Packet Loss Analysis in Optical Packet-Switched Networks with Limited Deflection Routing

Ireneusz Szcześniak
IITiS
Polish Academy of Sciences
ul. Bałtycka 5
44-100 Gliwice, Poland
Tel: +48 32 2317319
Fax: +48 32 2317026
Email: ijs@iitis.gliwice.pl

Tadeusz Czachórski
IITiS
Polish Academy of Sciences
ul. Bałtycka 5
44-100 Gliwice, Poland
Tel: +48 32 2317319
Fax: +48 32 2317026
Email: tadek@iitis.gliwice.pl

Jean-Michel Fourneau
PRiSM
Versailles University
45 avenue des Etats-Unis
78035 Versailles Cedex, France
Tel: +33 1 39254000
Fax: +33 1 39254057
Email: jmf@prism.uvsq.fr

September 27, 2008

Abstract

We present an approximate analytical method for the evaluation of packet loss probability in synchronous optical packet-switched networks which operate under limited deflection routing with the contention resolution method based on priorities. Packets are lost because they are removed by nodes. They are removed because they experience too many deflections and stay prohibitively long in the network. Such packets have to be removed because they will be ignored by the transmission protocols (like TCP) and because the quality of their optical signal is unacceptable. Presented are results for the network in the topology of the torus of the two-dimensional grid, which operates at a steady state with the uniform load u, $u \in \langle 0, 1 \rangle$. The strength of our analysis is its novel mathematical appraoch, which is capable of providing very low packet loss probabilities. For the network composed of 100 nodes, we predict the packet loss probability as low as 10^{-9} or lower, while simulation provided results only at the order of 10^{-6}. For a given permissible packet loss probability our analysis provides the maximal network load and the number of allowed deflections. We verify the analysis with simulation in the cases for which simulation gave results.

Keywords: optical packet switching, deflection routing, performance evaluation

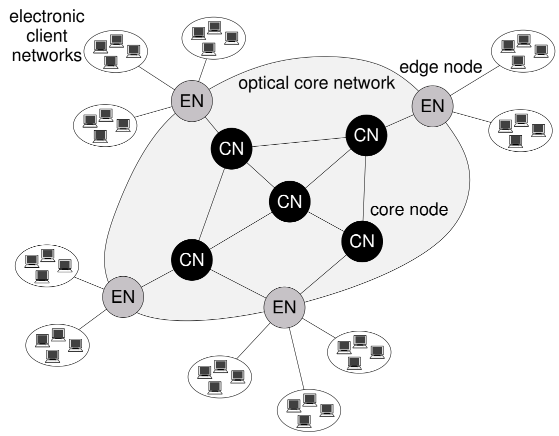

Figure 1: General architecture of an optical packet-switched network.

1 Introduction

The communication networks currently in operation exploit only a small fraction of the bandwidth available in optical fibers, because of their slow electronic components [1]. The already available technology for transparent optical packet switching (OPS) is faster [2], but several years must elapse before OPS networks start being deployed. Aside from the fact this technology still has not left laboratories, networks based on OPS are yet to be designed. New solutions (e.g.. routing strategies, congestion control mechanisms) must be invented and evaluated. Our work contributes to the evaluation of the packet loss probability (PLP) in limited deflection routing.

Several projects aimed at an OPS network, most notably [3, 4, 5]. A generic network architecture is shown in Fig. 1. A number of electronic client networks are connected to an electronic edge node. At an edge node arriving packets (mostly IP packets [6]) are aggregated and encapsulated in optical packets for more efficient utilization of the core network and for mitigation of adverse self-similarity [7, 8]. Then the packets' payload is converted from the electronic into optical domain and injected into the optical core. The packets hop between core nodes with the payload remaining optical to finally arrive at the destination edge node. At the destination edge node the payload is converted back to the electronic domain and delivered to the destination electronic client network.

Deflection routing is an attractive routing strategy for OPS networks [9], because it does not have to rely on optical buffering of packets. Nowadays optical buffering is implemented solely with optical delay lines (ODL), which are bulky and expensive and hence should be avoided whenever possible. However, with deflection routing a packet can live in the network for a long time before it eventually arrives at its destination due to a large number of deflections it can experience [10].

In limited deflection routing the number of packet's deflections (recorded in the packet header) is limited to the number S, called the threshold. A packet is removed from the network and lost when it experiences the Sth deflection. There are two reasons to employ limited deflection routing. First, packets that travel too long in the network can be removed because they anyway will be ignored by real-time applications (like video conference applications) or considered lost by the TCP protocol (data transfer) and retransmitted. Second, the optical signal quality of long traveling packets is unacceptable because of noise from optical amplifiers though which the packet has traveled.

Limited deflection routing suffers from packet loss. For a given network topology there are two parameters that influence packet loss: the threshold S, $S = 1, 2, \dots$, and the network load u, $u \in \langle 0, 1 \rangle$. The parameter u expresses the probability that a packet arrives along any link at any time slot.

