Efficient Packet Selection for Deflection Routing

Loubna Echabbi\textsuperscript{1}, Jean-Michel Fourneau\textsuperscript{2}, and Franck Quessette\textsuperscript{2}

\textsuperscript{1} INPT, Rabat, Morocco
\textsuperscript{2} PRiSM, Université de Versailles-Saint-Quentin, CNRS UMR 8144, Versailles, France

Abstract. Deflection routing is the usual algorithm proposed for all-optical packet networks. We study the selection part of this algorithm: how to choose the packets which must be misdirected when resources are not sufficient. Using graph and algorithmic arguments, we prove some optimal and fast selection techniques which can be easily implemented.

Keywords: deflection routing, optical packet networks.

1 Introduction

Due to the high bandwidth they could offer, all optical networks have received considerable attention during the last years (see for instance \[1\] and references therein). Optical Packet Switching is the most promising technology because it can manage flows of packets with a granularity smaller than a wavelength. One of the major drawback of this technology is the lack of large buffers which can be used to store packets waiting for a free link during the routing. Therefore routing algorithms are quite different of the algorithms designed for store and forward networks based on electronic buffers. Several packet routing strategies without intermediate storage of data packets have been designed in the literature \[2,3\] and deflection routing \[4\] is clearly the simplest solution proposed so far.

In shortest-path Deflection Routing, switches attempt to forward packets along the shortest hop path to their destinations. Each link can send a finite number of packets per time-slot (the link capacity). No packets are queued. At each slot, incoming packets have to be sent immediately to their next switch along the path. If the number of packets requesting a link is larger than the link capacity, then there is some contention. Only some of the packets will receive the link they ask for and the other ones have to be misdirected or deflected. Thus, deflected packets have to travel through longer paths to their destination. These routing algorithms are known to clearly avoid deadlocks but livelocks could occur (packets move but never reach their destination). Thus the average end to end delay is not sufficient to analyze the performance. The main question is the number of packets which are heavily deflected and which cannot arrive on time at destination. Simulations show that typically more than 1\% of the packets may suffer from an extremely large number of deflections \[5\]. Note that this may also trigger a domino effect: due to old packets in the network, new packets are not allowed to enter or experience a larger number of deflections. Deflection
routing algorithms do not explain how to choose the packets to be misdirected, or even the number of such packets. The selection and the routing decision are local choices without global knowledge and must be made in every node and at each slot. We consider that the packets in a switch are synchronous. Let $d$ be the output degree of a switch (i.e. the number of output channels). Let $v$ the number of links and $f$ the number of wavelengths multiplexed on the link. To model the routing problem, we consider a bipartite directed graph $G = (V_1, V_2, E)$. Nodes of $V_1$ represent the incoming packets while nodes of $V_2$ represent outputs. The edges represent the directions that a packet may use to follow a shortest path to its destination. Nodes of $V_1$ may have an output degree between 1 and $d$ because several links can be the starting step of a shortest path. Such a graph will be denoted as a routing configuration (see the left-upper part of Fig. 1). The nodes of $V_2$ have an arbitrary degree but every node $i$ of $V_2$ has a capacity $\lambda(i)$. This is the maximal number of packets which may use the output link represented by node $i$. The selection algorithm has to optimize the number of packets which are sent in a direction they request. If $\lambda(i) = 1$ for all $i$, the selection of the packets which join their wished output is a matching, and the optimal choice is a maximal matching. Remember that a matching of a bipartite graph is a regular subgraph of degree 1. A polynomial algorithm based on alternating paths to find a maximal matching in a bipartite graph is known for long. Although the complexity of the maximal matching algorithm is polynomial, it does not fulfill the time requirements of the optical switches (a few hundreds of nanoseconds).

![Routing configuration](image)

**Fig. 1.** Routing configuration $G$ with 5 packets and 3 links and capacity vector (upper-left), resulting routing affectation (upper-middle), associated graph $H$ (upper-right), 2D-grid and nodes at distance 3 of $D$ (lower-left), NSFNET (lower-right)