A Data Vortex Switch
for Optical Packet Switching

By Manuel Iván Macías Rodríguez

Graduation thesis
Carried out from September 2002 to April 2003

Supervisors:
Dr. ir. I. Tafur Monroy
Ir. A. Nowbakht Irani

Graduation professor:
Prof. ir. A. M. J. Koonen

April 2003
A Data Vortex Switch for Optical Packet Switching

Manuel Iván Macías Rodríguez

Graduation thesis, April 2003
Abstract

This thesis studies the application of a novel Data Vortex packet switch architecture in optical packet switched networks. Data Vortex principle of operation and characteristics are presented and its principal features and requirements are identified. The Data Vortex performance is evaluated by means of extensive computer simulations. Performance parameters such as the throughput, latency and latency distribution are analyzed in order to assess the Data Vortex efficiency and possible application in optical packet switching. Our results show that the generic Data Vortex, though feasible for packet switching, introduces a degree of latency. As a solution to this issue, we propose a new Data Vortex architecture that is much more efficient than the generic one. The new operation introduced results in a remarkable improvement in the latency performance and it is attractive for a compact optical packet switch realization both for telecom and supercomputing applications. Furthermore, the use of electroabsorption modulator technology is proposed for Data Vortex nodes implementation, taking into consideration the requirements imposed in terms of fast switching time, simplicity and prospect of integrability in a photonic circuit. Finally, we present the use of the Data Vortex switch for the optical packet switching scheme known as Photonic Slot Routing. The use of the Data Vortex as a Photonic Slot Routing switching node, results in a promising approach for bandwidth and cost efficient networks.
Acknowledgements

First and foremost, I would like to thank my parents for providing me an education that culminates at this moment with this graduation degree. I am grateful for their love and support along the years, in which they have always sacrificed themselves for me. It is a pleasure to dedicate my graduation thesis to them. I have to thank also my sister for her support and for enduring me along all these years.

I would like also to thank all my friends of the university: Luis Alberto, Clarisa, Juanan, Jorge, Gus, Gonzalo, Juan, Henar, Miguel, Judith for helping me during my degree and make me enjoy all my five years of study. Particularly, I would like to thank twice my untiring workmate, Juanan, for the fantastic team-work that we achieved during the last five years.

I have to thank the Eindhoven University of Technology and in particular my supervisor, dr. ir. Idelfonso Tafur, and my graduation professor, prof. ir. A.M.J Koonen, for giving me the opportunity to perform my graduation thesis in this prestigious university. I am very grateful to my supervisor, Idelfonso Tafur, who chose me to perform this graduation thesis. I really appreciate all his help and the enthusiasm shown during all my work. I am also grateful to Ali Nowbakht, my other supervisor, for his help and encouragement during my work here. I would like also to thank some members of the Electro-optical group: Ronelle Geldenhuys, Nicola Calabretta and Sander Jansen for all their helpful suggestions. I cannot forget to thank also Jesus for all his suggestions and programming tips. Also, I would like to thank dr. eng. I. de Miguel of the university of Valladolid for his help.

Finally I would like to thank all my friends and mates here in Eindhoven: Almudena, Diego, Bas, Juan Carlos, Jesus and Dani. Thank you for all the great moments we have spent here. Special appreciations go to Marta for her support, and for putting up with me all these months.
## Contents

1 Introduction .................................................................................................................. 1

2 Data Vortex Packet Switch Architecture ........................................................................ 5
   2.1 Concept ..................................................................................................................... 5
   2.2 Topology .................................................................................................................. 6
   2.3 Data Connection Pattern ......................................................................................... 8
   2.4 Contention Resolution Mechanism ........................................................................ 10
   2.5 Contention at the Switch Input and Output ............................................................ 11
   2.6 Switching Mechanism ............................................................................................. 12
   2.7 Synchronous Operation ......................................................................................... 14
   2.7 Summary ................................................................................................................. 16

3 Data Vortex Performance ............................................................................................. 17
   3.1 Parameters under study ......................................................................................... 17
   3.2 Performance Under Uniform and Random Traffic ............................................... 18
      3.2.1 Injection Rate .................................................................................................. 19
      3.2.2 Exit Rate ......................................................................................................... 22
      3.2.3 Mean Latency ................................................................................................. 23
      3.2.4 Latency Distribution ...................................................................................... 25
   3.3 Summary ................................................................................................................. 27

4 A Data Vortex Packet Switch Architecture with Improved Latency Performance ........ 29
   4.1 Generic Data Vortex ............................................................................................... 30
   4.2 Improved Data Vortex Configuration ..................................................................... 30
      4.2.1 Simulation Results ......................................................................................... 31
   4.3 Output Contention Resolution Method .................................................................. 34
      4.3.1 Principle of Operation .................................................................................... 35
      4.3.2 Simulation Results ......................................................................................... 36

5 Implementation Proposal for Data Vortex Packet Switch ............................................. 41
   5.1 Packet Slot Format ............................................................................................... 41
   5.2 Node Implementation Scheme ............................................................................... 42
   5.3 Node Implementation based on Electroabsorption Modulator Technology .......... 44
      5.3.1 EAM Implementation Proposal for Data Vortex Switching Nodes .............. 45
      5.3.2 Photodiode-EAM Optical Gate Characteristics ............................................. 46
      5.3.3 EAM Implementation Proposal Characteristics ............................................... 47
   5.4 Summary ................................................................................................................. 48
Chapter 1

Introduction

The current and overwhelming increase in the Internet traffic and the growth of new and different kind of services are calling for higher data rates and flexibility that current networks cannot offer. Optical packet switching has emerged as the solution to provide the required bandwidth by means of increasing the throughput, efficiency, transparency and flexibility. However, to reach these promising expectations it is necessary to perform the routing and switching operations in the optical domain, instead of doing it electronically, which results in a remarkable bottleneck. Given the current photonic technology, hybrid optical-to-electronic configurations are used, resulting in a performance penalty due to the realization of the signal processing in the electronic domain. Many obstacles and challenges have to be overcome on the way to achieve the so longed all-optical packet switching. The principal challenges are focused on optical data processing, buffering, contention resolution and synchronization, due mainly to the current lack of RAM memory in the optical domain [1]. Data Vortex is a novel optical packet switching architecture, originally conceived for large-scale computer interconnections, that has many attractive features for mitigating some of these limitations in optical packet switching.

Contention resolution is one of the key problems in the design of optical packet switches. Contention occurs when more than one packet have to be switched to the same output port at the same time. Three methods are proposed to resolve this contention: buffering, routing deflection and wavelength conversion; these methods can be single or combined deployed. Regarding to the use of buffering, the utilization of electronic buffers is not possible for obtaining a good performance. Optical buffering has to be deployed, with the major drawback that it does not have random access. Optical buffering, with a collection of fixed long fiber delay lines, controls and switches, results in a great amount of hardware and complex controls. Deflection routing is easier to implement and decreases in great quantity the number of buffers needed in case it is deployed together with the buffering strategy. On the other hand, it penalizes remarkably the performance in terms of latency, latency distribution and throughput. Data Vortex architecture joins the use of delay lines together with an inherent deflection strategy. Although Data Vortex uses delay lines, it avoids complex hardware control needed in common buffers. This is achieved due to the distributed and simple control that Data
Vortex deploys along its structure. Data Vortex achieves packet routing and buffering simultaneously along its structure, performing deflection along its connection lines. Regarding to wavelength conversion, it usually comes accompanied by wavelength routing. This strategy adds the wavelength domain as another dimension in order to resolve contention, thus eliminating part of the optical buffers. On the other hand, today’s wavelength routing and wavelength conversion devices result in an important increase in terms of hardware complexity and cost. Data Vortex takes advantage of WDM technology just to increase the throughput, performing the packet routing insensitively with respect to the wavelength. This fact results in a remarkable increase in the scalability and in a reduction in the number of hardware resources needed in comparison with individual wavelength switching. Optical signal processing is another important challenge in optical packet switching. Currently, optical signal processing is in a very premature stage, thus packet header processing has to be done in the electronic domain, with the corresponding performance penalty. Data Vortex switch deploys a simplified routing strategy that reduces the number of logic decisions and makes the switching operations very simple. This simplified routing together with the simplified distributed control suggests the possibility to make an all-optical implementation of the Data Vortex packet switch.

Goals of the thesis

The Data Vortex attractive features, explained above, prompted us to carry out a depth analysis of this Data Vortex architecture, studying its characteristics and its possible implementation and application efficiency for optical networks. The goals pursued by this thesis can be summarized as:

- Presentation of the novel Data Vortex concept, including the explanation of: the Data Vortex structure, principle of operation, principal characteristics, performance results and advantages and disadvantages for optical packet switching.
- Improvement of the Data Vortex generic architecture in order to achieve a better performance and more suitable application for optical packet switching.
- Technology proposal for an efficient Data Vortex packet switch implementation.
- Presentation of suitable networking scenarios in which Data Vortex switch can perform more efficiently than conventional switches.

Structure of the thesis

Chapter 2 presents the Data Vortex packet switch principle of operation and characteristics. Data Vortex concept, structure, distributed control and switching mechanism are all presented in detail. Besides, it also presents the cases of possible contention together with the synchronous operation that Data Vortex requires. Chapter 3 studies the Data Vortex performance under uniform traffic in terms of the throughput,
latency and latency distribution. The performance results are analyzed as a function of the traffic load and the Data Vortex configuration. Contention probability at the switch output is also studied. In chapter 4, we propose a new Data Vortex architecture, which results in a remarkable latency improvement with respect to the generic one and enables a compact Data Vortex realization. It is additionally presented how this configuration improves the packet order at the switch output and permits the implementation of a simple output contention resolution mechanism within the Data Vortex structure. In chapter 5, we propose the use of electroabsorption modulator (EAM) technology for the implementation of Data Vortex switching nodes. The proposed implementation is analyzed in terms of a collection of parameters that identify the EAM technology as an efficient alternative for Data Vortex implementation. In chapter 6, the use of Data Vortex switch for Photonic Slot Routing (PSR) is presented. We propose a PSR networking scenario in which the use of Data Vortex switch is advantageous. Finally, chapter 7 and chapter 8 present the summary conclusions and the suggested further work respectively.
Chapter 2

Data Vortex Packet Switch Architecture

Data Vortex is a novel optical packet switching architecture. It is a multi-stage network architecture that has been designed for large-scale computer interconnections in the field of supercomputing. Data Vortex has been designed as part of the HTMT (Hybrid Technology Multithread) architecture, in which it acts as an optical interconnection network with the objective to reach a Petaflops scale supercomputer system [2]. This novel architecture has some attractive features for solving some of the current limitations of optical packet switching, such as optical processing, buffering and contention resolution. Data Vortex architecture has been designed to perform packet-routing and buffering altogether within the same three-dimensional structure. This architecture deploys a binary-tree self-routing mechanism that simplifies the optical processing needed for packet routing. Besides, Data Vortex adopts a self- and distributed control, which avoids the use of a central and external control. This characteristic simplifies the control necessary for contention resolution and makes Data Vortex attractive to scale it to a large number of input and output ports. All these features have prompted us to study its architecture characteristics and principle of operation.

2.1 Concept

Data Vortex presents a three-dimensional multi-stage architecture that, as it has been said, simplifies both the routing strategy and the control for contention resolution. Data Vortex switch deploys a multi-hop packet routing through several stages along its switch structure in which the switching and logical operations are very simple. The control is done within the switch itself using distributed signaling inside the structure. This self-routing and self-control makes unnecessary the use of any external control. On the other hand, the principle of operation assumes synchronous time slot operation, in which the packet slots have a fixed length and arrive synchronously at the nodes of the structure.

Data Vortex architecture consists of small units called nodes placed on a collection of concentric cylinders. Each node is positioned on a certain cylinder, on a
specific height of that cylinder and at a certain angle of the ring on which they are placed (see Fig.2.1). It has to be noted that the rings are virtual; they are not the physical connections of the structure. The nodes are in charge of routing the packets throughout the structure toward the target output port. In this structure, the packets enter the switch fabric by the nodes of the outermost cylinder and they are routed through the structure following a binary tree fashion toward the nodes of innermost cylinder by which packets exit. Thus, the nodes of the outermost cylinder work as the input ports whereas the nodes of the innermost cylinder work as the output ports of the switch fabric. In this routing the packets are processed in a synchronous and parallel manner. Each packet is processed in a node and jumps to the next node within the given slot of time. The packets can hop either to a node of the next inner cylinder, going forward within the switch, or to another node placed in the same cylinder. When the packet is sent to the same cylinder it can be because of two reasons: in order to get the right way to reach its proper output port or because of the deflection mechanism to resolve contention. The inherent deflection mechanism deployed within the switch fabric guarantees that never two packets reach the same node in a given slot of time. Thus, each node switches only one packet in a given slot of time. Fig. 2.1 shows the routing of a packet whose target output is height h=11.

![Figure 2.1. Example of packet routing in a Data Vortex switch fabric with four heights H=4, binary coded, and with five nodes per height A=5.](image)

### 2.2 Topology [2-6]

The nodes are placed over several virtual rings, conforming the concentric cylinders structure. However, these rings are not the physical connections between the nodes. The pattern of physical connections will be presented next. The cylinders are characterized by the following three parameters:

- **Height** (H) refers to the number of heights of the cylinders. They are expressed in binary because of the binary tree routing fashion
implemented by the switch. Expressing them binary coded is a suitable way to understand the switching mechanism.

- **Angles** ($A$) refers to the number of nodes placed on the circumference of the virtual ring (this parameter is usually a small odd number less than 10).

- **Levels** ($J$) which is the number of concentric cylinders the structure has and depends on the $H$ value. Its value is given by $J = \log_2(H) + 1$.

