"We have a lot of empty space on our crossbar".

— Buffered Crossbars Vindicated†

E

A Modified Buffered Crossbar

In this appendix, we will prove Theorem 5.5 which states that a modified buffered crossbar can mimic a PIFO-OQ router with a fixed delay of $N/2$ time slots. In what follows, we will first show that the crux of the proof depends on showing that the size of the burst over any time period, to every output, is bounded by N cells. Then we prove the theorem for a buffered crossbar with N cells per output as shown in Figure 5.4, and we show that the theorem is trivially true for a buffered crossbar with N cells per crosspoint.

Bounding the size of the burst to any input: When header scheduling is performed, an input could receive up to N grants (one from each output) in a single output scheduling phase. Fortunately, over p consecutive phases the number of grants received by an input is bounded by $p+N-1$. This is because an input can communicate at most one header per input scheduling phase, and there are at most N outstanding headers (one for each crosspoint) per input. On the other hand, each output grants at most one header per scheduling phase. So there are at most p grants for an output over any p consecutive scheduling phases.

We are now ready to prove the following theorem:

Theorem 5.5. *(Sufficiency, by Reference) A modified buffered crossbar can emulate a PIFO-OQ router with a crossbar bandwidth of* $2NR$ *and a memory bandwidth of*

†Architectural Discussion, Campus Switching Group, Cisco Systems, California, May 2008.

$6NR$.

Proof. An input can receive at most $p+N-1$ grants over any p consecutive scheduling phases. If the input adds new grants to the tail of a *grant FIFO*, and reads one grant from the head of the grant FIFO in each scheduling phase, then the grant FIFO will never contain more than $N-1$ grants. Each time the input takes a grant from the grant FIFO, it sends the corresponding cell to the set of N crosspoints for its output. Because the grant FIFO is served once per phase, a cell that is granted at scheduling phase p will reach the output crosspoint by phase $p+N-1$.

We need to verify that the per-output buffers in the crossbar never overflow. If the crosspoint scheduler issues a grant at phase p, then the corresponding cell will reach the output crosspoint between phases p and $p+N-1$. Therefore, during scheduling phase p, the only cells that can be in the output crosspoint are cells that were granted between phases $p-N$ to $p-1$.

In a modified crossbar with N buffers per output, the buffers will never overflow, and each cell faces a delay of at most N scheduling phases, *i.e.*, $N/2$ time slots (because $S=2$). ❒

Note that in a modified crossbar with N cells per crosspoint, the buffers will also never overflow and the above theorem will also hold.