Modeling Fiber Delay Loops in an All Optical Switch

Ana Bušić[1], Mouad Ben Mamoun[1,2] and Jean-Michel Fourneau[1]
[1]*PRiSM, Université de Versailles Saint-Quentin-en-Yvelines, 78000 Versailles, France*
[2]*Université Mohammed V, B.P. 1014, Rabat, Maroc*
{*abusic, mobe, jmf*}*@prism.uvsq.fr*

Abstract

We analyze the effect of a few fiber delay loops on the number of deflections in an all optical packet switch. The switch is based on the ROMEO architecture developed by Alcatel. We use deflection routing because of the lack of optical memory. Some fiber delay loops allow the packets to be locally deflected instead of being sent on the network for much longer delays. As the model is numerically difficult, we apply stochastic bounds. First, we consider a partial ordering on the state space and we prove that the problem is monotone. Then we present a new method which strongly relies on this property. Note that this method is quite general as partial order monotone multicomponent systems are quite frequent in performance evaluation. The upper bounds are computed using robust numerical algorithms on a smaller state space. We also show how we can compute lower bounds to check the accuracy of the method.

1 Introduction

Recent technology advancements in optical packet switching [7, 9] give rise to an increasing need for performance evaluation methodologies. Semiconductor Optical Amplifiers achieve reconfiguration time in the order of a few nanoseconds or even a few hundred picoseconds [7]. These improvements allow to design in the near future switches based on Optical Packets rather than Optical Bursts. For instance, the ROM project [9] promoted by Alcatel has proved the feasibility of the optical components and the electronic control plane for an all optical packet core network. However, some performance issues are still crucial before one can completely design such switches. One of the major problems is related to routing without the buffers which are necessary for the store and forward principle. Here we assume that the switch is synchronous and that the packets have a constant size. Even if it is not completely true that the arrivals are synchronous, the technology to synchronize was shown to be available and the choice of a constant size packet offers many advantages (see the conclusions of the ROM project [9]).

Deflection routing is an attractive routing strategy for Optical Packet Switching networks since it does not rely on optical buffering of packets [2]. However, with deflection routing, a packet can stay in the optical network for an arbitrary long time due to the large number of deflection it experiences. In Shortest-Path Deflection Routing, switches attempt to forward packets along a shortest hop path to their destinations. Each link can send a finite number of packets per time-slot (the link capacity). Incoming packets have to be sent immediately to their next switch along the path. If the number of packets which require a link is larger than the link capacity, only some of them will use the link they ask for, and the others have to be misdirected or deflected and they will travel on longer paths.

Using simulations it has been shown that the average number of deflections is not that large but a significant fraction of the number of packets is heavily deflected when the traffic is unbalanced and the link capacity is small [3]. These packets constitute a real problem: they are never physically lost due to physical errors or buffer congestion but they can be logically lost because the transport delay is larger than the timeouts. As optical packets are very long and contain a lot of TCP packets, the loss of an optical packet will provoke a lot of TCP session slow starts. Thus, it is quite important to have the smallest transport time and the smallest deflection probability.

When the number of wavelengths per link is high, one can observe that the probability of deflection becomes smaller. But this is usually not sufficient to avoid long delays. Adding a few Fiber Delay Loops (FDLs in the following) will help to reduce the effect of a deflection. If a packet must be deflected, we send it in this loop instead of sending it in a wrong direction and it will be inputted again into the switch at the end of the loop. The FDLs have the length equivalent to an integer multiple of the time slot. When the packet comes back into the switch, it will compete with the other deflected packets, and with the transit packets and the fresh packets which have just entered. Using fiber delay loops is sometimes denoted as local deflection. The deflection probability is not much modified by the loop, but the effect of a deflection is now significantly less important. Indeed, a local deflection only takes a few time slots (i.e. the length of the loop), while a real deflection will add at least one propagation delay. Remember that in a core network the propagation delay due to the link lengths is quite important (typically tens of time slots).

Previous analytical studies of deflection algorithms and networks [5, 1] have proposed models for networks based on 2×2 switching blocks without FDLs. However, these models have a too simplified model for a switch and they do not consider the effect of FDLs to allow local deflections.

Another type of approach was recently proposed [12] to analyze the FDLs. However, the authors assume that the switch has an infinite number of FDLs and each loop has a different length. Clearly this assumption implies that the switch has an infinite bandwidth. If the number of FDLs becomes large, this approximation may be valid. But real switches do not have such a large bandwidth to provide for loops. For instance, the switch architecture developed with Alcatel (the ROMEO core switch) has only a few FDLs for packet recirculation and the infinite FDL assumption is not acceptable.

We develop here a detailed model of a ROMEO switch and we show how efficient the FDLs can be to avoid real deflections. Using some independence assumptions, we build a Markov chain representation of the number of packets in the loops. Unfortunately, this chain is numerically intractable. Thus we use stochastic comparison to state that these FDLs provide low real deflection probability.

Stochastic comparison of random variables has often been presented as a powerful technique in various areas of applied probability (see for instance the books by Stoyan [11], and Shantikumar [13]). When models are too complex to be solved efficiently, this method allows to study simpler or smaller models with the guarantee that the results are upper or lower bounds. The distributions or rewards of the simpler model are greater or smaller than the results of the original problem which remain unknown. Then we can directly prove that Quality of Service requirements are satisfied. Stochastic comparison of Markov Chains is also possible. Sufficient conditions for the existence of stochastic comparison of two time-homogeneous MCs are given by the stochastic monotonicity and bounding properties of their one step transition probability matrices [11]. An algorithmic derivation of their stochastic bounds have been introduced [6]. However, this theory is based on a total ordering of the state space, while many problems that we consider are based on a natural partial order. Of course, it is possible to transform this partial order into a total one, but this modification adds a lot of unnecessary constraints making the bounds less accurate.

Quite often, we analyze Markov chains which are specified by several components and their composition rules. These formalisms (for instance Stochastic Automata Networks, Queuing Networks or Stochastic Process Algebra) generally lead to multidimensional Markov chains which are clearly associated to partial order. Similarly, when the problems involve rewards, we have a total order on the rewards but only a partial order on the states if several states have the same rewards. In this paper we use the natural partial order monotonicity of the model to build a bounding system that is easy to analyze.

