, which means that it is blocked or, to introduce the terminology we use here, it is "time-blocked".

In fact, the problem of blocking can be decomposed into three factors:

1. there are no resources at the output: this is the traditional problem faced in circuit switching networks usually named **call blocking**;
2. resources are available at the output, but the internal structure of the switch prevents connecting the input and the output, this is normally called **space blocking**;
3. resources are available at the output, and the switch can be configured to connect the input and the output, but it is impossible to find a feasible schedule mapping the input TF to the output TF, this is **time-blocking**.

Time-blocking can be avoided (or at least reduced) by buffering, and indeed in TDS switches if k buffers(with k TFs per time cycle) are used, time-blocking is zero. In the following, we focus our attention on the time blocking and call blocking distributions, neglecting the space blocking. Since we assume switching matches as strictly non blocking architecture.

2.6 TDS Forwarding techniques

The forwarding of an SVP implies the research of free TFs in every link composing the path: if there is no possibility to find free TFs, the connection request will be rejected. There are various techniques for creation of an SVP ad they are called "Forwarding Techniques".

2.6.1 Immediate Forwarding Technique (IF)

In the Immediate Forwarding regime, data units of a given connection c in $TF(i)$ are moved to their output port and sent out in $TF(i+1)$. Once a TF is reserved on a wavelength of a link, this wavelength is not available any more for all the other connections which have to use a TF on their way towards the destination node. Moreover, if the $TF(i+1)$ of the output port is already occupied, no other connections can use that output port in $TF(i+1)$; also, these connections do not have the $TF(i)$ as an optional TF to be used (assuming to have only one lambda in output). The delay of the received packets delivery is equal to the number of the traversed switching elements multiplied per time frame duration d . We have to add the propagation delay from one node to another.

Although characterized by the advantage of no buffer requirements, one of the serious limitations of this technique is the high blocking ratio, even in the situations where there are enough free TFs. In case of available resources both at the input and output of the switch (but not adequate for IF), we used to call these TFs as not schedulable. The adoption of the IF-technique reduces the flexibility of the scheduling process, thus increasing the loss probability.

2.6.2 Non Inmediate Forwarding Technique(NIF)

Data units of a given connection c received in $TF(i)$ are moved to their output port and sent out in $TF(i + k_c)$ with $(k_c > 1)$. The flexible selection of k_c enables flexible scheduling with low time blocking probability. Note that if k_c is equal to the total number of time frames in the time cycle there is no time blocking, given by the possibility of using every free TF of a TC. Since with $k_c \leq k$ it is always possible to find a schedule at the expense of longer delay, while preserving constant jitter of one TF.

2.6.3 Comparison of the Forwarding Techniques

	Inmediate Forwarding (IF)	Non Inmediate Forwarding(NIF)
Buffering	Not needed	Very High with (Kc=k) High with(Kc < k)
Loss Probability	High	Low with (Kc = k) Medium (Kc < k)
Node Complexity	Low	Very High with (Kc =k) High with(Kc < k)
Maximun Delay	H*d	H*d*k with (Kc = k) H*d* Kc (Kc < k)
Resource Utilization	Low	Optimal with (Kc = k) High with(Kc < k)

Table 2.1: *Forwarding Techniques Comparison (H = #hops; k = #TFs in a TC; d = duration of a TF)*

In Table 2.1 we summarize the main characteristics of the following techniques, in terms of buffering, loss probability, node complexity, maximum delay and resource utilization.

In our study both techniques will be considered, in both the static (see chpater 3)and dynamic (see Chapter 6) traffic scenarios.

2.7 Wavelength Selection in TDS Networks

TDS operations can be performed in two modes:

- *No Wavelength Interchange (NWI)*, i.e. time frames belonging to the same SVP have to use the same colour;
- *Full Wavelength Interchange (FWI)*, i.e. one SVP can contain time frames on different colours on different links.

The usage of the wavelength converters increases the flexibility and the network scalability, but at the same time, impacts the node complexity which leads to a higher cost and power consumption. In our study the NWI is present in all analysis.

2.8 Routing scenarios for TDS

In our work the following scenarios are analised in detail under static and dynamic traffic conditions:

Single Path - Multiple Wavelengths (SP-MW): All the TFs associated to a connection are routed over the same path (single SVP) but they can use different wavelengths.

Single Path - Single Wavelength (SP-SW): All the TFs associated to a connection are routed over the same path (single SVP) and they have to use the same wavelength.

Multiple Path (MP): All the TFs associated to a connection are routed over different paths (i.e., different SVPs) and they can use different wavelengths. we analize all the aproches leaving FWI out of our study (see Chapter 6) .

2.9 TDS Related Works

Authors in [9] introduce Fractional Lambda Switching and studies its blocking issues. This work studies also the probability of call blocking as a function of link utilization, defining it as a ratio of available and non available TFs. [13] presents the mathematical formulation of the NIF scheduling problem, under a wide variety of networking requirements, then it introduces an efficient (i.e., having at most polynomial complexity) search algorithm that guarantees to find at least one schedule whenever such schedule exists. Authors in [12] study further blocking issues related to Time-Driven Switching (TDS). The performance analysis considers multiple scenarios with variety of optical transmission parameters and switch parameters. [11] shows that there is a huge potential for decreasing Future Internet power requirements by synchronizing the operation of routers and scheduling traffic in advance, thus reducing complexity (e.g., header processing, buffer size, switching fabric speedup and memory access bandwidth speedup). This article presents TDS as a possible approach for "trading" global time for electricity utilized by the global Internet. In [10], the authors provide a general closed-form analysis of the time-blocking probability in time-driven switching networks. The analysis yields the exact blocking probabilities for all possible scheduling delays and under different loading conditions for an isolated node.

Chapter 3

Models and heuristics for the design of TDS network

Since some years ago, research on optical networks has been investigating design and optimization techniques. The various proposed solutions can be classified into two main groups: heuristic methods and exact methods. The former return suboptimal solutions that in many cases are acceptable and have the advantage of requiring a limited computational effort. The latter are much more computationally intensive and do not scale well with the network size, being even not applicable in some cases. However since the exact methods are able to identify the absolute optimal solution, they play a fundamental role either as direct design network tools or as benchmarks to validate and test heuristic methods. In this chapter, we present the design optimization of TDS network applying the two kinds of methods mentioned above.

The TDS network design can be carried out with different techniques according to the type of traffic the network has to support. We investigate the static traffic case in which a known set of permanent connection requests is assigned *a priori* to the network, forming the offered traffic matrix virtual topology (alias virtual topology). Each request is for one or more point-to-point optical circuits (SVPs) able to carry a given capacity from the source termination to the destination termination.

With regard to the SVP treatment. We assume a periodic SVP (or periodic pipeline forwarding [9]), that is to say that the sequence of TFs along the path are periodically used in each TC, so that the wavelength capacity used is determined by the TFs used in a TC. For instance, with 100 TFs per TC and 80TC per super-cycle in a 10Gbps wavelength, if the same TF is periodically used in each TC then this reserves a hundredth of the wavelength capacity. In this way, it's not important to know the number of TC in a super-cycle.

3.1 TDS network optimization by ILP

As exact method, we decided to use the ILP due to its flexibility in varying the objective function. WDM network optimization by ILP has been widely studied in literature. We can subdivide research contributions in two groups according to the type of networks they are applied to: WDM networks with single-fiber links and multifiber WDM networks. We are in the first group where the problem consists in optimal Routing and Wavelength Assignment (RWA) of the lightpath. We analised TDS network optimization as an extended RWA problem called Routing, Wavelength and Timeframe Assignment (RWTA) of the SVPs. So, RWTA problem implies not only "wavelength continuity" (which we have decided to adopt for TDS (NWI)), but also an additional constraint: "time continuity" which must be guarantee to map in the right way the sequence of TFs between the source and destination.

A recent work [14] have shown a power consumption optimization of TDS newtwok by means of ILP formulation. The authors have treated the traffic as an amount of Gbit/s. Unlike that, we introduce the time issues in our formulations, i.e., TFs allocation and the correponding implications for transporting it (propagation delays across the links and optical switching delays in the nodes), as a result, the traffic is treated as an amount of needed TFs.

RWTA problem can be accomplished in two basilar and well-known approaches [15]: *flow formulation* (FF) and *route formulation* (RF). In the former the basic variables are the flows on each link relative to each source-destination OXC pair; in the latter the basic variables are the paths connecting each source-destination pair of one lightpath. Both these formulations have been used for the analysis with immediate forwarding technique, while only the former is considered for the analysis with non immediate forwarding technique. All the ILP formulatios are modelled only in a single path and multiple wavelength scenario described in chapter 2.

Let us formally define the RTWA problem: Given a single (optical) layer WDM network consisting of OXCs connected by WDM links and a set of traffic demands among the nodes, the total number of wavelength and number of transponders used in the network, are minimized, while satisfying two main constraints: i) all the requests have to be routed, and ii) each fiber supports limited number of wavelengths.

3.1.1 Flow Formulation

In this approach we consider the following inputs:

- $G = (N, A)$ is the physical topology of the network, consisting of a set N of nodes, corresponding to OCXs, and a set A of bidirectional arcs, corresponding to the fiber links connecting the nodes; moreover, we define N_m as the set of all nodes adjacent to node m, i.e., $N_m := \{n \in N | (m, n) \in A\}$;
- R is the set of traffic requests; source and destination nodes of request $r \in R$ and the corresponding required bandwidth are indicated as $s(r)$, $d(r)$ and t_r, respectively;
- T is the set of TFs. Its cardinality depends on the TF granularity chosen.
- L is the set of wavelengths carried by each fiber (for sake of simplicity, we consider mono-fiber links); C is the capacity, in TF per TC, of each wavelength;

- $E = \{1, 2\}$. Index $e \in E$ is used to characterize the propagation delay through links of the path. For each link (m,n), $e = 1$ means the allocation of traffic in the TF before delay and $e = 2$ means the allocation of traffic after undergoing the delay.

For the NIF formulation we define an additional set:

- B as the set of number of buffers that can be used at each node.

Immediate-Forwarding Flow Formulation (IF-FF)

Let us consider first the problem in which the use of buffers at the node is totally disabled, i.e. IF scenario. The problem formulation can be modelled with 3 sets of binary variables, $X^r_{m,n}$, $y^{r,e}_{m,n,l,t}$ and $\lambda_{m,n,l}$, and 3 sets of integer variables, Y^r_l, $S_{m,l}$ and $D_{m,l}$. These variables are defined as follows:

- $X^r_{m,n}$: Equal to 1 if link $(m, n) \in A$ is used by the request $r \in R$ (binary).
- $y^{r,e}_{m,n,l,t}$: Equal to 1 if TF $t \in T$ of the wavelength $l \in L$ is used to serve traffic demand of $r \in R$ on the link $(m, n) \in A$ at the end point $e \in E$ (binary).
- $\lambda_{m,n,l}$: Equal to 1 if at least one TF of the wavelength $l \in L$ is used in the link $(m, n) \in A$ (binary).
- Y^r_l: Amount of traffic (in TFs) of the request $r \in R$ routed on wavelength $l \in L$ (integer).
- $S_{m,l}$: Number of transmitting transponders at wavelength $l \in L$ installed in node $m \in N$ (integer).
- $D_{m,l}$: Number of receiving transponders at wavelength $l \in L$ installed in node $m \in N$ (integer).

The resulting formulation is:

$$minimize \quad \epsilon_1 \cdot \sum_{m\in N} \sum_{l\in L} (S_{m,l} + D_{m,l}) + \epsilon_2 \cdot \sum_{(m,n)\in A} \sum_{l\in L} \lambda_{m,n,l} \tag{3.1}$$

subject to:

$$\sum_{n \in N_m} X^r_{m,n} - \sum_{n \in N_m} X^r_{n,m} = \begin{cases} 1 & \text{if } m = s(r) \\ -1 & \text{if } m = d(r) \\ 0 & \text{otherwise} \end{cases} \qquad \forall m \in N, r \in R \tag{3.2}$$

$$\sum_{r \in R} y^{r,e}_{m,n,l,t} \leq 1 \qquad \forall (m,n) \in A, l \in L, t \in T, e \in E \tag{3.3}$$

$$y^{r,1}_{m,n,l,t} \leq X^r_{m,n} \qquad \forall (m,n) \in A, l \in L, t \in T, r \in R \tag{3.4}$$

$$\sum_{\substack{n \in N_m \\ t \in T}} y^{r,1}_{m,n,l,t+1} - \sum_{\substack{n \in N_m \\ t \in T}} y^{r,2}_{n,m,l,t} = \begin{cases} Y^r_l & \text{if } m = s(r) \\ -Y^r_l & \text{if } m = d(r) \end{cases} \qquad \forall m \in N | (m = s(r) \vee d(r)), l \in L, r \in R \tag{3.5}$$

$$\sum_{\substack{n \in N_m \\ m \neq d(r) \wedge s(r)}} y^{r,1}_{m,n,l,t+1} - \sum_{\substack{n \in N_m \\ m \neq d(r) \wedge s(r)}} y^{r,2}_{n,m,l,t} = 0 \qquad \forall m \in N, l \in L, r \in R, t \in T \tag{3.6}$$

$$\sum_{n \in N_m} y^{r,1}_{m,n,l,t} = \sum_{n \in N_m} y^{r,2}_{m,n,l,t'} \qquad \forall m \in N, r \in R, l \in L, t \in T \tag{3.7}$$

$$\sum_{l \in L} Y^r_l = t_r \qquad \forall r \in R \tag{3.8}$$

$$\sum_{\substack{t \in TF \\ r \in R}} \sum_{\substack{n \in N_m \\ m = s(r)}} y^{r,1}_{m,n,l,t} \leq S_{m,l} \cdot C \qquad \forall m \in N, l \in L \tag{3.9}$$

$$\sum_{\substack{t \in T \\ r \in R}} \sum_{\substack{n \in N_m \\ m = d(r)}} y^{r,2}_{m,n,l,t} \leq D_{m,l} \cdot C \qquad \in N, l \in L \tag{3.10}$$

$$\sum_{r \in R} \sum_{t \in T} y^{r,1}_{m,n,l,t} \leq \lambda_{m,n,l} \cdot C \qquad \forall (m,n) \in A, l \in L \tag{3.11}$$

$$\sum_{r \in R} \sum_{\substack{n \in N_m \\ m=s(r)}} y^{r,1}_{m,n,l,t} \leq S_{m,l} \qquad \forall m \in N, l \in L, t \in T \tag{3.12}$$

$$\sum_{r \in R} \sum_{\substack{n \in N_m \\ m=d(r)}} y^{r,2}_{m,n,l,t} \leq D_{m,l} \qquad \forall m \in N, l \in L, t \in T \tag{3.13}$$

The cost function (3.1) to be minimized is composed by two terms: the first term computes the number of transmitting and receiving transponders required to support the traffic demands and the second term determines the number of wavelength used in the network. The two terms are multiplied by two very small multiplicative constant. When the number of wavelengths is minimized, ϵ_1 is used, alternatively ϵ_2 is used to minimize the number of trasnponders. Constraints (3.2) and (3.4) ensure the traffic demand r uses a single path regardless of the wavelength used. Note that constraint (3.4) applies only to the end point of the link before propagation (i.e., $e = 1$), since this restriction is imposed indirectly to the end point after propagation $e = 2$ by applying constraint (3.7). Constraint 3.3 impose that the TF of a specific wavelength has to be used only once for each unidirectional link. This is considered at the both extreme nodes of the link (i.e. before and after propagation on the link).

Equations (3.5) and (3.6) together form a solenoidality constraint for physical flows. But here compared to the usual flow-conservation used, it is necessary to ensure not only flow conservation (i.e., total outgoing traffic should be equal to total incoming traffic except for source and destination nodes for a certain lambda l, ensuring also the wavelength continuity) but also time conservation (i.e. each TF in ingress must be mapped to the corresponding output TF, plus one due to switchig delay) at all nodes of the network.

Equation (3.7) is the propagation constraint on the link. As we mentioned earlier, index e is used to validate the propagation condition on the network, so that, for each TF on the link we make equal its allocation t before propagation delay and its allocation $t' = (t + w_{m,n} + 1)$[1] after crossing the link. The delay of the link (m, n) is has given *a priori*.