Thus, the switch fabric presents $H \cdot A$ input ports, $H \cdot A$ output ports, and $H \cdot A \cdot J$ nodes. Figure 2.2 shows a switch fabric structure characterized by $H = 4$, $A = 5$ and $J = 3$.

Each node of the structure can be identified by the three parameters $r$, $\theta$, and $z$ which determine its position in the switch:

- Parameter $r$ identifies the cylinder or level of the structure. The range of $r$ is: $r = 0 - (J-1)$. $r = 0$ identifies the innermost cylinder, whereas $r = J-1$ identifies the outermost cylinder.

- The parameter $\theta$ identifies the angle position on the virtual ring of the cylinder. The range of $\theta$ is: $\theta = 0 - (A-1)$.

- The parameter $z$ identifies the height on the cylinder. The range of $z$ is: $z = 0 - (H-1)$.

![Figure 2.2. Data Vortex switch structure. (a) Data Vortex structure identified by $r$, $\theta$, $z$. (b) Input Ports. (c) Output Ports](image)

For the case presented in Figure 2.2 a), the range of these parameters is: $r = 0 - 2$, $\theta = 0 - 4$ and $z = 0 - 3$. 
The three dimensional structure enables to make the packet routing and buffering altogether within the same structure, although it results in a large increase in the number of resources, in terms of the nodes and fiber required, especially when the number of input and output ports increase.

2.3 Data Connection Pattern [2-6]

A packet at a given switching node can be forwarded to the inner cylinder or to the same cylinder in each clock cycle. In both cases the packets go one angle forward. For the case in which the packet hops to the inner cylinder, the packet reaches an inner level, goes an angle forward, and remains at the same height. Figure 2.3 shows the pattern of these connections.

![Figure 2.3. Data Connections to the inner level](image)

In case that the packet hops to the same cylinder, the packet remains on the same cylinder, goes an angle forward, and reaches a different height in the cylinder. Figure 2.4 shows the interconnection scheme for nodes on the same cylinder. These connections are used to route the packet or to deflect it in order to resolve contention within the switch. Regarding to the packet routing, this scheme makes sure that every output port can be reached from any input port. In reference to the deflection technique, the interconnection pattern minimizes the packet deflection probability because of the reduced correlation among different cylinders. In this scheme the routing path of the packet is recovered in the next hop.

Below we generalize this data connection pattern for any switch fabric size \((H, A, J)\). Given a node \(B(r, \theta, z)\) placed on level \(r\), angle \(\theta\), and height \(z\), it can send data to node \(C(r-1, \theta+1, z)\) of the inner level, and also can sends data to node \(D(r, \theta+1, h_r(z))\) of the same level. \(h_r(z)\) is the height resulting from the transformation of the height \(z\) of the node \(B\) on level \(r\). This transformation depends on the level \(r\), as is explained below. The height is binary coded, thus, if the switch fabric height is \(H\), \(\log_2(H)\) bits are needed to code each height. Since the number of levels \(J\) is given by \(J=\log_2(H)+1\), we can write the height \(z\) in binary as \(z=[z_{J-1}, z_{J-2}, \ldots, z_r, z_{r-1}, \ldots, z_2, z_1, z_0]\). For the case of Figure 2.4, \(z=[z_3, z_1, z_0]\). The transformation \(h_r(z)\) consists first in reversing the \(r\) lower bits, then adding 1 in modulus 2\(^r\), and finally reversing the \(r\) lower bits again.
The transformation steps that result in the connection pattern of Figure 2.4 are presented:

- For the nodes on the outermost level, $r=3$:

  $z = 0 \rightarrow z = [000] \rightarrow [000] \rightarrow [001] \rightarrow h_3(0) = [100] \rightarrow h_3(0) = 4$
  $z = 1 \rightarrow z = [001] \rightarrow [100] \rightarrow [101] \rightarrow h_3(1) = [101] \rightarrow h_3(1) = 5$
  $z = 2 \rightarrow z = [010] \rightarrow [010] \rightarrow [011] \rightarrow h_3(2) = [110] \rightarrow h_3(2) = 6$
  $z = 3 \rightarrow z = [011] \rightarrow [110] \rightarrow [111] \rightarrow h_3(3) = [111] \rightarrow h_3(3) = 7$
  $z = 4 \rightarrow z = [100] \rightarrow [001] \rightarrow [010] \rightarrow h_3(4) = [010] \rightarrow h_3(4) = 2$
  $z = 5 \rightarrow z = [101] \rightarrow [101] \rightarrow [110] \rightarrow h_3(5) = [011] \rightarrow h_3(5) = 3$
  $z = 6 \rightarrow z = [110] \rightarrow [011] \rightarrow [100] \rightarrow h_3(6) = [001] \rightarrow h_3(6) = 1$
  $z = 7 \rightarrow z = [111] \rightarrow [111] \rightarrow [000] \rightarrow h_3(7) = [000] \rightarrow h_3(7) = 0$

- For $r=2$:

  $z = 0 \rightarrow z = [000] \rightarrow [000] \rightarrow [001] \rightarrow h_2(0) = [010] \rightarrow h_2(0) = 2$
  $z = 1 \rightarrow z = [001] \rightarrow [010] \rightarrow [011] \rightarrow h_2(1) = [111] \rightarrow h_2(1) = 3$
  $z = 2 \rightarrow z = [010] \rightarrow [010] \rightarrow [011] \rightarrow h_2(2) = [001] \rightarrow h_2(2) = 1$
  $z = 3 \rightarrow z = [011] \rightarrow [110] \rightarrow [111] \rightarrow h_2(3) = [000] \rightarrow h_2(3) = 0$
  $z = 4 \rightarrow z = [100] \rightarrow [100] \rightarrow [101] \rightarrow h_2(4) = [110] \rightarrow h_2(4) = 6$
  $z = 5 \rightarrow z = [101] \rightarrow [101] \rightarrow [110] \rightarrow h_2(5) = [111] \rightarrow h_2(5) = 7$
  $z = 6 \rightarrow z = [110] \rightarrow [110] \rightarrow [111] \rightarrow h_2(6) = [101] \rightarrow h_2(6) = 5$
  $z = 7 \rightarrow z = [111] \rightarrow [111] \rightarrow [100] \rightarrow h_2(7) = [100] \rightarrow h_2(7) = 4$

- For $r=1$:

  $z = 0 \rightarrow z = [000] \rightarrow [000] \rightarrow [001] \rightarrow h_1(0) = [001] \rightarrow h_1(0) = 1$
  $z = 1 \rightarrow z = [001] \rightarrow [001] \rightarrow [000] \rightarrow h_1(1) = [100] \rightarrow h_1(1) = 0$
  $z = 2 \rightarrow z = [010] \rightarrow [010] \rightarrow [011] \rightarrow h_1(2) = [011] \rightarrow h_1(2) = 3$
  $z = 3 \rightarrow z = [011] \rightarrow [011] \rightarrow [010] \rightarrow h_1(3) = [100] \rightarrow h_1(3) = 2$
  $z = 4 \rightarrow z = [100] \rightarrow [100] \rightarrow [101] \rightarrow h_1(4) = [101] \rightarrow h_1(4) = 5$
  $z = 5 \rightarrow z = [101] \rightarrow [101] \rightarrow [100] \rightarrow h_1(5) = [110] \rightarrow h_1(5) = 4$
  $z = 6 \rightarrow z = [110] \rightarrow [110] \rightarrow [111] \rightarrow h_1(6) = [111] \rightarrow h_1(6) = 7$
  $z = 7 \rightarrow z = [111] \rightarrow [111] \rightarrow [110] \rightarrow h_1(7) = [110] \rightarrow h_1(7) = 6$
For the nodes of the innermost level $r=0$:

$z = 0 \rightarrow z = [000] \rightarrow [000] \rightarrow [000] \rightarrow h_{0}(0) = [000] \rightarrow h_{0}(0) = 0$

$z = 1 \rightarrow z = [001] \rightarrow [001] \rightarrow [001] \rightarrow h_{0}(1) = [001] \rightarrow h_{0}(1) = 1$

$z = 2 \rightarrow z = [010] \rightarrow [010] \rightarrow [010] \rightarrow h_{0}(2) = [010] \rightarrow h_{0}(2) = 2$

$z = 3 \rightarrow z = [011] \rightarrow [011] \rightarrow [011] \rightarrow h_{0}(3) = [011] \rightarrow h_{0}(3) = 3$

$z = 4 \rightarrow z = [100] \rightarrow [100] \rightarrow [100] \rightarrow h_{0}(4) = [100] \rightarrow h_{0}(4) = 4$

$z = 5 \rightarrow z = [101] \rightarrow [101] \rightarrow [101] \rightarrow h_{0}(5) = [101] \rightarrow h_{0}(5) = 5$

$z = 6 \rightarrow z = [110] \rightarrow [110] \rightarrow [110] \rightarrow h_{0}(6) = [110] \rightarrow h_{0}(6) = 6$

$z = 7 \rightarrow z = [111] \rightarrow [111] \rightarrow [111] \rightarrow h_{0}(7) = [111] \rightarrow h_{0}(7) = 7$

Figure 2.4 shows that all the output nodes on the innermost level are reachable from every input node on the outermost level. As the packets go forward through the structure, the number of output nodes reachable from the levels is being divided by 2. For the case of Figure 2.4, the 8 output heights are reachable from any input node, because there is a path along the switch to reach them. However from a node placed on the level $r=2$ only 4 heights can be reached, from level $r=1$ only 2 heights, whereas on level $r=0$ the packet can only circulate along the same height. It can be expressed saying that a packet processed by a node on level $r$ can reach $2^r$ heights.

At the same time, node $B(r, \theta, z)$ receives data either from the node of outer level $E(r+1, \theta-1, z)$ or from the node of the same level $F(r, \theta-1, H_{r}(z))$. $H_{r}(z)$ is the height resulting from the transformation of the height $z$ of the node $B$ on level $r$, which logically verifies $H_{r}(z) = h^{-1}_{r}(z)$.

Finally, it has to be said that for proper operation, the connections to the same level are all the same length as are to be all the connections to the inner level.

### 2.4 Contention Resolution Mechanism [2-6]

The switching nodes, apart from the two data inputs and two data outputs, have one additional control input and one control output to receive and send control signal for resolving contention. Each node of the structure presents two data inputs, but it cannot receive more than one packet in a given slot of time, so a distributed control signaling mechanism is implemented to resolve the possible contention. When two nodes want to send a packet to the same node in the same slot of time, the priority is given to the node on the same level, that is, the packet going to the same level has priority over the packet going to the inner level. A control message from the node on the same level is sent to the node of the outer level before the packet arrives at it. This message warns the node of the outer level not to send the packet that is receiving, so contention is resolved because the node of the outer level has no choice but to deflect the packet to its same level. Thus, when a node sends a packet to the same level, it also sends a control message, which can be just a single bit, to the corresponding node of the outer level, warning about the blocking. Figure 2.5 a) shows how this contention resolution method operates. Node $A$ and node $B$ want to send a packet to node $C$. Node $A$ has the priority, as it is on the same level of node $C$, so it sends the packet to node $C$ and also sends a control message to node...
$B$ notifying to it not to send the packet to the inner level. Thus, node $B$ has no choice but sending the packet to the node to which it is connected on the same level, node $D$. On the other hand, Figure 2.5 b) presents the situation in which there is no contention and node $B$ can send the packet forward to the inner level.

![Figure 2.5. Control resolution mechanism. (a) Deflection used to resolve the contention. (b) No need of deflection.](image)

Thus, with this contention resolution mechanism packets do not need to carry any priority information, as the priority is given to the same level interconnections over the inner ones. Furthermore, since packets are routed from the outermost level to the innermost level, it is likely that a packet on an inner level is older (respecting to the time since entered the structure) than a packet on an outer level, and then priority is likely to be given to older packets. Consequently, the packet order at the output will remain almost invariable. This simple control mechanism assures that during a single time slot only one packet will enter a node. It will enter either from the same level or from the outer level, but never from both, that is, each node processes only one packet in a given time slot. Regarding to the control signal, as it just warns the node if it is possible or not to forward the packet to the inner node, just 1 bit signal is enough. Therefore, the use of this contention resolution method assures no packet get lost within the switch fabric, and makes unnecessary additional hardware to control the switch fabric packet routing.

### 2.5 Contention at the Switch Input and Output

The control mechanism deployed, which resolves the contention within the switch fabric, has been already presented. On the other hand, contention can occur at the switch input when packets try to enter the switch. Contention can also occur at the switch output when two or more output nodes share the same destination address. In order to resolve this contention, some sort of mechanism must be implemented. Electronic buffering at
the input and output has been proposed in [6] to control the data flow into and out of the switch. In chapter 4, we propose a solution that resolves the output contention within the Data Vortex structure itself.

Contention occurs at the switch input if the packet injection into the switch is not controlled by some sort of data flow control mechanism. Packets that try to enter the switch can collide with packets that are already within the structure on the outermost level. This situation will happen when a packet that is on the outermost level is deflected to the same level and reach a node that is also used as an input port by which another packet wants to be injected. The probability of this situation will be one of the main points of study in the next chapter. This study will be done in terms of the parameter *Injection rate*, which is defined as the ratio of successful packet injections over the total number of injection attempts. Logically, we can foresee that if the traffic load of packets that tries to enter the switch fabric is high and all the nodes of the outermost level are used as input ports, collisions will occur very frequently. Thus, a way to reduce or even avoid the use of buffering at the switch input is not using all the available input ports. This topic will be subject of a more in depth study in the next chapter.

