Chapter 9

QoS management in connection-oriented OPS networks

9.1 State-of-the-art

Recent studies have proposed enhanced contention resolution policies in order to provide quality-of-service (QoS) differentiation according to the DiffServ model [90]. The very limited queuing space and the impossibility of pre-emptying packets already buffered makes impossible the implementation of conventional fair queuing scheduling algorithms commonly used in electrical switches. At the same time, QoS management schemes must be kept very simple to be effective in OPS where each node must be able to schedule tens of Tbit/s of traffic.

Existing QoS schemes use basically the same method: 1) design a contention resolution algorithm which minimizes the Packet Loss Rate (PLR), thus 2) apply a QoS mechanism (some form of resources reservation on top of the contention resolution algorithm) able to differentiate the PLR among two or more classes. We can classify them as following:

- *Queueing priority* [67]. All incoming packets are stored into the buffer starting with the highest priority packets. If there are not free slots, remaining packets are dropped. This approach can be implemented only for synchronous networks where packets arrive at the same moment.

- *Threshold dropping.* Some resources (either wavelengths or delays) are reserved for higher priority class, while the rest is available for any class. From threshold dropping techniques are derived the following policies:

  - *Threshold-based priority policy* [19] [22]. A threshold value (which indicates that a possible congestion is imminent) is predetermined. If the network is not congested (queue occupancy above the threshold value) every packet is allowed to enter the queue.(i.e. no packet discard). When the queue length is longer than the threshold, only prior packets are allowed to enter the queue while low priority packets are discarded.

  - *Threshold-based with RED priority policy* [76]. The threshold mechanism can also be associated with a Random Early Detection (RED) strategy. When the buffer occupancy reaches the first threshold, the discard probability for lower priority packets begin to be mode than zero and its value is increased to 1 when the buffer occupancy reaches the second threshold.
  - *Wavelength-based priority policy* [18] [22] [12]. The lower priority packets can access only a subset of the wavelength resources and in any case they share it with higher priority packets.

- *Look-ahead* [41]. When a packet arrives, the header is extracted and the payload is delayed by a fixed amount of time at the input switch by means of an additional pool of optical buffers. This allows to the SCL to take scheduling decisions knowing a certain amount of packets arrivals. This solution to be effective needs additional hardware which means increase the cost. Also, the packet delay is affected since, to be effective, this scheme needs long FDLs as the results in [41] show.

- *Offset time-based* [106] [107] [41]. In this scheme, the edge node firstly sends a control packet and, after a given time called offset, sends the packet. The control packet is processed at each hop to determine the path of the packet. In order to provide QoS, different offset times are imposed at the edge node.

- *Use of electrical buffers* [12] [76]. In this case, to reduce the optical buffer requirements and make possible the use of random access, the optical packets can be converted in the electrical domain and stored electronically.

- *Queuing priority with overwriting* [56]. It is possible to build a complex buffer structure with two optical switches and a pool of FDLs in order to allow the overwriting of low-priority packets.

The problem of resource reservation schemes is that a fixed amount of resources are always reserved independently of the traffic profile. This implies a switch over-dimensioning with the relative cost increase. Moreover, such scheme does not present good enough results, for instance in [22] the PLR for low priority class is 10^{-2} with a load of 0.8. To achieve acceptable levels of PLR with this method, the scheduling requires very high computational complexity or very large optical memories.

The offset time method shows good results when applied to optical burst switching [107] where bursts comprise several packets. However, it seems not effective in OPS where the overhead of the control packets introduces considerable bandwidth wastage.

The hybrid E/O buffer method is not scalable with the bitrate since electronic devices cannot keep up with the speed of optical links and the E/O bottleneck is maintained.

The latter results costly and requires a computational-demanding scheduling. Indeed, no further investigations have been performed.