The objective of our work is to find the values of u and S for which some given level of the PLP is not exceeded. The ROM project [4] aimed at the PLP on the order of 10^{-9} for their "Premium" quality service. We provide the values of S and u such that the "Premium" service is of the requested quality. We had to obtain our results with mathematical analysis, since the sought PLP is very low, and software simulators are unable to provide results of desired precision within a reasonable time limit.

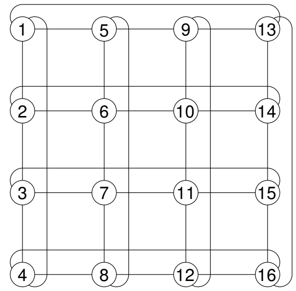

Figure 2: The torus of the two-dimensional grid with 16 nodes.

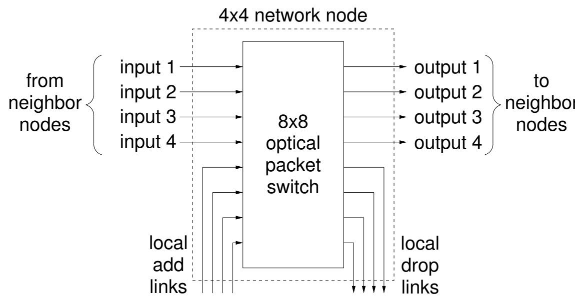

Figure 3: The network node of the 4×4 type.

As to previous works, in [11] unlimited deflection routing is examined with Markov chains. In [12] deflection routing is loopless, and for this reason the analysis there does not consider circulating packets. A detailed combinatorial analysis of packet deflection at a node is conducted in [13], but there the packet loss is caused by buffer overflows, not by limited deflection routing. Analysis of limited deflection routing in networks based on 2×2 nodes is presented in [14], which is an adapted analysis of unlimited deflection routing published in [15]. In [16] packet loss analysis is presented for asynchronous OPS networks with unlimited deflection routing. Regular networks built of 2×2 nodes running unlimited deflection routing are analyzed in [17].

2 Network Model

The network works synchronously, i.e. time is the same for every node and is divided into time slots. The network is homogeneous, i.e. every node functions both as a core node and an edge node. The node functions as an edge node for packets for which this node is not the destination. It functions as an edge node when it admits packets from the local source and drops packets to the local sink. At every node packets arrive from the source according to the Poisson process and ask for admission into the the network. They are admitted whenever there is an available slot in one of the output fibers. The destination of the packet asking for admission is uniformly distributed among every node in the network except the admitting node.

Nodes are connected with fibers that carry one wavelength only forming a network in the topology of the torus of the two-dimensional grid. The torus has l rows and l columns, and has $N = l^2$ nodes. A sample network with $N = 16$ nodes is shown in Fig. 2. This topology is regular or, using the graph theory nomenclature, node symmetric.

Each node is of the 4×4 type, i.e. each has four input and four output fibers. In this node a packet is lost only when its number of deflections has reached S. Contention resolution is based on priorities. More important packets are routed first, while the contention among equally important packets is resolved at random. We try to send a packet along one of its preferred outputs, which we chose at random.

There are four classes of packets: 1^{st}, 2^{nd}, 3^{rd}, and 4^{th}. The 1^{st}class packets are the most important, while the 4^{th}class packets are the least important. A packet is assigned a single class depending only on the number of preferred outputs it has. The preferred output is the one that follows a shortest path to the packet's destination. A packet's preferred output is also referred to as a packet's preference. For instance, let us suppose a packet residing at node 11 (Fig. 2) is destined to node 1. At node 11 the packet is of the 4th class with respect to node 1, because each of the four links at node 11 offers a shortest path to node 1. When the same packet arrives later at node 10, it is of the 3rd class with respect to node 1. A packet during its lifetime can be of various classes. The notion of packet class has been taken from [13]. In [11] it is referred to as the packet type.

We assume that traffic on all links in the whole network is statistically equivalent, and that the traffic at all nodes is statistically equivalent. One packet arrives along one link per one time slot.

We note that our model can be implemented with the KEOPS components [2], which are synchronous as opposed to the asynchronous components also being researched. In our work we chose the synchronous approach, because it receives more attention from the research community [18], and because it achieves lower contention probability in comparison with the asynchronous approach [3, 19]. Moreover, the KEOPS technology is presently the most mature [3].

Packets are of a fixed duration in time with the time slot of $t = 1.64\mu s$. Each link in our model is 200km long. The speed of light in fiber is roughly $2 \cdot 10^8$ m/s and therefore one time slot corresponds to 329.2m of fiber, while 608 time slots correspond to 200km of fiber, which is the length of a single link in our model.