The remaining of the paper is organized as follows. In Section 2 we present the architecture of the switch and the routing. Then we state the model of the system. In Section 3 we show how to compare multicomponent systems whose components are monotone. We also show that we can compute upper and lower bounds and how we can check the accuracy of the bound. Section 4 is devoted to the numerical analysis of the bounds. We have also computed numerically the deflection probability for a system without FDLs. We show that one or two loops are sufficient to provide a very low real deflection probability even at a very high load. Finally, in the conclusion, we stress that the approach is much more general and can be applied to most systems which we may define as partially ordered and monotone.

2 The model and its properties

We first describe the architecture of the switching nodes promoted by the ROMEO project conducted by Alcatel (for more details see [4]). The switch has 8 bidirectional links (see Figure 1). Each of them uses f wavelengths. In the usual configuration, 4 links are used for interconnection of switches according to the core network topology. Two links are connected to the edge network and are used for the add and drop mechanism (the input and the output of packets). Finally, the Fiber Delay Loops use the last remaining 2 links. Note that this long delay FDLs are different of the small delay loops which are associated to all links. These small loops allow to delay the optical payload while the optical header of the packet is decoded and transferred into the electronic control plane for routing.

Note that it is possible to exchange links between the set of add and drop links and the set of FDLs. We study in this paper configurations with 1 or 2 FDLs. The stochastic comparison result we prove in Section 3 does not require some particular number of FDLs or some assumptions on their lengths. For the sake of readability we will detail here only rather simple 2 FDL configuration with a link of length 1 and a link of length 2. Similarly, when we consider a 1 FDL configuration, we assume the FDL of length 1. Numerical results show that these simple configurations are sufficient to obtain very low real deflection probability.

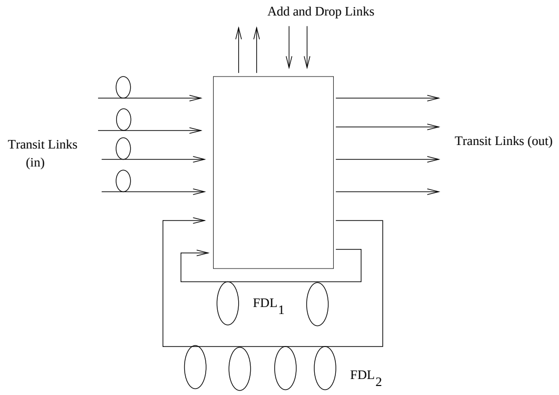

Figure 1. Architecture of the ROMEO switch

In the following we assume that the number of traffic links is c, that the number of add and drop links is m and that the number of FDL links is n (1 or 2). Remember that the link capacity is f. The packets have only one output link required at each step. This link is computed based on the destination and the current place of the packet. The list of links constitute the route followed by a packet. We assume that the packets are independent. The switching is processed as follows:

1. The packets from core network enter the switch. Let A_t be this random variable. The packets locally deflected in the former slots also enter the switch. Let InD_t be this number of packets.
2. The packets which are now arrived at their destination must leave. Let D_t be the number of packets which need to leave and T_t the packets in transit which have to be routed. As each packet routing decision is i.i.d. with probability d to leave the switch, we have:

   $$D_t \overset{d}{=} B(A_t + InD_t, d), T_t = A_t + InD_t - D_t, \quad (1)$$

   where $B(X, p)$ denotes the Binomial distribution.
3. But we only have $f * m$ output capacity. If D_t exceeds $f * m$, then some packets will be deflected because we do not have enough bandwidth to leave the core network. The number of packet to be deflected due to this output bandwidth limitation is $(D_t - f * m)^+$. As usual, $x^+ = \max(x, 0)$.
4. Then, new packets may enter. Denote the offered load by E_t. As the global bandwidth is $(c+n)f$ and the transit or deflected packets have higher priority, some new packets may be blocked and do not enter the switch. Let N_t be the number of new packets which really enter the switch, it is clearly the minimum between the offered load and the available bandwidth:

   $$N_t = \min(E_t, (c+n)f - T_t - (D_t - f * m)^+). \quad (2)$$
5. We now have to route $T_t + N_t$ packets according to their destinations. Let $U_t^1, \ldots, U_t^c$ be the partition of $T_t + N_t$ packets into c directions (i.e. the c outputs of the switch connected to the core). Let $p_1, \ldots p_c$ the probability distribution for the routing. Again we assume that the packets are independent. Thus,

   $$(U_t^1, \ldots, U_t^c) \overset{d}{=} M(T_t + N_t, p_1, \ldots, p_c) \quad (3)$$

   where $M(X, p_1, \ldots, p_c)$ denotes the Multinomial distribution.
6. Again each link has capacity f and some packets must be deflected. Thus $(U_t^i - f)^+$ is the number of packets deflected due to link i.

Clearly the number of deflected packets, $OutD_t$ is:

$$OutD_t = (D_t - f * m)^+ + \sum_{i=1}^{c} (U_t^i - f)^+. \quad (4)$$

Local deflection is always preferred to real deflection and we use the shortest FDL first. We now have to establish the connection between the packets entering at time t after a local deflection and the packets locally deflected in the past. We only present here a simple 2 FDL configuration. The general case is straightforward but it requires more notations. The first loop (FDL_1) adds a delay of 1 time slot, while the second loop (FDL_2) keeps the packets for two time slots. Note that a Fiber Delay Loop is a constant delay. When the packet is injected in the FDL of length 2 at time t, it will enter again the switch 2 time slots later, even if we have bandwidth available in the good direction at time $t+1$. The output of this FDL at time $t+2$ are exactly the packets which enter the loop at time t. On the FDL of size 2, we can send another packet at time $t+1$ on the same wavelength we have used to send a packet at time t.

Let us denote by $F1_t$ the number of packets at time t in FDL_1. Let $F2_t$ be the number of packets sent at time t in FDL_2. If the number of deflection is smaller than f, we have only local deflections and they all use FDL_1. If there are more deflections, up to f packets are sent in FDL_2. Finally, if we have more than $2f$ deflected packets, they are really deflected. Thus,

$$\begin{cases} F1_t & = & \min(f, OutD_t) \\ F2_t & = & \min(f, (OutD_t - f)^+) \\ RD_t & = & (OutD_t - 2f)^+ \end{cases} \quad (5)$$

where RD_t is the number of really deflected packets.