9.2 The Service Category-to-Algorithm Switching Selection technique

In this thesis, we propose a novel strategy able to improve the switch performance and provide the required QoS. It consists of defining different OPS service categories, like it was done in ATM networks, and the strategy is based on the fact that, in a QoS environment, it is not practical to provide the best handling to a traffic class that does not really require it. Therefore, if a set of K categories of service is available in the network, each category should be handled according to its requirements. For this reason we suggest to implement a set of K different handlings (i.e., algorithms) in the switches. When a packet belonging to an OVC with category i arrives to a switch, the SCL will execute the corresponding algorithm i to forward the packet which guarantees only the required service. We refer to this technique as *Service Category-to-Algorithm Switching Selection* (SCASS).

9.2.1 Example of defining three different OPS service categories

For this study, we consider a system with the following three categories of service:

- **Best Effort** (BE) with no specific QoS requirements.
- **Loss Sensitive** (LS) for multimedia broadcasting applications which requires bounded losses;
- **Real Time** (RT) for interactive applications which requires strict performance (very low PLR and very short delay);

Others could be defined, but the point here is the definition of the different service categories, not the categories themselves.

We hence design three algorithms to be implemented in the SCL. The algorithms are the following:

- **Two-State Wavelength Selection** (TSWS) for supporting the BE category of service;
- **Losses Bounding Wavelength Selection** (LBWS) for supporting the LS category of service. It can also support the BE category when there are low LS connection demands;

  Sequence Keeping Wavelength Selection (SKWS) for supporting the RT category of service. It can also support the BE and LS categories when there are low RT connection demands.

The aim of the TSWS algorithm is to reduce the control overload (low FO) while maintains an acceptable level of the PLR. This algorithm tries to improve the performance of the static approach assigning two wavelengths to the OVC during the

When a packet belonging to OVC l arrives:

1. Look-up the forwarding table to determine $n^{out}(l)$ and wavelengths $\lambda_0^{out}(l)$ and $\lambda_1^{out}(l)$ assigned to l;
2. Determine the minimum delay D_j available for $\lambda_0^{out}(l)$ and $\lambda_1^{out}(l)$;
   (a) If $D_j > D_M$ (i.e., both wavelength are busy), the packet is lost;
   (b) If not, select the FDL j to send the packet to.

Figure 9.1: TSWS algorithms.

setup procedure (i.e., the GRP algorithm is executed twice). This assignment is kept constant all over the OVC life and single packets are always forwarded to the less congested wavelengths. This means that the wavelength searching step of the contention resolution algorithm is never needed (FO is always 0%).

Figure 9.1 shows the steps to follow when running this algorithm.

The aim of the LBWS algorithm is to achieve a bounded PLR. Each OVC is assigned to a wavelength at setup using the GRP algorithm. This assignment may change if the OVC experiences a PLR above of a predetermined value R (*required PLR*). For this scope, a window T is defined. Every T the algorithm computes the PLR of each OVC. These PLRs are then ordered in descending way; starting from the higher value, the algorithm compares the PLRs with R, if it is higher, a new GRP algorithm is executed to reassign the OVC to another wavelength. Clearly, the value of T affects the switch performance: high values may not guarantee the required PLR; contrarily, low values can increase the control overload with an extreme situation of executing a new GRP algorithm per each incoming packet. It is important to notice that the value of R can be different from one OVC to another since their requirements can be distinct. For sake of simplicity, in this work we assume the same value for all OVCs that use this algorithm.

Figure 9.2 shows the steps to follow when running this algorithm.

The aim of the SKWS algorithm is to achieve excellent level of PLR maximizing the resource utilization and throughput. This is achieved taking per-packet decisions. At the same time, SKWS needs to control the delay preserving the correct packet sequence belonging to the same OVC. Indeed, since very short optical buffers are available in OPS networks, the delay is only due to the propagation delay and to rebuild the original information at the edge of the optical network. The latter may introduce considerable delays if extensive reordering operations are needed due to the out-of-order delivery of the packets [71]. For this reason, the design of the SKWS algorithm also considers the maintenance of the correct sequence of the packets belonging to the same OVC.