The second contention problem occurs at the switch fabric output. When more than one output node have the same address and are connected to the same fiber, contention at the output will occur if no mechanism is implemented to resolve it. For example, in the Data Vortex switch fabric studied in [3] every output node on the same height has the same address. Thus, if two or more packets exit the switch in a given clock cycle by output ports on the same height, collision will occur. Therefore, some kind of contention resolution method should be implemented at the output. The use of buffers outside the structure to control the data flow at the switch output is proposed in [3]. The solution we propose in chapter 4 consists in using the buffering resources of the Data Vortex structure itself. The strategy deployed is to circulate the contending packets along the innermost ring until they can exit without contention. This solution will reduce or even make unnecessary the use of buffering outside. Another possibility could be the implementation of lines of feedback toward the input ports of the switch fabric. In the next chapter, the probability of collisions due to this output contention is obtained.

### 2.6 Switching Mechanism [2-6]

Packets contain a header in which several bits identify the output port by which the packet has to exit. The switch fabric processes this header routing information bit by bit along the structure throughout the different levels. Each bit of the header is read by each one of the cylinders or levels, that is, every node on the same level reads the same single bit of the header. In this way, the packet is being routed inside the switch, going through the different levels until exiting by the proper output port. The packet payload remains transparent along the routing. The switching mechanism follows a binary tree mechanism accompanied by the simple contention control explained before. This switching mechanism ensures that every packet that enters the switch is going to come out properly by the correct output port. As explained previously, the packets are switched
in a strictly parallel and synchronous manner by means of a clock that controls the operation, being each packet processed at a single node in a given a slot of time. The simple switching and logical operation at each node will enable to minimize the switching time. This switching mechanism is explained in detail below.

Each packet consists of several header bits that identify the height of the target output node, and depending on the configuration it can also specify the angle position of the output node. A given node \( A(r,0,z) \), on level \( r \), different from 0, and at height \( z \), receives a packet and reads the corresponding header bit of the packet. It compares it with the bit at position \( r \) of its height \( z \) codified in binary. If the bits do not coincide the packet cannot be sent to the inner level because the packet can not reach the target output from that height \( z \) on an inner level. Thus, the packet is sent to the same level changing in this way the height. Due to the connection pattern explained before, it is sure that sending the packet to the same level it will reach a height from which on the inner level will be able to reach the target output. In case that both bits coincide, it also has to be taken into account the control bit, which warns if there is blocking in the inner level. If there is blocking, the packet is sent to the same level, deflecting the packet to resolve contention. Otherwise, if no blocking occurs the packet can be sent to the inner level. Thus, in this self-routing scheme, each node has to read one bit of the header, and also the control bit. Thus, the packet routed by node \( A(r,0,z) \) is sent from level \( r \) to the next inner level \( r-1 \) as long as the following two conditions are met:

1) The target output is accessible from the height \( z \) on level \( r-1 \).
2) The packet is not blocked by another packet on the level \( r-1 \).

When the packet reaches level \( r = 0 \) it is already routed in case the angle position of the output node is not specified. In case that all the output ports at the same height are connected to the same fiber, the packet could circulate along this innermost ring to avoid contention. Otherwise, if the angle is specified as output port, the nodes of level cero would have to look at the corresponding bits to enroute the packet to the specific angle output. Thus the innermost cylinder can circulate the packet along the ring to provide temporal buffering or to route the packet to a specific angle.

Figure 2.6 a), b) and c) presents an example of the packet routing explained above. In this example, all the output nodes on the same height have the same address, so the packet header contains just two bits to select one of the four heights as target output. The value of the packet header is 11, which means that the target output is any of the five nodes on the innermost level at height \( z=11 \). The packet enters the switch by an input node on height \( z=01 \). This node looks at the first bit of the header, which value is 1, and compares it with the first bit of its height, which is 0. Since they do not coincide the packet is sent to the node on the same level (see figure 2.6 a)). The next node on height \( z=11 \), compares the same first bit of the header with the same bit of its height, which in this case is 1. Since both bits coincide and the control bit received from the inner level (represented in the figure 2.6 b) by the black arrow) warns there is no block, the packet is sent to the inner level (see figure 2.6 b)). The next node looks at the second bit of the header, which is 1, and compares it with the second bit of its height, which in this case is
1 as well. Thus, as the control bit that receives from the inner level says that there is no block, the packet is sent to the inner level, reaching the target output by which it exit the structure (see figure 2.6 c)).

Figure 2.6. Data Vortex packet routing

2.6 Synchronous Operation [6]

Data Vortex principle of operation relies on a synchronous scheme that allows switching the packets in the parallel way presented above. This synchronous operation is necessary since it is the key so the contention resolution mechanism works properly. There must be a timing communication schedule within the switch fabric in which the data packets and control messages arrive the nodes at appropriate moments of time. The synchronous operation is explained below.

It has been explained that for contention resolution the transmissions to the same level have priority over the transmissions to the inner level. Nodes placed on a given level have to receive a control message from the inner level, in order to route the packet it receives. Therefore, this control message has to arrive the node at least at the same time the packet arrives. Once the node has received the packet and the control message, it is able to decide the next hop for the packet. According to this decision, it sets the corresponding control message that it has to sent to the corresponding node on the outer level. This control message will have to arrive at least at the same time the packet arrives to the node on that outer level. Thus packets must arrive earlier at the inner levels. The inner the level is, the earlier the packets arrive at its nodes. Also it is necessary that all the nodes placed on the same level receive packets at the same time. Figure 2.7 shows in detail the synchronous operation that is specifically explained below.

Figure 2.7 presents a Data Vortex switch fabric seen from its top view and characterized by $H = 4$, $A = 3$ and $J = 3$. It is not strictly a top view since some nodes of the structure do not appear because have been skipped in order to show more clearly the synchronous operation. Thus, not all the nodes of the figure are the ones placed at the top
of the switch fabric. In order to identify the nodes of the structure they have been tagged with the parameters $r$, $\theta$ and $z$. In the example proposed it is considered that not only the height is specified in the packet destination address, but also the angle by which the packet will exit. The figure shows that the nodes on the innermost cylinder can circulate the packet until it reaches the output node at the target angle. The synchronous operation is based on:

- Packet time slots must have the same length $T$ and must be aligned in time before entering the switch fabric. Between packets there has to be a guard time necessary to change the switch state at each node. We define here the packet time slot $T$ to include the guard time necessary for the node processing time.

- Connections to the same level add a latency equal to the packet time slot $T$. This connection includes the part of fiber used as a delay line inside a node to have time to make the logic operations that determine the switching state of the node. On the other hand the connections to the inner level add a latency $T - \xi$, including the same time for determining the switch state. Finally the control connections have a length so that a time $\xi$ takes a control message to reach the corresponding node on the outer level.

- The packet time slot $T$, not only determines the minimum length of the connection lines, but also the minimum duration of the clock cycle. In this way, a packet inside the switch fabric is switched only by a single node of the structure in a given clock cycle since it does not reach the next node until it has been switched entirely by the current node.

![Figure 2.7. Synchronous operation scheme of the Data Vortex switch fabric](image)

Since the contention control is made level by level, from the innermost to the outermost level, two packets arrive at two nodes on consecutive levels with a time gap equal to $\xi$, which is the time that takes the control message to arrive the outer level. In the Figure 2.7, $t_0$, $t_1$ and $t_2$ show this time gap between the packets arrivals at the nodes on the different cylinders of the structure. When a packet is forward to the inner level, the
difference in time $\xi$ is compensated by the shorter connection line to the inner level, which adds a latency equal to $T - \xi$ instead of $T$. Thus the packet arrives at the inner level at the proper time, as it is shown the figure.

2.7 Summary

In this chapter, Data Vortex architecture has been presented. Data Vortex concept, structure and principle of operation have been explained. We have shown that the Data Vortex three-dimensional structure together with the distributed control deployed enables to make the packet routing and buffering within the same structure without the need of any external control. This distributed control makes Data Vortex attractive to scale it to large switch fabric, in which an external control would become very complex. On the other hand, the multi-dimensional structure deployed results in a large increase in the number of resources and bulkiness inside the switch when the number of input and output ports increases. Thus, a very simple, integrable and economical implementation of the switching nodes must be achieved to consider a feasible implementation of Data Vortex switch. It has been also presented the Data Vortex time slotted and multi-hop operation. The strict synchronous operation has been presented. It has been also presented the self-routing operation which results in very simple switching logic operations needed for the packet routing. This suggests the possibility to make an all-optical Data Vortex implementation. We have also presented the contention resolution method, which guarantees that no packet within the structure gets lost, and the connection pattern that minimizes the probability of contention. However, contention problems at the switch input and output have been identified. Furthermore, the multi-hop operation together with the inherent deflection technique deployed will result in a latency penalty that has to be further studied.
Chapter 3

Data Vortex Performance

Data Vortex switch principle of operation and characteristics have been presented in chapter 2. A generic Data Vortex performance analysis has been published in [3,6]. In this chapter, we confirm previously released results by extensive computer simulations. As the developed simulator is shown to work properly, relevant topics like the output packet contention, mentioned in the previous chapter, will be studied in depth within our performance analysis. We also contribute with several new and interesting results. The performance will be analyzed under uniform and random traffic in terms of the throughput, latency and latency distribution for a given switch fabric configuration, by a collection of parameters that will characterize the switch fabric performance.

3.1 Parameters under study

The performance analysis is performed in terms of the following collection of parameters:

- **Injection Rate** characterizes the throughput performance when no contention is considered at the switch output, as the switch studied in [3], or when some sort of buffering at the output resolves it. It is defined as the ratio of successful packet injections over the total number of packet injection attempts. Since the switching operation assures that no packet is lost within the switch fabric and if no contention is considered at the output, the only packet loss will occur at the switch input, in case no buffering is implemented at the input. Thus, Data Vortex architecture assures no packet loss within the switch fabric at the expense of not allowing every packet injection. For a certain packet loss (if no input buffering is implemented), the injection rate gives the maximum throughput the switch fabric can support.

- **Exit Rate** is a parameter that in conjunction with the injection rate will characterize the throughput of the switch fabric when there is contention at the switch output and when it is not resolved by some sort of buffering outside the
switch. Contention occurs when more than one output node have the same destination address, as the switch studied in [3] (where all the output nodes on the same height have the same address), and are connected to the same fiber. If no buffering outside the switch resolves this contention, packet will get lost. In our study, no buffering outside is considered and then when more than one packet is routed to the same output fiber at the same time, collision will occur and it is considered that only one of them contributes to the switch fabric throughput. Thus we define the Exit Rate as the ratio of packets that successfully exit the switch fabric over the total number of exit attempts. Therefore for a specific packet loss rate, the Exit Rate gives the maximum throughput that the switch fabric can handle if no data flow control is implemented at the output. Thus the lowest value given by the injection rate and exit rate determines the throughput of the switch fabric.

- **Latency** is the average time since packets enter the switch until they are routed to the target output port and can exit the switch. It is studied in terms of the average number of hops that a packet makes to reach the target output port. In order to calculate the latency in terms of time, first it has to be estimated the switching times at the nodes and the latency added by the connection lines.

- **Latency distribution** determines the latency variance of the packets. It is presented as the percentage of packets as a function of the number of hops they make. Small latency variance results in small packet disorder at the output, which is important to ease the future packets reassembly at the receiver. In addition, it is important that packets hops roughly the same number of times, so there is no important differences in the degradation of the signals between the packets.

### 3.2 Performance under Uniform and Random Traffic

The performance is analyzed under uniform and random traffic. The destination addresses distribution is uniform, thus every output port has the same probability to be the target output port of the injected packet. The packets are injected synchronously in time slots to the different input nodes. The probability of packet presence in a given slot of time is random and characterized by the parameter load. The load characterizes this probability and is the same for all the input ports. If the load is 1.0 there will be a packet at every slot of time, that is, in every slot of time a packet is injected to every input port. In case the load is 0.5, half of the slots in average will carry packets. Another traffic characteristic is that packet injections to a given input port are independent of the injections to the other input ports.

The performance is analyzed for a Data Vortex switch in which all the output nodes on the same height have the same address, as in [3]. Thus packet destination addresses identify just the target height of the switch fabric and packets will be able to
exit by any of the \( A \) output nodes on that target height. Also it is considered that the nodes per height used as input ports are positioned at the same angle position on every height, thus contention at the input and within the switch is minimized.

### 3.2.1 Injection Rate

First, the performance is analyzed in terms of the injection rate in order to estimate the sustained throughput of the Data Vortex switch fabric. This injection rate will be mainly determined by the traffic load and by the number of nodes used as input ports. Thus the injection rate is studied as a function of the traffic load offered, the number of nodes used as input ports and also in function of the switch fabric size. Figure 3.1 shows the injection rate versus the load for different switch fabric sizes, with five nodes per height \( A = 5 \) and all the outermost nodes per height used as input ports \( A' = A = 5 \). As shown in the figure, the injection rate decreases remarkably when the load increases. Since all the input nodes available are used, the contention at the outermost level becomes notably larger when the load increases, and thus more packet injections are rejected. In addition, it is shown in figure 3.1 that the injection rate decreases when the switch fabric height becomes larger. The increase in the switch height means an increase in the number of input and output ports and in the number of levels to reach the target output. This results in an increase in the traffic congestion within the switch fabric, and so at the outermost level, thus more packet injections are rejected. Figure 3.1 shows that the difference in the injection rate performance becomes very small when the switch height becomes larger than 32, \( H > 32 \). It means that the Data Vortex switch is scalable to a large number of input/output ports maintaining a similar injection rate even under heavy loads. On the other hand, as shown in the figure, the values of the injection rate are

![Figure 3.1. Injection rate versus traffic load offered for different switch fabric heights, with five nodes per height \( A = 5 \) and all of them used as input ports \( A' = 5 \).](image-url)
very low, even under low loads, so it is clear that for a good performance it is not possible to make use of all the available input ports.