The locally deflected packets return to the switch after one time slot if they are in FDL_1 or two time slots for packets in FDL_2. Therefore, $InD_{t+1} = F1_t + F2_{t-1}$. Thus if A_t and E_t are i.i.d., and if the deflected packets do not have memory we can model the system with an order 2 Markov chain. As usual, we can obtain a Markov chain adding a new component in the chain description: $(F1_t, F2_t, F2_{t-1})$. The reward is the number of real deflections RD. Remark that the lack of memory for deflected packets is a key property to get the Markov property here. Once they enter again in the switch the deflected packets choose again if they try to leave or along which link they try to route.

Remember that $F2_t$ is positive only when $F1_t$ is equal to f. Thus the state space has size $(2f+1)(f+1)$ when we model the above described configuration with 2 loops. Typically the number of wavelengths is 64 or 128. When we consider systems with a larger number of FDLs or longer FDLs the state space becomes much larger. It is possible to perform some exact aggregation but the state space is still too large to be handled efficiently. Indeed, the eigenvalues are badly distributed and iterative algorithms do not converge easily. So we use a stochastic comparison approach to bound the number of really deflected packets RD by rewards computed on a smaller chain. This small chain is solved by a direct method known for its accuracy (i.e. the GTH algorithm [8]).

The system with only one delay loop is easily derived from the former equations. The first set of equation describing the quantities involved in the routing is kept unchanged while the FDL description is now simplified. The Markov chain is much simpler as we only need to represent one loop: $F1_t$ is sufficient. We have: $F1_t = \min(f, OutD_t)$, $RD_t = (OutD_t - f)^+$ and $InD_{t+1} = F1_t$.

Note that the comparison theorem we state in the next section is established for an arbitrary number of loops. We do not need at this time assumptions about the arrival process. The comparison results only assume independence and identical distribution among the packets. This last assumption is not necessary but it is quite natural ant it helps to derive a simpler proof of our main result.

3 Stochastic Bounds for Monotone Multicomponent Systems

We now present the basic methodology we use to obtain the comparison results. It is worthy to remark that this method is more general and can be used to establish bounds of rewards and distributions for a wide set of models which exhibit some partial order monotonicity properties that we will now define. In particular the theorem we proved does not take into account the nature of traffic and the number of links.

We will first define the strong stochastic order with respect to the partial ordering on the state space. Then we will introduce the monotonicity property for transition matrices of homogeneous discrete time Markov chains (DTMC). The monotonicity property together with the transition matrix comparison (Definition 3) are sufficient conditions for stochastic comparison of two DTMC (Theorem 1). More details on stochastic orders and monotonicity properties on partially ordered spaces can be found in [10, 11].

Definition 1 *Let $(S, \preceq)$ be a partially ordered space and X and Y two random variables on S. X is smaller than Y in a strong stochastic sense, $X \preceq_{st} Y$ if, provided that the expectations exist,*

$$\mathbf{E}[f(X)] \leq \mathbf{E}[f(Y)],\ \forall f \textit{ increasing function.}$$

In the following, we will consider only a finite partially ordered space S, and we will use interchangeably $X \preceq_{st} Y$ and $x \preceq_{st} y$, where x and y denote the probability distribution vectors of random variables X and Y.

A subset $U \in S$ is called an upper set if its indicator function $\mathbf{1}_U$ is increasing. It follows that U is an upper set if and only if $x \in U$ and $x \preceq y$ imply $y \in U$. The following characterization is often used as definition of $\preceq_{st}$-order on a partially ordered space S [13].

Proposition 1 *$X \preceq_{st} Y$ if and only if $P(X \in U) \leq P(Y \in U)$, for all upper sets $U \subset S$.*

For example, let us consider the space $S = \{1, 2, 3, 4\}$ with the partial order defined by $1 \preceq 2 \preceq 4$ and $1 \preceq 3 \preceq 4$. Then $\emptyset, \{4\}, \{2, 4\}, \{3, 4\}, \{2, 3, 4\}, S$ are all upper subsets of S. If we consider the random variables X, Y and Z with the following distribution vectors $(0.3, 0.4, 0.1, 0.2)$, $(0.3, 0.1, 0.3, 0.3)$ and $(0.1, 0.2, 0.3, 0.4)$, then we have $X \preceq_{st} Z$ and $Y \preceq_{st} Z$, but X and Y are not comparable in the $\preceq_{st}$-sense since $P(X = 4) = 0.2 < P(Y = 4) = 0.3$ but $P(X \in \{2, 4\}) = 0.6 > P(Y \in \{2, 4\}) = 0.4$.

Let us now introduce the monotonicity property and the comparison for transition matrices of homogeneous DTMC.

Definition 2 *A transition matrix P of a homogeneous DTMC $\{X_t\}_{t\geq 0}$ is monotone if for all probability vectors x and y, $x \preceq_{st} y$ implies $xP \preceq_{st} yP$.*

The monotonicity property for a transition matrix of a homogeneous DTMC simply states that if the distributions at time t (x and y in the former definition) are ordered, the relation is kept at time $t+1$. Of course the monotonicity property strongly relies on the order considered. The example below shows that a chain can be monotone for a partial order and not monotone with a total order on the state space. Let us denote $P_{i,*}$ the row i of the transition matrix P. We have the following characterization of $\preceq_{st}$-monotonicity (see [11]).

Proposition 2 *Let $\{X_t\}_{t\geq 0}$ be a homogeneous DTMC with a partially ordered state space $(S, \preceq)$. The transition matrix P of $\{X_t\}$ is $\preceq_{st}$-monotone if for all $i, j \in S$ such that $i \preceq j$, $P_{i,*} \preceq_{st} P_{j,*}$, i.e if $\sum_{k\in U} P_{i,k} \leq \sum_{k\in U} P_{j,k}$ for all upper sets U.*

Definition 3 *For transition matrices P and Q we say that $P \preceq_{st} Q$ if $P_{i,*} \preceq_{st} Q_{i,*}$ for all $i \in S$, i.e if $\sum_{k \in U} P_{i,k} \leq \sum_{k \in U} Q_{i,k}$ for all upper sets U.*

We give now the classical comparison theorem for two homogeneous DTMC. The proof of this theorem can be found in [11].