For the purpose of this work, given a stream of ordered packets at the switch input, we define the packet n to be out-of-order when the first bit of packet n leaves the switch before the last bit of packet $n-1$. This may happen in general in a DWDM

When a packet belonging to OVC l arrives:

1. Look-up the forwarding table to determine $n^{out}(l)$ and $\lambda^{out}(l)$ assigned to l;
2. Determine the minimum delay D_j available for $\lambda^{out}(l)$;
   (a) If $D_j > D_M$, the packet is lost;
   (b) If $D_j \leq D_M$, select the FDL j to send the packet to.

Every window T:

1. Determine the PLR of the OVCs: PLR^T;
2. Order (descending) the PLR^T;
3. From $l = 1$ to $L \times W \times N$:
   (a) If $PLR_l^T > R$, execute a new GRP algorithm

Figure 9.2: LBWS algorithms.

environment that is equivalent to a bunch of links in parallel. Strictly speaking the packet sequence is also maintained as long as the first bit of packet n does not leave the switch before the first bit of packet $n-1$, meaning that the head of packet n is allowed to partially overlap with the tail of packet $n-1$. Unfortunately such less restrictive case is more difficult to control, especially when considering a cascade of switches. In fact, taking into account that, in general, the optical packets can aggregate more than one IP packet, the relative position of subsequent IP packets included in two subsequent optical packets cannot be controlled if overlapping is permitted. Therefore, a strict sequence keeping (i.e. avoiding packet overlapping) represents the unique procedure that assure the maintenance of sequence both at the optical packet level and at the IP packet level. Consequently in this work we have adopted this procedure.

To keep the correct packet order, the SCL stores the time-stamps t^{out} (one per each OVC) at which the last bit of the last packet is scheduled to leave the switch. This time is calculated as the sum of the packet arrival time, its duration and the delay assigned in the buffer. When a packet belonging to the OVC l arrives, the SCL recalls the time $t^{out}(l)$ and determine if the new packet needs additional delay to keep the order. This delay must be equal at least as long as the residual transmission time of the previous packet belonging to the same OVC l. Due to the discrete number of delays provided by the optical buffer, the additional delay is calculated as the integer multiple of D greater than $t^{out}(l)$.

Figure 9.3 shows the steps to follow when running the SKWS algorithm.

When a packet with duration d belonging to OVC l arrives at time t:

1. Look-up the forwarding table to determine $n^{out}(l)$, $\lambda^{out}(l)$ and $t^{out}(l)$ assigned to l;
2. If $t^{out}(l) \geq t$, the previous packet has already leave the node:
   (a) If $\lambda^{out}(l)$ is busy, search for the set of wavelength $\Lambda \in n^{out}(l)$ not busy:
   (b) If $\Lambda = \emptyset$, the packet is lost;
   (c) If $\Lambda \neq \emptyset$, determine the wavelength w with the minimum delay D_j and select the FDL j to send the packet on;
3. If $t^{out}(l) < t$, the previous packet is still in the node:
   (a) Calculate the minimum delay D_{min} to add to the packet to preserve the order $D_{min} = \left\lceil \frac{t^{out}(l)-t}{D} \right\rceil D$;
   (b) If $\lambda^{out}(l)$ cannot provide D_{min}, search the set of FDL F able to provide a delay $D_j \geq D_{min}$
   (c) If $F = \emptyset$, the packet is lost;
   (d) If $F \neq \emptyset$, select the FDL with minimum D_j, if more than one is available, select the wavelength w that introduces the minimum gap between two subsequent packets;
4. Update the forwarding table $t^{out}(l) = t + d + D_j$.

Figure 9.3: SKWS algorithms.