Figure 3.2, shows the injection rate performance for a switch fabric with five nodes per height $A=5$, which uses two of these nodes as input ports $A'=2$. As shown, there is a remarkable improvement in the injection rate, compared with the case in which all the five nodes per height are used as input ports. However, for switch fabric heights larger than 8, $H>8$, the injection rate decreases remarkably under medium and heavy loads (load>0.6). It is shown that the injection rate is not very high under conditions of heavy load. For example, under load = 0.9, a switch fabric with $H = 64$ has an injection rate equal to 69.76%. If no data flow control were implemented at the input, around 3 out of 10 packets would get lost.

![Injection Rate vs Load, A=5 A'=2](image)

Figure 3.2. Injection rate versus traffic load offered for different switch fabric heights, with five nodes per height $A=5$ and two of them used as input ports $A'=2$.

On the other hand, Figure 3.3 shows that when only one out of five nodes per height are used as input ports ($A=5$, $A'=1$) the injection rate increases extraordinarily. The best performance is achieved for height $H=4$ because every packet is injected successfully even under maximum load=1.0. For the rest of heights, the resulting injection rate is around 100% even under heavy loads. For example, under maximum load (load = 1.0), a large-scale switch fabric with $H=2048$ presents an injection rate of 97.31%. Figure 3.3 also shows how the injection rate for switch fabrics with $H>4$ practically remains the same, illustrating Data Vortex good scalability. The good injection rate performance achieved in this case is at the expense of not using all the resources available, with the corresponding increase in the hardware cost of the switch fabric. Thus for a given number of input ports, this configuration increases the hardware cost by a factor of five.
Figure 3.3. Injection rate versus traffic load offered for different switch fabric heights, with five nodes per height $A=5$ and one of them used as input port $A'=1$.

Figure 3.4 shows how the injection rate increases when the number of resources increases. It shows the injection rate vs. the load for a switch fabric with $H=32$, and 32 input ports. The different curves plotted correspond to the different number $A$ of nodes per height that the switch fabric has. As shown the difference in the performance for the different number of resources used becomes more remarkable under heavy loads. In this case, for $H=32$ and under heavy load, when the number of resources is increased by a factor of three (the number $A$ of nodes per height changes from 1 to 3), the injection rate improves extraordinarily by a factor slightly lower than three. And when the number of resources increases by a factor of five, the injection rate almost increases by a factor of four, reaching almost 100% of injection rate.

Figure 3.4. Injection rate versus traffic load offered for switch fabrics with $H=32$, one node per height used as input port and different number of nodes per height.
Thus, at the moment to implement a Data Vortex switch fabric a balance between the injection rate performance and the hardware cost must be reached. Under uniform and random traffic, it has been shown that in order to get a the throughput near 1.0 the ratio \( \psi \) equivalent to the number of nodes per height used as input ports over the number of nodes per height has to be equal or less than 1/5, resulting in an increase in the number of hardware resources. As consequence, the number of switching nodes within the switch fabric will be large. Thus, one of the major goals is to implement the switching nodes in a very simple, economical and integrable way. In Chapter 5 we propose some technologies in order to reach this objective.

### 3.2.2 Exit Rate

The exit rate is an interesting parameter in order to study the traffic offered at the switch output. If more than one switch output node have the same address and are connected to the same fiber, some sort of mechanism must be implemented to resolve this contention. The exit rate indicates the grade of contention at the switch output that will determine the number of resources needed if some kind of buffering is implemented at the output to resolve the output contention.

The exit rate is analyzed for switch configurations with ratio \( \psi =1/5 \) that as shown achieve an injection rate near 100 \% for every switch fabric height. Figure 3.5 shows in particular the exit rate versus the traffic load offered for different switch fabric heights with five nodes per height, and one of them used as input port, \( A=5, A'=1 \). Since it is considered that the packet destination distribution is uniform and packets are injected by only one input port per height, the probability that more than one packet exit by the same height in the same clock cycle is low, although it is not insignificant, as shown in the figure. It is shown that the exit rate remains the same although the switch fabric height increases. The reason is that, for the configuration under study with \( \psi =1/5 \), the injection rate is roughly the same for all the switch fabric heights and thus the same traffic load is offered. And also because the packet destination distribution remains uniform independently of the number of heights.

As shown, the exit rate decreases with the load. Since more packets are injected, more throughput is sustained and the probability that more than two packets exit by the same height in the same clock cycle increases. This contention at the output is very relevant especially under heavy loads. Under maximum load more than 30\% of the packets would get lost for switch fabric heights \( H>4 \) (see in the figure that the exit rate is around 67\%). These figures logically will become lower if a non-uniform packet destination distribution is applied or/and bursty traffic is offered to the switch since the probability of contention at the output will increase extraordinarily.

This exit rate will determine the throughput that the switch fabric can support in case no mechanism is implemented to resolve the output contention. For the configuration under study, the traffic load offered to the switch should be below 0.2, in
order to have a packet loss around 5%, which completely penalizes the throughput supported by the switch fabric. Therefore if all the output nodes on the same height are connected to the same output fiber, as in the configuration studied in [3], a contention resolution mechanism should be implemented. In chapter 1 we mentioned some possibilities, like the packet circulation along the innermost ring or the use of lines of feedback toward the input ports of the switch fabric. In chapter 4 it is proposed and analyzed one mechanism to resolve this output contention.

3.2.3 Mean Latency

The switch latency is studied in terms of the average number of hops that packets make to reach the target output. This mean number of hops is studied as a function of the traffic load offered, the switch fabric height, and the ratio $\psi$. First is analyzed the mean latency performance for the configuration with five nodes per height $A=5$ and one of them used as input port $A'=1$, which, as shown, support a throughput near 100%. Figure 3.6 presents the mean latency versus the load offered for the different switch fabric heights. Logically the number of hops increases when the switch fabric height increases, since more levels have to be traversed by the packets to be routed to a larger number of heights. The mean latency also increases with the load, since more traffic congestion within the switch will result in an increase in the contention, in the number of deflection hops and so in the overall number of hops. However, as figure 3.6 shows, when the ratio of input ports used $\psi$ is $1/5$ the increase in the number of hops is not very important, even under heavy loads. This is because, with a ratio $\psi=1/5$, the traffic congestion is not so high, and the connection pattern, which provides reduced correlation among the different cylinders of the structure, minimizes the packet deflection probability and thus few deflections occur. As an example, under maximum load=1.0, for a switch fabric with
H=64, the mean latency is less than 13 hops (see figure). This configuration with 64 heights has 7 levels, and thus the minimum number of hops a packet would have to make to reach the output is 6. However, for the routing mechanism this case only occurs if the packet destination address coincides with height by which the packet entered and no contention within the switch occurs. Since the destination address distribution is uniform and independent of the input port, the average number of hops a packet would have to do to reach the output if contention never occurs is 9 hops. Thus in this case 4 out of the 13 hops are due to deflections. Figure 3.6 also shows that for switch fabrics with bigger heights, the mean latency increases more when the load offered increases than in switch fabrics with smaller heights. The reason again is the bigger congestion within switch fabrics with bigger size.

![Figure 3.6. Mean number of hops vs. traffic load offered for different switch fabric heights, with five nodes per height A=5 and one of them used as input port A'=1.](image)

Next we analyze the mean latency performance as the function of the ratio $\psi$. Figure 3.7 presents the mean latency versus the height for three different ratios of input ports used $\psi=1/5$, $3/5$ and $5/5$, under maximum load=1.0. As shown in the figure, the mean latency is linearly proportional to $\log(H)$. It is also shown that the mean latency does not increase with the height at the same pace for the different ratios. The increase in the mean latency when the height increases is very low for the ratio $\psi=1/5$ compared when the ratio is $3/5$ and $5/5$. The reason again dwells on the smaller congestion within the switch when only one input port per height is used. Also it is shown that the difference in the mean latency between the configurations with different ratios $\psi$ becomes more noticeable when the height increases. For example, the difference in the latency for a configuration with $H=32$ between ratios $\psi=1/5$ and $5/5$ is around 4 hops, whereas for $H=256$ the increase is around twice, 8 hops. It confirms that switch fabrics with bigger sizes present more congestion. Another important feature figure 3.7 shows is that the increase between ratios $\psi=1/5$ and $\psi=3/5$ is very large, whereas the increase between ratios $\psi=3/5$ and $\psi=5/5$ is much more narrow. As a consequence, the latency
performance results have shown that for a ratio $\psi = 1/5$ Data Vortex has very small latency and has very good scalability to large-scale switch fabrics.

![Mean Latency vs Height, $A=5$, Load=1.0](image)

Figure 3.7. Mean number of hops vs. switch fabric height for different ratios $\psi = 1/5$, $3/5$ and $5/5$, under maximum load=1.0.

### 3.2.4 Latency distribution

The latency distribution of the packets is another important aspect within the switch fabric performance. It gives an idea of the grade of packet disorder at the switch output, and thus it is interesting that the latency variance keeps small in order to ease the packet reassembly at the receiver. In addition if the latency is distributed close to the mean values, the signal degradation will be similar for every packet at the switch output. The latency distribution is plotted as a histogram in which it is represented the percentage of packets as a function of the number of hops they make. The latency distribution has been obtained for different switch fabric heights, different traffic loads offered and different ratios $\psi$ used. Figure 3.8 shows the latency distribution for different switch fabric heights under maximum load=1.0 and with a fixed ratio $\psi = 1/5$ ($A=5$, $A'=1$). As shown, the latency distribution gets broader when the switch fabric height increases. When the height increases the number of hops to reach the target output increases, since more levels and more routing decisions have to be done to reach the target output. The increase in the number of hops, together with the probability that two packets do not make the same number of hops, makes the latency variance bigger. Moreover switch fabrics with bigger sizes have more traffic congestion that enhances the number of packet deflections, which contributes to increase the number of hops, enlarging the latency distribution. However under maximum load, for a ratio $\psi = 1/5$, the latency remains narrowly distributed even for big switch fabric sizes. For example, for $H=2048$, the latency variance is around 15 hops for percentages of packets higher than 1%.
Figure 3.8. Latency distribution under maximum load=1.0 for different switch fabric heights with 5 nodes per height $A=5$ and one of them used as input ports $A'=1$.

Figure 3.9 presents the latency distribution for different traffic loads offered for a switch fabric with $H=64$ and a ratio $\psi=1/5$ ($A=5$, $A'=1$). It shows how the latency distribution gets wider when the load increases. Since the load increases more traffic congestion occurs, thus the larger number of deflection hops contributes to increase the number of hops, enlarging also the latency distribution. It is also shown that this latency distribution enlargement is more significant between load 0.5 and load 1, when the congestion is more notably.

Figure 3.9. Latency distribution for different traffic loads offered for a switch fabric with $H=64$, 5 nodes per height $A=5$, and one of them used as input ports $A'=1$. 
Figure 3.10 shows the latency distribution for different ratios $\psi$ used. It is shown that the latency distribution becomes wider when more nodes per height are used as input ports. Since more traffic congestion is within the switch fabric, the latency distribution gets wider. The latency distribution has been obtained for a switch configuration with $H=256$ under maximum load. It is shown a very remarkable enlargement in the latency distribution between the case of using one node as input port $A'=1$ and using two nodes as input ports $A'=2$. However the difference from $A'=2$ and $A'=5$ is much smaller. Therefore, it is shown a remarkable better performance when the ratio $\psi$ is 1/5 in contrast with larger ratios.

![Figure 3.10. Latency distribution for different ratios $\psi$, under maximum load=1.0, for a switch fabric with $H=256$ and $A=5$.](image)

### 3.3 Summary

We have studied the Data Vortex switch performance under uniform Bernoulli traffic. The results have shown that the switch performance is a function of the traffic offered, the switch fabric dimensions and the number of input ports. The performance has been characterized by the throughput, latency and latency distribution. We have studied the throughput by means of the parameters injection rate and exit rate. With these two parameters it has been analyzed the contention at the input and output of the switch. The introduced parameter exit rate has shown the grade of output contention, characterizing the traffic at the switch output. The results obtained have shown that the contention at the switch output must be resolved in order to sustain a high throughput. And the ratio $\psi$, defined as the number of nodes per height used as input ports over the total number of nodes per height, must be kept equal or less than 1/5 to sustain high throughput (injection rate $>97\%$, even for large scale switch fabrics with $2k$ ports). Latency results have also shown that for a ratio $\psi=1/5$ Data Vortex switch presents smaller latency (in terms of the
average number of hops) and good scalability. And also for $\psi = 1/5$ a narrow latency distribution with good scalability has been obtained. The overall performance results under uniform traffic have shown a remarkable improvement and good scalability when the ratio $\psi$ is $1/5$ in contrast when higher ratios are used. Moreover, performance results under bursty traffic have shown that a ratio $\psi = 1/5$ is at least necessary to have small performance degradation and maintain the good scalability [7]. However when the ratio $\psi$ is $1/5$, the number of hardware resources increases remarkable especially for large scale switch fabrics. Thus in order to have a feasible realization of the Data Vortex switch fabric, the switching nodes have to be implemented in a very simple, economical and integrable way.
Chapter 4

A Data Vortex Packet Switch Architecture with Improved Latency Performance

The main characteristic of the Data Vortex switch is the use of a multi-hop and self-routing mechanism in combination with an inherent deflection technique for contention resolution. The packets are routed along the switch structure hopping between the Data Vortex nodes via connection lines in a synchronous slot operation. No internal buffering is implemented in the Data Vortex nodes, although the whole structure can be considered as a delay-line buffer in itself. When two packets compete for the same node input of the structure, one of them is deflected and has to make additional hops along the structure thus avoiding contention. This packet multi-hop routing, which includes packet deflection in itself, brings about a latency determined by: the number of hops that a packet has to make along the structure, the time a hop takes and the switching time at the nodes. Since switching operations are very simple, the latency added by the node processing can be made insignificant with respect to the latency added by the connection lines. Thus the length of the connection lines mainly determines the total latency of the Data Vortex packet switch. In addition, the fixed length connections within the generic Data Vortex introduce signal degradation and make the size of the switch fabric large due to the bulkiness and amount of optical fiber required. In this chapter, we present a new Data Vortex operation mode that allows the use of shorter connection lines and consequently results in an improved latency performance and also results in a decrease in the grade of packet disorder at the switch output. Furthermore, the new operation mode results in a less bulky implementation, making it attractive the compact realization of WDM/TDM optical packet switches for telecom and supercomputing applications. Corresponding simulation results confirm the improved latency performance and the improvement of packet order at the switch output.