Theorem 1 *Let $(S, \preceq)$ be a partially ordered space and let $\{X_t\}$ and $\{Y_t\}$ be two DTMC and P and Q be their respective transition matrices. If $X_0 \preceq_{st} Y_0$, at least one transition matrix P or Q is $\preceq_{st}$-monotone and $P \preceq_{st} Q$, then $X(t) \preceq_{st} Y(t),\ for\ all\ t > 0$. If X and Y have steady-state distributions π_X and π_Y, then $\pi_X \preceq_{st} \pi_Y$.*

We will now show that the Markov chain of the FDL based switch has a monotone transition matrix with respect to a partial order that we will define. Then, by using Theorem 1 we can design both upper and lower bounding chains that are numerically easier to analyze. We illustrate this step in the next section.

First, let us define the state space. Without loss of generality we can suppose that FDL links are indexed increasingly in their length. Remember that the packets to be locally deflected are put to the shortest FDLs first. We suppose that the number of loops is n and we denote by w_i the length of the FDL i, $1 \leq i \leq n$. Let w be the maximal FDL length ($w = w_n$).

Let us denote by $F_t^{(i,j)} = F_{t-j+1}^{(i)}$ the number of packets that entered FDL_i at time $t - j + 1$. Then the state space S can be given by the vector $F_t = (F_t^{(i,j)})_{1 \leq i \leq n, 1 \leq j \leq w_i}$ which describes the number of packets in each step of the loops.

The state space S is rather simple. However, we do not need all the information contained in vector F_t. It is sufficient to know the total number of packets in all FDLs that will return to the switch after k slots, for each k. More formally, let $G_t^{(k)}$ be the total number of packets which have already been locally deflected and that will return to the switch after exactly k slots:

$$G_t^{(k)} = \sum_{i | w_i \geq k} F_t^{(i, w_i - k+1)},\ \forall k, 1 \leq k \leq w, \qquad (6)$$

We only take into account the packets which are already present in some loops. Of course, the length of the loop must be larger than k; otherwise the packets will leave before k. For instance, for the 2 FDL system described in Section 2 with $f = 3$, if the state is $(F^{(1,1)} = 3, F^{(2,1)} = 1, F^{(2,2)} = 2)$, we have 5 packets leaving next slot $G^{(1)} = 3 + 2 = 5$ and 1 packet already in the loops leaving after two time slots $G^{(2)} = F^{(2,1)} = 1$. $G^{(k)}$ represents the contribution of the past to the traffic entering the switch in k slots.

Based on this property, we now define a function g on S: for $x = (x^{(i,j)})_{1 \leq i \leq n, 1 \leq j \leq w_i} \in S$, we define $g(x) = (g^{(1)}(x), \ldots, g^{(w)}(x))$, where

$$g^{(k)}(x) = \sum_{i | w_i \geq k} x^{(i, w_i - k+1)}, \forall k, 1 \leq k \leq w.$$

We can show that $g(x)$ contains all the information needed concerning the FDLs for the state x. For a random variable X and an event A, we will denote by $[X|A]$ a random variable that has as its distribution the conditional distribution of X given A.

Lemma 1 *For $x, y \in S$ such that $g(x) = g(y)$,*

$$[(G_{t+1}^{(1)}, \ldots, G_{t+1}^{(w)}) | F_t = x] = [(G_{t+1}^{(1)}, \ldots, G_{t+1}^{(w)}) | F_t = y].$$

Proof. Remember that we have supposed that FDL links are indexed increasingly in their length and that the packets to be locally deflected are put to the shortest FDL first. Thus we have

$$F_{t+1}^{(i,1)} = \min(f, (OutD_{t+1} - (i-1) * f))^{+}). \qquad (7)$$

Remark also that for all i, $F_{t+1}^{(i,j)} = F_t^{(i,j-1)}$, for all $j > 1$. Thus, for all k,

$$\begin{aligned} G_{t+1}^{(k)} &= \sum_{i | w_i \geq k} F_{t+1}^{(i, w_i - k+1)} \\ &= \sum_{i | w_i > k} F_{t+1}^{(i, w_i - k+1)} + F_{t+1}^{(k,1)} \\ &= G_t^{(k+1)} + F_{t+1}^{(k,1)}. \end{aligned} \qquad (8)$$

The second term is a function of (G_t, A_{t+1}, E_{t+1}). As A_{t+1}, E_{t+1}, and G_t are mutually independent, it follows that for $x, y \in S$, such that $g(x) = g(y)$, $[G_{t+1} | F_t = x] = [G_{t+1} | F_t = y]$. □

Thus, we can aggregate all the states with the same value of g into one state. We will denote this smaller state space by S'. Notice that S' can be seen as the image of function g defined on S. Thus, the state at instant t of our model is fully described by the vector $G_t = (G_t^{(k)})_{1 \leq k \leq w}$.

We will denote by $\preceq_*$ the usual product partial order on the state space S':

$$r, s \in S', r \preceq_* s \text{ if and only if } r_k \leq s_k, 1 \leq k \leq w. \qquad (9)$$

We give first some technical lemma and then state the monotonicity result for the FDL model (Theorem 2). We recall that, on a product space endowed with the usual product order, a function ϕ is increasing if and only if it is increasing in each variable.

Lemma 2 *1. If $Z = (Z_1, \ldots, Z_k) \preceq_{st} Z' = (Z'_1, \ldots, Z'_k)$ then $\phi(Z) \preceq_{st} \phi(Z')$ for each increasing function $\phi : \mathbf{R}^k \to \mathbf{R}^j$.*

2. *If Z_1 and Z_2 are independent random vectors of size k_1 and k_2, and $Z_i \preceq_{st} Z_i'$, $i = 1, 2$, where Z_1' and Z_2' are also independent, then $\phi(Z_1, Z_2) \preceq_{st} \phi(Z_1', Z_2')$ for each increasing function $\phi : \mathbf{R}^k \rightarrow \mathbf{R}^j$, $k = k_1 + k_2$.*

The proof can be found in [13, 11].

Lemma 3 1. *$B(X, p)$ is an increasing function of random variable X.*

2. *$M(X, p_1, \ldots, p_c)$ is an increasing function of random variable X.*

Proof. Follows from the fact that sum is an increasing function and that a Binomial distribution $B(X, p)$ can be seen as number of success of X independent identically distributed (i.i.d.) Bernoulli trials.