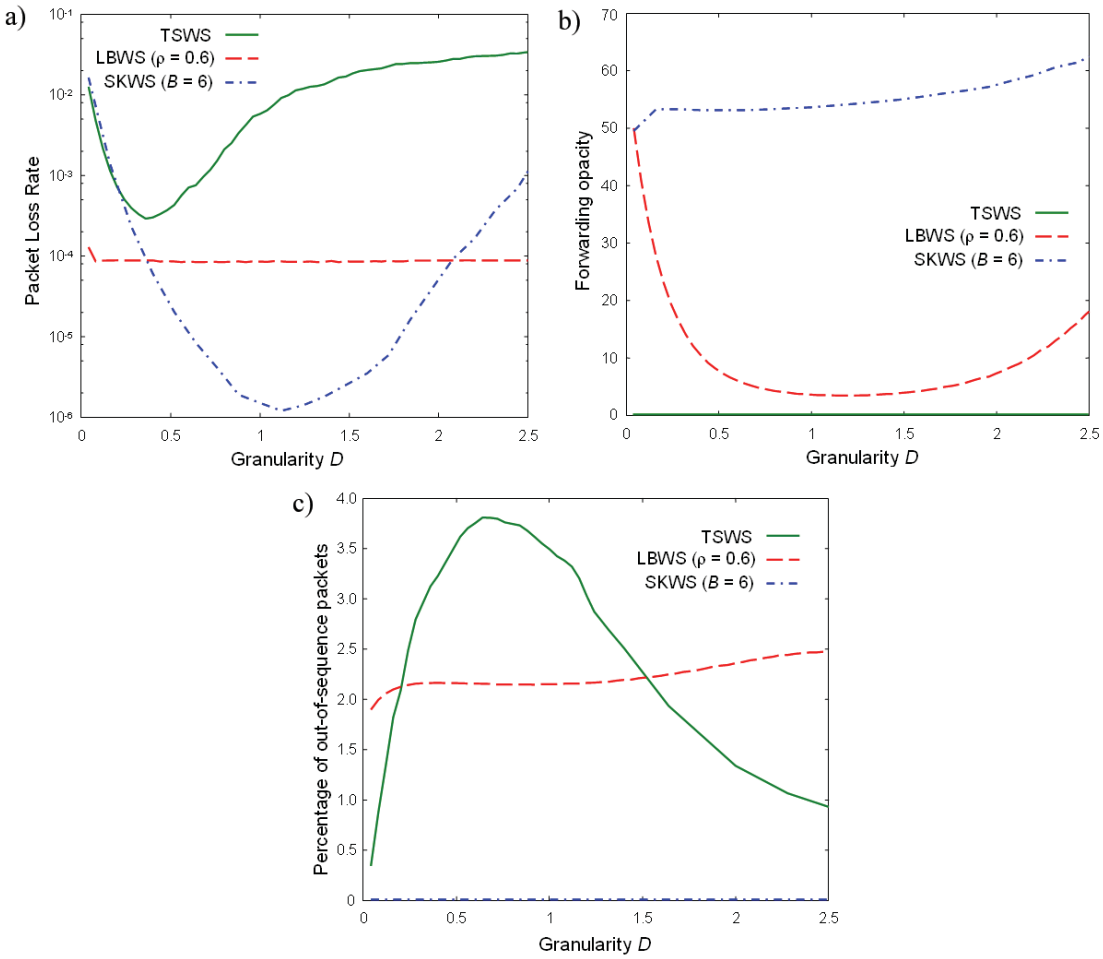

Figure 9.4: a) Packet loss rate, b) Forwarding opacity, and c) Out-of-sequence packets as a function of D normalized to the average packet duration, comparing TSWS, LBWS and SKWS

9.2.2 Performance evaluation

In the this section, the algorithms are evaluated separately in order to find their specific characteristics. Afterwards, we integrate them in the same switch and evaluate the SCASS technique.