Constraint (3.8) defines, for each request, the amount of traffic to be routed over each wavelength. For each wavelength, equations (3.9) and (3.10) compute the number of transmitting and receiving transponder used in each node, respectively; constraint (3.11) identifies which wavelengths have been used in each link. Finally, constraints equation (3.12) satisfies an important condition at source nodes, respectively, that is: for the source nodes, if the outgoing traffic does not exceeed the wavelength capacity (i.e., use a single transponder), then the TF t of all links adjacent to the node can be used only for one link, so, this constraint prevents the source nodes from using more transponders than it should use, since TDS switches are able to forward traffic along different liks by means of one transponder. The analogous constraint (3.13) is applied for the incoming traffic of the destination nodes.

It's worth noting that wavelength capacity constraints are not necessary, since implicitly the index $t \in T$ in $y^{r,e}_{m,n,l,t}$ indicates the capacity on each link.

Non-Immediate-Forwarding Flow Formulation (NIF-FF)

In this model, it is considered that the nodes are equipped with buffers. The new variabile $y^{r,u,e}_{m,n,l,t}$ replaces the variable $y^{r,e}_{m,n,l,t}$ in all the constraints, and it is defined as:

- $y^{r,u,e}_{m,n,l,t}$: Equal to 1 if TF $t \in T$ is used to serve traffic demand of $r \in R$ in wavelength $l \in L$ on the link $(m, n) \in A$ at the end point $e \in E$. The index u is introduced to represent the delay inserted by the node m and can take values from 0 to $|B|$.

The objective function is the same expounded for IF-FF and the set of constraints is the following:

[1]all arithmetic operations involving TF numbers are modulo C (the wavelengnth capacity); for example, if i is a TF number, then $(i + 1)$ means $(i + 1) \bmod C$;

$$\sum_{n\in N_m} X^r_{m,n} - \sum_{n\in N_m} X^r_{n,m} = \begin{cases} 1 & \text{if } m = s(r) \\ -1 & \text{if } m = d(r) \\ 0 & \text{otherwise} \end{cases} \qquad \forall m \in N, r \in R \tag{3.14}$$

$$\sum_{r\in R} y^{r,0,e}_{m,n,l,t} + y^{r,1,e}_{m,n,l,(t-1)} + y^{r,2,e}_{m,n,l,(t-2)} + \cdots + y^{r,U,e}_{m,n,l,(t-U)} \leq 1 \qquad \forall (m,n) \in A, l \in L, t \in T, e \in E \tag{3.15}$$

$$y^{r,u,1}_{m,n,l,t} \leq X^r_{m,n} \qquad \forall (m,n) \in A, l \in L, t \in T, r \in R, u \in B \tag{3.16}$$

$$\sum_{n\in N_m} y^{r,0,1}_{m,n,l,t} + y^{r,1,1}_{m,n,l,t} + \cdots + y^{r,U,1}_{m,n,l,t} \leq 1 \qquad \forall m \in N, l \in L, t \in T, r \in R \tag{3.17}$$

$$\sum_{\substack{t\in T\\ n\in N_m}} (y^{r,0,1}_{m,n,l,t+1} + \cdots + y^{r,U,1}_{m,n,l,t+1}) - \sum_{\substack{t\in T\\ n\in N_m}} (y^{r,0,2}_{n,m,l,t} + \cdots + y^{r,U,2}_{n,m,l,t}) = \begin{cases} Y^r_l & \text{if } m = s(r) \\ -Y^r_l & \text{if } m = d(r) \\ 0 & \text{otherwise} \end{cases} \tag{3.18}$$

$$\sum_{n\in N_m} (y^{r,0,1}_{m,n,l,t+1} + \cdots + y^{r,U,1}_{m,n,l,t+1}) - \sum_{n\in N_m} (y^{r,0,2}_{n,m,l,t} + \cdots + y^{r,U,2}_{n,m,l,t+U}) = 0 \qquad \forall m \in N, l \in L, r \in R, t \in T \tag{3.19}$$

$$\sum_{l\in L} Y^r_l = t_r \qquad \forall r \in R \tag{3.20}$$

$$\sum_{n\in N_m} y^{r,u,1}_{m,n,l,t} = \sum_{n\in N_m} y^{r,u,2}_{m,n,l,t'} \qquad \forall m \in N, l \in L, t \in T, e \in E, r \in R, u \in B \tag{3.21}$$

$$\sum_{\substack{t\in T\\ r\in R\\ u\in B}} \sum_{\substack{n\in N_m\\ m=s(r)}} y^{r,u,1}_{m,n,l,t} \leq S_{m,l}\cdot C \qquad \forall m\in N, l\in L \tag{3.22}$$

$$\sum_{\substack{t\in T\\ r\in R\\ u\in B}} \sum_{\substack{n\in N_m\\ m=d(r)}} y^{r,u,1}_{m,n,l,t} \leq D_{m,l}\cdot C \qquad \forall m\in N, l\in L \tag{3.23}$$

$$\sum_{r\in R}\sum_{t\in TF}\sum_{u\in B} y^{r,u,1}_{m,n,l,t} \leq \lambda_{m,n,l}\cdot C \qquad \forall (m,n)\in A, l\in L, t\in T \tag{3.24}$$

$$\sum_{\substack{r\in R\\ u\in B}} \sum_{\substack{n\in N_m\\ m=s(r)}} y^{r,u,1}_{m,n,l,t} \leq S_{m,l} \qquad \forall m\in N, l\in L, t\in T \tag{3.25}$$

$$\sum_{\substack{r\in R\\ u\in B}} \sum_{\substack{n\in N_m\\ m=d(r)}} y^{r,u,1}_{m,n,l,t} \leq D_{m,l} \qquad \forall m\in N, l\in L, t\in T \tag{3.26}$$

$$\sum_{\substack{t\in T\\ r\in R\\ l\in L}} y^{r,u,1}_{m,n,l,t} \leq B_u \qquad \forall u\in B \tag{3.27}$$

Constraints (3.14) and (3.16) have the same function of (3.2) and (3.4). Moreover, since each node can use buffers, it is necessary to sum all the variables that represent the same TF t and then, enforce this sum to be less than or equal to 1, in this way, constraint (3.15) imposes that the TF is used only once. This must be done at both end points of the link. At the output of each node on the link, equation (3.17) ensures that only one of all possible values of buffers ($u \in B$) is used for each TF $t \in T$ in wavelength $l \in L$ and also for each request $r \in R$. The expressions in (3.18) and (3.19) together form solenoidality constraint, but unlike the constraints (3.5) and (3.6) in the IF-FF, here in ingress at intermediate node m, all the variables that represent the same TF t (i.e., $y^{r,0,2}_{n,m,l,t} + \cdots + y^{r,U,2}_{n,m,l,t+U}$) must be added and equal to the variables of TF $t+1$ (i.e., $y^{r,0,1}_{m,n,l,t+1} + \cdots + y^{r,U,1}_{m,n,l,t+1}$) in egress.

Constraints (3.20)-(3.26) are similar to the previous (3.8)-(3.13) and in the end, we can find the constraint (3.27) that computes the number of buffers used in each node.

3.1.2 Route Formulation

The Flow Formulation stated above can be rewritten as a route-based formulation. In the Route Formulation, if it is acceptable that routing is peformed in a constrained way, then the solution complexity can be controlled. For example, all the TFs can be constrained to be routed along the first k-shortest paths connecting the source to the destination. Differently from the flow formulation, the complexity of which is strictly dependent on physical and virtual topologies, the complexity of the route formulation decreases with the number of paths that can be employed to route the TFs.
Therefore, constrained routing is able to overcome this complexity limitation, but the solution they produce is only an approximation of the actual optimal network design. This route formulation aims at reducing the computational burden but leads us to an approximated solution instead of the exact one (as indeed does flow Formulation).

Immediate-Forwarding Route Formulation (IF-RF)

To begin, we introduce the following definitions and inputs:

- *inputs:*
  - $G = (N, A)$ is the physical topology of the network, consisting of a set N of nodes, corresponding to OCXs, and a set A of bidirectional arcs, corresponding to the fiber links connecting the nodes; moreover, we define N_m as the set of all nodes adjacent to node m, i.e., $N_m := \{n \in N | (m, n) \in A\}$;
  - R is the set of traffic requests; source and destination nodes of request $r \in R$ and the corresponding required bandwidth are indicated as $s(r)$,

$d(r)$ and t_r, respectively;

- T is the set of TFs. Its cardinality depends on the TF granularity chosen.

- L is the set of wavelengths carried by each fiber (for sake of simplicity, we consider mono-fiber links); C is the capacity, in TF/TC, of each wavelength;

- P is set of paths that connect the source $orig(p)$ and destination $dest(p)$ nodes of the graph.

- $P_{m,n}$: set of paths that traverse link $(m, n) \in A$.

- P_r: set of paths that connect source destination nodes of traffic demand $r \in R$. Three shortest paths between each source-destination couple are evaluated *a priori*.

- $\Delta^p_{(m,n)}$: is the accumulated delay of the path $p \in P$ that is computed *a priori* from the source node to link (m, n).

- <u>*variables:*</u>

  - $X^r_{p,l,t}$: Equal to 1 if path $p \in P$ is used with TF $t \in T$ on wavelength $l \in L$ for request $r \in R$. It is important to note that t represent the TF chosen at the beginning of the path (binary).

  - y^r_l: variable that represents allocated traffic of r on the wavelength $l \in L$ (integer).

  - x_p: Equal to 1 if the path $p \in P$ is used. Unlike $X^r_{p,l,t}$, this is just an auxiliary variable to guarantee that a request r uses a single path (binary).

  - $S_{m,l}$ *e* $D_{m,l}$: the usual variables of the number of transponders in transmission and in reception, respectively (integer).

The objective function is to find the number of wavelength and transponders deployed in the TDS network (as shown in IF-FF), such that:
such that:

$$\sum_{p \in P_r, t \in T} X^r_{p,l,t} = y^r_l \qquad \forall r \in R, l \in L \tag{3.28}$$

$$\sum_{l \in L} y^r_l = t_r \qquad \forall r \in R \tag{3.29}$$

$$\sum_{\substack{p \in P_{(m,n)} \\ r \in R}} \sum_{t \in T | (t + \Delta^p_{(m,n)} = t')} X^r_{p,l,t} \leq 1 \qquad \forall (m,n) \in A, l \in L, t' \in T \tag{3.30}$$

$$\sum_{\substack{t \in T \\ r \in R}} \sum_{p \in P | orig(p) = m} X^r_{p,l,t} \leq S_{m,l} \cdot C \qquad \forall m \in N, l \in L \tag{3.31}$$

$$\sum_{\substack{t \in T \\ r \in R}} \sum_{p \in P | dest(p) = m} X^r_{p,l,t} \leq D_{m,l} \cdot C \qquad \forall m \in N, l \in L \tag{3.32}$$

$$\sum_{\substack{r \in R \\ t \in T}} \sum_{p \in P_{(m,n)}} X^r_{p,l,t} \leq \lambda_{m,n,l} \cdot C \qquad \forall (m,n) \in A, l \in L \tag{3.33}$$

$$\sum_{r \in R} \sum_{p \in P | orig(p) = m} X^r_{p,l,t} \leq S_{m,l} \qquad \forall m \in N, l \in L, t \in T \tag{3.34}$$

$$\sum_{n \in N} \sum_{\substack{r \in R \\ p \in P | dest(p) = m}} \sum_{t \in T | (t + \Delta^p_{(m,n)} = t')} X^r_{p,l,t} \leq D_{m,l} \qquad \forall m \in N, l \in L, t' \in T \tag{3.35}$$

$$\sum_{p \in P_r} x_p \leq 1 \qquad \forall r \in R \tag{3.36}$$

$$r^r_{p,l,t} \leq x_p \qquad \forall r \in R, l \in L, t \in T, p \in P \tag{3.37}$$

The flow balance constraint as always are the ones to start with. It ensures that, the sum of all available paths for all TFs t is equal to the amount of traffic y^r_l requested (in TFs) by r on every wavelength l (3.28). In addition, the sum of all amounts of traffic on all the wavelengths has to be equal to the total traffic requested (3.29).

Constraint (3.30) ensures that the TF t' is used only once in the following way: for each link (m, n), wavelength l and TF $t' \in T$[2], the sum of all the TFs $t \in T$[3] that are the TF t' after adding the propagation delay ($\Delta^p_{(m,n)}$) at that link, has to be less than or equal to 1. Constraints (3.31) and (3.32) calculate the number of transmitting and receiving transponder used in each node, respectively; and constraint 3.33 identifies which wavelengths have been used in each link (m, n). Equations (3.34) and (3.35) in IF-RF have the same goal of (3.12) and (3.13)[4] in IF-FF, respectively. To keep non biforcated traffic condition, constraints (3.36) and (3.37) are applied for each request r.

Non Immediate Forwarding Route Formulation (NIF-RF)

Now, we explain why a NIF-RF was not developed. Imagine the following situation, we have a path composed by nodes 1-3-4-8, in this scenario each node could delay the traffic i.e., buffering (except 8 which is the destination node), so, if we adapt propagation constraint (3.29), we should add another term to the expression $t + \Delta^p_{(m,n)} = t'$ that takes into account the buffering done in previous nodes to link (m,n). In this way, we could control the delay added by each node to the traffic, however, this term is a variable and can not be added to the set of specifications of a summation in ILP.

[2] t' is the current TF at the link (m, n)

[3] t represents the TF in the wavelegnth l at the beginning of the path p for the request k

[4] To capture the same idea expressed in the constraint (3.13) for IF-RF, it's necessary at the destination node m to identify which TFs t of all possible paths represent the TF t' in all the adjacent links (n, m)

3.2 TDS network design optimization by heuristic algorithms

In this section, we provide a heuristic algorithm for minimizing the number of the wavelengths deployed for a traffic matrix in TDS network. This algorithm takes into consideration: forwarding techniques such as IF and NIF and scenarios such as SP-SW and MP expounded in Chapter 2.

3.2.1 Heuristic Overview

Firstly, a initial RWTA is obtained by adopting a set of heuristic criteria. In this phase the physical topology is dimensioned so that all the SVPs can be set up and all the TF requested are satisfied. Then, the network is globally optimized by trying to improve the first RWTA. As we have already pointed out in ILP formulations, we chose to consider a monofiber network in which all the fibers in the network host the same number of wavelengths W, chosen a priori. The variable capacity of the physical topology is represented by the number of wavelengths in each link. Therefore the cost function to be minimized is the total number of wavelengths in the network.

The routing, wavelength assignment and TFs allocation (or, RWTA) are performed by means of the algorithms BF-based, SP-SW-A, SP-MW-A and MP-A developed for dynamic traffic condition[5]. These algorithms permit to explore different scenarios of routing in the TDS architecture[6]. Note that the routing criteria is applied both in the first RWTA and in the optimization cycle phases.
In the whole design procedure a particular representation of the network has been adopted which is the *layered graph* introduced by [22],[23].

[5]Details of the three algorithms are given in Chapter 4
[6]The definition of these scenarios are in Section 2.8

3.3 Description of the algorithm

The network design procedure is composed by two main phases:

- Phase 1. the initial greedy RWTA and
- Phase 2. the subsequent optimization cycle.

3.3.1 Phase 1. Initial greedy RWTA

The initial greedy RWTA is obtained starting from the idle physical topology. All the connection requests of the logical topology are set up in sequence one after the other until all have been satisfied. To do this the initial idle physical topology is oversized: each link initially contains a large number of wavelengths W, which can be considered infinite. In this way we are guaranteed that the greedy TFs mapping leads to a feasible solution. Clearly, W must not be too large to limit unnecessary memory occupation. An appropriate value of W can be estimated by inspection. The initial TFs mapping requires that all the requested connections of the logical topology are sorted. There is no way to decide a priori how to sort connections in an optimal way. Therefore we proceeded in a heuristic sense by trying different sorting rules and deducing the best rule from the results. In particular, we took into account the following rules:

- "longest first": priority is given to connections whose source and destination nodes are farthest apart;
- "busiest pair first": the node pair requesting the maximum number of TFs is selected;
- "balanced": the previous rules are combined: the node pair to be served is selected according to the product of the distance and the number of TFs requests;

After the initial sorting, the connection requests are processed. The first request in the sorted list is satisfied by setting up an SVP in a greedy way i.e. by exploiting the available capacity at the best. Then the second is satisfied with the same procedure: the resources available for the second setup comprise the whole network minus those TFs already allocated to the first connection. Then the next connection request is done and so on until all the connections are activated.