The statement for multinomial distribution uses similar arguments. □

Lemma 4 *For all $t \geq 0$, with respect to the usual product order, $OutD_t$ is an increasing function of (InD_t, A_t, E_t).*

Proof. First, notice that D_t and T_t are both increasing functions of (InD_t, A_t). This follows directly from (1), Lemma 3 and the fact that $T_t \overset{d}{=} B(A_t + InD_t, 1 - d)$. Thus, $(D_t - f * m)^+$ is also an increasing function of (InD_t, A_t) as a composition of increasing functions. Denote by $R_t = (D_t - f * m)^+$.

Let us now denote by W_t the packets that will go to the FDL links (n links) or to the next switching node (c links). We have:

$$W_t = R_t + T_t + N_t.$$

Now, from (2) it follows that $W_t = \min(R_t + T_t + E_t, (c + n) * f)$, thus W_t is an increasing function of (InD_t, A_t, E_t).

Finally, we will show that $OutD_t$ is an increasing function of (R_t, W_t). Then, $OutD_t$ is an increasing function of (InD_t, A_t, E_t) as a composition of increasing functions.

Equation (4) can be written as:

$$OutD_t = R_t + \alpha(W_t - R_t), \tag{10}$$

where $\alpha(X) = \sum_{i=1}^{c} (U_t^i - f)^+$ and $(U_t^1, \ldots, U_t^c) \overset{d}{=} M(X, p_1, \ldots, p_c)$. Lemma 3 implies that $\alpha(X)$ is an increasing function of X. Moreover, for $X, Y \geq 0$, from the properties of multinomial distribution, it follows easily that $\alpha(X + Y) \overset{d}{=} \alpha(X) + \alpha(Y)$.

Consider now $(R_t^1, W_t^1) \leq (R_t^2, W_t^2)$ and denote by $Z_t^i = W_t^i - R_t^i \geq 0,\ i = 1, 2$. If $Z_t^1 \leq Z_t^2$, then $OutD_t^1 \leq OutD_t^2$ follows trivially from (10) and the fact that α is increasing. Finally, if $Z_t^1 > Z_t^2$, denote by $\Delta_t = Z_t^1 - Z_t^2$. Then, $OutD_t^2 = R_t^2 + \alpha(Z_t^2) \geq R_t^1 + \Delta_t + \alpha(Z_t^2) \geq R_t^1 + \alpha(\Delta_t) + \alpha(Z_t^2) \overset{d}{=} OutD_t^1$. Thus, $OutD_t$ is an increasing function of (R_t, W_t) and, consequently, an increasing function of (InD_t, A_t, E_t). □

Theorem 2 *Let $\{A_t\}_{t \geq 0}$ and $\{E_t\}_{t \geq 0}$ be two mutually independent i.i.d. processes modeling respectively the packets from core network and the offered load of new packets. Under the independence hypothesis of the packets, the Markov chain of the FDL model has a monotone transition matrix in the strong stochastic sense with respect to the partial order $\preceq_*$ on the state space S'.*

Proof. Let $r, s \in S'$ such that $r \preceq_\star s$. We need to show that

$$[G_{t+1} | G_t = r] \preceq_{st} [G_{t+1} | G_t = s] \tag{11}$$

with respect to the partial order $\preceq_\star$ on S'. Then, by Proposition 2 it follows that the transition matrix of our model is $\preceq_{st}$-monotone with respect to the partial order $\preceq_\star$. Recall that (see (8)) $G_{t+1}^{(k)} = G_t^{(k+1)} + F_{t+1}^{(k,1)}, \forall k$. From (7) it follows that $F_{t+1}^{(k,1)}$ is an increasing function of $OutD_{t+1}$. Now from Lemma 4 and the fact that $InD_{t+1} = G_t^{(1)}$ it follows that $G_{t+1}^{(k)}$ is an increasing function of (G_t, A_{t+1}, E_{t+1}). As A_{t+1}, E_{t+1}, and G_t are mutually independent, the theorem follows now from Lemma 2. □

Let us remark that the proof of this theorem does not use the fact that all the E_t, and respectively A_t, are identically distributed. However, by taking E_t i.i.d. and A_t i.i.d., we assure the homogeneous property of our Markov model. Theorem 2 remains valid even in the case of a non-homogeneous Markov chain.

4 Numerical Results

We now add more assumptions to compute the bounds, check the accuracy of the method, and show that adding a few FDLs is an efficient way to avoid useless propagation delay.

4.1 Traffic Assumptions

We present the two arrival processes we have considered in this study and we explain why we have considered such processes to model traffic in an all optical switch. As the transit traffic has a higher priority than entering traffic the transit arrivals are always accepted while fresh packets may be blocked.

First, the arrivals from the core network are slotted and the number of wavelengths per link is f. As the ROMEO core switch has 4 transit links, the number of input packets A_t is upper bounded by $4f$. We assume i.i.d. batch arrivals for transit packets, and we consider two different batch distributions: a truncated Poisson distribution and a "simple batch". Let us now describe this last arrival process. We assume that we have a full batch (i.e. $4f$ packets) with probability λ and no packets with probability $(1 - \lambda)$. We do not claim that the real arrivals follow such a traffic model.

But we may expect that the real arrival process is less bursty than this batch process and that this batch process provides some kind of conservative analysis. On the other hand, the truncated Poisson distribution for the batch arrivals may be a valid approximation for core network traffic. Indeed, if some measurements show that Internet traffic is not Poisson today at TCP packet level, it is very difficult to predict the traffic properties in the future in an all optical core network which aggregates a lot of sessions.

The arrivals of new optical packets (i.e. the E_t process) follow a batch process that can be either a truncated Poisson process or a "simple batch". We assume that the traffic assumptions are constant. We use the same type of batch for core arrivals and fresh arrivals. Note, however, that the maximal batch size for fresh arrivals is $2f$. To fix the average of this batch, we consider two cases according to the mean exchange of packets between the edge and the core. As packets are independent, the average number of packets which try to leave the core is $d\mathbf{E}(A)$, where d is the probability that a packet will try to leave the network. As the core network is quite small (typically between 15 and 30 nodes), the average distance is small. Thus, the departure probability is quite large. In all experiments shown here, $d = 0.2$. The average offered load is $\mathbf{E}(E)/f$. The two different cases we consider are:

- Node in Equilibrium : $\mathbf{E}(E) = df\mathbf{E}(A)$
- Node with more inputs than outputs $\mathbf{E}(E) = 2df\mathbf{E}(A)$

We consider two set of parameters for the routing probabilities. We assume that the traffic matrix is uniform or close to uniform. To model an uniform traffic matrix, we assume that the routing probabilities $p_1, \ldots, p_c$ are all equal. The close to uniform routing model is defined by: $p_1 = 1/c + (c-1)\epsilon, \;\; p_i = 1/c - \epsilon \;\; \forall\, i \neq 1$.