3 Evaluation of a stand-alone 4×4 node

In this section we evaluate the network node when it functions as a 4×4 core node, where a packet is only switched, not added or dropped. We do not evaluate the workings of the 8×8 optical packet switch, but only the logical function of the 4×4 core node. The evaluation considers a separate node, without any reference to the network. We only assume that the traffic offered to every input link is statistically identical, and that when a packet requests an output link, then every output link is requested with an equal probability, i.e. on the average no output link is favored.

In order to evaluate the packet loss in the whole network we need to evaluate how vulnerable to deflection a packet is at a stand-alone 4×4 node. At a node a packet is of one of the four classes, and we must calculate the deflection probability for each class. The probability of deflecting a packet of class i is denoted by d_i, $i = 1, 2, 3, 4$, and all the deflection probabilities together as an array $D = \langle d_1, d_2, d_3, d_4 \rangle$. In this article an array is a one-dimensional array commonly used in programming. These values d_i we plug into the transition matrix T.

The calculation of deflection probabilities d_i for a 4×4 node is intricate, because there are many possible situations in which packets arrive and in which they are routed. Some ideas for this tedious combinatorial computation were taken from [13]. We employ the brute force approach: we consider every situation that may take place at a node, and calculate its corresponding deflection probabilities. Then we calculate deflection probabilities d_i based on the deflection probabilities for every separate situation.

The parameters for the evaluation are the network load u and the distribution of packet classes given by the array V of probabilities v_i, $V = \langle v_1, v_2, v_3, v_4 \rangle$. The probability in which a packet is of class i is given by v_i, provided that the packet arrived (conditional probability). Therefore the probability of getting a 1st class packet at any input link at any time slot is uv_1. The array V is computed in the next section.

At a node, along each of the four incoming links a packet can arrive with probability u, so at a node there can arrive from 0 to 4 packets. The probability with which k packets arrive along $n = 4$ links is given by the binomial distribution:

$$P_u(k|n)\Big|_{n=4} = \binom{4}{k} u^k (1-u)^{4-k}. \tag{1}$$

Each of the k packets is of one of the $n = 4$ different classes, and therefore we not only obtain k packets, but a multiset of packet classes. A multiset is a set where the order of the elements does not matter, but the number of repeated elements does. For instance, three packets, $k = 3$, can form combination {1st, 1st, 2nd}, i.e. two packets are of the 1st class and one of the 2nd. From now on we refer to a combination of packet classes by a *class combination*.

In combinatorics this way of choosing packet classes is called combinations with repetitions (or multichoosing) of k elements from the set of $n = 4$ elements. The number of multisets of k elements on $n = 4$ packet classes is:

Table 1: Number of class combinations and preference arrangements at a 4×4 node.

	number of packets 1	2	3	4	TOTAL
number of class combinations	4	10	20	35	69
number of preference arrangements	15	147	1195	8763	10120

Table 2: All preference combinations for a 4×4 node.

number of preferences	preference combinations
1	$\langle 1\rangle$, $\langle 2\rangle$, $\langle 3\rangle$, $\langle 4\rangle$
2	$\langle 1,1\rangle$, $\langle 1,2\rangle$, $\langle 1,3\rangle$, $\langle 1,4\rangle$, $\langle 2,2\rangle$, $\langle 2,3\rangle$, $\langle 2,4\rangle$, $\langle 3,3\rangle$, $\langle 3,4\rangle$, $\langle 4,4\rangle$
3	$\langle 1,2,3\rangle$, $\langle 1,2,4\rangle$, $\langle 1,3,4\rangle$, $\langle 2,3,4\rangle$
4	$\langle 1,2,3,4\rangle$

$$\overline{C}_n^k = C_{n+k-1}^k = \binom{n+k-1}{k}\Bigg|_{n=4} = \binom{k+3}{k}. \tag{2}$$

For instance, there are 10 class combinations of two packets, $\overline{C}_4^2 = C_5^2 = 10$: {1st, 1st}, {1st, 2nd}, {1st, 3rd}, {1st, 4th}, {2nd, 2nd}, {2nd, 3rd}, {2nd, 4th}, {3rd, 3rd}, {3rd, 4th}, and {4th, 4th}. Table 1 lists the number of class combinations for $k = 1, 2, 3, 4$.

A class combination is characterized by an array K, $K = \langle k_1, k_2, k_3, k_4\rangle$, where k_i is the number of packets of class i. Therefore $K = \langle 2, 1, 0, 0\rangle$ characterizes class combination $X = \langle 1, 1, 2\rangle$. The number k of packets equals:

$$k = \sum_{i=1}^{4} k_i. \tag{3}$$

The order of packets in a multiset is unimportant, and so multisets {1st, 1st, 2nd}, {1st, 2nd, 1st} represent the same class combination. In order to uniquely refer to a class combination we introduce the array representation of class combinations. An array $X = \langle x_1, \ldots, x_k\rangle$ represents a class combination of k elements, where x_i can be 1, 2, 3 or 4 to denote the packet class. The array is sorted in nondecreasing order, i.e. $x_i \le x_{i+1}$. For example, class combination {1st, 2nd, 1st} is represented by $X = \langle 1, 1, 2\rangle$.