The path routing (BF-based) chooses a path according to different metrics such as the number of hops (minimum hop - mH) or the sum of the physical lengths (minimum length - mL). The typical wavelength assignment criteria are: "Pack", "Spread" and "First Fit". "Pack" and "Spread" consider consider the utilization of wavelengths on the network and define a priority order promoting, respectively, the most and the least used wavelength in the network. "First Fit" creates an arbitrary and preset priority order for wavelength selection which is kept unchanged throughout the whole network. Finally, only First-Fit can be applied to TF selection, because we can not consider a sub-utilization of a TF for the other two criteria. In our algorthm, "First Fit" has been applied for both wavelength assignment and TF assignment. We decided to explore different scenarios of routing in the TDS architecture such as

3.3.2 Phase 2. Optimization cycle

After the greedy RWTA phase has been completed (starting from the idle network), all the empty wavelengths are pruned from the physical topology; correspondingly, all the TFs of the pruned wavelengths are "disabled". Then the optimization iteration begins. An optimization counter k is defined and initialized to 1. Any wavelength on each link containing only k occupied and $W - k$ unused TFs (let us name it a k-wavelength) is detected. Then the SVPs using one or more TFs in a k-wavelength are sequentially considered. Every SVP is temporary deallocated (i.e., all the TFs belonging to the SVP are dealloacted including those found in other wavelengths, if applicable) and the k-wavelength is temporary disabled. Each deallocated SVP is reallocated by performing a second RWTA adopting the same routing criteria of the greedy RWTA phase. In this case there is no guarantee of success of the reallocation, since the physical network has a constrained capacity.

If all the deallocated SVPs can be rerouted then the wavelength is permanently disabled; otherwise, initial SVPs are restored (i.e., TFs associated to SVPs are restored including those in other wavelength, if applicable) and the k-wavelength is preserved.

The rationale of this optimization cycle is the following. Given our cost function, it's inefficient to have partially empty wavelengths in the network; we attempt to free first the wavelengths with lower occupation i.e., wavelengths with a small number of TFs used (low k), and then the more highly loaded wavelengths where many TFs are assigned (high k).

We summarise the proposed heuristic optimization algorithm in the flow-chart of Figure 3.1. Since the number of iterations of the for cycle is limited and the number of k-wavelengths inspected in each iteration is finite, this algorithm terminates deterministically.

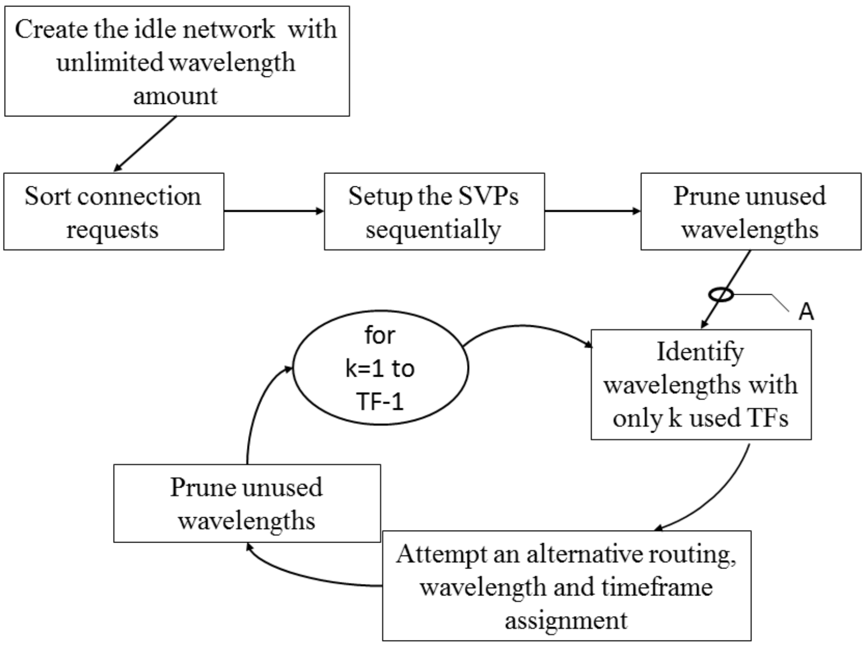

Figure 3.1: *Flow-chart describing the main steps of the heuristic algorithm for the design of TDS networks (A: Number of wavelengths used)*

The optimization cycle is applied to the network until no significant changes are presented in the total number of wavelengths used. We do this because once a connection is re-routed, the TFs of the wavelength (for SP-SW) or wavelengths (for SP-MW) in each link of the old path that were occupied by the connection, are freed. Thus, new opportunities for re-routing are generated for other connections.

Finally, we shall mention a limitation of the heuristic approach. The success of finding a new route in optimization cycle is highly dependent on the number of links available after the pruning. Therefore,as we increase the number of requests with different source and destination, the number of links availaible will also increase and then we could get a better solution in terms of wavelengths used.

3.4 Heuristic Complexity

We now analyze the computational complexity of the proposed algorithm. The single-shortest path for each TF is found using Bellmann Ford's algorithm which can be implemented in $O(|N| \cdot |N|)$ where $|N|$ and $|M|$ are the number of nodes and links, respectively. Therefore, the time complexity of the algorithm for each connection is $O(|W||T|(|N| \cdot |N|))$ as it is run once for each TF $t \in T$ of the wavelength ($w \in W$).

Chapter 4

Static Traffic: Results and Discussion

In this section, we will show and discuss illustrative results obtained by our ILP formulations and heuristic for TDS-based network routing, TF and wavelength allocation problem. The ILP was solved using a standard solver, namely ILOG CPLEX 11.0.1, based on the branch-and-bound method, whereas heuristic was solved using a C++ simulator we have developed.
The metric used for minimization in both approaches is *minimum Hop*.

4.1 8-node Network

Our first case-study is a 8-node network (Figure 4.1) that has 10 bidirectional links (in total 20 links) and average nodal degree of 20/8=2.5. We decided to use this small network mainly to face the IF Flow Formulation which has the largest number of variables and constraints, along with NIF Flow Formulation.

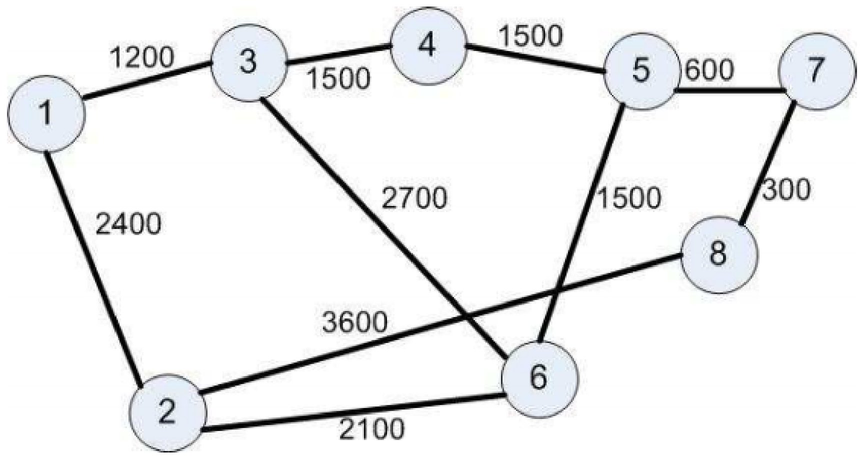

Figure 4.1: *8-node network topology (links lengths in km)*

	1	2	3	4	5	6	7	8
1	0	400	400	200	2200	600	800	2400
2	0	0	600	0	200	0	0	800
3	0	0	0	1000	1200	800	0	6200
4	0	0	0	0	0	200	0	0
5	0	0	0	0	0	0	0	200
6	0	0	0	0	0	0	200	0
7	0	0	0	0	0	0	0	600
8	0	0	0	0	0	0	0	0

Figure 4.2: *Traffic Matrix 1 for 8-node network (each entry is in units of Mbps)*

4.1.1 8-node network - IF Flow Formulation vs Route Formulation with variable TF granularity

We will use a non-uniform traffic matrix (as in Figure 4.2). The traffic matrix has the following characteristics:

- Number of requests: 18;
- Total traffic load: 19 Gbps
- Average traffic carried by one of 18 requests: 1.05 Gbps;

The matrix has a small amount of traffic because of flow formulation complexity.

A wavelength capacity is equal to 10Gbps and three different granularities will be considered:

- *Case 1.* 10 TF per TC, which represents a capacity of 1 Gbps per TF.
- *Case 2.* 20 TF per TC, which represents a capacity of 500 Mbps per TF.
- *Case 3.* 50 TF per TC, which represents a capacity of 200 Mbps per TF.

In general, we expect that the number of wavelengths used is smaller in case 3, as traffic can be groomed in a more efficient way. For example, for source-destination pair 1-3 the required traffic is 400 Mbps, this corresponds in the three cases to:

- *Case 1.* 1 TF requested, where a residual capacity of 600 Mbps is wasted.
- *Case 2.* 1 TF requested, here 100 Mbps capacity is wasted.
- *Case 3.* 2 TF requested, no capacity is wasted.

Before showing the comparison between IF-FF and IF-RF, we analyse the effect of the granularity in the final solution, when the traffic load increases (Figure 4.3). The three curves correspond to the three cases mentioned above and have been provided by IF Flow Formulation. In order to serve the traffic, four wavelengths in each link are used for T, 2*T and 4*T, while five and six wavelengths are used for 8*T and 10*T, respectively. We observe that:

- For 50TF/TC, we obtain the lowest values of the number of wavelengths used. So, this confirms that using a higher granularity (i.e., more TFs per TC) helps to have a better occupation of the network capacity.
- When the bandwidth request starts to grow, the effectiveness of having higher granularity is lost since waste capacity is negligible.

Now, we compare the Flow and Route Formulations with respect to computational time, and validate the Route Formulation using absolute and relative errors.

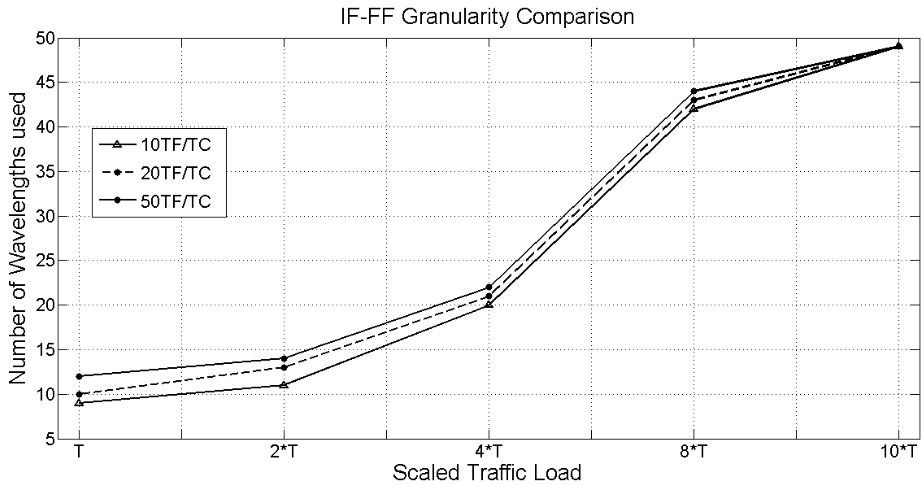

Figure 4.3: *8-node Network: Granularity comparison (with T=traffic load of traffic matrix in Figure 4.2)*

	Case 1. |T|=10 TF/UTC second		Case 2. |T|=20 TF/UTC second		Case 3. |T|=50 TF/UTC second	
Traffic Load	IF-FF	IF-RF	IF-FF	IF-RF	IF-FF	IF-RF
T	5,2d	21,9h	9,6d	2,5d	12,3d	4,7d
2*T	4,9d	21,3h	**9,2d**	2,1d	**12,1d**	4,4d
4*T	4,1d	18,4h	**7,4d**	1,5d	10,6d	2,7d
8*T	3,4d	15,3h	6,9d	1,2d	**10,5d**	2,3d
10*T	2,1d	6,1h	4,6d	12,2h	8,6d	1,3d

Table 4.1: *8-node Network Optimization: computational time (h and d stand for hours and days, respectively)*

With regard to computational time, Figure 4.1 summarises all the results, where computational times in bold are associated to runs failing in finding an optimal result. To clearly understand the reported data, a particular aspect of the ILP must be clarified. The branch-and-bound algorithm progressively occupies memory with its data structure while it is running. When the optimal solution is found, the algorithm stops and the computational time and the final memory occupation can be measured. In some cases, however, available memory is filled up before the optimal solution can be found. Sometimes the integer solution, when it is forced to be returned because of the limited amount of memory, is associated to the so-called gap parameter that expresses the percentage difference between the integer solution found and the minimal possible value the solution could reach (i.e. a lower bound returned by branch and bound algorithm). This parameter returns an estimation of the quality of the non-optimal integer solution found in terms of maximal possible distance from the optimum. From Table 4.1 we can see that Route Formulation always requires a shorter run duration than Flow Formulation.

Regarding computational performance of Route Formulation, the following parameters are introduced:

- W_{FF} (W_{RF}): total wavelength number returned by the ILP Flow (Route) Formulation;

- ϵ_W: percent relative error $100 \cdot (W_{FF} - W_{RF})/(W_{FF})$;

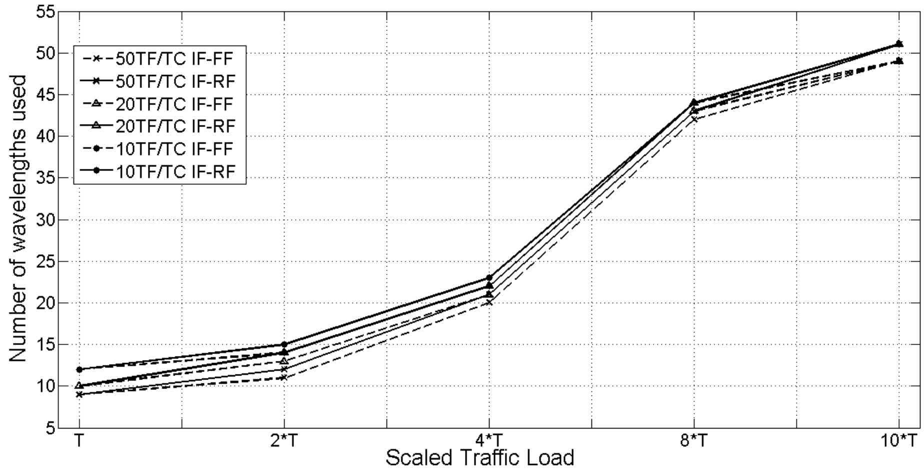

Figure 4.4: *8-node Network: Flow-Route comparison on the total number of wavelength used*

The percentage relative error between the two formulations obtained are represented in Figure 4.5 as functions of traffic. Convergence between the two formulations occurs in all the three cases for traffic load T. Route Formulation can be up to 9.1% worse than Flow Formulation, as shown for granularity of 50TF/TC,

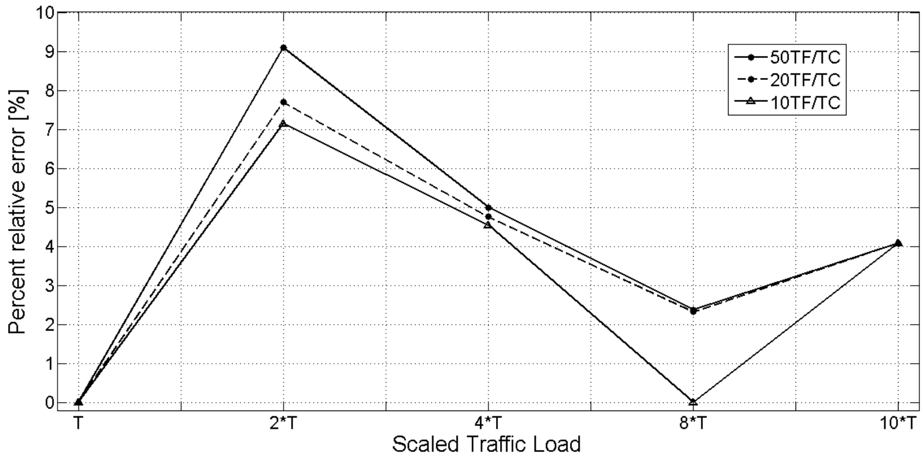

Figure 4.5: *8-node Network: Flow-Route comparison on the total number of wavelength used as percent relative error* ϵ_W

that is related to the paths availables for Route Formulation, since better solution could be found in another path we do not consider.