Remember that the proof of our main theorem on the comparison of models does not depend on all these parameters which can take arbitrary values.

4.2 Configuration without FDLs

We first consider the case without any FDL link ($n = 0$). Clearly, $InD_t = 0$ and the transit traffic comes only from A_t. The number of packets D_t which need to leave and the number of packets in transit T_t which have to be routed are simply: $D_t \overset{d}{=} B(A_t, d)$ and $T_t = A_t - D_t$. Let us remark that all deflections are real deflections in this case, i.e. $RD_t = OutD_t$. As $m = c = 4$, we have enough bandwidth to allow all the D_t packets to leave the switch. Thus, the first term in relation (4) is zero. The number of fresh packets that might try to enter the switch at time t, E_t, is upper bounded by $4f$, and the number of new packets N_t which really enter the switch is the minimum of the offered load E_t and the available bandwidth, $N_t = min(E_t, 4f - T_t)$. Finally, $T_t + N_t$ packets are routed according to their destinations, $(U^1, \ldots, U_t^4) \overset{d}{=} M(T_t + N_t, p_1, \ldots, p_4)$. The number of deflected packets is then: $RD_t = OutD_t = \sum_{i=1}^4 (U_t^i - f)^+$. Let us remark that RD_t is upper bounded by $3f$, as $T_t + N_t$ is upper bounded by the bandwidth size $4f$. As RD_t depends only on A_t and E_t which are independent and both i.i.d., the random variables RD_t are also i.i.d. Once the distributions A and E (for A_t and E_t) are known, distribution RD can be easily computed following the above steps.

4.3 One FDL link configuration

For the case with only one FDL link of length 1, the state space of the corresponding homogeneous DTMC can be completely described by the number of locally deflected packets $F1_t$. Remark that the partial order $\preceq_*$ introduced in previous section is a total order in the case of only one FDL of length 1. The size of the state space is $f + 1$ so we can compute directly the exact values of the steady-state distribution $F1$. After solving the linear system for $F1$, we can easily compute the distribution of the number of real deflections RD in the steady-state by following the similar steps as described for the case of zero FDL links (all the distributions are in the steady-state):

$$\begin{aligned}
&D \overset{d}{=} B(A + F1, d) \quad and \quad T = A + F1 - D,\\
&N = \min(E, 5f - T - (D - 3f)^+),\\
&(U^1, \ldots, U^4) \overset{d}{=} M(T + N, p_1, \ldots, p_4),\\
&OutD = (D - 3f)^+ + \sum_{i=1}^{4} (U^i - f)^+,\\
&RD = (OutD - f)^+.
\end{aligned}$$

4.4 Deriving Upper Bounds

In order to analyze the case with two FDL links described in Section 2, we will use the monotonicity property of the FDL model (Theorem 2). We will use the homogeneous DTMC comparison theorem (Theorem 1) and design a bounding model that is stochastically larger in the $\preceq_{st}$-sense with respect to the partial order $\preceq_*$. Since our FDL model is already stochastically monotone, we only have to satisfy the comparison constraints. Let us remark here that constructing a bounding model in the sense of Theorem 1 allows us to obtain bounds for means of all increasing rewards. Notice that the reward we are interested to bound must be increasing in the sense of the same partial order on the state space we used to establish the comparison and monotonicity results (the partial order $\preceq_*$ in our case).

Using similar arguments as in Section 3, we can easily prove the following property.

Property 1 *The number of real deflections RD is an increasing reward with respect to the partial order $\preceq_\star$.*

Let the matrix P be the transition matrix of the FDL model. Let us suppose that P is ergodic and let π_P be the steady-state distribution of P. Note that this is the case for both truncated Poisson and "simple batch" arrival distributions. Suppose that we know how to build an ergodic transition matrix Q such that $P \preceq_{st} Q$ with respect to the partial order $\preceq_*$ on the state-space. Let us denote by π_Q the steady-state distribution of Q. Then from Theorem 1 and Property 1 it follows that

$$\mathbf{E}_{\pi_P}(RD) \leq \mathbf{E}_{\pi_Q}(RD).$$

Note that the bounding matrix Q does not need to be $\preceq_{st}$-monotone, as we proved in Theorem 2 the $\preceq_{st}$-monotonicity of matrix P.

Additionally, we want the bounding model to be simpler to solve. Specifically, we will construct a bounding model that has an ordinary lumpable transition matrix. Recall that a DTMC is ordinary lumpable with respect to a given partition $C_k, k = 1, 2, \ldots, M$ of the state space if its transition matrix P satisfies the following: for all states a and b which belong to the same arbitrary macro state C_k, $\sum_{j \in C_i} p_{a,j} = \sum_{j \in C_i} p_{b,j}, \text{ for all } C_i, \; 1 \leq i \leq M$.

By taking the lumped version Z of a lumpable upper bounding chain for a monotone DTMC X, we can compute bounds for increasing rewards by defining a reward on the macro-states as a maximal reward for the individual states. Denote by r the reward on the individual states and by s the new reward on the macro-states. Then, $\mathbf{E}_X(r)$ is upper bounded by $\mathbf{E}_Z(s)$, for both transient and steady-state rewards. Remark that the actual computations are done on a much smaller chain.

We will now define the partition into macro-states for the model with two FDL links. The defined partition takes into account the order $\preceq_*$. Remark that we use the chain description $(G_t^{(1)}, G_t^{(2)}) = (F1_t + F2_{t-1}, F2_t)$ (see Section 3). We will denote the reachable state space by S'. Recall that $G_t^{(1)} = F1_t + F2_{t-1}$ represents the packets in the FDL links that will return to the switch in the next slot, while $G_t^{(2)} = F2_t$ represents the packets that will return to the switch in two time slots. We choose to give more importance to the packets $G_t^{(1)}$ in the considered partition:

1. The states $x = (x_1, x_2) \in S'$ with $x_2 = 0$ are not aggregated.
2. The other states are first grouped according to the value of x_1.
3. Each group from the previous step is divided into b macro-states according to the value of x_2. For example, if $f = 128$ and $b = 8$, then for each $k = 0 \ldots 7$, we put the states with $x_2 = k * 16 + i, 1 \leq i \leq 16$ into the same macro-state.