Furthermore, additionally to this new mode of operation, we propose a simple output contention resolution mechanism for Data Vortex switch fabrics that have more than one output node with the same address and connected to the same fiber, as the switch fabric studied in [3]. The mechanism we propose works within the Data Vortex structure using no additional hardware for buffering or control from what the general Data Vortex switch fabric already makes use of. This mechanism proposed guarantees a
packet loss rate equal to zero and will permit to simplify or even make unnecessary any buffering at the outside of the switch. Corresponding simulation results show the performance when this output contention resolution mechanism is implemented.

4.1 Generic Data Vortex

A generic Data Vortex packet switch architecture processes fixed-length packets in a synchronous, time-slotted, and parallel manner. Packets enter the switch by the nodes of the outermost cylinder and they are routed following a binary tree fashion towards the nodes of the innermost cylinder via which they exit. Not all the nodes on the outermost cylinder have to be used as input ports, and the use of only a fraction of them has been considered for better performance in the chapters before. For proper synchronous operation, the connections to the same level are all of the same length as are all the connections to the inner level. In a given clock cycle the packets can hop either to a node of the next inner cylinder, moving forward within the structure, or to another node placed in the same cylinder, in order to get properly routed or in order to resolve contention. In a generic Data Vortex the packet length \( u \) determines the minimum length of the connection lines and the minimum duration of the clock cycle (we define the packet time slot \( u \) to include the guard time necessary for the node processing time). In this way, a packet inside the switch fabric is switched only by a single node of the structure in a given clock cycle since it does not reach the next node until it has been switched entirely by the current node. Thus a generic Data Vortex switch adds a minimum latency per hop equivalent to a packet time slot \( u \), as the minimum of the clock cycle has to be \( u \).

4.2 Improved Data Vortex Configuration

Alternatively, we propose a Data Vortex configuration in which connection lines have a length shorter than the equivalent duration \( u \). Thus the clock cycle becomes a fraction of the packet time slot, with the consequence that a packet is processed within the switch fabric by several nodes at the same time. In the configuration we propose, the length of each connection line is decreased by a factor \( K \), given by the number of nodes used as input ports and the number of nodes per height, as explained below. As a result the latency decreases by this same factor \( K \) and the connection lines add a latency of \( u/K \) instead of \( u \). The improved Data Vortex architecture is a generic structure with the following configuration:

1. The distance between two consecutive input ports of the switch fabric that are on the same height must be equivalent to the packet time slot duration. This distance can be expressed in terms of the number of angles that two consecutive input ports are separated. In the case that only one node per height is used as input port the distance is the total number of angles per height. As a result the connection lines have a length that adds a latency equivalent to the packet slot of time divided by the number of angles \( K \) of distance. It must be noted that the minimum length of the connection lines is limited by the node processing time in order to configure the switch state at each node.
2. The input ports on the different heights must be aligned at the same angle positions.

These requirements enable the packets inside the switch fabric always to travel aligned in the angle dimension. It means that two packets arriving at two nodes which are at the same angle position arrive in the same clock cycle. This angle alignment is a condition required for correct operation. When a packet arrives at a node \( X \), it is routed and a control message is sent to a node \( Y \) at the same angle position on the outer level. Because of the angle alignment the packets that reach nodes \( X \) and \( Y \) arrive in the same clock cycle with a gap of time equivalent to the time it takes the control message to reach \( Y \). And since the packets have the same size, the control message that \( Y \) receives remains the same until it completes switching the packet. Thus the switch state of each node, which depends on the control message received, stays in the same state for the entire packet duration, avoiding packet splitting. The proposed configuration results in a remarkable decrease in connection line lengths and latency, while at the same time guaranteeing that the angle alignment condition is met for any switch fabric size under any traffic type and any offered traffic load.

4.2.1 Simulation Results

4.2.1.1 Injection Rate, Latency and Latency distribution

To demonstrate the improved latency operation we present simulation results for the case of a switch fabric with five nodes per height \( A=5 \), one of them used as input port \( A'=1 \), placed on every height at the same angle, and connections that add a latency equal to 1/5 of the packet time slot, thus decreasing the latency added by a factor of five. Figures 4.1, 4.2 a) and b) show the performance of this configuration for different switch fabric heights under uniform traffic in terms of the injection rate, mean number of hops to reach the output and latency distribution respectively.

![Figure 4.1. Injection rate vs. uniform offered traffic load for different switch fabrics with different heights (H), functioning according to the new operation mode proposed.](image-url)
32 A Data Vortex Switch for Optical Packet Switching

Figure 4.2. Performance under uniform traffic for different switch fabric heights (H). (a) Mean number of hops vs offered load. (b) Latency distribution under maximum load=1.0.

The performance curves of the injection rate, mean number of hops and latency distribution for the modified configuration coincide with the generic ones presented in the chapters before. The reason is that the switch fabric routing operations remain the same, but with the added advantage that now the latency and fiber length have decreased by a factor of five.

4.2.1.2 Packet Disorder at the Switch Output

It has been shown that the curves of the latency distribution for the new mode coincide with the generic ones. However the new operation mode results in a decrease in the number of packets disordered at the output of the switch, for the reason we explain below. For the new operation mode, each hop adds a latency equal to a fraction 1/K of the packet time slot. Thus, only if a packet #2 makes K or less than K hops less than a packet #1 that has entered the switch immediately before, packet #2 will reach its target output port at the same time or before than the packet #1, respectively. If packets #1 and #2 belong to the same communication, i.e. with the same source and destination addresses, the fact exposed results in one packet disordered at the switch output. Figures 4.3 a) and b) describe the situations of disorder that can occur for the new operation mode, when the Data Vortex switch fabric has more than one output node with the same address, as the switch studied in [3] (in which all the innermost nodes on the same height have the same address). Fig 4.3 a) presents a situation in which packet #1 has made K hops more than packet #2, and thus both arrive at the output at the same time. This kind of situation can only occur for a switch that uses more than one node per height as input port. For the new operation mode, if the switch fabric had only one node per height used as input port, two packets could never reach two output nodes on the same height at the same clock cycle. Fig 4.3 b) presents a situation in which packet #1 makes K+2 hops more than packet #2. For this case the switch fabric could have one or more nodes per height being used as input ports. Thus with the new operation mode, one packet #1 is disordered at the output if it makes at least K hops more than packet #2. Whereas for the
generic Data Vortex, since packet slots are one complete clock cycle long and one hop adds a latency equal to one clock cycle, one additional hop made by packet #1 is sufficient to disorder it. For this reason, the number of packet disordered at the output for the new operation mode is smaller than for the generic Data Vortex, although the curves of latency distribution coincide.

In order to estimate the percentage of packets that exit the switch disordered, we study the probability that the two situations described in Figures 4.3 a) and b) occur. We calculate the probability $P_{i\text{ disorder}}$ that a packet #1, which makes $i$ hops, exits disordered as the probability that packet #2, which enters the switch by the same port and with the same destination address, makes $K$ or less than $K$ hops less than packet #1. This probability under uniform Bernoulli traffic is given by:

$$P_{i\text{ disorder}} = \frac{1}{H} (P_i \cdot \sum_{j=1}^{i} P_j)$$  \hspace{1cm} (1)$$

$H$ is the number of the different switch output addresses, which in our case coincides with the number of heights. Since the distribution of destination addresses is uniform, the probability that packet #2 has the same address than #1 is given by $1/H$. Whereas $P_i$ is the probability that one packet makes $i$ hops. This probability $P_i$ is given by the curves of latency distribution. Thus the total probability of disorder is given by:

$$P_{T\text{ disorder}} = \sum_{i=1}^{\infty} P_{i\text{ disorder}}$$  \hspace{1cm} (2)$$

This value of $P_{T\text{ disorder}}$ multiplied by 100 will give the percentage of packets that exit disordered. Figure 4.4 shows a comparison of the disorder at the output between the new operation mode and the generic one. The curves represent the percentages of disordered packets versus the different switch fabric heights for the two modes of operation for a switch with five nodes per height $A=5$, and one of them used as input port $A'=1$. As shown, the new operation mode decreases remarkably the grade of packet disorder at the output. The most remarkable difference occurs when the switch height is
\(H=4\) since the percentage of disordered packets for the new operation mode becomes zero.

![Percentage of disordered packets for different switch heights](image)

Figure 4.4. Percentage of disordered packets for the different switch heights, for the generic and new mode proposed.

### 4.3 Output Contention Resolution Method

One characteristic of the Data Vortex switch fabric is the contention problem at the switch output when two or more output nodes have the same address and are connected to the same fiber. Figures 4.3 a) and b) are examples of this output contention, which would result in packet loss at the switch output if no contention resolution method is implemented. It has been shown in sections before that when no output contention resolution, for an acceptable packet loss rate the switch throughput becomes completely penalized. Therefore for this case, an output contention resolution mechanism must be implemented. The use of buffers located at the output is proposed in [3] for controlling the data flow out of the switch. However this proposal results in the use of additional hardware in terms of delay lines and logical controls. Moreover the number of hardware resources needed to implement this buffering will depend on the traffic and will increase remarkably in order to make the packet loss rate near zero. Alternatively, together with the new mode of operation introduced, we propose a simple output contention resolution mechanism within the Data Vortex structure in which no additional hardware for buffering or control is used apart from the one that the general Data Vortex switch fabric already makes use of. It guarantees that no packet that entered the switch get lost, achieving packet loss rate equal to zero under any kind of traffic with any offered traffic load, with the same number of hardware resources used. However the implementation of this mechanism will result in a decrease in the injection rate performance. Thus, together with the principle of operation, simulation results are presented.
4.3.1 Principle of Operation

The mechanism we propose is based on the use of deflection inside the switch fabric, following the same operation principle as the general Data Vortex. The new operation mode allows using this same principle without the need to complicate the simple generic Data Vortex control.

We study the case in which all the output nodes of the switch on the same height have the same address, as in [3], and all are connected to the same fiber, since it is considered that every output node with the same address is connected to the same fiber. The output contention resolution mechanism we propose imposes one requirement in terms of the number of nodes used as input ports and number of fibers connected to the output nodes. For the case of our study, the mechanism requires that only one node per height used as input port, that is \( A' = 1 \). The new operation mode together with this requirement assures that two packets will never reach two output ports with the same address at the same clock cycle. Thus contention never occurs because two packets want to start exiting the structure by the same fiber at the same time. This characteristic is the one that enable to implement a simple mechanism to avoid contention at the output without complicating the Data Vortex simple control. Now the contention only occurs when a packet starts to exit the structure and then in the next cycles another packet is routed to the same output address before the current packet has entirely exited. Thus the output contention resolution mechanism avoids that in the subsequent cycles when the packet is still exiting, another packet reaches an output node on the same height. The mechanism consist on generating and delivering the standard Data Vortex control messages from the innermost ring by which a packet is exiting to the outer level in order to prevent from another packet to reach the same ‘output ring’ until the current packet has completely exit. After a packet starts exiting, control messages are generated and sent to the outer level as if the packet circulates along the output ring until it reaches an output node at an angle before the actual output node.

We present an example of how the output contention resolution mechanism functions for a configuration with 3 nodes per height \( \Lambda = 3 \), every output node on the same height with the same address, one fiber connected to every output node on the same height, and one input node per height used as input port \( A' = 1 \):

1. At a given clock cycle \( T=0 \) a packet \( P \) reaches an output node \((h,0)\) on height \( h \) and at angle \( \theta = 0 \). This output node \((h,0)\) starts exiting the packet by the output fiber. However it sends a ‘busy’ control message to the corresponding node on the outer level, as if it circulates the packet \( P \) to the same level, in order to prevent that at the next clock cycle \( T=1 \) the start of a new packet \( B \) reaches output node \((h,1)\), at angle \( \theta = 1 \) on the same height \( h \).

2. At clock cycle \( T=1 \), since the mechanism functions as if the packet is circulating along the same level, nodes \((h,0)\) and \((h,1)\) send ‘busy’ control messages to the outer level.
3. At T=2 the supposed circulated packet would reach the output node (h,2) at angle θ = 2, an angle before the actual output node. The mechanism works as if packet A would start exiting the structure by this output node. Thus at T=2 nodes (h,0) and (h,1) continue sending 'busy' control messages, because the packet is still circulating by them, but node (h,2) does not send a 'busy' control message because the packet is supposedly going to the inner level, that is exiting the switch. Thus a packet B on the outer level could reach the output node θ = 0 at the next clock cycle T=3, when the current packet has entirely exited, and the same mechanism would start.

4. At T=4 the supposed circulating packet would have finished to be routed by node (h,0), and so the only node that sends the 'busy control message' is node (h,1).

5. Finally at T=5 the supposed circulating packet would have finished to be routed by node (h,1) and no 'busy' control messages are sent to the outer level.

As shown, at the start of cycle T=3 the packet has actually exited entirely and another packet can reach an output node with the same address. The control messages generated and delivered in the next clock cycles are in order to maintain the angle alignment, thus avoiding any packet splitting.