In summary, Route Formulation is a good approach not only for optimal solutions but also for computational times, hence, it can provide a useful benchmark to evaluate the performance of a heuristic approach.

4.1.2 Heuristic approach

We first benchmark the quality of our heuristic over the same network configuration as the ILP route formulation. As we said in Chapter 4, better results are obtained when there is a considerable amount of requests in the network[1], thus we introduce a new matrix that was randomly generated with probabilities of choosing traffic request of 200, 400 and 600 Mbps higher than for 800, 1000 and 1200 Mbps (Figure 4.6). This matrix has the following attributes:

[1]We also observed results with the previous small matrix that corroborate this assertion about heuristic

	1	2	3	4	5	6	7	8
1	0	400	400	200	1000	600	200	400
2	0	0	600	600	200	200	200	800
3	0	0	0	1000	1200	800	600	200
4	0	0	0	0	600	200	200	400
5	0	0	0	0	0	200	600	200
6	0	0	0	0	0	0	200	200
7	0	0	0	0	0	0	0	600
8	0	0	0	0	0	0	0	0

Figure 4.6: *Traffic Matrix 2 for 8-node network (each entry is in units of Mbps)*

- Number of requests: 28;
- Total traffic load for 28 requests: 13 Gbps;
- Average traffic carried by one of 28 requests: 464.28 Mbps;

Once again, we plot the number of wavelengths used (Figure 4.7) and the percentage relative error (Figure 4.8) obtained with the heuristic approach compared to the Route Formulation case. The most significant facts are the following:

- the results of the two techniques are quite close: the heuristic approach is able to provide good optimal and sub-optimal results[2] compared to Route Formulation;
- the maximum error is 15.39% and is obtained for traffic load of 4*T.
- On the other hand, we can see that after the maximum error, the curve starts to decrease. This means that in general the error tends to be low, so we can use it for realistic cases.

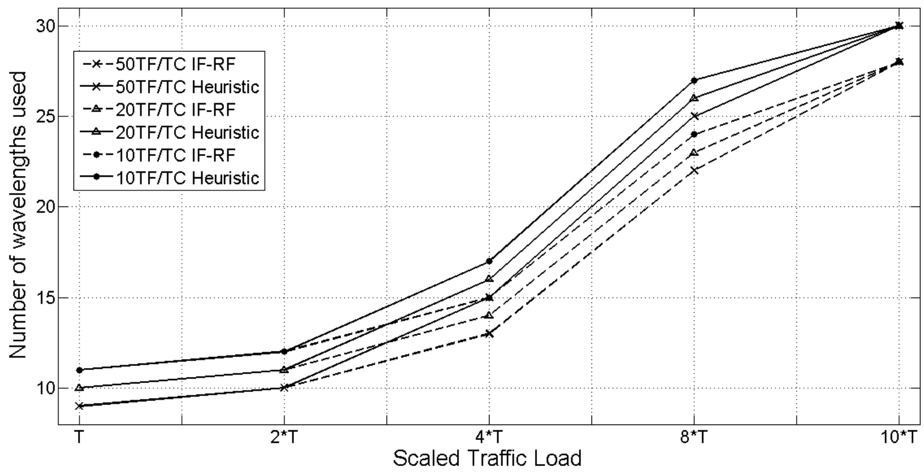

Figure 4.7: *8-node Network: Route Formultion - Heuristic comparison on the total number of wavelength used*

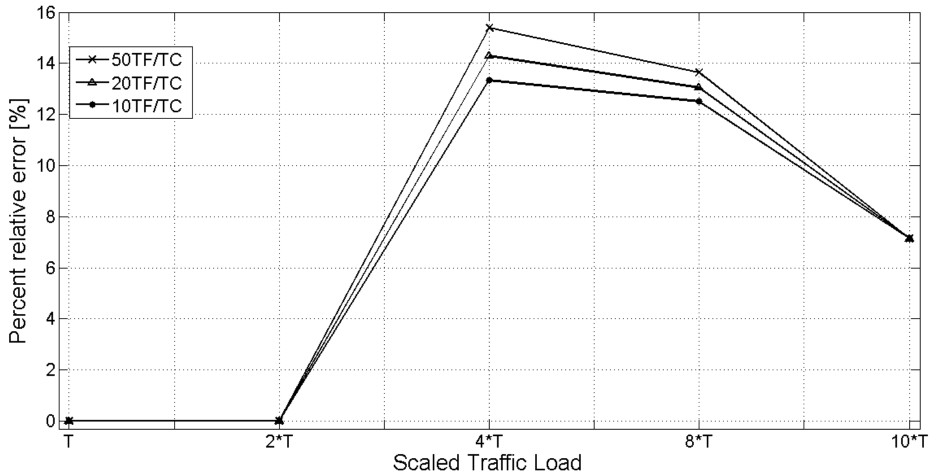

Figure 4.8: *8-node Network: Route Formultion - Heuristic comparison on the total number of wavelength used as percent relative error* ϵ_W

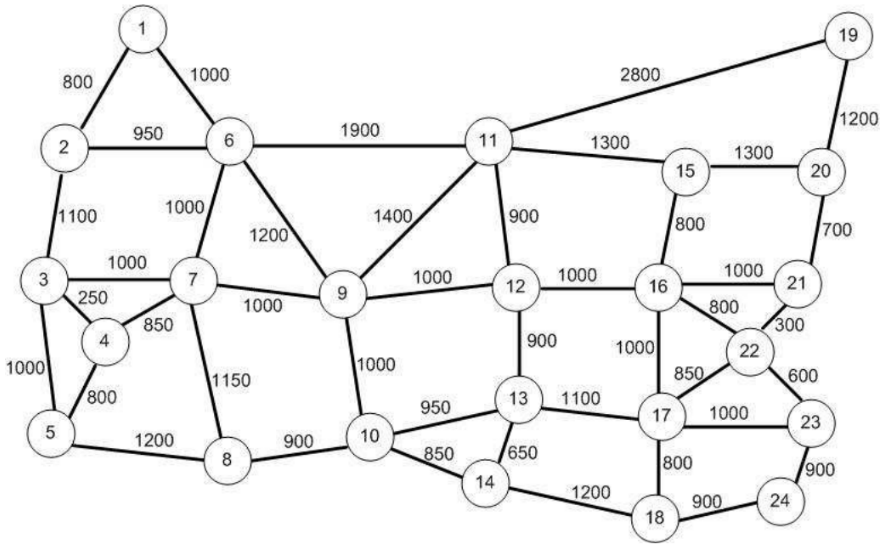

Figure 4.9: *24-node network topology (link lengths in km)*

4.2 24-node US backbone network

We now present results using heuristic algorithm for a larger network topology, i.e., the 24-node US backbone network (shown in Figure 4.9). This network has 86 bidirectional links (in total 43 links).

We are given a base traffic matrix in table 4.2:

- Number of requests: 453;
- Total traffic load for 453 requests: 99.3 Gbps
- Average traffic carried by one of 453 requests: 219 Mbps;
- Average nodal degree: 20/8 = 3.6

This topology is not only greater than the previous one, but also has a higher average nodal degree, i.e., it is 20/8 = 3.6.

[2] Remind that Route Formulation has a constrained routing, so, heuristic can obtain some similar values

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	0	100	100	100	100	300	200	200	0	200	500	100	200	0	100	200	100	0	200	500	0	700	1400	0
2	100	0	100	100	100	200	200	200	0	100	100	0	100	0	100	100	0	100	100	100	200	200	200	0
3	100	100	0	200	200	300	100	200	0	200	200	100	100	0	100	100	100	300	300	1700	300	500	200	100
4	100	100	200	0	300	300	100	200	0	300	200	100	100	0	100	100	100	200	100	100	400	100	0	0
5	100	100	200	300	0	200	100	100	0	200	100	0	100	0	100	100	100	1300	300	100	200	0	100	100
6	300	200	300	300	200	0	300	300	200	300	500	300	300	1700	300	500	200	500	100	200	200	100	500	200
7	200	200	100	100	100	300	0	300	0	200	200	100	100	0	100	200	100	300	0	200	200	100	100	100
8	200	200	200	200	0	300	300	0	0	200	300	100	200	0	100	200	100	100	0	100	200	100	200	0
9	0	0	0	0	0	200	0	0	0	0	100	100	0	0	0	100	0	1000	700	200	100	0	0	300
10	200	100	200	300	200	300	200	200	0	0	200	100	100	0	100	200	100	100	100	300	100	100	0	100
11	500	100	200	200	100	500	200	300	100	200	0	100	200	0	200	200	200	200	100	200	200	200	200	100
12	100	0	100	100	0	300	100	100	100	100	100	0	100	0	100	200	100	100	200	500	300	100	100	0
13	200	100	100	100	100	300	100	200	0	100	200	100	0	0	200	200	200	100	100	100	300	200	200	300
14	0	0	0	0	0	1700	0	0	0	0	0	0	0	0	0	0	0	100	0	100	200	100	200	200
15	100	100	100	100	100	300	100	100	0	100	200	100	200	0	0	200	100	300	300	200	300	500	300	300
16	200	100	100	100	100	500	200	200	100	200	200	200	200	0	200	0	200	100	100	0	100	200	100	200
17	100	0	100	100	100	200	100	100	0	100	200	100	200	0	100	200	0	0	100	200	100	200	300	0
18	0	100	300	200	1300	500	300	100	1000	100	200	100	100	100	300	100	0	0	200	300	100	200	400	100
19	200	100	300	100	300	100	0	0	700	100	100	200	100	0	300	100	100	200	0	100	100	300	100	0
20	500	100	1700	100	100	200	200	100	200	300	200	500	100	100	200	0	200	300	100	0	100	1400	300	300
21	0	200	300	400	200	200	200	200	100	100	200	300	300	200	300	100	100	100	100	100	0	100	100	0
22	700	200	500	100	0	100	100	100	0	100	200	100	200	100	500	200	200	200	300	1400	100	0	100	200
23	1400	200	200	0	100	500	100	200	0	0	200	100	200	200	300	100	300	400	100	300	100	100	0	100
24	0	0	100	0	100	200	100	0	300	100	100	0	300	200	300	200	0	100	0	300	0	200	100	0

Table 4.2: *24-node network Traffic Matrix*

In the following, we show the results in terms of total number of wavelengths used and average utilization[3], varying the offered traffic. The following scenarios are compared:

- Single path - Multiple wavelengths (SP-MW) vs Single path - Single wavelength (SP-SW): In the latter, there is a lower jitter as the information travels along a single path, but, this scenario is more restricted.
- Single path - Multiple wavelengths vs Multiple paths: In the latter, the jitter is inevitable because the traffic can use multiple paths. On the other hand, we expect the number of wavelengths decreases.

Comparisons will be made with IF and NIF, because it is important to identify the benefits of using buffer or not, and also how many buffers it is worth spending. Besides, the TF granularity is set to 50TFs per TC.

SP-MW scenario

Similar to the case of 8-node network, we generate different scaled traffic load by multiplying the traffic matrix (in Table 4.2) with different factors. Table 4.10

[3] The average utilization is intended as the sum of all the busy TFs of all the links, divided by the number of network links

illustrates how the number of wavelengths used decreases when buffering is employed. This confirms that using a NIF techniques helps in a better distribution of traffic offered. In particular cases, we obtain inconsistent results, e.g., with traffic load T, in the NIF (Z=50) scenario a higher number of wavelengths is used, compared to than NIF (Z=20 and 10) scenario. We attribute this error to the local minimum in the implemented greedy algorithm. To put it differently, the computed paths obtained with NIF (Z=50) by greedy algorithm, may be different from those obtained with NIF (Z=20), because, NIF (Z=50) has more chance of routing the traffic along shorter paths. But this is by no means a good aspect if the aim is to minimize the number of wavelengths as the pruning process could invalidate much faster some links which would yield smaller number of wavelengths used.

As regards the network occupation, we see that the average utilization is inversely proportional to the number of wavelengths used. This is perfectly reasonable; since certain requests, initially using a path, have been re-routed along new longer paths after optimization cycle. Moreover, we observe that most of the NIF (Z=50) results have a higher average utilization than the other NIF results even if NIF (Z=50) presents a lower number of wavelengths than the others.

SP-MW	IF		NIF-Z=10		NIF-Z=25		NIF-Z=50	
Traffic Load	Average Utilization [TFs]	# Wavelengths	Average Utilization [TFs]	# Wavelengths	Average Utilization [TFs]	# Wavelengths	Average Utilization [TFs]	# Wavelengths
T	24,85	68	25,12	63	25,81	61	26	64
2 T	37,55	113	39,94	105	40,02	102	40,07	101
4 T	74,09	190	76,49	176	76,74	173	76,35	176
8 T	146,05	353	147,4	334	149,11	332	147,58	328
10 T	182,56	424	183,08	403	183,84	399	185,17	404
20 T	358,15	789	362,21	778	361,91	781	362,28	775

Figure 4.10: *Results for US Network for scaled traffic load in SP-MW scenario*

The absolute number of wavelengths saved, as well as the percentage of saving are shown in Table 4.3 for the three NIF scenarios, with respect to the IF scenario.

Traffic Load \ SP-MW	NIF-Z=10	NIF-Z=25	NIF-Z=50
T	5 (7.35)	7 (10.29)	4 (5.88)
2 T	8 (7.08)	11 (9.73)	12 (10.62)
4 T	14 (7.37)	17 (8.95)	14 (7.37)
8 T	19 (5.38)	21 (5.95)	25 (7.08)
10 T	21 (4.95)	25 (5.90)	20 (4.72)
20 T	11 (1.39)	8 (1.01)	14 (1.77)

Table 4.3: *SP-MW: Saved number of wavelength in the three NIF scenarios respect to IF scenario. Percentage savings are shown within round parenthesis*

The trend we can notice is that NIF techniques generally give an enhanced traffic distribution. The number of wavelengths increases with the traffic load until the point where the traffic request becomes higher and then, the heuristic doesn't manage to re-routing it (as shown with NIF Z=10 and 25 for 20T). On the other hand, comparing the forwarding techniques, the percentage of savings shows that the number of wavelengths drops sharply with NIF (Z=10), while, with NIF (Z=25 and 50) the diminution is not significant.

SP-SW scenario

As we analysed for the SP-MW, we now apply the same analysis on the SP-SW scenario. It is worth noting that we find feasible solution (as shown in table 4.11) only for scaled traffic load with requests which do not exceed the wavelength capacity.

SP-SW	IF		NIF-Z=10		NIF-Z=25		NIF-Z=50	
Traffic Load	Average Utilization [TFs]	# Wavelengths	Average Utilization [TFs]	# Wavelengths	Average Utilization [TFs]	# Wavelengths	Average Utilization [TFs]	# Wavelengths
T	25,01	67	25,41	61	25,81	61	26,11	64
2 T	38,95	112	39,50	100	39,98	104	39,29	100
4 T	74,40	190	75,40	173	75,93	177	75,88	173

Figure 4.11: *Results for US Network for scaled traffic load in SP-SW scenario*

Qualitatively, this scenario is sligthly better than SP-MP, it is surpassed in the number of wavelengthso only in 3 over 12 results(especially IF, NIF Z=10 and NIF Z=50 provide a better response). The saved number of wavelengths and percentage savings (Table 4.4) do not show additional reduction with NIF Z=50 compared to the other allowable buffering, but again here, it is worth reminding the limitations of the heuristic approach.