We now show how to build the transitions to obtain a lumpable matrix which is an upper bound. We only consider here macro states which are not singleton. Assume that b divides f and let $a = f/b$. For any value of x_2, we denote by $k(x_2)$ the smallest multiple of a greater or equal to x_2. The transitions from state (x_1, x_2) are now changed and are exactly the transitions out of state $(x_1, k(x_2))$. Clearly, the chain is lumpable and it is quite simple to prove that the new matrix is $\preceq_{st}$-larger than the transition matrix of the original chain.

To check the accuracy of the method we can also compute a lower bound as follows: we change the transitions out of (x_1, x_2) to the ones from state $(x_1, k(x_2) - a)$. Note that for aggregated states $k(x_2) \geq a$.

4.5 Analysis

First we show that the method is accurate. We compute the average number of deflections for a small system with 2 FDLs and with 64 wavelengths. As the state space is small, it is possible to solve the exact problem. We just perform the computation of the bound to illustrate the accuracy of the method.

Table 1. Truncated Poisson distribution, f=64, node at equilibrium

rate	Mean real deflection		
	exact	lower bound	upper bound
0.85	5.9049e-32	5.9018e-32	5.9625e-32
0.9	1.4165e-27	1.3899e-27	1.7744e-27
0.95	3.5154e-23	2.5144e-23	1.5199e-22
0.99	2.0322e-21	1.1456e-21	1.1913e-20

We give in Table 1 the exact values and the bounds for various arrival rates for A_t process for a node in equilibrium. We assume that the routing is uniform. We only present here the results for truncated Poisson batch distribution. Clearly, the results are very accurate. We also present in Table 2 the results for a node which sends more packets than it receives, still accurate with a truncated Poisson batch process.

Table 2. Truncated Poisson distribution, f=64, node not at equilibrium

rate	Mean real deflection		
	exact	lower bound	upper bound
0.8	9.7646e-14	6.1356e-14	2.2838e-13
0.85	1.8134e-08	6.4540e-09	6.4907e-08
0.9	3.6084e-05	1.5443e-05	5.8506e-05
0.95	2.1806e-04	3.6083e-05	2.2537e-04
0.99	2.4905e-04	2.1707e-04	2.5230e-04

Then we show that our method remains accurate and efficient when we have a larger state space. We consider a system with 2 FDLs but we now have 128 wavelengths per link. We consider the model with uniform routing probabilities but the node sends more packets than it receives from the core. In Table 3 we consider a truncated Poisson distribution for the batch arrivals and give bounds on mean real deflection for various loads. For the case f=128, the size of the aggregated bounding Markov chain is 1289 for all the results presented. Thus, an algorithm like GTH can solve such a problem is a few seconds on an ordinary PC. In Table 4 we give the results for "simple batch" process. Clearly the bounds are very close.

Table 3. Truncated Poisson distribution, f=128, node not at equilibrium, block size=16

rate	Mean real deflection	
	lower b.	upper b.
0.8	1.3634e-26	2.0339e-25
0.85	4.5848e-16	4.4175e-14
0.9	1.5349e-09	1.5737e-08
0.95	6.0196e-08	7.9197e-08
0.99	8.3536e-08	9.1247e-08

Table 4. Simple batch, f=128, node at equilibrium, block size=16

rate	Mean real deflection	
	lower b.	upper b.
0.8	5.1781e-09	5.6603e-09
0.85	6.9154e-09	7.5504e-09
0.9	9.1268e-09	9.9284e-09
0.95	1.1915e-08	1.2881e-08
0.99	1.4641e-08	1.5716e-08

Let us now check the effect of the macro-state definition. We change the size of the macro-state and perform the same analysis (see Table 5) for a system with 2 FDLs and 128 wavelengths. As expected, the more blocks we keep, the more accurate the results are. However, this effect is not that important.

Table 5. Simple batch, f=128, node not at equilibrium, rate=0.7

block size	Mean real deflection	
	lower b.	upper b.
8	7.6550e-09	8.1552e-09
16	7.2840e-09	8.4278e-09
32	6.5486e-09	9.2386e-09

Finally, we check the method when we change the assumptions about the routing probabilities. We give here an example of non uniform traffic with $p_1 = 0.31$, $p_2 = 0.23$, $p_3 = 0.23$ and $p_4 = 0.23$. The bounds for the mean real deflection are still very close:

$$5.4765\ 10^{-6} \leq \mathbf{E}[RD] \leq 5.8848\ 10^{-6}.$$

As the routing is not uniform, we introduce some variance and we now have larger deflection probabilities.

In the following we study the effect of the FDLs. We compare systems without FDLs (column D0) with systems with 1 FDL (column D1) or with 2 FDLs (the last two columns) with 128 wavelengths. One can easily compute these averages exactly for 0 or 1 FDL. For two FDL configuration, we use the stochastic bounds. For all the cases presented here, the routing is uniform. In Table 6 we consider truncated Poisson arrival process and we present the results for the three configurations when we vary the load of transit packets. We only consider heavy loaded systems as we want to check the efficiency of the FDLs for the worst case scenario. Clearly, in Table 6 the configuration with 1 FDL is sufficient to avoid almost all real deflections, even if the load is 0.99.

Table 6. Truncated Poisson batch distribution, f=128, node at equilibrium

rate	Mean real deflection			
	D0	D1	D2 lower	D2 upper
0.8	1.0261e-01	2.3681e-29	<1e-39	<1e-39
0.85	6.2365e-01	9.6248e-24	<1e-39	<1e-39
0.9	2.5137e+00	8.6971e-19	<1e-39	<1e-39
0.95	6.9018e+00	3.9545e-14	<1e-39	<1e-39
0.99	9.4589e+00	1.0607e-11	<1e-39	<1e-39

This is mainly due to the low variance of the processes involved in the model, due to the independence assumption on the routing of customers. When we change the add and drop arrivals to send more packets (Table 7), the configuration with 2 FDLs is the only one which avoids almost all real deflections.