### 4.3.2 Simulation Results

The performance under uniform and random traffic is analyzed when the output contention resolution mechanism is implemented. The performance is studied in terms of the throughput, latency, latency distribution, and grade of packet disorder at the output. We present simulation results for the case of a switch fabric with five nodes per height A=5, one of them used as input port A'=1, placed on every height at the same angle, and connections that add a latency equal to 1/5 of the packet time slot.

#### 4.3.2.1 Throughput

Now with the output contention resolution mechanism used, the exit rate becomes 100%, since no packet is lost at the output. Thus, since no packets are lost at the output or within the switch, the throughput is given by the injection rate. Figure 4.5 shows the injection rate for different switch fabric heights versus the traffic load offered. It is shown a slightly better performance for bigger switch fabric sizes when the traffic load offered remains smaller than 0.7, although as seen before the traffic congestion inside the switch fabric becomes bigger when the switch fabric size increases. The reason is that now the traffic congestion caused by the output contention resolution method emerges at the level before to the innermost level and then it is transmitted to the outer levels. This traffic congestion gets slightly and gradually smoother to the outermost level through the different levels, due to the Data Vortex connection pattern. Thus since switch fabrics with
larger number of heights have more number of levels, the traffic congestion at the outermost level is slightly smaller. On the other hand when the load becomes bigger the increase in the congestion caused by the increase in the switch size becomes more remarkable. And the increase in the congestion becomes a more important factor compared with the one exposed before. Thus when load becomes bigger than 0.6, the performance is better for smaller switch fabric sizes.

Respecting to the throughput that the switch fabric can support Fig. 4.5 shows that the maximum throughput is around 0.6. In order to increase the throughput beyond 0.6 some sort of combination of the kind of mechanism proposed with some buffering at the input or at the output must be implemented. The implementation of an output contention resolution mechanism within the switch fabric as has been proposed, will simplify the extra buffering needed at the outside of the switch fabric, since it achieves an exhaustive use of the delay lines resources of the Data Vortex architecture.

4.3.2.2 Latency, Latency Distribution and Packet Disorder

We analyze the latency in terms of the mean number of hops that packets have to make in order to reach the target output. Fig. 4.6 a) shows the mean number of hops versus the traffic load offered for different switch fabric heights. As expected since the number of deflection hops increases the mean number of hops increases. This increase becomes more remarkable when the traffic load reaches 0.7, since congestion becomes larger. On the other hand the new operation mode makes decrease the latency per hop, so the overall latency performance penalty is not so important. Fig. 4.6 b) shows the latency distribution in terms of the percentages of packets that make certain number of hops for different switch sizes under maximum load. It is noticeable that now the curve of the latency distribution becomes wider than when no output contention resolution method is
deployed since more deflection hops occur (see Fig. 4.2 b). However with the new operation mode and additionally with the output contention mechanism implemented these curves do not reflect actually the grade of packet disorder at the output. Now a packet #1 would have to make at least 2·K hops more than packet #2 to exit disordered (see Fig. 4.7). Now the minimum number of additional hops is 2·K instead of K, because now the output contention resolution method do not allow the situations shown in Figs. 4.3 a) and b).

Figure 4.6. Performance under uniform traffic for different switch fabric heights (H). (a) Mean number of hops vs offered load. (b) Latency distribution under maximum load=1.0.

Figure 4.7. Packet disorder at the output. Packet #1 made 2·K hops more than packet #2.

Figure 4.8 shows the actual disorder at the output for the new mode of operation with output contention resolution method implemented, in comparison with the new mode of operation without contention resolution method and with the generic mode, for the different switch fabric heights. As shown, the disorder has remarkably increased with respect to the new mode without output contention resolution method (OCRM) but is still much lower than for the generic mode.
4.4 Summary

We have presented a new operation mode for the Data Vortex packet switch that allows the use of shorter physical connection lines and has improved latency performance and improved packet ordering at the output. Moreover, the proposed switch fabric configuration operates as the generic Data Vortex switch fabric, i.e. without any kind of internal contention under any kind of traffic with any offered traffic load. Computer simulations have shown the same performance as the conventional operation mode under uniform traffic. Furthermore, the proposed switch structure, due to its shorter connection line requirement, is attractive for a compact realization in WDM/TDM optical packet switches for telecom and supercomputing applications. Together with the new operation mode, we have proposed a mechanism to avoid contention at the output within the Data Vortex structure in which no additional hardware for buffering or control is used apart from the generic one. Computer simulation results have shown its performance in terms of the throughput, latency and packet disorder at the output. To achieve a better performance some sort of additional buffering at the input or output of the switch must be implemented as well.
Chapter 5

Implementation Proposal for Data Vortex Packet Switch

Data Vortex performance imposes several requirements on the switching nodes implementation in terms of simplicity, integrability, cascadability and switching time. We have concluded in chapter 3 that the ratio \( \psi \) (the number of nodes used as input ports over the total number of nodes per height) cannot be larger than 1/5 in order to have a good Data Vortex performance. As a consequence the number of hardware resources increase. Thus, in order to have a feasible realization of the Data Vortex switch fabric, the switching nodes have to be implemented in a very simple, economical and integrable way. In addition, the Data Vortex multi-hop switching mechanism requires small switching time, good extinction ratio and low insertion loss at the Data Vortex switching nodes. These are requirements needed to have an overall low latency and keep packet and control signals degradation under control in order to perform the routing mechanism properly. In this chapter, we propose a Data Vortex node implementation based on electroabsorption modulator (EAM) technology that meet these requirements. This technology is presented as a promising alternative to the use of linear optical amplifiers (LOA), previously proposed in [8].

5.1 Packet Slot Format

We have presented in chapter 2 how Data Vortex operates in a parallel, time-slotted and wavelength-insensitive way. This parallel and time-slotted operation forces the packets to be aligned and organized in time slots of equal duration. Besides, since the packet switching is done insensitively with respect to the wavelength, DWDM technology can be used to increase the network capacity. These Data Vortex characteristics make appropriate the use of DWDM packets, to satisfy the time-slotted operation requirements and at the same time increase the network bandwidth. Figure 5.1 shows a DWDM packet in which one packet occupies a time slot composed of multiple wavelength channels. The packet is composed of a framing wavelength, a set of wavelengths representing the addressing bits, and a set of wavelengths carrying the
packet payload. The packet payload is WDM/TDM multiplexed, where each wavelength channel is modulated at a certain data rate, like for example 40 Gbit/s. The payload can be fixed at very high data rates, since the optical routing is transparent with respect to payload. The throughput is increased by the number $w$ of wavelength channels, thus it is interesting to have as many as possible wavelength channels available. Regarding the packet header, the addressing bits are also codified in the wavelength domain. In the DWDM packet proposed in [3], each wavelength channel carries a single addressing bit during the entire time slot period. Thus, the addressing wavelength channel has to stages, ‘on’ or ‘off’, to represent the corresponding addressing bit (see Fig. 5.1 a)). This header wavelength codification simplifies the switching nodes implementation, since they can easily read the corresponding header bit by just using a passive wavelength filter. The packet format proposed has a framing bit whose value is always ‘1’ and which indicates the packet presence. This bit is also codified in a single wavelength channel that carries a continuous wave during the entire packet duration.

![Figure 5.1. (a). DWDM packet format. (b) DWDM/SCM packet format.](image)

This packet format explained above simplifies the routing and switching node implementation but, on the other hand, results in a waste of the wavelength spectrum. Since single wavelength channels are used to transmit one addressing or framing bit, the network capacity decreases remarkably. We propose an alternative DWDM/SCM packet format in which all the addressing bits and the framing bit are carried in a single wavelength channel (see Fig. 5.1 b)) The addressing and framing bits are codified by using subcarrier multiplexed on the same wavelength. Thus, the DWDM/SCM packet presented results in a more efficient use of the wavelength spectrum, since more wavelengths are available for carrying payload data. Below we present the node implementation schemes both when DWDM and DWDM/SCM packets are used.

### 5.2 Node Implementation Scheme

Below we present an implementation scheme of the Data Vortex switch fabric switching nodes, based on the switching operation presented in chapter 2. In a given time slot, each node receives one packet, either from a node on the outer level or on the same level, and one control signal. The switching node reads the corresponding packet-addressing bit and the control bit signal, and depending on their values, the node routes the packet to the same level or to the inner level. The switching decision is based on the
control bit value and the comparison between the addressing bit and the height bit of the node. It is defined the height bit of a node on level \( r \) and height \( z \), as the bit in position \( r \) of the binary height \( z \). For example, the height bit of a node on level \( r = 2 \) and height \( z = 11 \) ('1011' binary coded) is '0'. If the addressing and height bit coincide and the control signal indicates that there is no blocking inside the packet is sent to the inner level. Otherwise the packet is sent to the same level. The switching node also sends a control bit signal to the outer level. It sends a control bit with value '1', meaning blocking inside, if it routes the packet to the same level, whereas it sends a bit equal to '0' if it routes the packet to the inner level. In Table 5.1 is shown the routing decision for nodes with height bit equal to '1'.

<table>
<thead>
<tr>
<th>CONTROL</th>
<th>1 bit address ( (\lambda_z) )</th>
<th>OUTPUT PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>SAME LEVEL</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>SAME LEVEL</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>INNER LEVEL</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>SAME LEVEL</td>
</tr>
</tbody>
</table>

Table 5.1. Logic routing table for nodes with height bit equal to '1'.

Figure 5.2 shows the scheme proposed for nodes with height bit '1' for DWDM packets used. As shown, the packet is switched to the inner or to the same level transparently by a 1×2 optical switch as a result of the routing decision obtained from the logical operation between the addressing and the control bit. The node receives the WDM packet and the optical control bit signal at the same wavelength \( \lambda_r \) as the framing bit. Firstly, a small power of the packet is tapped, and a passive wavelength filter is used to filter the wavelength \( \lambda_1 \) that carries the corresponding addressing bit. These addressing and control bits are the input of the logical gate, whose output drives the optical switch. When the control bit is '0' and the addressing bit is '1', the result of the logic operation is '1' and the optical switch is configured to switch the packet signal to the inner level. Otherwise the result of the logic operation is '0' and the packet is switched to the same level.

![Figure 5.2. Node implementation scheme for a node with height bit equal to '1'](image)

As shown in the figure, the packet is delayed by means of a fiber delay line until the optical switch state is configured. Since the addressing and control bits are coded during the entire packet duration, the optical switch state remains invariable during the complete duration of the packet. Also, the node has to send the optical control bit signal
to the outer level. In the proposed scheme, this optical signal is generated by tapping part of the output power of the logic gate, and inverting its value. The control signal generation could be also done by tapping part of the packet power of the signal when it is switched to the same level. However, the power tapping of the packet signal is not recommended because of the several stages packets have to go through.

Table 5.2 shows the routing decision for a node with height bit ‘0’, and Fig. 5.3 shows its scheme. The only difference lies in the logic gate needed, whose result is ‘1’ when the control and the addressing bits are both equal to ‘0’.

<table>
<thead>
<tr>
<th>CONTROL</th>
<th>1 bit address (I, b)</th>
<th>OUTPUT PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>SAME LEVEL</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>SAME LEVEL</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>SAME LEVEL</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>INNER LEVEL</td>
</tr>
</tbody>
</table>

Table 5.2. Logic routing table for nodes with height bit equal to ‘0’

When SCM/WDM packets are used, the implementation scheme remains the same, except for the need of very sharp optical filtering demultiplexing in order to filter the subcarrier that carries the corresponding addressing bit. This filtering could be done by means of some state-of-the-art sharp fiber-grating filter, or alternatively it can be done in the electronic domain.

5.3 Node Implementation based on Electroabsorption Modulator Technology

We propose the use of electroabsorption modulators (EAM) as an efficient and promising technology to implement the Data Vortex switching nodes. This technology has been considered based on its promising compactness, stability, integrability, and ultra-fast operation in OTDM systems. The use of the EAM technology will result in fast switching times at the Data Vortex nodes and we believe it will also enable a simple and easily integrable node implementation.
5.3.1 EAM Implementation Proposal for Data Vortex Switching Nodes

EAMs are devices that have the property to absorb or let pass the optical signal fed into its input, depending on the reverse bias voltage applied to them. Because of its way of operation they are usually used for modulating and demultiplexing purposes. In our case, we propose the utilization of two EAMs as gates and operating as one 1×2 optical switch. In the configuration proposed, the EAMs are arranged in such a way that while one of them lets pass the optical signal the other absorbs it. As shown in figure 5.4, one of the EAMs is controlled by the inverted signal that controls the other one. This control signal determines the reverse voltage applied to the EAMs. As shown in the figure, it is the result of the logical operation $A\overline{B}$ or $\overline{A}B$ between the addressing and the control bits, respectively for a node with height bit ‘1’ and for a node with height bit equal to ‘0’. In one of our proposals, the logic operations are done in the optical domain using semiconductor optical amplifier (SOA) gates. To carry out the operation $A\overline{B}$, we propose the use of one single SOA using its cross-gain modulation effect. Whereas the logic operation $\overline{A}B$, needed in nodes with height bit ‘0’, can be carried out by an all-optical ‘NOR’ gate using two SOAs [9]. After the SOA gate, a optical filter at $\lambda_1$ is placed to filter the result of the logic operation and part of this signal power is tapped and sent as the control signal to the outer level. After that, since the EAM driving signal is electrical, an optical-to-electrical conversion is used to obtain the signal that will drive the two EAMs. Two implementations are proposed, namely one all-optical and the other electro-optical. Figure 5.4 shows our first all-optical implementation proposal when DWDM packets are used.