A class combination is more likely to take place proportionally to the number of ways in which it can be permuted with repetitions, which equals $k!/\sum_{i=1}^{4} k_i!$.

Therefore, given array V and provided that k packets arrived, the probability with which a class combination characterized by array K arrives is equaled to:

$$p_{cmb}(K, V) = \frac{k!}{\sum_{i=1}^{4} k_i!} \prod_{i=1}^{4} v_i^{k_i} \tag{4}$$

Class combination X can refer to packets with various preferences. We number arbitrarily from 1 to 4 the output links of the 4×4 node. A packet's preferences are grouped into an array of integers sorted in nondecreasing order. For instance, $\langle 1, 4\rangle$ represents the packet with preferred outputs 1 and 4. Table 2 lists four groups of preference combinations. Preference combinations within one group are equally likely as no output is favored as a preference. The number of preference combinations with k preferences is:

$$C_4^k = \binom{4}{k}. \tag{5}$$

Algorithm 1 given u and V, calculate D

```
D = ⟨d_1, d_2, d_3, d_4⟩ ⇐ ⟨0, 0, 0, 0⟩
C = ⟨c_1, c_2, c_3, c_4⟩ ⇐ ⟨0, 0, 0, 0⟩
for k ⇐ 1 to 4 do
  p_ber ⇐ P_u(k|4)
  for each combination X = ⟨x_1, ..., x_k⟩ do
    p_cmb ⇐ p_ber · p_cmb(K, V)
    for each arrangement Y of combination X do
      p_arr ⇐ p_cmb · p_arr(K)
      A = ⟨a_1, ..., a_k⟩ ⇐ def(Y)
      for i ⇐ 1 to k do
        c_{x_i} ⇐ c_{x_i} + p_arr
        d_{x_i} ⇐ d_{x_i} + p_arr · a_i
      end for
    end for
  end for
end for
for i ⇐ 1 to 4 do
  d_{x_i} ⇐ d_{x_i} / c_{x_i}
end for

return D
```

When we replace every packet class x_i in a class combination X with an array of packet preferences y_k, then we get *preference arrangements* denoted by $Y = \langle y_1, \dots, y_k \rangle$. There may be many preference arrangements for a single class combination. For example, class combination $X = \langle 1, 1, 2 \rangle$ refers to preference arrangements $Y = \langle \langle 1 \rangle, \langle 3 \rangle, \langle 2, 4 \rangle \rangle$ and $Y = \langle \langle 2 \rangle, \langle 2 \rangle, \langle 1, 2 \rangle \rangle$ among many others.

The number of preference arrangements for a class combination described by array K is:

$$\prod_{i=1}^{4} (C_4^i)^{k_i}. \tag{6}$$

Since no output is favored, every preference arrangement is equally likely, and therefore the probability of getting a specific preference arrangement for a class combination characterized by K is:

$$p_{arr}(K) = \frac{1}{\prod_{i=1}^{4} (C_4^i)^{k_i}}. \tag{7}$$

Up to this point in this section we presented the elementary combinatorial expressions utilized in the evaluation of a 4×4 node. We derived the probability $p_{cmb}(K, V)$ with which a specific combination takes place and the probability $p_{arr}(K)$ with which an arrangement occurs.

It remains to calculate the deflection probabilities for arrangement Y. We need a function $def(Y)$ which takes arrangement Y as an argument and returns array $A = \langle a_1, \dots, a_k \rangle$, where a_i is the probability of deflecting packet y_i. The function iterates over every routing scenario (i.e. assignments of outputs to which each packet should be sent) and calculates average deflection probabilities. The detailed description of this function is beyond the page limit of this article. Please consult [20] for more details.

Finally, based on the results of this section we present Algorithm 1. The algorithm iterates over every combination and every arrangement, and computes corresponding deflection probabilities. Each arrangement contributes to the sought deflection probabilities with a specific weight.

Table 3 presents sample analysis and simulation results for a stand-alone 4×4 node. The analytical four values have been computer by Algorithm 1, while the simulation values by a simulation of the node for 10^6 time slots. We can see that the analysis and simulation results for the four classes match very closely. Note that packets of the 4^{th}class are never deflected, as they prefer every output.

Table 3: Simulation and analysis results for a separate 4×4 node with $u = 0.1$ and $V = \langle 0.4, 0.2, 0.1, 0.3\rangle$.

	1^{st} class	2^{nd} class	3^{rd} class	4^{th} class
analysis	0.007755	0.001044	0.000062	0.000000
simulation	0.007753	0.001110	0.000089	0.000000

4 Analysis

The goal of the analysis is to obtain the probability distribution of packet loss as a function of the number of hops. The parameters of the analysis are the threshold S, the load u, and the number of nodes in the network. The analysis is based on the method presented in [14], and concentrates on the tagged packet.