In the following figures, we set up the simulator (described in Chapter 7) with $N = 4$, $W = 16$, $C = 10$ Gbps, and $L = 3$. The buffer configuration is a degenerate buffer $\mathbf{Q}_8$ (i.e., the length is $B = 8$) except for SKWS which uses a shorter buffer $\mathbf{Q}_6$. The offered load is $\rho = 0.8$, except for LBWS where it is $\rho = 0.6$ because it is not possible to bounding the PLR of high amount of traffic maintaining an acceptable control complexity. R is set to 10^{-4} and T to 20 D which are reasonable values offering a good trade-off between complexity and PLR.

Figure 9.4 shows the simulation results when independently evaluating the three algorithms.

Figure 9.4a) plots the PLR as a function of D normalized to the average packet duration, comparing the TSWS, LBWS and SKWS algorithms. In this figure we can see that SKWS achieves the better PLR of 10^{-6} with $D = 1.2$. Contrarily to the usual concave behavior shown by other algorithms, LBWS exhibits constant values less than 10^{-4} which is the value set as required. TSWS presents the worst PLR but it is important to remark that its aim is to have low control complexity.

Figure 9.4b) plots the FO measure comparing the TSWS, LBWS, and SKWS. It is clear that SKWS imposes the higher overload on the switch control; while LBWS shows low computational requirements reaching values close to 4%. The LBWS curve indicates that keeping bounded PLR require less computations for value of D ranging between $D = 1$ and $D = 1.4$, with a minimum in $D = 1.2$. Finally, TSWA does not need to reconfigure its OVC-to-wavelength assignment; therefore FO is always 0%.

Figure 9.4c) shows the percentage of out-of-sequence packets comparing the TSWS, LBWS, and SKWS algorithms. As expected, SKWS maintains the correct sequence delivering. LBWS presents values around $2 \div 2.5\%$ while TSWS exhibits a concave behavior with a maximum of 3.7% in $D = 0.7$.

9.2.3 Optical buffer architecture to integrate different SCASS

The results previously obtained assess the goodness of the proposed algorithms indicating that their aims have been fully accomplished: TSWS imposes low control overload and reaches acceptable PLR; LBWS requires low control overload and is able to guarantee a bounded PLR; finally, SKWS requires high control overload but achieves very good PLR maintaining the correct order of the packet sequence. The next step is hence the integration of these algorithms in the same SCL and the verification of the mutual impacts on the performance measures.

The integration is not trivial because the previous results also indicate that the algorithms achieve the better performance with different values of the fiber granularity D, the optimum D for LBWS and SKWS is 1.2 while it is 0.4 for TSWS (see Figure 9.4a and Figure 9.4b).

Note that the rate between these two values ($D = 1.2$ and $D = 0.4$) is exactly 3. Exhaustive simulations (not presented here for lack of space) show that this peculiar factor of 3 is valid for whatever traffic matrix. It only depends on the traffic characteristics like average packet size. Based on this factor, the integration of the different contention resolution algorithms can be done using the following buffer architecture. Firstly, we fix $D = 0.4$ and set up two degenerate buffers: $\mathbf{Q}'$ with $D_j = jD$ delays and length B' for the BE packets and $\mathbf{Q}''$ with $D_j = 3jD$ delays and length B'' for RT and LS packets. Then, these buffers are merged in a non-degenerate buffer $\mathbf{Q} = \mathbf{Q}' \cup \mathbf{Q}''$ in such a way that the delays that are common in $\mathbf{Q}'$ and $\mathbf{Q}''$ are available for any category. Figure 9.5 shows an example with $B' = B'' = 4$, and a resulting length $B = 6$ of buffer $\mathbf{Q}$.