![Figure 5.4. Node implementation proposal using EAMs technology for DWDM packets and SCM/DWDM packets if optical subcarrier filtering is available](image)

The implementation, when SCM/DWDM packets are used, will be generally the same if a sharp fiber-grating filter could enable the subcarrier optical filtering. In the case that the subcarrier has to be filtered in the electronic domain, the wavelength channel that carries all the addressing bits is converted to the electronic domain in order to filter the corresponding subcarrier. The control bit would also be converted to electrical and thus the two-bit logic operation would be done in the electronic domain. In this case the control signal sent to the outer level will not be part of other signal power. On the contrary, it is generated by means of a laser controlled by the electrical signal resulted
from the two-bit logic operation. Figure 5.5 shows the node implementation when the subcarrier are filtered in the electronic domain.

The key for the good operation of this implementation lies in the dynamic response of the EAM to multiple wavelength channels, modulated at a certain bit rate, applied to its input. In [10] it has been demonstrated the simultaneous RZ modulation of four 40 Gbit/s NRZ-modulated wavelength channels (1547 nm -1557 nm, 2.5 nm spaced between each other) using EAM technology. This result confirms that high bit-rate wavelength channels, as the ones of the proposed DWDM packets, can be switched simultaneously with the EAM technology, provided that the total power does not saturate the EAM's absorption. It has been demonstrated that the EAM static response is roughly flat for a sufficiently large wavelength range when high reverse voltages are applied. A similar response is expected in the dynamic domain, thus we suggest in our configuration to apply a high reverse voltage, between -6V and -10 V, to drive the EAMs, in order to have a similar degradation in all the packet wavelength channels [11].

Furthermore, an ultra-fast monolithic photodiode-EAM optical gate working at 200 Gb/s has been demonstrated in [12,14] and even at 320 Gbit/s [14], in which the photodiode acts as the optical to electrical converter. For the case of our node implementation in which no electronic filter is needed, two of these PD-EAM optical gates could be used to drive each EAM, integrating in one of them the corresponding inverter. Below we present the characteristics of this recently introduced ultra-fast optical gate.

5.3.2 Photodiode-EAM Optical Gate Characteristics

In [12,14] has been presented a novel ultrafast optical gate which monolithically integrates a uni-traveling-carrier photodiode and a traveling-wave electroabsorption modulator. In this configuration the photodiode directly drives the EAM without the need of electrical amplifiers and electrical connections whose bandwidth would limit the speed of the electro-optical switch. In this configuration the photodiode anode is connected to
the signal line of the TW-EAM, and thus an optical input generates a positive signal to the EAM, constructing the transmission gate that let pass the optical input signal. It has been demonstrated the demultiplexing of optical RZ signals at a 200 Gbit/s data rate with an extinction ratio of 18.6 dB. A gate opening time of 2.3 ps has been demonstrated and the chip size obtained is of 1 mm x 0.4 mm. The extinction ratio of 18.6 dB was obtained for a configuration in which the optical input signal fed into the EAM was fixed at 0.1 pJ/pulse, the EAM bias voltage at -3.5V and the control pulse energy fixed at 8.6 pJ/pulse. It was demonstrated that for a lower energy of the control pulse of 3.9 pJ/pulse, the extinction ratio remains high with a value of 16.7 dB.

5.3.3 EAM Implementation Proposal characteristics

Below we present the main switching characteristics of the EAM based implementation proposal:

- **Switching time.** The switching time, together with other factors like wavelength dispersion, determines the guard time necessary between packets in order to have a good operation. This guard time must be equal or larger than the switching time in order that: (i) the switch state recovers from the state set by the packet before and (ii) the new switch state determined by the next packet can be established. This switching state remains during the entire packet duration.

  We estimate the total switching time considering the data rate at which the different devices of the implementation operate. We analyze the switching time when the logic operation (for the routing decision) is done both in the optical domain and in the electronic domain (that is, when is not possible to filter the subcarrier in the optical domain, as explained above).

  When the logic operation is done in the electronic domain, the wavelength that carries the different subcarriers and the control signal, have to be converted to the electronic domain by an optical to electrical converter (O/E). The operation of the O/E converter has been demonstrated at 200Gbit/s [12,13], thus it adds a switching time of 5 ps. Then, the logic operation, done in the electronic domain, can operate at 40 Gbit/s, adding a switching time of 25 ps. Finally, the EAM, demonstrated at 40 Gbit/s [10], adds a switching time of 25 ps. Thus, the total switching time is around 55 ps.

  Suppose the logic operation is done in the optical domain, by using the optical logic gate demonstrated at 20Gbps, adds a switching time of 50 ps. And the PD-EAM optical gate, demonstrated at 200 Gbit/s [12], adds a switching time of 5 ps. The total switching time would be also around 55 ps. New technologies may result in fastest all-optical switching times [15].

  Table 5.3 summarizes the total switching time for the different considered implementations. Both configurations have really small switching time, one of the
Data Vortex requirements in order to achieve a low latency. Besides, the necessary guard time between packets imposed by the switching time will be very small. Thus, the switching time will not be a factor that results in bandwidth decrease. On the other hand, it is expected that the guard time will be fixed by the dispersion factor.

<table>
<thead>
<tr>
<th>Logic operation in electronic domain</th>
<th>Logic operation in optical domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>O/E converter (200Gbps) ~ 5 ps</td>
<td>Optical Logic Gate (20 Gbps) ~ 50 ps</td>
</tr>
<tr>
<td>Electronic Logic Gate (40 Gbps) ~ 25 ps</td>
<td>PD-EAM (200 Gbps) ~ 5 ps</td>
</tr>
<tr>
<td>EAM (40 Gbps) ~ 25ps</td>
<td>Total Switching time ~ 55 ps</td>
</tr>
</tbody>
</table>

Table 5.3. Switching time summary, based on published results in literature.

- **Integrability.** We estimate a very good possibility for integrability of the EAM implementation proposed because of the small number of devices used and especially because the suitability for integration of the EAM technology. For example, the integration of SOAs and EAMs has been already published in literature [16].

- **Extinction ratio.** The extinction ratio is determined by the EAM extinction ratio, which typical value is around 20 dB, and which also depends on the reverse bias voltage applied. As explained before, the reverse bias voltage is applied when absorption is needed and must be high to obtain a roughly the same extinction ratio for the total number of wavelength channels that compose the DWDM packet.

- **Cascadability.** Since packets have to be routed through several switching nodes, signal amplification would be necessary at some of the switch stages to compensate for the insertion loss of the EAM. SOAs could be used for this purpose. Besides, equalization would be necessary in switch fabrics with a large number of stages in order to compensate for the power variations of the different wavelength channels.

### 5.4 Summary

We have proposed a Data Vortex node implementation using electroabsorption modulator technology. We have chosen the use of an electro-optical switch because of its simplicity, stability, integrability and wide bandwidth it provides. Experimental results reported in literature have confirmed the feasibility of gating operation of the EAM technology when multiple modulated wavelength channels are applied to its input. Besides, EAM technology has become an ultra-fast device in OTDM system. Its operation has been demonstrated at 200 Gbit/s and even at 320 Gbit/s (monolithically integrated with a photodiode), which could make the switching time of our implementation very small. Moreover, the EAM technology provides a very good extinction ratio, and roughly flat response for a large range of wavelengths when high reverse voltage is applied, thus enabling its cascadability. In addition, EAM has very
good intergrability, and integration with SOAs has been also published in literature. SOA technology could be used for amplification inside the switching nodes. All these facts meet the Data Vortex node requirements in terms of small switching time, simplicity, integrability and cascadability. Additionally, the implementation proposed allows the routing of packets in which all the addressing bits are carried in the same wavelength but at different subcarriers, resulting in a better use of the optical spectrum and a throughput increase.
Chapter 6

Photonic Slot Routing using a Data Vortex Packet Switch

The way in which Data Vortex Packet Switch works in a time-slotted WDM/TDM network scheme has been presented. Packets occupy fixed-length time slots multiplexed in DWDM. In this scheme, the packets are not carried in individual wavelength channels and they are switched in an insensitive way with respect to the wavelength. Thus, the routing is not wavelength dependent and the wavelength domain is not used for contention resolution either. DWDM technology is just used to increase the throughput and simplify the packet routing mechanism. These Data Vortex switching characteristics matches perfectly with the OTDM packet switching approach known as Photonic Slot Routing (PSR) proposed in [17]. In the PSR scheme, photonic slots (PS) of fixed length and composed of multiple wavelengths carry multiple packets and are routed as a single entity throughout the PSR network. Contrary to Individual Wavelength Switching, the different wavelengths are not switched individually, which results in several advantages in terms of hardware complexity, cost and scalability of the switching nodes. Although Individual Wavelength Switching can offer a better performance in forms of the throughput, PSR represents currently a cost-effective solution to deploy all optical Access and Metro networks that satisfies the increasing bandwidth demand. In this chapter, we present the Data Vortex switch fabric as an efficient proposal for photonic slot routing. In fact, Data Vortex operates in PSR networks, without the need of modification in its operation, and provides some advantages in respect of other switches proposed in [18-20].

6.1 PSR Concept [17-21]

Photonic Slot Routing is an optical packet switching (OPS) concept originated from the idea of making packet switching independently with respect to the wavelength. It has been introduced in [18] as a cost-effective and scalable solution to achieve all-optical packet switching in access and metro networks in contrast to the use of individual wavelength switching (IWS) which would be costly unaffordable outside the core
network domain like in access and metropolitan area networks. PSR has been also proposed for wide area networks in [19].

The aim of PSR scheme is to perform the packet switching without the need of sensitive wavelength devices and thus achieve a remarkable decrease in the number of hardware resources needed at the switching nodes of the WDM network. PSR uses a time-slotted scheme in which packets at different wavelengths form photonic slots, which are routed as a whole entity throughout the PSR network until reaching its destination. The time-slotted operation requires that fixed-length packets, at different wavelengths, are aligned parallelly in time forming the photonic slots. Since PSR nodes switch the photonic slots as individual units, independently of the wavelengths they carry, it is necessary that every packet arranged in the same slot shares the same PSR node destination.

6.1.1 PSR Advantages

In PSR, the optical packet switching is done insensitively with respect to the wavelength. Thus, PSR switching nodes can be realized by means of wavelength-insensitive optical technology, which is already proven. These wavelength-insensitive devices have low cost compared with the sensitive optical devices that IWS need. Besides, since the photonic slots are routed as single entities throughout the PSR network, the wavelength channels are switched all together and there is no need to use multiplexing, demultiplexing and dedicated control and hardware to switch each individual wavelength channel. Another advantageous feature lies in the fact that each photonic slot carries a single header that is enough to route along the PSR network all the packets included in it and which are on the different wavelength channels. This fact, considering the complexity of optical processing, is a great advantage since it is just necessary to process the single header of the photonic slot to route the total number of packets it carries. These characteristics result in a remarkable lower complexity, cost and switching time at the PSR switching node compared with an IWS one. The PSR switching node complexity, in terms of the number of hardware resources and control needed, is independent of the total number \( w \) of wavelengths and it is \( w \) times smaller than at an IWS one. Furthermore the capacity network can be upgraded just increasing the number of wavelengths within the pass-band of the switching nodes, without the need to increase the number of hardware resources. Finally, PSR has also the advantage that the switching crosstalk between contiguous wavelength channels is avoided, since the wavelengths are switched all together.

6.1.2 PSR requirements and constraints

PSR architecture imposes several important requirements and constraints because of its way of operation. One important issue in PSR networks, as in every time-slotted scheme, is the need of a strict synchronization. Photonic slots must arrive at the same time at the input ports of one switching node. In order to control this requirement, a
general synchronization system as the one proposed in [1] should be established at the switching nodes input. Furthermore, the synchronization issue includes the need of packet alignment in time to form the photonic slots.

Dispersion is another important constraint for the proper operation in PSR networks. It can result in photonic slot overlapping, thus sufficient time guards must be established between photonic slots, resulting in a decrease in the network bandwidth. The use of non-zero dispersion fibers to cover the distances and dispersion compensating sections located at the switching nodes is proposed in [18] to resolve this problem. Due to this dispersion problem makes PSR concept more suitable to implement in access or metropolitan networks than in wide area networks.

Finally one of the main problems of PSR networks is finding an efficient way to allocate the packets into photonic slots. Photonic slot creation is indeed a hot issue and must be based on an efficient protocol in order to achieve a good use of the WDM spectrum. Since packets in the same photonic slot must share the same PSR node destination, the challenge is to fill nearly the complete number of the photonic wavelength channels to exploit the maximum network capacity and satisfy the traffic demands. Thus, the complexity now appears at the PSR network periphery at which a control protocol has to be established to form the photonic slots to be routed throughout the PSR network. Regarding to the photonic slots composition, in [22] a technology proposal for multiplexing the packets into the slots has been presented.

### 6.2 Data Vortex Switch Fabric used in PSR networks

As presented in previous chapters, Data Vortex is a packet switch fabric that switches DWDM/TDM packets parallel and in a time slot operation way. Each packet is encoded in DWDM/TDM and occupies an entire fixed length time slot composed of multiple wavelength channels. Packets are switched insensitively with respect to the wavelength and the DWDM technology is used to increase the throughput. The total number of wavelength channels enhances the packet time slot capacity. The difference with respect to PSR lies in the fact that whereas Data Vortex switches a single packet included in each time slot, PSR switches several packets included in the same photonic slot. However, since both concepts consist in switching wavelength-insensitively WDM/TDM time slots as a function of its slot header, Data Vortex switching mechanism would switch photonic slots exactly in the same way as it switches packet time slots.