In the previous section the variables k and n had their frequent combinatorial meaning, i.e. k denoted the number of chosen elements and n the number of elements to choose from. However, in this section k denotes the number of a time slot, and n the number of a node. Later in this section the symbol K is ascribed a different meaning to its previous one.

We trace the tagged packet from the time slot it enters the network ($k = 0$), through consecutive time slots ($k = 1, 2, \dots$) up to the time slot when the packet cannot make more hops ($k = K$). The tagged packet makes one hop at a time slot. The value K is the upper bound on the number of hops a packet can make [14]:

$$K = S \cdot (\text{network's diameter}). \tag{8}$$

Since the network is regular and the traffic is uniform, the tagged packet is traced during its journey to node 1 instead of to every node. In a regular network every node is equivalent to every other node, which together with the assumption of traffic uniformity allows the analysis to be carried out for one node only (e.g. node 1) as the destination.

In the analysis we operate on the probabilities that the tagged packet is present at node n, $n = 1, \dots, N$, after it made k, $k = 1, \dots, K$, hops. For instance, we might be interested in the probability that the tagged packet resides at node $i = 5$ after it made $k = 3$ hops. We also need to know the probabilities which correspond to the number s, $s = 0, \dots, S - 1$, of deflections which the packets experiences during its travel. For example, we might need to know the probability that the tagged packet resides at node $n = 5$ after it made $k = 3$ hops and that it has experienced $s = 2$ deflections on the way. We express these probabilities of the packet presence by a polynomial $p_{k,n}(x)$ of the degree at most $S - 1$:

$$p_{k,n}(x) = \sum_{s=0}^{S-1} p_{k,n,s} x^s, \tag{9}$$

where the coefficient $p_{k,n,s}$ expresses the probability that the tagged packet resides at node n after it made k hops and experienced s deflections.

From the mathematical viewpoint, polynomials $p_{k,n}(x)$ belong to the ring of polynomials modulo x^S over the field $\mathbb{R}$ of real numbers. This is formally expressed in the following way:

$$p_{k,n}(x) \in \mathbb{R}[x]/x^S \tag{10}$$

We do not need, however, to think in the terms of abstract algebra. In this article a polynomial $p_{k,n}(x)$ is a tool to remember the numbers of deflections a packet has experienced and their corresponding probabilities. If we only seek the probability that the tagged packet resides at node n after k hops, we obtain it by summing the coefficients of the polynomial $p_{k,n}(x)$, which is the same as the evaluation of the polynomial at point $x = 1$: $p_{k,n}(x = 1)$. Using values for x other than one has no purpose. We do not have to treat x as a variable, since it is only an object on which we perform algebraic manipulations. This polynomial approach does not have anything in common with the polynomial approximation.

We group the polynomials into vectors $P_k = (p_{k,n}(x))$ of N elements, where element n, $p_{k,n}(x)$, relates to node n in the network:

$$P_k = \left(p_{k,1}(x), \dots, p_{k,n}(x), \dots, p_{k,N}(x)\right)^T . \tag{11}$$

$$T = \begin{pmatrix}
0 & p_1 & 0 & p_1 & p_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & p_1 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{2}p_2 & 0 & 0 & \frac{1}{2}p_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2}p_2 & 0 & 0 \\
0 & \frac{1}{3}d_1x & 0 & \frac{1}{3}d_1x & 0 & 0 & \frac{1}{3}p_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3}p_3 & 0 \\
0 & 0 & \frac{1}{2}p_2 & 0 & 0 & 0 & 0 & \frac{1}{2}p_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2}p_2 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{2}p_2 & 0 & \frac{1}{2}p_2 & \frac{1}{2}p_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{3}d_1x & 0 & 0 & \frac{1}{3}d_1x & 0 & \frac{1}{3}p_3 & 0 & 0 & \frac{1}{3}p_3 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{2}d_2x & 0 & 0 & \frac{1}{2}d_2x & 0 & \frac{1}{2}d_2x & 0 & 0 & \frac{1}{4}p_4 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{3}d_1x & \frac{1}{3}d_1x & 0 & \frac{1}{3}p_3 & 0 & 0 & 0 & 0 & \frac{1}{3}p_3 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{3}d_1x & 0 & 0 & 0 & 0 & \frac{1}{3}p_3 & 0 & \frac{1}{3}p_3 & \frac{1}{3}d_1x & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{2}d_2x & 0 & 0 & \frac{1}{2}d_2x & 0 & \frac{1}{4}p_4 & 0 & 0 & \frac{1}{2}d_2x & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & d_3x & 0 & 0 & d_3x & 0 & d_3x & 0 & 0 & d_3x & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2}d_2x & \frac{1}{2}d_2x & 0 & \frac{1}{4}p_4 & 0 & 0 & 0 & 0 & \frac{1}{2}d_2x \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2}p_2 & 0 & 0 & 0 & 0 & \frac{1}{2}p_2 & 0 & \frac{1}{2}p_2 \\
0 & \frac{1}{3}d_1x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3}p_3 & 0 & 0 & \frac{1}{3}d_1x & 0 & \frac{1}{3}p_3 & 0 \\
0 & 0 & \frac{1}{2}d_2x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{4}p_4 & 0 & 0 & \frac{1}{2}d_2x & 0 & \frac{1}{2}d_2x \\
0 & 0 & 0 & \frac{1}{3}d_1x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3}p_3 & \frac{1}{3}d_1x & 0 & \frac{1}{3}p_3 & 0
\end{pmatrix} \tag{14}$$