SP-SW / Traffic Load	NIF-Z=10	NIF-Z=25	NIF-Z=50
T	6 (8.96)	6 (8.96)	3 (4.48)
2 T	12 (10.71)	8 (7.14)	12 (10.71)
4 T	17 (8.95)	13 (6.84)	17 (8.95)

Table 4.4: *SP-SW: Saved number of wavelength in the three NIF scenarios respect to IF scenario. Percentage savings are shown within round parenthesis*

Since the differences are not significantly greater, we conclude that if we consider either the number of wavelengths used or the average utilization, SP-SW scenario has a better performance under static conditions.

MP scenario

The values presented in Table 4.5 are for MP scenario.

MP	IF		NIF-Z=10		NIF-Z=25		NIF-Z=50	
Traffic Load	Average Utilization [TFs]	# Wavelengths	Average Utilization [TFs]	# Wavelengths	Average Utilization [TFs]	# Wavelengths	Average Utilization [TFs]	# Wavelengths
T	24,58	65	25,02	62	25,78	61	26,08	64
2 T	37,70	114	40,07	101	39,76	104	38,95	106
4 T	73,80	185	75,37	174	76,28	175	77,8	175
8 T	146,27	347	147,36	327	147,78	329	148,54	325
10 T	181,91	422	183,38	404	184,29	399	185,52	398
20 T	364,42	763	362,28	757	363,32	752	365,23	736

Table 4.5: *Results for US Network for scaled traffic load in MP scenario*

MP / Traffic Load	NIF-Z=10	NIF-Z=25	NIF-Z=50
T	3 (4.62)	4 (6.15)	1 (1.54)
2 T	13 (11.40)	10 (8.77)	8 (7.02)
4 T	11 (5.95)	10 (5.41)	10 (5.41)
8 T	20 (5.76)	18 (5.19)	22 (6.34)
10 T	18 (4.27)	23 (5.45)	24 (5.69)
20 T	11 (1.44)	6 (0.79)	27 (3.54)

Table 4.6: *MP: Saved number of wavelength in the three NIF scenarios respect to IF scenario. Percentage savings are shown within round parenthesis*

The most significant differences among values of SP-MW and MP scenarios (shown in Table 4.10 and Table 4.5) are the following:

- For all the traffic loads with IF and NIF techniques, on average, MP provides better solutions in terms of number of wavelengths used.
- As far as average utilization is concerned, we find in some cases (e.g., for traffic load 4T with IF) that MP has not only a lower number of wavelengths, but also a lower average utilization respect to SP-MW. Then MP exploits the available resource in efficient way as well.
- Furthermore, the percentage wavelength savings for MP scenario (in Table 4.6) indicate that buffering has less beneficial effect compared with the analogous values for SP-MW (Table 4.3).

4.2.1 24-node US backbone network : Analysis of TDS architecture by varying the granularity

We now analyse the flexibility of TDS architecture employing three different TF granularities:

- 50 TF/TC - with a TF capacity of 200 Mbps for 1 TF
- 20 TF/TC - with a TF capacity of 500 Mbps for 1 TF
- 10 TF/TC - with a TF capacity of 1 Gbps for 1 TF

We use the same traffic matrix demand (shown in Figure 4.2) for all three granularities and we also consider a wavelength capacity of 10Gbps.

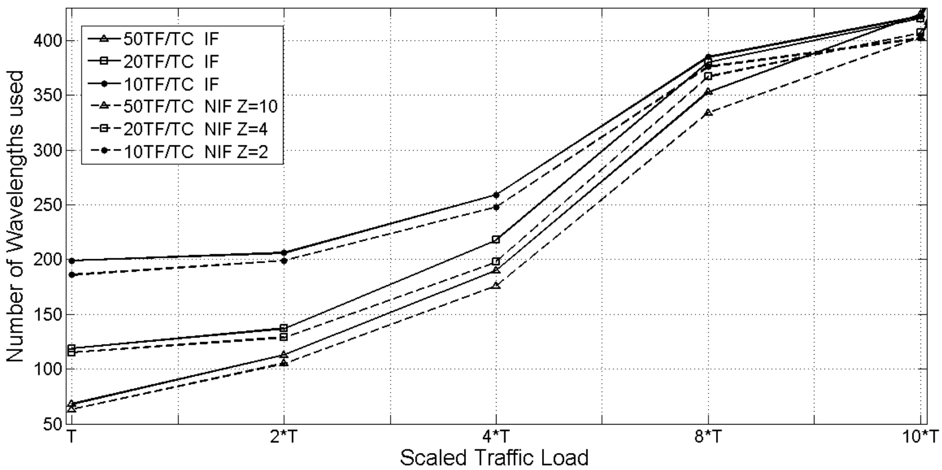

Figure 4.12: *Granularity comparison using the IF and NIF techniques*

We note that (in figure 4.12) for high traffic load (10T and 20T), the three curves are close one another, as a great amount of traffic tends to represent the same proportion of occupation for the three granularities (proportion intended as a number of required TFs divided by the total number of TFs in the specific granularity). When high bandwidth is offered for each request, it is less frequent to perform traffic grooming and then a higher granularity (as 50TF/TC or more) becomes useless. However, we see an interenting feature[4]; although higher traffic load is offered, additional reduction in wavelengths used is provided when buffering are available (e.g., for 20T up to 7.20% using 20TF/TC (Z=20)).

On the other hand, for low traffic load, the granularity of 50TF/TC shows a remarkable improvement in terms of number of wavelengths used (up to about

[4]disregarding heuristic flaws which provokes worse solution in presence of much more available buffers

42% and 66% less than the 20TF/TC and 10TF/TC, respectively), this can be easily explained considering that the minimun bandwidth required by a connection matches better with a TF capacity when there are more TFs per TC.

MP – 50TF/TC	IF	NIF-Z=10
Traffic Load	Average Utilization [TFs]	Average Utilization [TFs]
T	24,85 (0,497)	25,12 (0,502)
2 T	37,55 (0,751)	39,94 (0,799)
4 T	74,09 (1,482)	76,49 (1,530)
8 T	146,05 (2,921)	147,4 (2,948)
10 T	182,56 (3,651)	183,08 (3,662)
20 T	358,15 (7,163)	362,21 (7,244)

Table 4.7: *MP-50TF/TC: Average utilization in TFs (in wavelengths)*

MP – 20TF/TC	IF	NIF-Z=4
Traffic Load	Average Utilization [TFs]	Average Utilization [TFs]
T	18,14 (0,907)	18,33 (0,916)
2 T	22,02 (1,101)	22,53 (1,127)
4 T	34,64 (1,732)	35,14 (1,757)
8 T	65,81 (3,291)	66,6 (3,330)
10 T	73,05 (3,653)	73,65 (3,683)
20 T	144,98 (7,249)	145,07 (7,254)

Table 4.8: *MP-20TF/TC: Average utilization in TFs (in wavelengths)*

Tables 4.7, 4.8 and 4.9 display the average utilization in TFs and in order to compare the three different granularities we include in the tables the ratio between the average utilization in TFs and the total number of TFs per TC on each granularity. We note that:

- Starting from low traffic load, the average utilization decreases significantly with the growth in the number of TFs per TC. For instance, observing an offered traffic T with IF and 50TF/TC, we get 50% of wavelength occupation, whereas with 20TF/TC, 90% of occupation, because the efficiency of the resource utilization is strongly affected by TF granularity.

- The difference in terms of average utilization between granularities tends to decrease for higher values of traffic loads. This is because the same reason expounded for the trend of the number of wavelengths.

MP – 10TF/TC	IF	NIF-Z=2
Traffic Load	Average Utilization [TFs]	Average Utilization [TFs]
T	16,69 (1,669)	17,08 (1,708)
2 T	17,65 (1,765)	17,77 (1,777)
4 T	22,15 (2,215)	22,44 (2,244)
8 T	34,22 (3,422)	34,40 (3,440)
10 T	36,51 (3,651)	36,90 (3,690)
20 T	72,42 (7,242)	72,58 (7,258)

Table 4.9: *MP-10TF/TC: Average utilization in TFs (in wavelengths)*

Chapter 5

TDS Routing Algorithms

This chapter is structured as follows.

- In Section 2.1, the Yen based (Yen-based) and Bellman-Ford Extended (BF-Extended) algorithms are described. These algorithms are used to generate a list of k paths.
- In Section 2.2, the list of k-paths is used as an input of the following algorithms SP-SW-A (Single Path - Single Wavelenght Algorithm), SP-MW-A (Single Path - Single Wavelenght Algorithm) and MP-A (Multiple Path Algorithm).
- In Section 2.3, the complexity of the algorithms is briefly discusses. The scenario in which we work is a dynamic traffic environment. Connection requests arrive at random time instants, remains in the network for a certain period of time and then resources dedicated to them are released.

5.1 k-Paths algorithms

The goal of our Yen-based and BM-Extended algorithms is to find k paths that will be used to route the connection request, always in the absence of wavelengths converter (NWI). To achieve this goal, the idea of a multi-layer graph is used, where instead of considering one layer for each wavelenght, we consider each layer relates to a time frame (TF). Typically, in these cases, the solution is to apply the routing algorithms layer by layer and then choose the best solution (although not with all the algorithms is possible to adopt this solution), but in our simulator a single layer is used in order to reduce the computational complexity. Yen-based and BF-Extended algorithms are developed under this conditions.

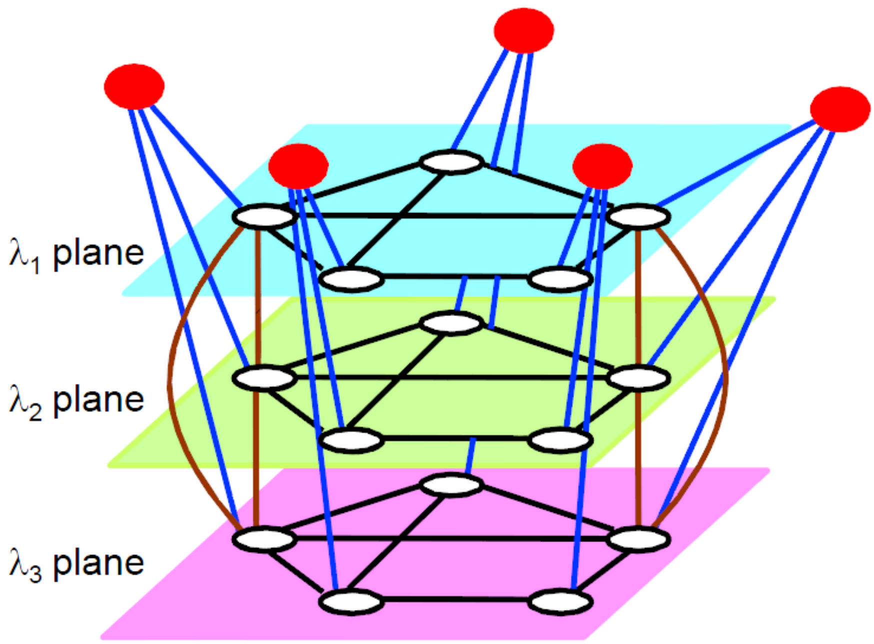

Figure 5.1: *Multilayer graph*

Yen based algorithm for TDS (Yen-based) uses the concepts of the original Yen algorithm. It search k-shortest paths (i.e.the first k paths with minimum length in the network that can be found between a source-destination pair considering the available capacity of the links). In order to find those paths, Yen-based uses the Bellman-Ford algorithm that we modified for adapting to the TDS scenario (BF-based) .

5.1.1 BellmanFord based Algorithm (BF-based):

In this algorithm two cases are considered for TDS: with Immediate Forwarding (i.e., it inludes only the propagation and switching time) and Non-immediate Forwarding (i.e., it includes optical buffering). The Inputs of the problem can be listed as follows:

1. Let $G = (V, E, C, \lambda)$ be a graph where V is the number of vertexes, E is the number of links, C is the capacity, λ is the number of wavelengths at each link;

2. Connection request $c = \langle s, d, Tfreq, t_h \rangle$, where s is the source node, d is the destination node, $Tfreq$ is the required bandwidth (an integer amount of time frames) and t_h is the holding time of connection;

3. T is the number of time frames per time cycle;

4. TF is the set of all time frames belonging to a link (with a cardinality iqual to $T * \lambda$);

5. $nBuffers$ is the set of all possible delay values (each elemtent represents an integer number of time frames);

6. *Delaylink* is the propagation delay at each link (an integer number of time frames);

7. *LinkCost* is the cost of each link (km);

This BF-based algorithm inherits from the Bellman-Ford algorithm(BF) the updating process at nodes and follows the same principles:
Initially all adjacent nodes to source are analyzed for setting up a cost and each node which gets a cost update is added to a list of nodes to be explored; at that point a node is randomly chosen from the list and its adjacent nodes are analyzed. The procedure ends when there are no more nodes to explore in the list.

The differences between these two algorithms(BF-based and BF) are the labels used at each node: in BF algorithm scalar quantities are used in order to recorder the cost of reaching a node and at the same time the previous node is stored. On the other hand in BF-based vectors are used; it is necessary because it has been introduced a graularity to the wavelengtht and is neccesary to find an available route for each of time frames.

We decided to use vectors on a single graph instead of using parallel graphs as mentioned above because it is less heavier computationally than generate all graphs for each time frame at each time.

Variables are inizializated as:

a. $NodeCost$

b. $PrevNode$

c. $DelayNode$

d. $ToVisit$

e. $UpdateNode$

The first three vectors are present at each node and represent: the cost to reach the node (of a certain time frame that belongs to a wavelength and is identified by the position in the vector), the predecessor node in the path (always relative to a time frame identified by the position in the vector) and the delay which time frame is subjected to (considering optical switching of the previous nodes, propagation delays subjected to the previous links and the fiber lines delays in the previous nodes). The fourth variable is global and represents the set of all those nodes that have been updated of cost in at one of the time frames and the fifth variabile rappresent if cost has been updated at node.

For all nodes, all elements of $NodeCost$, $PrevNode$, $DelayNode$ e $UpdateNode$ are initialized $UNREACHEABLE$, $NULL$, 0 and $FALSE$ with the exception

of the source node where the costs are zero and the Updates are true; initially in $ToVisit$ is inserted the source node.
The algorithm works in the following way:

After initialization, we enter in the main loop of the algorithm, that is composed by a while cycle which continue until $ToVisit$ has at least one element; soon after is selected a node i belonging to $ToVisit$, eliminating him from this set, and for all nodes adjacent to this node (indicated by the variable *Adji*), is verified if is possible to improve the cost of these nodes (only when there is capacity available to route the time frame), if so, as well as update the cost vectors and predecessors on the destination node, the destination node is inserted in $ToVisit$. Once out of the while loop, the costs of the nodes have stabilized to the correct values, then is examined the $NodeCost$ in the destination node, looking on all the $t \in TF$, searching for the time frame with the lower cost (if costs are identical, is chosen the time frame with the minor index), once found, the path is reconstructed going among the nodes using the vector $PrevNode$ in the predecessors of the nodes (always looking at the appropriate TF).

Algoritmo 1 - BF-based

Input: Let $G = (V, E, C, \lambda)$ be a graph, a connection request $c = \langle s, d, Tfreq, t_h \rangle$, the number of time frames per time cycle T, is the set of time frames beloging to a link TF, is the set of all possible delay values $nBuffers$, the propagation delay at each link $Delaylink$, cost of each link $LinkCost$.

Output: a shortest path with a Time Frame and wavelength assigned.