We propose the Data Vortex packet switch as an efficient switching node for PSR. Data Vortex routes the photonic slots in a transparent all-optical way, simplifying both the optical header processing and the logical operations needed. We propose the use of the Data Vortex switch as a switching node in a metropolitan area network scenario, interconnecting a large number of local networks and other PSR nodes.
6.2.1 Data Vortex as a PSR Switching Node in Metropolitan Area Networks

In [18,19] it has been proposed the PSR architecture in access, metropolitan and wide area networks with regular and mesh topologies. We propose the use of the Data Vortex switch with improved latency as a switching node within a PSR Metropolitan Area Network. Our proposal consists in the use of the Data Vortex switch as a PSR switching node for interconnecting multiple local area networks (LANs), other MAN PSR nodes and also nodes to access wide area networks. Figure 6.1 shows the setting scheme. As shown in the figure, the PSR network consists of End Nodes placed at the network periphery and Data Vortex Switching Nodes in the core of the PSR network.

![Diagram of Data Vortex switch as a switching node in a PSR Metropolitan Network.]

The End Nodes operate as the sources and destinations of the PSR network. They are in charge of the photonic slot generation, delivery to the PSR network and reception from the network. These End nodes will be normally gateways or bridges to local networks, thus the photonic slots will be created from the packets received from the local network. The photonic slots received from the PSR network will be split up into packets to deliver them individually to the local network. These local networks could be made all-optically, with each terminal of the network transmitting and receiving at a specific wavelength channel. Thus, the slot creation and split up would be based on DWDM multiplexing and demultiplexing. In addition, these End Nodes will perform TDM multiplexing to increase the data rate of the photonic slot. As an example, if there is a group of \( w \) terminals connected to the same End Node, transmitting packets at 10 Gbit/s,
each one at one of the $w$ wavelength channels of the photonic slot, the End Node could multiplex the packets using DWDM and TDM in order form photonic slots with $w$ channels transmitting at 40 Gbit/s each, i.e. achieving a $w \times 40$ Gbit/s transmission.

The Data Vortex switches with improved latency (described in chapter 4) act as switching nodes in the PSR network in charge of the photonic slot routing. Each Data Vortex switch receives photonic slots from the End Nodes and from other switching nodes of the PSR network. Data Vortex operation, as well as PSR, imposes the photonic slots to arrive synchronously to its input ports. This could be done by fixing the optical links length at a multiple of the photonic slot length to ease the synchronization, a synchronization stage has to be established at the switch fabric input. After the synchronization stage the photonic slots are switched transparently throughout the Data Vortex structure based on the slot header toward the corresponding output port destination. In order to simplify the Data Vortex routing and implementation, and at the same time achieve an efficient use of the wavelength spectrum, the slot header bits will be transmitted during the entire time slot at the same wavelength, each one at different subcarriers, as presented in chapter 5. Thus, the Data Vortex nodes will just use passive wavelength filtering to read the corresponding addressing bit.

The contention resolution is done by means of the distributed control mechanism that Data Vortex deploys. Since PSR architecture does not make use of wavelength conversion, the wavelength domain is not used as an additional resource to resolve contention. In [19,20] the contention is resolved by pre-arranging the photonic slot transmission in a non-blocking manner. Thus, the photonic slots are allocated in transmission frames in a repeated and cyclic fashion avoiding that two photonic slots contend for the same output port at the same time. However, this pre-arranged cyclic fixed TDM photonic allocation into frames results in a significant bandwidth waste. Therefore, in order to obtain a better throughput performance, a photonic routing without fixed allocation using switching nodes with some sort of buffering is needed. The Data Vortex switch, with improved latency and without output contention that was proposed in chapter 4, together with some small buffering at its input, will allow the slot contention without the need to schedule the TDM photonic slot transmissions. The Data Vortex three-dimensional structure, together with its distributed control mechanism, enables to perform the slot buffering and slot routing at the same time throughout Data Vortex structure.

Data Vortex, as mentioned before, has a self and distributed control without any central and external control. This fact makes Data Vortex really attractive to scale to a large switch fabric size with great number of input and output ports. An external and central control would be a very complex solution if the number of input and output ports becomes very large. Thus Data Vortex switch is a very appropriate solution to be in charge of interconnecting a large number of local networks and other PSR nodes. Figure 6.2 shows another proposed scenario in which a single PSR Data Vortex will be enough for the interconnection of a very large number of local networks. Furthermore, the use of the improved Data Vortex configuration introduced in chapter 4 would add low latency and would enable a compact realization of the PSR switching nodes.
6.3 Summary

We have considered the Data Vortex Packet Switch with improved latency as an efficient proposal for Photonic Slot Routing Networks. The Photonic Slot Routing concept has been studied, explaining its advantages and requirements for proper operation. It has been explained why the wavelength-insensitive routing makes PSR an all-optical cost-effective solution for access and metropolitan networks contrary to the use of individual wavelength switching. It has been shown how the Data Vortex switch can work perfectly as a PSR switching node without the need to change its operation way and at the same time maintaining all its attractive features. We have presented the metropolitan network scenario in which Data Vortex would be more efficient. It has been under consideration its distributed control and good scalability to propose it as a prominent solution to interconnect a large number of local networks and other PSR nodes. In addition, it has been identified that, apart from simplifying the optical processing and packet routing, the Data Vortex switch eliminates the need to pre-arrange the photonic slot transmission in a non-blocking manner, thus increasing the network bandwidth with respect to other proposals published [18,19].
Chapter 7

Conclusions

In this thesis an extensive study of a novel Data Vortex packet switch architecture has been performed. This study has been focused on the Data Vortex efficiency and possible application for optical packet switching. As a result of the depth analysis performed and the knowledge acquired, we have proposed a new Data Vortex architecture with notably improved latency performance and prospect compact realization in comparison with the generic Data Vortex. Besides, in this thesis we have studied the electroabsorption modulator technology as a promising proposal for the realization of a wavelength non-sensitive switching. Finally, we have proposed the use of the new and improved Data Vortex for the OTDM packet switching scheme known as Photonic Slot Routing.

Data Vortex Architecture Study

First, in order to study the efficiency and possible application of Data Vortex concept for optical packet switching, Data Vortex principle of operation and characteristics were studied in detail. An exhaustive and strict presentation of the architecture has been performed in chapter 2, mainly as a result of the study of the Data Vortex invention patent, apart from other literature published. Packets are switched following a binary tree multi-hop along the Data Vortex structure. This binary tree routing operation is similar to the one deployed in Banyan or Shuffle networks. However, the Data Vortex 3-D structure offers many advantages comparing to these two-dimensional architectures. Data Vortex topology and principle of operation enable to perform the packet routing and buffering throughout the same structure. The connections used for the multi-hop packet routing are also used as delay lines for contention resolution. An inherent deflection technique along the structure is deployed. The self-routing mechanism is accompanied by a simple, self and distributed control. Thus, the need of any central and external control is avoided. This characteristic makes Data Vortex very attractive to scale it to a large switch fabric, in which an external control would become very complex. In addition, the simple switching logic operations needed for the packet self-routing and contention resolution suggests the possibility to make an all-optical Data Vortex implementation. On the other hand, we have identified the principal requirements and disadvantages of the Data Vortex architecture. A strict
synchronous scheme is necessary for the Data Vortex time-slotted operation. Besides, the Data Vortex multi-dimensional structure calls for a very simple, integrable and economical implementation of the small units that form the switch in order to have a cost-effective Data Vortex implementation. We have also identified that, although contention is resolved within the structure, some contention occurs at the output if more than one output node have the same destination address. Also, the multi-hop operation together with the inherent deflection technique deployed will result in a latency penalty. As a way to solve it, we have proposed a new operation mode that remarkably improves the latency performance.

During this thesis, a computer simulator has been built, by which we have characterized the Data Vortex performance. The performance has been analyzed under Bernoulli traffic. Results have shown that the contention at the switch output must be resolved in order to sustain a high throughput. The overall performance results have shown also that an asymmetric I/O operation mode equal or smaller than 1/5 has to be adopted for a good performance in terms of the throughput, latency and latency distribution and for a good switch scalability.

**New Data Vortex Architecture with Improved Latency**

As a result of the depth study of the Data Vortex architecture, we have proposed a new Data Vortex architecture with improved latency performance. In the new operation mode allows a packet to be processed within the switch fabric by several nodes at the same time. Thus, the new operation mode proposed enables the use of remarkably shorter physical connection lines. Since Data Vortex deploys a multi-hop packet routing, this results in a remarkable improvement in the latency performance. Besides, the new architecture proposed generally keeps the packet order at the switch output. The proposed switch fabric configuration operates as the generic one, i.e. without any kind of internal contention under any kind of traffic with any offered traffic load. Computer simulations have shown the same performance as the conventional operation mode under uniform traffic, demonstrating that there is no performance penalty. Furthermore, the proposed switch structure, due to its shorter connection line requirement, is attractive for a compact realization in WDM/TDM optical packet switches for telecom and supercomputing applications. Together with the new operation mode, we have proposed a mechanism to avoid contention at the output within the Data Vortex structure itself. This mechanism proposed needs no additional hardware for buffering or control.

**Data Vortex Technology Implementation Proposal**

A survey of possible technologies for the Data Vortex implementation was carried out. As a result, the use of electroabsorption modulators (EAM) is proposed for the implementation of the switching nodes of the Data Vortex structure. Two implementations are proposed, one all-optical and the other electro-optical. The use of this ultra-fast EAM technology meets all of the Data Vortex node requirements in terms
of fast switching time, simplicity and integrability. Amplification and equalization stages must be implemented along Data Vortex structure in order to achieve the required cascadability. Besides, the implementation proposal switches packets having the header bits encoded in subcarriers, resulting in a throughput increase due to the better use of the optical spectrum.

**Photonic Slot Routing using Data Vortex Switches**

Finally, we have considered the Data Vortex Packet Switch with improved latency as an efficient proposal for Photonic Slot Routing (PSR). It has been shown how the Data Vortex switch works perfectly as a PSR switching node without the need of modifications. A PSR metropolitan area scenario in which Data Vortex interconnects a large number of local area networks and other PSR nodes has been proposed. In these scenarios Data Vortex has advantageous features such as no need to pre-arrange the photonic slot transmission in a non-blocking manner, thus increasing the network bandwidth with respect to other proposals published.
Chapter 8

Further Work

In this thesis we have studied the performance and possible application of Data Vortex concept in optical packet switched networks. In a Data Vortex switch, packet switching is performed insensitively with respect to the wavelength. Thus, we have identified the Photonic Slot Routing (PSR) as one good scenario in which Data Vortex architecture can operate efficiently. This PSR scheme has been presented as a cost-effective solution to deploy all-optical access and metro networks in contrast with individual wavelength switching (IWS). Regarding to this topic, we consider that a realistic comparison between IWS and wavelength-insensitive switching concepts is important. This comparison in terms of the performance, number of resources necessary and cost will give a good idea where to apply each concept in optical packet switched networks.

Regarding to the Data Vortex implementation as an optical packet switch for optical networks, some issues have to be resolved. One is the way to perform the packet routing along the entire optical network. Multi Protocol Label Switching (MPLS) is the strategy that should be used for this task. The question would be the use of one single label for the entire packet routing or the use of re-labeling at the different Data Vortex switches. Due to the Data Vortex operation principles, we consider more appropriate to avoid label swapping and use a single label to route the packets. For example, a scheme implementation that enables different switches to route the same-labeled packet to different output ports would be a good solution. In addition, as explained during this thesis, the implementation of some small buffering at the switch input or output is necessary. In this thesis we have quantify the grade of contention at the switch output and input. By means of these results, determining the buffering strategy and number of resources needed would be very interesting, especially if we want to make a realistic comparison with other packet switching architectures. In particular we suggest to perform a realistic comparison between the Data Vortex packet switching and other IWS optical packet switching architectures. This comparison should be focused on the performance and number of resources necessary of each architecture to provide a given capacity. We consider a strict and realistic comparison complex, since the current premature stage in which optical packet switching architectures are.
Regarding to the novel Data Vortex architecture with improved latency proposed, we analyzed its performance under uniform traffic. Computer simulations showed that the new mode of operation have no performance penalty. We propose to analyze the performance under bursty traffic to complete the new operation mode evaluation. This new operation mode will not have exactly the same performance as the generic one. However, a very similar performance is expected, with the added improved latency and compact realization advantages of the novel operation mode. This issue should be confirmed.

The EAM technology has been proposed for WDM/TDM packet switching. This proposal was based on the demonstration of a simultaneous RZ modulation of four 40 Gbit/s NRZ-modulated wavelength channels (1547 nm -1557 nm, 2.5 nm spaced between each other) using EAM technology. The realization of a specific experiment in which the dynamic response of the EAM could be characterized is suggested. In this experiment several wavelengths modulated at a certain bit rate would be multiplexed and applied to the EAM input. The modulation pattern of each wavelength should be different to simulate WDM/TDM packets at the EAM input. Different reverse voltages would be applied to the EAM to study its output response. Our interest lies in obtaining a similar absorption for each of the modulated wavelength channels. The control voltage would be maintained continuous, since we are not testing the EAM for multiplexing or modulation purposes as those performed for OTDM. The aim of this experiment is to confirm the possible use of EAM for WDM/TDM packet gating.

We propose to study the implementation of the slot merging function in the Data Vortex architecture. The slot merging consists in grouping photonic slots that are switched to the same output port. This slot merging is possible when the merged slots are compatible, i.e., they do not carry data on the same wavelength channel. This function will increase the optical packet switch throughput. The Data Vortex architecture presented does not support this kind of functionality. However, the Data Vortex principle of operation is not incompatible with the slot merging function, although its implementation will result in a more complex packet processing. The distributed control would need to carry additional information of the wavelengths occupied in the WDM packets. Thus, the objective would be to include the slot merging function while maintaining the original simple Data Vortex routing operation.
References