For the evaluation under multi-category, we set $N = 4$, $W = 16$, $C = 10$ Gbps, $\rho = 0.8$, $L = 3$, and, finally, the required PLR and measure window for LS packets to $R = 10^{-5}$ and $T = 20\ D$, respectively. Regarding the distribution of traffic, in Figure 9.6, Table 9.1 and Figure 9.7 we assume that 50% of the OVCs transport BE

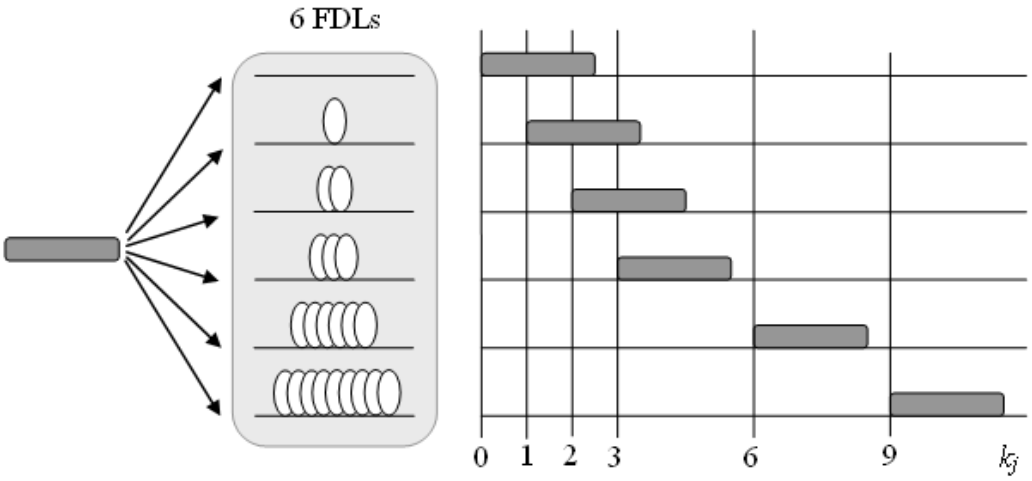

Figure 9.5: Non-degenerate buffer configuration with 6 FDLs. BE packets can use delays $\{0, D, 2D, 3D\}$, while the RT and LS packets can use delays $\{0, 3D, 6D, 9D\}$

packets, 30% transport RT packets, and the rest LS packets. In Figure 9.8 we analyze the PLR changing this distribution.

Figure 9.6 plots the PLR for the entire system as a function of D normalized to the average packet duration. In the figure, we include secondary x-axis which indicates the granularity perceived by SKWS and LBWS algorithms (exactly 3 times D). As expected, any categories of service achieves the optimal PLR in correspondence of $D = 0.4$. Hence, we use this value to obtain the following results.

In Table 9.1, we compare the SCASS technique with the *Empty Queue Wavelength Selection* (EQWS) algorithm [21] - the best performed dynamic algorithm - and the *Minimum Gap* (MINGAP) algorithm [15] - the best performed connectionless algorithm. Both EQWS and MINGAP use the buffer threshold approach [22] to provide QoS (the values of D and of thresholds are those providing the lowest PLRs).

The results show that the SCASS technique provides the lowest PLR for both LS and BE traffic. Moreover, as expected, the higher control complexity is required to forward the RT traffic (FO is 66.14%) while LS and BE impose low overload (5.93% and 0% respectively). In contrast, MINGAP imposes the same (very high) FO for any category, while EQWS requires higher FO for BE traffic which is an evident nonsense. At the same time, the packet sequence of RT traffic is preserved using the SCASS technique, while it reaches 2% and 5% using EQWS and MINGAP, respectively. Previous studies [5] [65], confirm that even a small percentage of out-of-sequence (like that caused by EQWS algorithm) may impact harmfully on the network performance. We must also consider that this percentage is counted at the output of a single switch; by assuming n switch in series along a path this percentage increases accordingly.

Figure 9.7 plots the PLR as a function of the buffer depth B for any category. The results indicate that a significant improvement of the performance can be obtained with a small increase of the number of FDLs B of buffer $\mathbf{Q}$.