Vector P_k provides the complete information about the tagged packet presence after it made k hops. Vector P_0 provides the probabilities of packet presence at the moment it is admitted to the network. The packet is destined to node 1, and starts its journey from every other node with an equal probability $1/(N-1)$:

$$P_0 = (0, 1/(N-1), \ldots, 1/(N-1))^T \,. \tag{12}$$

To obtain vectors P_k we employ the following equation:

$$P_k = TP_{k-1}, k = 1, \ldots, K \tag{13}$$

where T is the transition matrix that describes the transitions the tagged packet can make when making one hop. The elements $t_{i,j}(x)$ of the matrix are also polynomials. The element $t'_{i,j}(x)$ expresses the probability of the tagged packet transition from node j to node i, provided that the tagged packet resides at node j.

The elements of the transition matrix are deduced as follows. If a link between nodes j and i does not exist, then the tagged packet cannot make a transition, and for that reason $t_{i,j}(x) = 0$. The transition of a class i packet with deflection is represented by $t_{i,j}(x) = \frac{1}{4-i}d_i x$, while the transition along a preferred link is represented by $t_{i,j}(x) = \frac{1}{i}p_i$, where $p_i = 1 - d_i$. These rules do not apply to node 1, the destination node, where the tagged packet is absorbed, and so $t_{i,1}(x) = 0$. The transition matrix for the network in Fig. 2 is given in (14), where node 1 is the destination of the tagged packet.

The probabilities of packet loss are derived from vectors P_k. The probability with which the tagged packet is lost after it makes k hops is derived only from vector P_k, and equals:

$$\sum_{n=1}^{N} p_{k,n,S-1} \cdot d_{c_n}, \tag{15}$$

where c_n is the class of the tagged packet at node n.

In the previous section the array $V = \langle v_1, v_2, v_3, v_4 \rangle$ was introduced. Its elements are calculated as follows:

$$v_i = \frac{\sum_{k=0}^{K-1} \sum_{n=2}^{N} p_{k,n}(x=1)\delta_{ic_n}}{\sum_{k=0}^{K-1} \sum_{n=2}^{N} p_{k,n}(x=1)}, i = 1, 2, 3, 4, \tag{16}$$

where δ_{ic_n} is the Kronecker delta for i and c_n.

The values of v_i are needed to calculate the values of d_i. The values of d_i are put into the transition matrix T. Having the transition matrix T, we compute the vectors P_k. Finally, having the vectors P_k, we compute the new values of v_i. Thus the values of v_i for the transition matrix are obtained by successive approximations.

The initial guess for the values of v_i is the ratio of the number of nodes at which the tagged packet is of class i to the number of all nodes. For the next approximated value of v_i the values of d_i is calculated

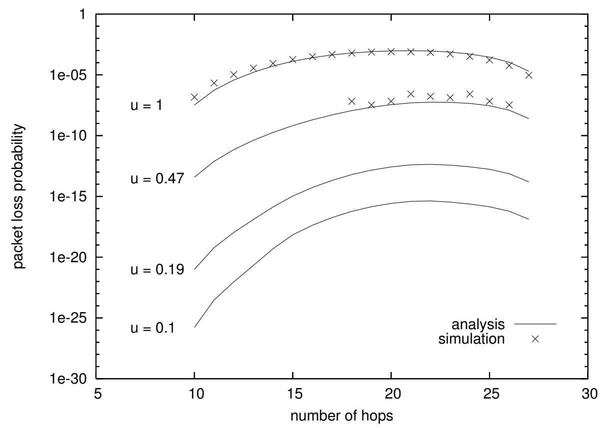

Figure 4: The PLP as a function of the number of hops for three values of the network load u and the fixed threshold $S = 10$.