Variables:

- $\boldsymbol{NodeCost^n}$:$\{NodeCost^n_{t_1}, NodeCost^n_{t_2}, ...NodeCost^n_{t_{|TF|}} | t_i \in TF\}$ $\forall n \in V$
- $\boldsymbol{PrevNode^n}$:$\{PrevNode^n_{t_1}, PrevNode^n_{t_2}, ...PrevNode^n_{t_{|TF|}} | t_i \in TF\}\}$ $\forall n \in V$
- $\boldsymbol{DelayNode^n}$:$\{DelayNode^n_{t_1}, DelayNode^n_{t_2}, ..DelayNode^n_{t_{|TF|}} | t_i \in TF\}$ $\forall n \in V$
- $\boldsymbol{UpdateNode^n}$:$\{UpdateNode^n_{t_1}, UpdateNode^n_{t_2}, ..UpdateNode^n_{t_{|TF|}} | t_i \in TF\}$ $\forall n \in V$
- $\boldsymbol{TFoccupation}$ is the the current state occupation of the network in each link, for the corrisponding $t \in TF$; is equal to 1 if the time frame is available and is $UNRECHEABLE$ if the time frame is busy.

1. Initialization of variables:

   a. $\forall \langle n \in V |n\neq s$, t$\in TF \rangle$: $NodeCost^n_t = UNREACHEABLE$, $Prevnode^n_t = NULL$, $Delay^n_t = 0$, $UpdateNode^n_t = FALSE$.

      For $n = s$:

      $\forall \langle t \in TF \rangle$: $NodeCost^s_t = 0$, $Prevnode^s_t = NULL$, $UpdateNode^s_t = TRUE$

   b. $ToVisit = s$

2. Algorithm:

   Adj_i= $\{j \in V / i \in V, (i, j) \in E\}$ $\forall i \in V$

while $ToVisit \neq \emptyset$ **do**
  Assign index i at the first element in $ToVisit$
  for $j \in Adj_i$ **do**
    for $t \in TF$ **do**
      if $UpdateNode_t^i$=TRUE **then**
        for $z \in nBuffers$ **do**
          x= $(t + DelayNode_t^i + 1 + z)\%T + (t - t\%T)$
          $NewCost = NodeCost_t^i + LinkCost * TFoccupation(x)$
          $NewDelay = (DelayNode_t^i + Delaylink + 1 + z)$
          if $NewCost < NodeCost_t^j$ **then**
            $NodeCost_t^j = NewCost$
            $PrevNode_t^j = i$
            $Delay_t^j = NewDelay$
            add the node j to the set $ToVisit$
          $ToVisit$= $ToVisit$ $\cup$j
          end if
          if TFoccupation(x)=1 **then**
            break
          end if
        end for
      end if
    end for
  end for
  $ToVisit = ToVisit - \{i\}$
end while
search $t|min_t\ NodeCost_t^d$.
The path is reconstructed using the vector $PrevNode_t^d$. At the end, the corresponding wavelength and Time Frame is assigned to the path: $wlassigment = (t - (t\%T))/T$, $TimeFrame = t$

5.1.2 Yen based algorithm (Yen-based)

This algorithm finds kth shortest paths ($Kpaths$), each of them with a Time Frame assigned. The algorithm receives the following inputs:

1. Let $G = (V, E, C, \lambda)$ be a graph where V is the number of vertexes, E is the number of links, C is the capacity, λ is the number of wavelengths at each link;

2. Connection request $c = \langle s, d, Tfreq, k, t_h \rangle$, where s is the source node, d is the destination node, $Tfreq$ is the required bandwidth(an integer amount of time frames), K is the number of shortest path and t_h is the holding time of connection;

3. T is the number of time frames per time cycle;

4. TF is the set of all time frames belonging to a link (with cardinality iqual to $T * \lambda$);

5. $nBuffers$ is the set of all possible delay values (each elemtent represents an integer number of time frames);

6. $Delaylink$ is the propagation delay at each link (an integer number of time frames);

7. $LinkCost$ is the cost of each link (km);

The algorithm works in the following way:

A shortest path, considered the first of the k-shortest paths(called $CurrentPath$), is found using the BF-based algorithm. From all the nodes of this path, excluding the destination node, new routes(Spurs) are computed as alternative paths to reach the destination, each spur S_i^k is associated with a root(R_i^k) that contains all the nodes between the request's source node and the corresponding spur's source node i. The calculation of each pair (R_i^k, S_i^k) is made using the BF-based with the following constraints:

- it is forbidden to cross the root's nodes of the $CurrentPath$ in order to avoid finding the S_i^k by means of cycles
- if the $Kpaths$ or the $CurrentPath$ has the same root R_i^k, is forbidden cross the next link associated to node i of those paths.

This constraints developed by Yen above described, in which BF-based works and named BF-based* in advance, works in the following way: By means of the appropriate restrictives flags placed on the links and nodes, is possible satisfy the above conditions, avoiding that the new routes, (R_i^k, S_i^k) don't have loops or follows routes that have been explored before. Once all the pairs (R_i^k, S_i^k) have been found, a new k-th $CurrentPath$ is chosen from them with the minimun cost, which is placed as a new k-th shortest path. The procedure is repeated using the new k-th path(called $CurrentPath$) and continues until they have found the desired number of paths.

Always to adapt Yen to the TDS case, the following changes were introduced. We have Extended the dimensionality of the labels on the costs for reaching the node (vector $NodeCost$) and its predecessors(vector $PrevNode$), and the addition of a accumulated delay (vectore $DelayNode$) as previously explained in BF-based. Usually the initialization of the labels for the calculation of a path (R_i^k, S_i^k) that will become a K-shortest path, is to put UNREACHABLE for all the elements of $NodeCost$, ZERO for all the elements of the vector $DelayNode$, and NULL for all the elements of $PrevNode$.

In the other hand when Path (S_i^k) is calculated from node $i = s$, you must put to ZERO all elements of $NodeCost$ at that node to avoid making cycles, but when is calculated a path (S_i^k) from a node $i \neq s$, we identify all the time frames that have the same root of the $CurrentPath$, and for all root's nodes the corrisponding elements of $NodeCost$ will be put to ZERO; besides the $PrevNode$ and $DelayNode$ will be the values found by BF-TDS, for the remaining elements of the vector will be executed the normal initialization.

Algoritmo 2 - Yen-based

Input: $G = (V, E, C, \lambda)$, a conecction request $c = \langle s, d, Tfreq, K, t_h \rangle$, the number of time frames per time cycle T, is the set of time frames beloging to a link TF, is the set of all possible delay values $nBuffers$, the propagation delay at each link $Delaylink$, cost of each link $LinkCost$, the number of shortest paths k.

Output: **K** shortest paths $Kpaths$ with a Time Frame and wavelength assigned.

1. *Initialization of variables:*

   a. $Kpaths = \emptyset$.

   b. $k = 0$.

2. *Algorithm:*

   $CurrentPath$: is the shortest path computed from s to d throught BF-based.

   $RootSpurPaths = \emptyset$;

   while $k <$ **K** **do**

   for each $i \in N$ of the $CurrentPath$-$\{Dst\}$ **do**

   for each $t \in TF$ **do**

   Be R_i^k the root of the k-th $CurrentPath$(i.e. a subpath from node s to node i).

   Be S_i^k is the spur of the k-th $CurrentPath$(i.e the last part computed from node i to destination node throught BF-based*.)

   end for

   if $(R_i^k, S_i^k) \notin RootSpurPaths$ **then**

   $RootSpurPaths = RootSpurPaths \cup \{(R_i^k, S_i^k)\}$

   end if

   end for

   $Kpaths = Kpaths \cup \{CurrentPath\}$.

   $CurrentPath$= The minimun cost path between all the paths inside of $RootSpurPaths$.

k++

end while

5.2 Belman-Ford Extended algorithm (BF-Extended):

This algortihm finds $|TF|$ paths, one path for each element belongs to TF. In this case the inputs of the algorithm are the following:

1. Let $G = (V, E, C, \lambda)$ be a graph where V is the number of vertexes, E is the number of links, C is the capacity, λ is the number of wavelengths at each link;

2. Connection request $c = \langle s, d, Tfreq, t_h \rangle$, where s is the source node, d is the destination node, $Tfreq$ is the required bandwidth (an integer amount of time frames) and t_h is the holding time of connection;

3. T is the number of time frames per time cycle;

4. TF is the set of all time frames belonging to a link (with cardinality iqual to $T * \lambda$);

5. $nBuffers$ is the set of all possible delay values (each elemtent represents an integer number of time frames);

6. *Delaylink* is the propagation delay at each link (a integer number of time frames);

7. *LinkCost* is the cost of each link (km);

Variables are inizializated as:

a. $NodeCost$

b. $PrevNode$

c. $DelayNode$

d. $ToVisit$

e. $UpdateNode$

f. $TFvalid$

BF-Extended follows the same prnciples of BF-based and the inizialization of variables with the follow differences: the last variable $TFvalid$ is introduced in the algorithm as a vettore to validate/invalidate the time frames in each link, and is inizializated as $TRUE$.
The algorithm works in the following way:

After initialization, we enter in an external **for** loop that is executed every $t \in TF$ then is initiated a while which continue until $ToVisit$ has at least one element, the first element is the source node s; a node i selected from $ToVisit$, eliminating him from this set, and for all nodes adjacent to this node (indicated by the variable Adj_i), we verified if is possible to improve the cost of these nodes (only when there is capacity C available to route the time frame), if so, as well as update the cost vectors and predecessors on the destination node, the destination node is inserted in $ToVisit$.

Once out of the while loop, the costs of the nodes in the corrisponding t are stabilized to the correct values, then is examined the $NodeCost$ in the destination node, looking on the particular $t|t \in TF$, if it have found a valid path, the shortest path is reconstructed going among the nodes using the vector $PrevNode$ in the predecessors of the nodes (always looking at the appropriate t). Then, using $TFvalid$, the time frames that have been chosen in each link are invalidated and it's finished until splore all the $t|t \in TF$.

Algoritmo 3 Bellman-Ford Extended algorithm(BF-Extended):):

Input: $G = (V, E, C, \lambda)$, a conecction request $c = \langle s, d, Tfreq, t_h \rangle$, the number of time frames per time cycle T, the set of all time frames belonging to a link TF, the set of all possible delay values $nBuffers$, the propagation delay at each link $Delaylink$, cost of each link $LinkCost$.

Output:: A list of paths ordered from lowest to highest cost($PathsTDS$). *Variables*:

- ***NodeCost***$^{\boldsymbol{n}}$:$\{NodeCost^n_{t_1}, NodeCost^n_{t_2}, ...NodeCost^n_{t_{|TF|}} | t_i \in TF\}$ $\forall n \in V$
- ***PrevNode***$^{\boldsymbol{n}}$:$\{PrevNode^n_{t_1}, PrevNode^n_{t_2}, ...PrevNode^n_{t_{|TF|}} | t_i \in TF\}\}$ $\forall n \in V$
- ***DelayNode***$^{\boldsymbol{n}}$:$\{DelayNode^n_{t_1}, DelayNode^n_{t_2}, ..DelayNode^n_{t_{|TF|}} | t_i \in TF\}$ $\forall n \in V$
- ***UpdateNode***$^{\boldsymbol{n}}$:$\{UpdateNode^n_{t_1}, UpdateNode^n_{t_2}, ..UpdateNode^n_{t_{|TF|}} | t_i \in TF\}$ $\forall n \in V$
- ***TFValid***$^{\boldsymbol{i,j}}$: $\{TFValid^{i,j}_{t_1}, TFValid^{i,j}_{t_2}, ..TFValid^{i,j}_{TF} | t_i \in TF\}$ $\forall (i, j) \in E$
- ***TFoccupation*** is the the current state occupation of the network in each link, for the corrisponding $t \in TF$; is equal to 1 if the time frame is available and is $UNRECHEABLE$ if the time frame is busy.

1. *Initialization of variables:*

   a. $\forall \langle n \in V | n \neq s, t \in TF \rangle$:$NodeCost^n_t = UNREACHEABLE$, $Prevnode^n_t = NULL$, $Delay^n_t = 0$, $UpdateNode^n_t = FALSE$, $\forall \langle (i, j) \in E, t \in TF \rangle$: $TFValid^{i,j}_t = TRUE$
   For $n = s$:
   $\forall \langle t \in TF \rangle$: $NodeCost^s_t = 0$, $Prevnode^s_t = NULL$, $UpdateNode^s_t = TRUE$

   b. $ToVisit = s$

c. $PathsTDS = \emptyset$

2. <u>*Algorithm:*</u>

$Adj_i = \{j \in V | i \in V, j \in V, (i,j) \in E\}\ \forall i \in v$
for $t \in TF$ **do**
  while $ToVisit$ **do**
    Assign index i at the first element in $ToVisit$
    for $j \in Adj_i$ **do**
      if $UpdateNode_t^i$ = TRUE **then**
        for $z \in nBuffers$ **do**
          $x = (t + Delay_t^i + 1 + z)\%T + (t - t\%T)$
          the followings variables $NewCost$ and $NewDelay$ are updated in the node j if is verified:
          a. Free occupation for the time frame x in $TFoccupation$.
          b. $TFValid_x^{i,j}$ is $TRUE$.

- $$NewCost = NodeCost_t^i + LinkCost$$
- $$NewDelay = (Delay_t^i + Delaylink + 1 + z)$$

          if $NewCost < NodeCost_t^j$ **then**
            $NodeCost_t^j = NewCost$
            $PrevNode_t^j = i$
            $Delay_t^j = NewDelay$
            $Update_t^j = true$
            add to the set in$ToVisit$ the node j
          $ToVisit$=$ToVisit$∪j
          end if
          if $TFoccupation(x) = 1$ & $TFValid_x^{i,j} = true$ **then**
            break
          end if
        end for
      end if

end for

$ToVisit = ToVisit - \{\mathrm{i}\}$

end while

if $NodeCost_t^d! = UNREACHEABLE$ **then**

we obtain the path starting from $PrevNode_t^d$ and going back throught the other nodes. Then is assigned the corresponding wavelength: $wlassigment = (t - (t \mod T))/T$

After obtaining the *path*, all the Time Frames chosen by the path are invalidated by placing $TFValid_t^{i,j}$=$FALSE$.

$PathsTDS = PathsTDS \cup path$

end if

end for

5.3 Routing and resource allocation algorithms in the network:

The MP-A routes the required bandwidth using the list of paths provided by the BF-Extended algorithm explained in the previous section. MP-A chooses from the list $PathsTDS$ the shortest paths to satisfy the Tfreq. In this configuration the required bandwith can use different routes in order to reach the destination.

The algorithms SP-MW-A and SP-SW-A described below have as input the list $PathsTDS$ given by BF-Extended algorithm in order to found a feseable path to satisfy the incoming connection request.

5.3.1 Single Path - Multiple Wavelenght Algorithm (SP-MW-A):

This algorithm enables us to do an unsplitable routing (i.e., routing in the same route) forwarding all the required bandwidth over all the wavelenghts that have a fiber, using also IF and NIF tecniques. The Inputs of the problem can be listed as follows:

(a) Let $G = (V, E, C, \lambda)$ be a graph where V is the number of vertexes, E is the number of links, C is the capacity, λ is the number of wavelengths at each link;

(b) Connection request $c = \langle s, d, Tfreq, t_h \rangle$, where s is the source node, d is the destination node, $Tfreq$ is the required bandwidth(an integer amount of time frames) and t_h is the holding time of connection;

(c) T is the number of time frames per time cycle;

(d) TF is the set of all time frames belonging to a link (with cardinality iqual to $T * \lambda$);

(e) $nBuffers$ is the set of all possible delay values (each elemtent represents an integer number of time frames);

(f) $PathsTDS$ is the list of paths computed by (SR) oredered to avoid find the same paths;

(g) $t* \in TRreq$ is one of the time frame asked in $TRreq$;

Variables are inizializated as:

$TFvalid$ is a vector to validate/invalidate the Time Frames in each link, is initialized as $TRUE$;
The algorithm works in the following way:

We enter in a for which scoured every $Path \in PathsTDS$, in order to route the connection request, if it is possible forwarding the required time frames $t* \in TRreq$ in that $Path$, the positions taken on each link by a time frame are invalidated through the $TFvalid$; the algorithm continues until the finds a valid path that belongs to $PathsTDS$ in oder to forward all $t* \in TRreq$.