Table 7. Truncated Poisson batch distribution, f=128, node not at equilibrium

rate	Mean real deflection			
	D0	D1	D2 lower	D2 upper
0.8	8.0738e+00	1.3051e-08	1.3634e-26	2.0339e-25
0.85	1.3710e+01	5.0881e-04	4.5848e-16	4.4174e-14
0.9	1.5474e+01	1.1942e-02	1.5349e-09	1.5737e-08
0.95	1.5622e+01	1.6336e-02	6.0196e-08	7.9196e-08
0.99	1.5624e+01	1.6553e-02	8.3536e-08	9.1247e-08

Similarly, when we use a more bursty arrival process (the "simple batch" process), the configuration without FDL has poor performance. Even if the average number is small, it will introduce very large delay for the packet transport time in the core. Do not forget that a real deflection implies a delay of several tens of time slots while a local deflection only cost one or two time slots. Clearly, with 2 loops, almost all deflections are now local when the node is at equilibrium (Table 8) or not (Table 9).

Thus we have reached our objectives: a two FDL system

Table 8. Simple batch, f=128, node at equilibrium

rate	Mean real deflection			
	D0	D1	D2 lower	D2 upper
0.8	6.3136e-01	2.8297e-03	5.1781e-09	5.6603e-09
0.85	6.7080e-01	3.1944e-03	6.9154e-09	7.5504e-09
0.9	7.1024e-01	3.5813e-03	9.1268e-09	9.9283e-09
0.95	7.4968e-01	3.9903e-03	1.1915e-08	1.2880e-08
0.99	7.8123e-01	4.3334e-03	1.4641e-08	1.5715e-08

Table 9. Simple batch, f=128, node not at equilibrium

rate	Mean real deflection			
	D0	D1	D2 lower	D2 upper
0.8	1.2560e+00	8.6668e-03	2.4402e-08	2.6877e-08
0.85	1.3345e+00	7.9806e-03	3.2359e-08	3.5079e-08
0.9	1.4130e+00	7.1627e-03	4.2038e-08	4.4673e-08
0.95	1.4914e+00	6.3889e-03	5.3517e-08	5.5563e-08
0.99	1.5542e+00	5.6594e-03	6.3994e-08	6.5104e-08

allows to keep almost all deflections local. This has been shown by all the analysis we have performed.

5 Conclusion

We must stress that the result is much more general. It can be applied to more complex sets of FDLs and to various scheduling algorithms under different traffic assumptions. Theorem 2 applies as soon as Lemma 4 holds and we have an induction on $G^{(k)}$ based on increasing functions. Note that within the proof of Lemma 1 we have derived such a relation: $G^{(k)}_{t+1} = G^{(k+1)}_t + F^{(k,1)}_{t+1}$.

For more complex examples of FDLs we obtain more complex relations but they are all increasing. Indeed, these relations state how the traffic already in the loops exits from the loops and also how it enters the loops. Clearly, the more packets we have to switch, the more packets we will have to deflect. This is also true from random variables if we use stochastic comparison arguments. As all the packets which enter the FDLs must leave (FDLs are lossless), these packets will add more traffic in the future of the process. This reason is the key reason of monotonicity in this model and in its multiple generalizations. Clearly, if we change the traffic assumptions, we will still have this property about the influence of traffic. If we need a chain to modulate the arrival process, the result is still true but we must add the modulating chain in the Markovian model of the system and we must modify the order accordingly. This is quite simple if we use a partial order. However, this is much more difficult when one uses a total order on a multicomponent state space. Here, with the partial order approach, one can also generalize this approach with modulated traffic like Switched Batch Bernoulli Process.

We plan to generalize the presented model by taking explicitly into account the type of packets which are stored inside the FDLs. Such a model will have a larger state space but it is expected that it will be still monotone for a natural partial ordering which will be unfortunately more complex. This extension of the model will allow to remove the independence assumption on the traffic inside the loops.

Acknowledgment: This work was supported by the research project ROMEO from the RNRT and by the research project SMS.

References

[1] J. Bannister, F. Borgonovo, L. Fratta, and M. Gerla. A versatile model for predicting the performance of deflection routing networks. *Performance Evaluation*, 16:201–222, 1992.

[2] P. Baran. On distributed communication networks. *IEEE Transactions on Communication Systems*, CS-12:1–9, 1964.

[3] D. Barth, P. Berthomé, A. Borrero, J.-M. Fourneau, C. Laforest, F. Quessette, and S. Vial. Performance comparisons of Eulerian routing and deflection routing in a 2d-mesh all optical network. In *15th European Simulation Multiconference*, pages 887–891, 2001.

[4] D. Barth, J.-M. Fourneau, D. Nott, and D. Chiaroni. Routing and QoS in an all optical packet network. In *IEEE Workshop on Optical Burst Switching 2005, Boston*, 2005.

[5] A. Bonnoni and P. P. Prucnal. Analytical evaluation of improved access techniques in deflection routing networks. *IEEE/ACM Trans. on Networking*, 4(5), 1996.

[6] J.-M. Fourneau and N. Pekergin. An algorithmic approach to stochastic bounds. In *Performance Evaluation of Complex Systems: Techniques and Tools, Performance 2002, Tutorial Lectures*, volume 2459 of *LNCS*, pages 64–88. Springer, 2002.

[7] C. M. Gallep and E. Conforti. Reduction of SOA switching times by preimpulse step-injected current technique. *IEEE Photon. Technology Letter*, 14:902–904, 2002.

[8] W. K. Grassman, M. I. Taksar, and D. P. Heyman. Regenerative analysis and steady-state distributions for Markov chains. *Oper. Res.*, 33:1107–1116, 1985.

[9] P. Gravey, S. Gosselin, C. Guillemot, D. Chiaroni, N. Le Sauze, A. Jourdan, E. Dotaro, D. Barth, P. Berthomé, C. Laforest, S. Vial, T. Atmaca, G. Hébuterne, H. El Biaze, R. Laalaoua, E. Gangloff, and I. Kotuliak. Multiservice optical network: Main concepts and first achievements of the ROM program. *Journal of Ligthwave Technology*, 19:23–31, 2001.

[10] W. A. Massey. Stochastic orderings for markov processes on partially ordered spaces. *Math. Oper. Res.*, 12(2):350–367, 1987.

[11] A. Muller and D. Stoyan. *Comparison Methods for Stochastic Models and Risks*. Wiley, New York, NY, 2002.

[12] W. Rogiest, K. Laevens, D. Fiems, and H. Bruneel. A performance model for an asynchronous optical buffer. *Performance evaluation*, 62:313–330, 2005.

[13] M. Shaked and J. G. Shantikumar. *Stochastic Orders and their Applications*. Academic Press, San Diego, CA, 1994.