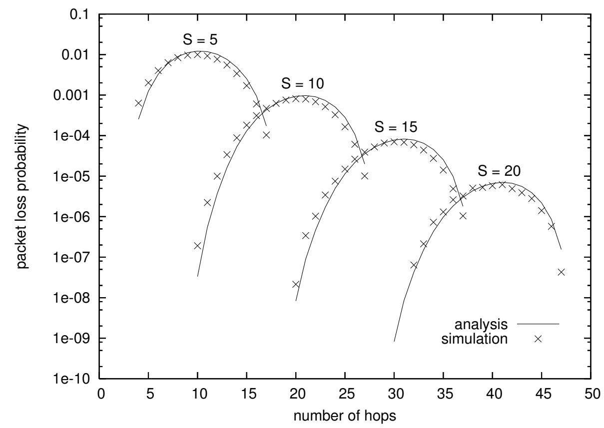

Figure 5: The PLP as a function of the number of hops for three values of the threshold S and the fixed load $u = 0.47$.

as described in Section 3, a new matrix T is generated, then vectors P_k are computed using (12), (13), and finally new and refined values of v_i are obtained from (16).

The process is repeated the necessary number of times until the desired precision of d_i is reached. In our calculations a very high precision was attained after a few iterations, where the successive values of d_i converged smoothly and fast.

5 Results

We present analysis and simulation results obtained with the OPUS software package [20]. The OPUS simulator has been implemented in OMNeT++ [21]. Figures 4 and 5 present the results of the analysis and simulation for the network with 100 nodes. The simulation results are represented by crosses, whereas the analysis results by lines.

There are two groups of simulation and analysis results. In the first group the threshold S was fixed at $S = 10$, while the load u varied, $u = 0.1, 0.47, 1.0$. The results are shown in Fig. 4. The load u influences only the PLP, not the number of hops at which packets are lost. We note that the results of packet loss presented in Fig. 4 for $u = 0.1$ are impractical, since they are very low (10^{-26}) and neither experiments nor simulation could corroborate our findings.

In the other group the load u was fixed at $u = 0.47$, while the threshold S varied, $S = 5, 10, 15$. The results are shown in Fig. 5. For different values of the threshold S packets are lost at different numbers of hops. As we increase the value of S, packets are lost later, they are more likely to be delivered to their destinations, and as the result the PLP is smaller.

Fig. 6 presents the aggregated PLPs (an aggregated PLP is obtained by adding together the PLPs for all possible numbers of hops) obtained with the analysis for the network of 100 nodes, where the threshold S and load u vary: $S = 1, 2, \ldots, 10$, $u = 0.1, 0.2, \ldots, 1.0$. In total there are 100 values of aggregated PLP that compose the curved grid: one value per a grid point. The figure also shows a plane of the allowed PLP at the value of 10^{-9}. Only a small part of the PLP surface is below the plane. For instance, the load $u = 0.1$ yields the PLP below 10^{-9} provided that $S = 7, 8, 9, 10$.

6 Conclusion

We analyzed the packet loss probability in uniformly loaded optical packet-switched networks of regular topology with limited deflection routing. No previous work has studied limited deflection routing in networks built of 4×4 nodes, which are more complicated than the extensively studied 2×2 nodes.

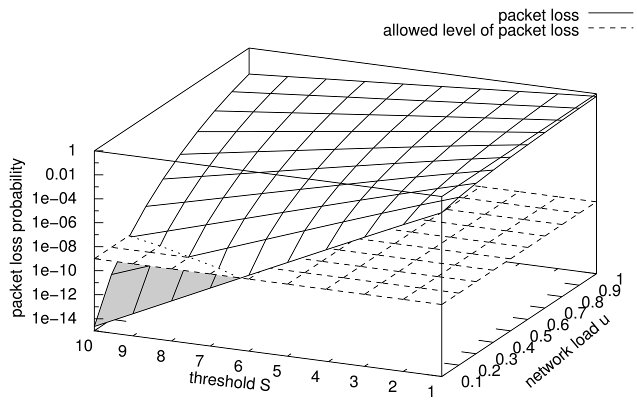

Figure 6: Probability of packet loss in a network of $n = 100$ nodes as a function of network load u and threshold S.

We found out that as the number of allowed deflections increases, the value of the packet loss probabilities decreases at the cost of a higher upper bound on the packet delivery time. Moreover, the greater the network load, the greater the packet loss probability. For a given permissible packet loss probability our analysis provides the maximal network load and the number of allowed deflections.

The strength of our analysis is its novel mathematical appraoch, which is capable of providing very low packet loss probabilities. Such low probabilities cannot be obtained with simulation. The analysis is verified by simulation only when simulation gives results. Based on this we believe that our analytical results are correct also for very low packet loss probabilities for which simulation provides no results. Our modeling technique has strong links with discrete time Markov chains and we plan to explore these links in the future.

Although our technique has been proven useful in the context of a simple network topology (the torus of the two-dimensional grid), in its present form it is inapplicable to the irregular topologies of real communication networks. It promises prospects of accurate evaluation of optical packet-switched networks with realistic characteristics, and therefore our ongoing work concentrates on extending this technique so that it can assist in designing practical networks.

Acknowledgment

This work has been partially supported by the CNRS-PAN project number 14495 and the EuroNGI Network of Excellence.