Single Path - Multiple wavelenght Algorithm (SP-MW-A):

Input: Let a graph $G = (V, E, C, \lambda)$, a connection request $c = \langle s, d, TFreq, t_h \rangle$, the number of wavelenghts λ, the number of time frames for each wavelenght T, the set of all time frames belonging to a link TF, the set of all possible delay values $nBuffers$, the delay in each link $Delay$, a list of paths $PathsTDS$ computed by (BF-Extended). *Output:* A $Path$ with an available capacity in order to route the TFreq.
variabili:

- $Delay_t^i$ is the delay of $t \in TF$ calculated from the source node s to node i, considering optical switching, the fiber lines delays of the previous nodes and the propagation delays.
- $\boldsymbol{TFValid^{i,j}}$: $\{TFValid_{t_1}^{i,j}, TFValid_{t_2}^{i,j}, ..TFValid_{TF}^{i,j} | t_i \in TF\}$ $\forall (i,j) \in E$
- ***TFoccupation*** is the the current state occupation of the network in each link, for the corrisponding $t \in TF$; is equal to 1 if the time frame is available and is $UNRECHEABLE$ if the time frame is busy.

(a) *Initialization of variables:*
$\forall \langle (i,j) \in E, t \in TF \rangle$: $TFValid_t^{i,j} = TRUE$

(b) *algorithm:*
for $Path \in PathsTDS$ **do**
for $t \in TF$ **do**
for $(i,j) \in Path$ **do**
for $z \in nBuffers$ **do**
$x = (t + Delay_t^i + 1 + z)\%T + (t - t\%T)$
The time frame t is available in the $Path$ if it is verified the following conditions:

a. there is free occupation for the time frame x in $TFoccupation$.

b. $TFValid_x^{i,j}$ is $TRUE$.

if the previous conditions are verified then break

end for

end for

end for

if the total $|TRreq|$ found an available rout, the request can be sent in the $Path$, and we will stop searching in the other paths.

end for

5.3.2 Single Path - Single wavelenght Algorithm (SP-SW-A):

This algorithm enables us to do an unsplitable routing of the requests, forcing the use of an unsplit Wavelength modality (i.e. a modality where the required bandwith is routed for the same wavelength). When there is the possibility of buffering we have the presence of jitter, from the delay introduced in each node.

- Let a graph $G = (V, E, C, \lambda)$, where V is the number of vertex, E is the number of links, C is the capacity of each link, λ is the number of wavelenghts of each link e TF is the number or time frames of each link and T is the number of time frames in each wavelenght, is the delay of each link($Delaylink$);
- Connection request $c = \langle s, d, Tfreq, t_h \rangle$, where s is the source node, where d is the destination node, $Tfreq$ is the bandwidth request and t_h is the duration time of a connection;
- $t* \in TRreq$ is one of the time frame asked in $TRreq$.
- $TFvalid$ represent the avalibity of the time frames in each link.
- $nBuffers$ is the set of all possible delay values (each elemtent represents an integer number of time frames);
- $PathsTDS$ is the list of paths computed by (BF-Extended) ordered to avoid find the same paths;

Variables are inizializated as:
a. $TFvalid$ is inizializated as true.

The algorithm works in the following way:

this algorithm is similar to SP-MW-A, with the difference that after entering in a for which scoured every $Path \in PathsTDS$ we search in each wavelength available capacity, in order to route the requered bandwith in the wavelenght chosen, if is possible forward the request in a specific λ and that$Path$, the positions taken on each link by a time frame are invalidated through the $TFvalid$; the algorithm continues until the requestis finds a valid path in $PathsTDS$.

Single Path - Single wavelenght Algorithm (SP-SW-A)

Input: $G = (V, E, C, \lambda)$, dove V is the number of vertex, E is the numberof links, C is capacity of each link and λ is the number of wavelenghts for each link, a equest of conecction $c = \langle s, d, Tfreq, k, t_h \rangle$, the number of time frames for each wavelenght T, the set of all time frames belonging to a link TF, the set of all possible delay values $nBuffers$, the delay in each link $Delay$, a list of paths $PathsTDS$ computed by (BF-Extended).
Output: a $Path$ with an available capacity in order to route the TFreq.

variables:

- $Delay_t^i$ is the delay of $t \in TF$ calculated from the source node s to node i, considering optical switching, the fiber lines delays of the previous nodes and the propagation delays.
- $\boldsymbol{TFValid^{i,j}}$: $\{TFValid_{t_1}^{i,j}, TFValid_{t_2}^{i,j}, ..TFValid_{TF}^{i,j} | t_i \in TF\}$ $\forall (i,j) \in E$
- ***TFoccupation*** is the the current state occupation of the network in each link, for the corrisponding $t \in TF$; is equal to 1 if the time frame is available and is $UNRECHEABLE$ if the time frame is busy.

(a) *Initialization of variables:* $\forall \langle (i,j) \in E, t \in TF \rangle$: $TFValid_t^{i,j} = TRUE$

(b) *Algorithm:*

for $Path \in PathsTDS$ **do**
  for $Lambda \in \lambda$ **do**
    for $t \in \lambda$ **do**
      for $(i,j) \in Path$ **do**
        for $z \in nBuffers$ **do**
          i= is the source node of the Links
          $x = (t + Delay_t^i + 1 + z)\%T + (t - t\%T)$

The time frame t is available in the $Path$ if it is verified the following conditions:

a. there is free occupation for the time frame x in $TFoccupation$.

b. $TFValid_x^{i,j}$ is $TRUE$.

if the previous conditions are verified then break

end for

end for

end for

end for

if($t* \in TRreq$) is accomplished, the request can be sent by that route $Path$, and we will stop searching in the other paths.

end for

Chapter 6

Dynamic Traffic: Results and Discussion

In order to analyse the performance of TDS in differents scenarios, a C++ simulation has been developed and used to simulate the behaviour of an optical network with dynamic connections. In our simulations, the connection-arrival process is Poisson based and connections holding time follows a negative exponential distribution with average normalized to unity. Connections are uniformly distributed among all node pairs, all the nodes have full grooming capability but no wavelength-conversion capability. Once the S/D node pair is chosen, four types of connection requests can be chosen, OC-3, OC-12, OC-48 and OC-192 with probabilities 1/19 for the latter and 6/19 for the three remaining OC-3,OC-12 and OC-48. All simulations are performed guaranteeing a confidence interval of 95 % with an error of 5 % for all the considered parameters.

6.1 Metrics for performance evaluation

It is necessary to define a metric for making the comparisons before showing the results:

- Blocking Porbability: regarding the performance of the algorithms, we chose the blocking probability as a benchmark. That is defined as the amount of bandwidth blocked (i.e., number of time frames of the blocked requests) over the total amount of bandwidth offered (i.e., number of time frames of the requests). In the graphs we will refer to this parameter with the term p_b;
- Average Utilization: is the ratio between the total amount of the required bandwidth of all connections that traverse the bottleneck link and the link capacity.

The different algorithms described in chapter 5 are compared. Typically the results present in the abscissa the arrival rate with values from a minimum of 350 connections/s to a maximum of 550 connections/s. The algorithms will be indicated by their acronyms, specifically with "SP-MW" we refer to the Single Path - Multiple Wavelength algorithm, with "SP-SW" refer to Single Path - Single Wavelegnth algorithm, while " MP" is used for Multiple Path algorithm.

6.2 USA24 network topology

The network used in the simulations (shown in figure 6.1), is an american network topology called *USA24* which is composed by 24 nodes and 43 bidirectional links. The number of wavelengths used in eacj link of the transport network is equal to 8. Each wavelength has a capacity of the OC-192(10 Gb/s).

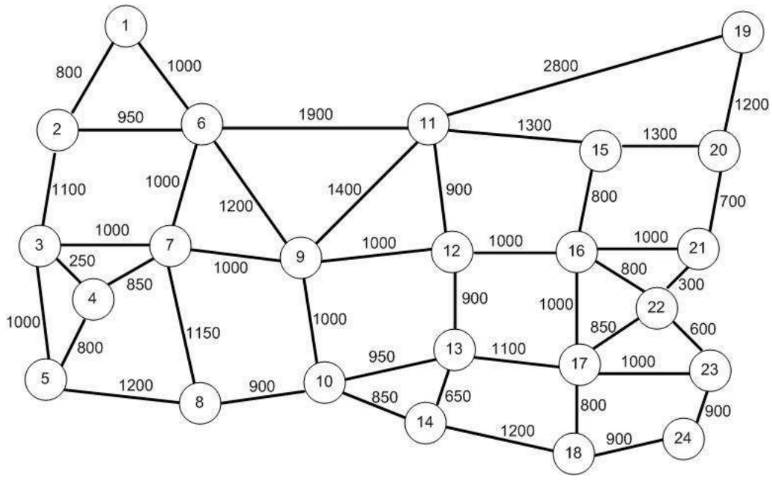

Figure 6.1: *USA24 network topology with link length marked (km)*

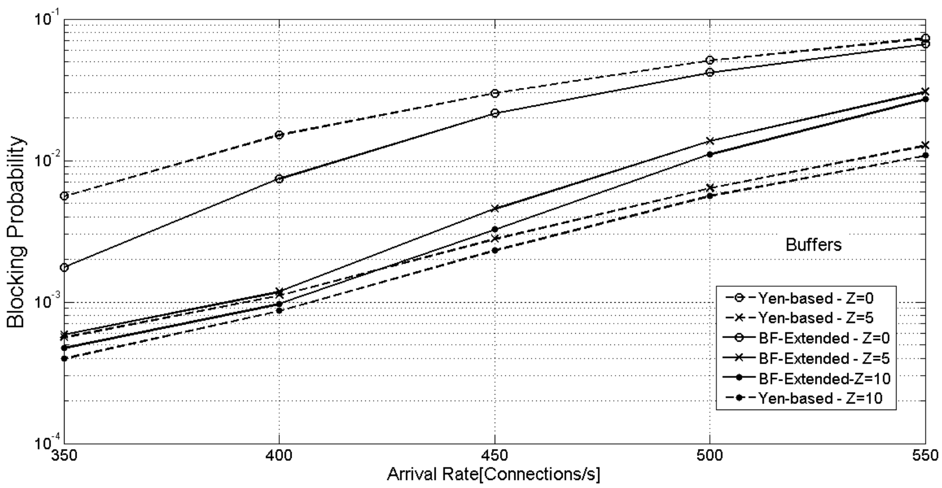

Figure 6.2: *Performance of Yen-based and BF-Extended*

6.3 Analysis of the two k-paths proposed algorithms

Figure 6.2 depicts the performance curves with IF and NIF(Z=5,10). We can see that BF-Extended implementing IF has a better performance respect to Yen-based; observing an arrival rate (400 con/s) where the biggest difference is reported, we find a percentage gain in P_b of 50% between these two algorithms. On the other hand, Yen-based outperforms BF-Extended passing from IF to NIF(z=5), where the percentage gain in blocking probability, for arrival rate of 550 connections/s is in the order of 82%, while BF-Extended has a percentage gain in blocking probability in order of 53%.

This behaviour is due to the same nature of the algorithms: in Yen-based, when the spur-paths are calculated, the following restrictions are imposed: 1) it is forbidden to cross the root's nodes of the current Path in order to avoid finding the spur by means of cycles; 2) if the k-th paths or the current path has the same root, it is forbidden to cross the next link associated to the source node of the spur of those paths. Thus using an IF technique it is more likely not to find a correspondence between the inlet and the outlet (which implies not to find a possible path to being one of the k-th shortest path) at the source node of the spur. On the contrary, using NIF this problem occurs in fewer occasions. In Bf-Extended there is no special restriction on the links, the paths are calculated from source to destination just with the condition of available capacity.

Despite of yen show us a general better performance, in the followings graphs all scenarios uses BF-Extended, since as explained in the previous chapter 6 this is less heavier computationally.

6.4 SP-MW vs MP: Results Comparison

In figure 6.3, we compare the performance of the MP and SP-MW algorithms using the IF and NIF(Z=20) techniques with a granularity of 100 TFs/TC.

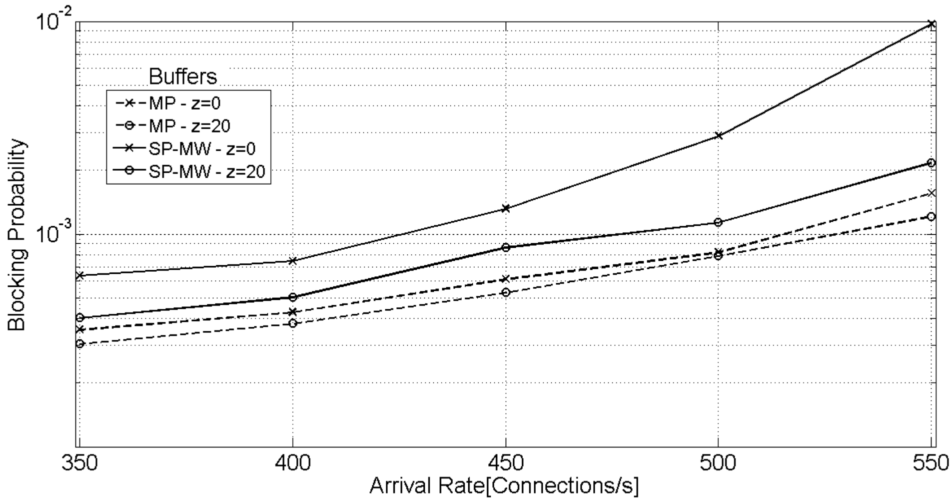

Figure 6.3: *SP-MW vs MP using IF and NIF (Z=20)*

We notice that, compared with MP scenario, the SP-MW scenario presents a higher P_b. Using the IF technique for arrival rate of 400 con/s, we can observe that the percentage gain in P_b of MP scenario is in the order of 42% with respect to SP-MW scenario and the corresponding percentage gain utilization of about 19%; this percentual tends to increase for higher values of the arrival rate in the considered range. For example for arrival rate equal to 550 con/S, we obtain a percentage gain in P_b of approximately 84% with a gain link utilization(28%).

In the SP-MW scenario, the P_b significantly improves when we pass from IF (Z=0) to NIF (Z=20), especially when the arrival rate increases. For arrival rate equal to 550 con/s we get a percentage gain in P_b equal to 77%. By contrast, in a MP scenario, the improvements are less significant, with a percentage gain in P_b equal to 20%. This behaviour is due to the same nature of the algorithm: MP tries to forward the required bandwith along different paths if necesary, so that the benefits introduced with buffering are reduced.

Finally, there is an interesting aspect regarding these scenario. we find that the difference between MP scenario with IF technique and SP-MW scenario with NIF (Z=20) is more than 7%, nevertheless MP has an aditional delay because of using mutiple paths.

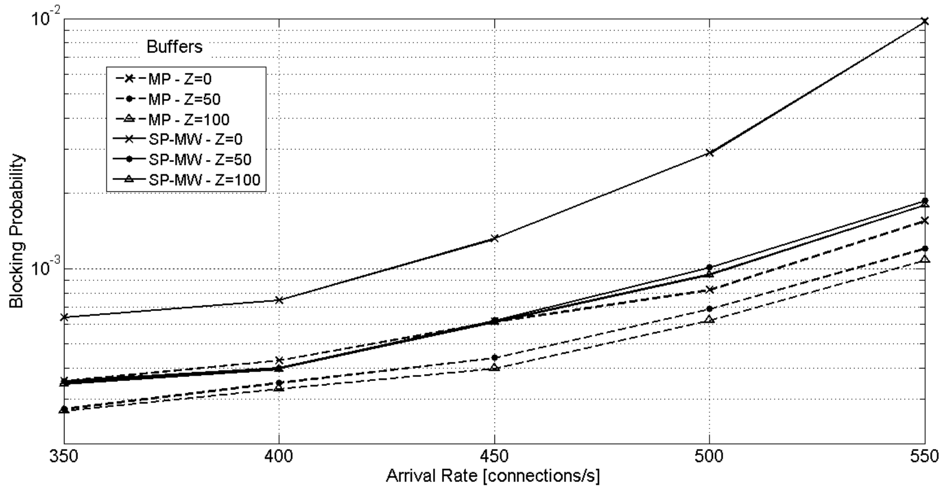

Figure 6.4: *SP-MW vs MP using IF and NIF (Z=50 and 100)*

Now, we would like to show the effect of employing NIF (Z=50) and (Z=100). The last one is called Full Forwarding (FF) as it permits to map any TF in ingress with any free TF in egress at each node. In figure 6.4, we plot the P_b results comparing the same scenarios used in the previous comparison. It can be seen that for the SP-MW scenario passing, from IF to FF (Z=50), the maximum gain in P_b and utilization is 80% and 6,6%, respectively, with arrival rate 550 conn/s; in addition, passing from IF(Z=0) to FF(Z=100) the percentage gain is 81% with an utilization gain of(7,2%); therefore the employment of FF is not much more advantageous as well as from the technological point of view.