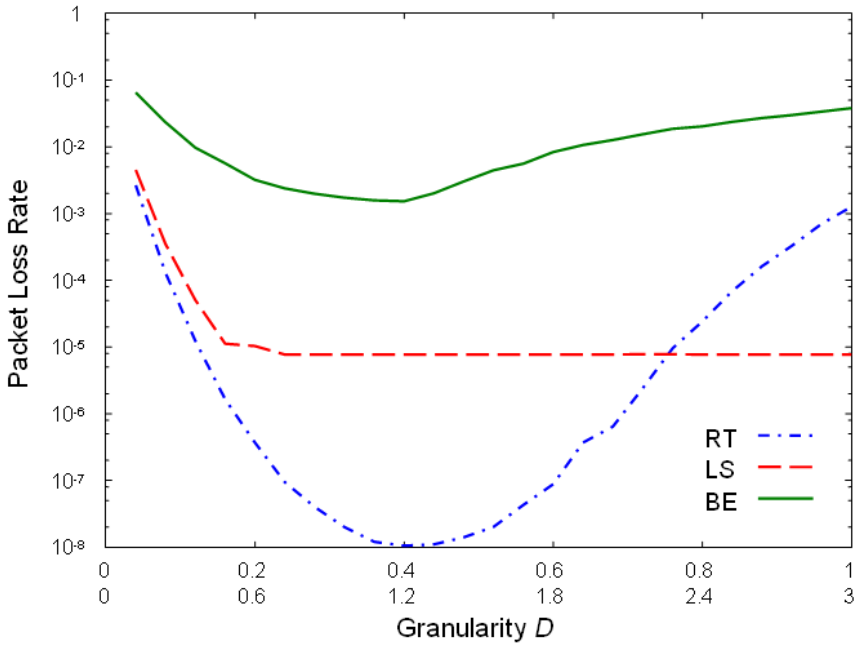

Figure 9.6: Packet loss rate as a function of D normalized to the average packet duration.

Table 9.1: PLR, FO and OS comparing SCASS technique with EQWS and MINGAP both adopting a buffer threshold technique

	SCASS		
Category	**PLR**	**FO**	**OS**
RT	$1.08\ 10^{-8}$	66.14%	0%
LS	$7.68\ 10^{-6}$	5.93%	1.76%
BE	$1.55\ 10^{-3}$	0%	3.29%
	EQWS		
RT	$3.00\ 10^{-8}$	16.20%	2.02%
LS	$2.75\ 10^{-4}$	30.82%	2.33%
BE	$5.24\ 10^{-2}$	52.51%	3.41%
	MINGAP		
RT	0	81.33%	5.39%
LS	$9.78\ 10^{-4}$	81.05%	5.03%
BE	$3.96\ 10^{-3}$	80.92%	4.62%