References

[1] P. Green: Progress in Optical Networking, IEEE Communications Magazine, vol. 39, no. 1, January 2001, pp. 54–61.

[2] C. Guillemot et al.: Transparent Optical Packet Switching: The European ACTS KEOPS Project Approach, IEEE/OSA Journal of Lightwave Technology, vol. 16, no. 12, December 1998, pp. 2117–2134.

[3] L. Dittmann et al.: The European IST project DAVID: A Viable Approach Toward Optical Packet Switching, IEEE Journal on Selected Areas in Communications, vol. 21, no. 7, September 2003, pp. 1026–1040.

[4] P. Gravey et al.: Multiservice Optical Network: Main Concepts and First Achievements of the ROM Program, IEEE/OSA Journal on Lightwave Technology, vol. 19, no. 1, January 2001, pp. 23–31.

[5] S. J. B. Yoo et al.: High-Performance Optical-Label Switching Packet Routers and Smart Edge Routers for the Next-Generation Internet, IEEE Journal on Selected Areas in Communications, vol. 21, no. 7, September 2003, pp. 1041–1051.

[6] M. Listanti, V. Eramo, R. Sabella: Architectural and Technological Issues for Future Optical Internet Networks, IEEE Communications Magazine, vol. 38, no. 9, September 2000, pp. 82–92.

[7] S. Yao, B. Mukherjee, S. J. B. Yoo, S. Dixit: Electrical Ingress Buffering and Traffic Aggregation for Optical Packet Switching and Their Effect on TCP-Level Performance in Optical Mesh Networks, IEEE Communications Magazine, vol. 40, no. 9, September 2002, pp. 66–72.

[8] F. Xue, S. J. B. Yoo: Self-similar Traffic Shaping at the Edge Router in Optical Packet-Switched Networks, Proc. of IEEE ICC 2002, (Anchorage, AK, USA, April 2002), vol. 4, pp. 2449–2453.

[9] L. Xu, H. G. Perros, G. Rouskas: Techniques for Optical Packet Switching and Optical Burst Switching, IEEE Communications Magazine, vol. 39, no. 1, January 2001, pp. 136–142.

[10] J. T. Brassil, R. L. Cruz: Bounds on Maximum Delay in Networks with Deflection Routing, IEEE Transactions on Parallel and Distributed Systems,vol. 6, no. 7, July 1995, pp. 724–732.

[11] T. Czachórski, J. M. Fourneau: Performance Evaluation of an Optimal Deflection Routing Algorithm on an Odd Torus, Proc. of HET-NETs 2004, (Ilkley, UK, July 2004), P63.

[12] J. P. Jue: An Algorithm for Loopless Deflection in PhotonicPacket-Switched Networks, Proc. of IEEE ICC 2002, (Anchorage, AK, USA, April 2002), vol. 5, pp. 2776–2780.

[13] M. Decina, V. Trecordi, G. Zanolini: Throughput and Packet Loss in Deflection Routing Multichannel-Metropolitan Area Networks, Proc. of IEEE GLOBECOM 1991, (Phoenix, AZ, USA, December 1991), vol. 2, pp. 1200-1208.

[14] I. Szcześniak: Analysis of a Finite Number of Deflections in Fully and Uniformly Loaded Regular Networks, Proc. of Networking 2004, (Athens, Greece, May 2004), pp. 675–686.

[15] A. S. Acampora, S. I. A. Shah: Multihop Lightwave Networks: A Comparison of Store-and-Forward and Hot-Potato Routing, IEEE Transactions on Communications, vol. 40, no. 6, June 1992, pp. 1082–1090.

[16] S. Yao, B. Mukherjee, S. Dixit: PLATO: A Generic Modeling Technique for Optical Packet-Switched Networks, International Journal on Wireless & Optical Communications, vol. 1, no. 1, June 2003, pp. 91–101.

[17] J. Bannister, F. Borgonovo, L. Fratta, M. Gerla: A Versatile Model for Predicting the Performance of Deflection-Routing Networks, Performance Evaluation, vol. 16, no. 1, November 1992, pp. 201–222.

[18] T. S. El-Bawab, J. D. Shin: Optical Packet Switching in Core Networks: Between Vision and Reality, IEEE Communications Magazine, vol. 40, no. 9, September 2002, pp. 60–65.

[19] S. Bregni, G. Guerra, A. Pattavina: Optical Packet Switching of IP Traffic, Proc. of Sixth Working Conference on Optical Network Design and Modeling, (Torino, Italy, February 2002), pp. 135-149.

[20] I. Szcześniak: OPUS: Optical Packet Switching Unified Solver, version 0.1, http://www.iitis.gliwice.pl/opus, November 2004.

[21] A. Varga: The OMNeT++ Discrete Event Simulation System, Proc. of European Simulation Multiconference, (Prague, Czech Republic, June 2001), pp. 319-324.

List of Figures

List of Tables