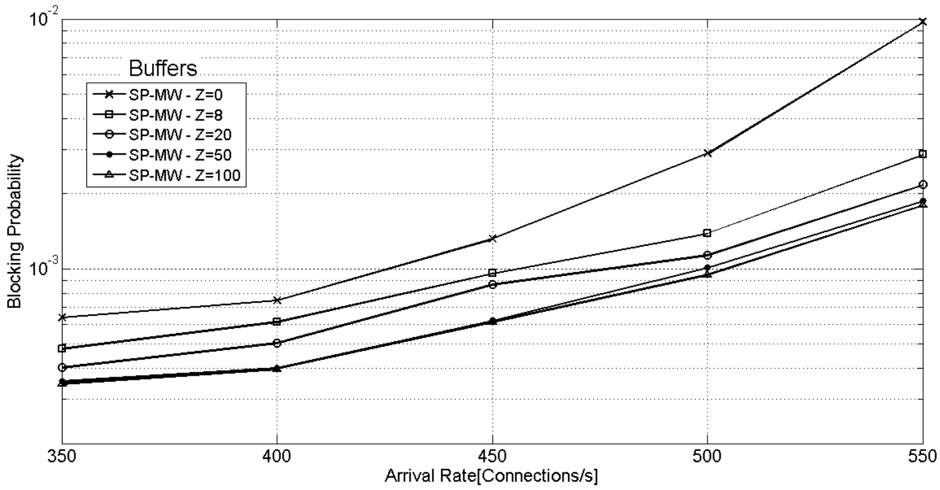

Figure 6.5: *Performance SP-MW*

We introduce an additional NIF with Z=8, in order to see a more realistic NIF technique. Figure 6.5 displays NIF (Z=8) along with the already mentioned techniques (IF and NIF(Z=20,50,100))

In the SP-MW scenario with an arrival rate equal to 550 conn/s, it can be seen that passing from IF (Z=0) to NIF (Z=8) we obtain a percentage P_b gain equal to 71%, whereas passing from IF(Z=0) to NIF(Z=20) we get a percentage P_b gain of 77%. Although, the difference between the two NIF techniques taken into consideration is of 6%, NIF (Z=8) is clearly less expensive in terms of buffers and more practical from technology point of view.

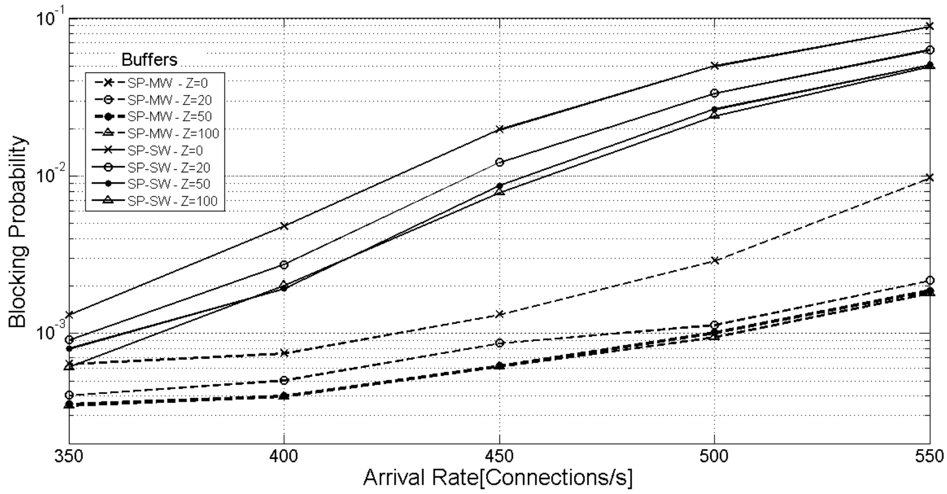

Figure 6.6: *Performance SP-MW vs SP-SW*

6.5 SP-MW vs SP-SW: Results Comparison

In Figure 6.6, we plot the blocking probabilities curves, comparing SP-MW performance with the SP-SW scenario. We observe that:

- in SP-SW scenario the blocking probability is remarkably higher than in the SP-MW scenario, so that, SP-MW outperforms the other algorithm especially for moderate arrivals rate in the range consider;
- with regard to IF technique with the lowest arrival rate (400), the percentage P_b gain of SP-MW scenario is in the order of 84% with a percentage utilization gain nearly to 12,34% respect to SP-SW; in addition, for the arrival rate (500) which reports the biggest difference between the two scenarios in comparison, SP-MW scenario gains in P_b up to 94,2% with a percentage utilization of 12,50% respect to SP-SW;
- for the highest arrival rate (550), we saw that in SP-MW, the P_b significantly improves passing from IF to NIF (Z=20) with a percentage

gain of 77%. By contrast, in a SP-SW scenario the improvements are not singnificant with a percentage gain equal to 28%.

- concluding, in the SP-SW, the employment of NIF Z=50 and Z=100, shows similar trends. Both of them gain up to 47 in P_b respect to IF.

Conclusions

We have presented and discussed a novel switching technique, called Time-Driven Switching (TDS), in order to find adequate tools to make today's telecommunication infrastructure more environment friendly, i.e. less environmental dangerous in the years to come. TDS has been proposed as a possible way to "green" today's Internet, referring mostly to the increasing of the network efficiency in utilization of limited resources and the overall power consumption. Thanks to TDS's ability to accomplish switching and transmission operations in optical domain, it is now possible to envision a sustainable development of the telecom network systems characterized by constant growth, providing at the same time adequate service to end users. After introducing the state-of-art of the present situation, we explicate the network model based on the TDS technique with its relevant features. In particular, we evaluate the TDS performance applying different forwarding techniques on different scenarios under static and dynamic conditions.

As for the static traffic demand scenario, TDS have been studied, implementig ILP formulations which have been verified using commercial optimization software CPLEX for smaller-sized networks. One heuristic algorithm with different scenarios have been proposed to solve the RTWA problem in TDS networks. The performance of the algorithm have been compared with the results obtained by solving the ILP, with cases studies on a 8-node network and the 24-node US backbone network. The performance of the heuristics relative to the ILP was found to improve as the network size increases. We have seen the benefits of using the different scenarios in the

TDS networks. In the first comparison, we conclude that the values (either wavelength or average utilization) are relatively similar to each SP-SW and SP-MW scenarios, thus, if we want to avoid the jitter due to the use of different wavelengths to carry the traffic, SP-SW then would be the best option; clearly, this last statement is valid only if the offered traffic by an request does not exceed the capacity of the wavelength. The second comparison shows that MP scenario has a noticeable better performance than SP-MW, not only for reducing the number of wavelength but in some values also for exploiting the network resource. In addition, we have analysing the effect of varying the number of TF per TC. As expected, the granularity affects the network utilization (for example, in 50TF/TC, 66% less wavelengths used than 10TF/TC), even more than the deployment of FDLs. However, if high traffic load is taken into consideration, then only FDLs help to reduce the number of wavelengths used (around 7% for 20T).

As far as dynamic traffic is concerned, we introduced differents scenarios as SP-MW, SP-SW and MP that take account of the time and are able to identify the delays along the path; this algorithms uses the propagation delay at each link, as well as the swtiches and buffers of each node. We use this scenarios to observe blocking probability and average utilization in WDM networks. We see a significant performance gain by employing MP than the other aproaches here exposed. As we observed this approach is not allowed in a multimedia streaming services because has a higher delays caused of using multiple paths. In the other hand we notice that using buffer tecnique in a SP-MW scenario, we can be closer to the performance of the MP scenario.

List of Figures

1 *Network Power Distribution* x

1.1 Block Diagram of a Network Router 3

1.2 *IP-over-WDM Network Architecture* 8

1.3 N^2 *Problem* 9

2.1 *TDS Synchronization with UTC* 14

2.2 *The Common Time Reference (CTR) with* $d = 12.5\mu s$ 16

2.3 *The OCS* N^2 *Problem is solved in TDS networks* 17

2.4 *TDS Switch Architecture and Operating Stages* 18

2.5 *Parallel Networks on the same Fiber Infrastructure with TDS* 20

3.1 *Flow-chart describing the main steps of the heuristic algorithm for the design of TDS networks (A: Number of wavelengths used)* 42

4.1 *8-node network topology (links lengths in km)* 45

4.2 *Traffic Matrix 1 for 8-node network (each entry is in units of Mbps)* 45

4.3 *8-node Network: Granularity comparison (with T=traffic load of traffic matrix in Figure 4.2)* 47

4.4 *8-node Network: Flow-Route comparison on the total number of wavelength used* . . . 48

4.5 *8-node Network: Flow-Route comparison on the total number of wavelength used as percent relative error* ϵ_W . . . 49

4.6 *Traffic Matrix 2 for 8-node network (each entry is in units of Mbps)* . . . 50

4.7 *8-node Network: Route Formultion - Heuristic comparison on the total number of wavelength used* . . . 51

4.8 *8-node Network: Route Formultion - Heuristic comparison on the total number of wavelength used as percent relative error* ϵ_W 51

4.9 *24-node network topology (link lengths in km)* . . . 52

4.10 *Results for US Network for scaled traffic load in SP-MW scenario* . . . 54

4.11 *Results for US Network for scaled traffic load in SP-SW scenario* 55

4.12 *Granularity comparison using the IF and NIF techniques* . . . 58

5.1 *Multilayer graph* . . . 62

6.1 *USA24 network topology with link length marked (km)* . . . 87

6.2 *Performance of Yen-based and BF-Extended* . . . 87

6.3 *SP-MW vs MP using IF and NIF (Z=20)* . . . 89

6.4 *SP-MW vs MP using IF and NIF (Z=50 and 100)* . . . 90

6.5 *Performance SP-MW* . . . 91

6.6 *Performance SP-MW vs SP-SW* . . . 92

List of Tables

2.1 *Forwarding Techniques Comparison (H = #hops; k = #TFs in a TC; d = duration of a TF)* 23

4.1 *8-node Network Optimization: computational time (h and d stand for hours and days, respectively)* 47

4.2 *24-node network Traffic Matrix* 53

4.3 *SP-MW: Saved number of wavelength in the three NIF scenarios respect to IF scenario. Percentage savings are shown within round parenthesis* 55

4.4 *SP-SW: Saved number of wavelength in the three NIF scenarios respect to IF scenario. Percentage savings are shown within round parenthesis* 56

4.5 *Results for US Network for scaled traffic load in MP scenario* . 56

4.6 *MP: Saved number of wavelength in the three NIF scenarios respect to IF scenario. Percentage savings are shown within round parenthesis* 57

4.7 *MP-50TF/TC: Average utilization in TFs (in wavelengths)* . . 59

4.8 *MP-20TF/TC: Average utilization in TFs (in wavelengths)* . . 59

4.9 *MP-10TF/TC: Average utilization in TFs (in wavelengths)* . . 60

Bibliography

[1] C. Lange, A. Gladisch, *"On the energy consumption of FTTH access networks,"* Optical Fiber Communication, March 2009.

[2] R. Tucker, *"The Role of Optics and Electronics in High-Capacity Routers,"* Lightwave Technology, Journal of, vol. 24, pp. 4655-4673, December 2006.

[3] D. Simeonidou, M. Pickavet, *"Seventh Framework Programme - D21.2b:Intermediate TP Report on First Research Results and Planned Activities: Building the Future Optical Network in Europe (BONE),"* BONE, December 2009.

[4] T. T. Ye, L. Benini, G. De Micheli, *"Analysis of power consumption on switch fabrics in network routers,"* Design Automation Conference, 2002. Proceedings. 39th, pp. 524-529, 2002.

[5] P. Nga and U. Killat, *"Resource efficient logical topology design for IP-over-WDM backbone networks,"* Computer Communications Vol. 31, No. 16, pp 3771-3777, 25 Oct. 2008.

[6] Y. Chen, C. Qiao, X. Yu, *"Optical Burst Switching (OBS): A New Area in Optical Networking Research,"* Network, IEEE, vol. 18, no. 3, pp. 16-23, May 2004.

[7] G. M. Lee, B. Wydrowski, M. Zukerman, J. K. Choi, C. H. Foh, *"Performance Evaluation of an Optical Hybrid Switching System,"* GLOBECOM, December 2003.

[8] I.P. Kaminow, T. Li, A.E. Willner, *"Optical Fiber Telecommunications V B: Systems and Networks,"* Academic Press, 2008.

[9] M. Baldi, Y.Ofek, *"Fractional Lambda Switching - Principles of Operation and Performance Issues,"* SIMULATION: Transactions of The Society for Modeling and Simulation International, vol. 80, pp. 527-544, October 2004.

[10] V. Nguyen, R. Lo Cigno, M. Telek, *"Time Blocking Analysis in Timedriven Switching Networks,"* INFOCOM, 2008.

[11] M. Baldi, Y. Ofek, *"Time for a "Greener" Internet,"* GREENCOMM, June 2009.

[12] A. Pattavina, M. Bonomi, Y. Ofek, *"Performance Evaluation of Time Driven Switching for Flexible Bandwidth Provisioning in WDM Networks,"* 2004.

[13] T. Truong, M. Baldi, Y. Ofek, *"An Efficient Scheduling Algorithm for Time-Driven Switching Networks,"* DIT-07-2002.

[14] F. Musumeci, F. Vismara, V. Grkovic, M. Tornatore, A. Pattavina, *"On the Energy Efficiency of Optical Transport with Time Driven Switching,"* Communications (ICC), 2011 IEEE International Conference on, pp. 1-5, June 2011.

[15] N. Wauters and P. M. Deemester, *"Design of the optical path layer in multiwavelength cross-connected networks,"* IEEE Journal on Selected Areas in Communications, vol. 14, pp. 881891, june 1996.

[16] B. Mukherjee, *"Optical communication networks,"* McGraw-Hill, 1997.

[17] R.Ramaswami, K.N.Sivarajan, *" Optical Networks: A Pratical Perspective,"* Morgan Kaufmann, 1998.

[18] R.Ramaswami, K.N.Sivarajan, *"Design of logical topologies for wavelength-routed optical networks,"* IEEE Journal on Selected Areas in Communications, vol. 14, pp. 840851, June 1996.

[4] B. Mukherjee, Optical communication networks, McGraw-Hill, 1997.
[5] R.Ramaswami and K.N.Sivarajan, Optical Networks: A Pratical Perspective, Morgan Kaufmann, 1998.

[19] G. Maier, *"Optical Switching -1-,"* Materiale didattico. [Online]. Available: http://home.dei.polimi.it/pattavina.

[20] M. Tornatore, G. Maier, A. Pattavina, *"WDM Network Design by ILP Models Based on Flow Aggregation,"* Networking, IEEE/ACM Transactions on, vol. 15, pp. 709-720, June 2007.

[21] R. Ramamurthy, Z. Bogdanowicz, S. Samieian, D. Saha, and B. Rajagopalan, *"Capacity performance of dynamic provisioning in optical networks,"* Journal of Lightwave Technology, vol. 19, no. 1, pp. 4048, Jan. 2001.

[22] I. Chlamtac, A. Farago, and T. Zhang, *"Lightpath (Wavelength) Routing in Large WDM Networks,"* IEEE Journal on Selected Areas in Communications, vol. 14, no. 5, pp. 909913, June 1996.

[23] C. Chen and S. Banerjee, *"A new model for optimal routing and wavelength assignment in wavelength division multiplexed optical networks,"* in Proceedings, IEEE INFOCOM 96, 1996, vol. 2a.4.1-8, pp. 164171.