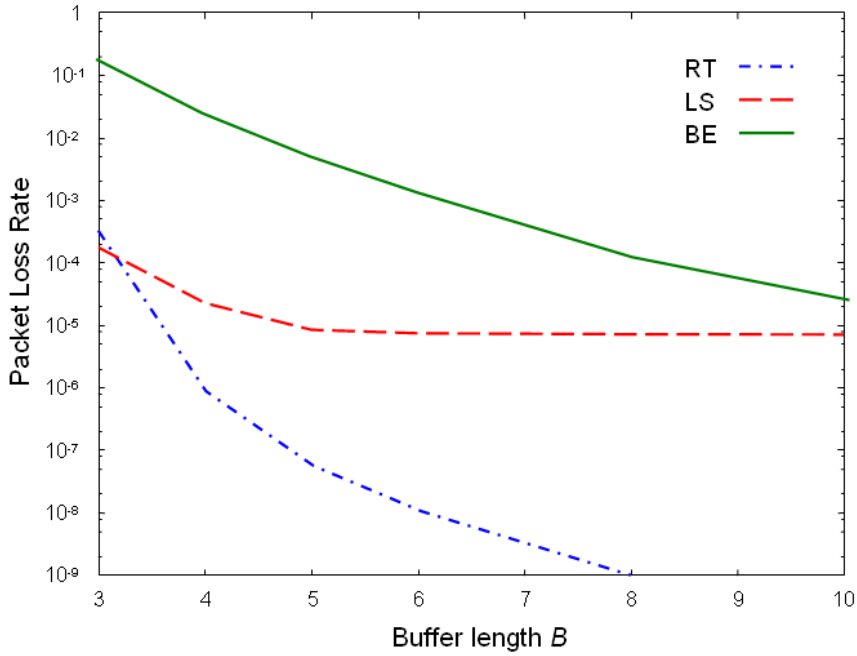

Figure 9.7: Packet loss rate as function of the buffer length B.

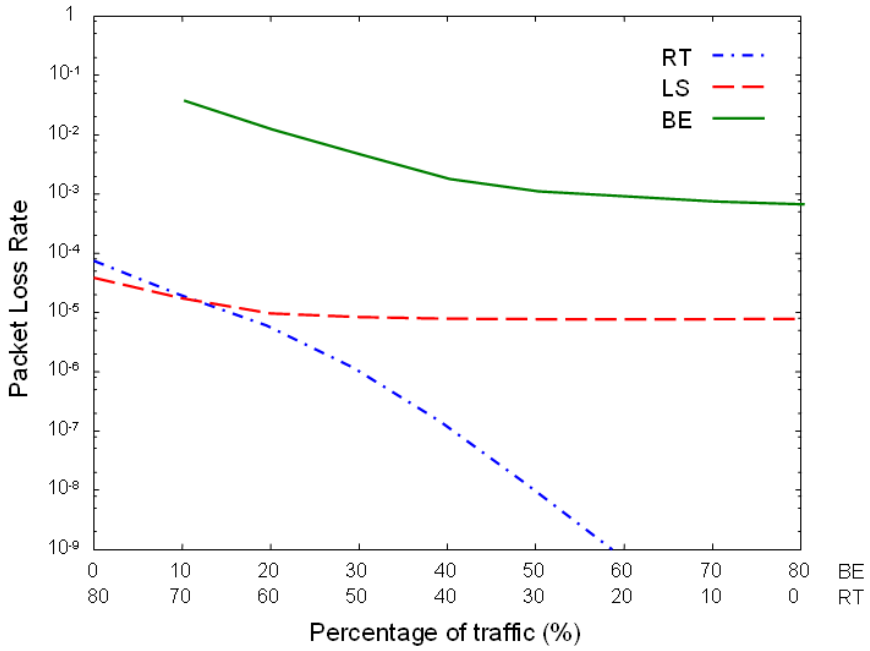

Figure 9.8: Packet loss rate as function of traffic relative load percentage.

Finally, Figure 9.8 shows the PLR changing the percentage of the relative load between the RT and BE traffic while maintaining fixed to 20% the relative load of LS traffic. This means that for instance, when the percentage of RT is 20%, the percentage of BE is 60%. We can see that the PLR of LS cannot be guaranteed if there is a high percentage of RT traffic (i.e., more than 60%). On the other side, if RT is not present, BE traffic is not able to fully exploit the switch capacity and the PLR remains relatively high. A way to improve the performance of the BE traffic when RT and LS present low loads is to apply the SKWS algorithm also to some BE OVCs. In this case, a smart algorithm should be developed in order to decide when, which, and how many BE OVCs can be forwarded according to the SKWS algorithm. This study is not developed here and is let for future investigations.

Future works will deal with the integration of the quality differentiation method with the SCASS technique in order to obtain a more flexible environment. At the same time, SCASS opens up future interesting developments on the routing problem for a whole network scenario.