Applicability of MIMO as Metric for Frame Latency

Arjan Durresi[1], Raj Jain[2], Gojko Babic[1]

[1]Department of Computer and Information Science, The Ohio State University, 2015 Neil Ave., Columbus, OH 43210

Phone: 614-688-5610, Fax: 614-292-2911, Email: {durresi, babic}@cis.ohio-state.edu

[2]Nayna Networks, Inc., 157 Topaz St., Milpitas, CA 95035

Phone: 408-956-8000X309, Fax: 408-956-8730, Email: jain@acm.org

Abstract- **Frame latency is one of the main QoS parameters and consequently an important target in performance testing. We compared MIMO, a new metric for measuring frame latency, versus other metrics and in particular versus LCD. Analysis of examples as well as measurement results demonstrated that MIMO is less affected by test workload and is a better indicator of the switch performance, in particular for out-of-service testing. We showed how to aggregate MIMO of a network of switches. The expression of MIMO aggregation clearly indicates the contribution of each network element and their interconnections to the total frame latency.**

I. INTRODUCTION

Frame latency is of particular interest as a QoS parameter, because the latency at frame level is more likely to influence the application latency. Cell-level metrics do not very often reflect the performance as experienced by end users. For example, a video user sending 30 frames/sec would like frames to be completely delivered every 33 ms and it does not matter whether the cells belonging to a frame arrive back-to-back or regularly spaced. Therefore, it is the frame delay and its variation that matters, not cell delay.

A frame is defined here as the ATM Adaptation Layer (AAL) protocol data unit (PDU). Also as shown in [1], to reflect accurately the system performance, the test probes of the monitoring equipment should be placed at the entrance and the exit of the system to be measured.

The main goal in designing a good metric for performance testing is that the metric should be, as much as possible, representative of real network situations. Also it should be independent of switch architecture and test workload. Based on the above criteria, there are shown in [1, 2] the advantages of using MIMO (Message In Message Out) instead of other metrics like FILO, LIFO and LILO. In this paper after introducing a new definition for MIMO we focus in comparing MIMO versus LCD (last cell delay) by analyzing several examples and measurement results. This comparison is done based on the criteria of accountability, additivity, simplicity and non-negativity. Our conclusion is that MIMO is less affected by test workload and is a better indicator of the switch performance, in particular for out-of-service testing. Also we show that MIMO can be aggregated and explain the meaning of different components in the aggregated MIMO. A clear understanding of different components of total latency is very helpful in designing networks with latency constraints.

II. MIMO DEFINITION AND ZERO LATENCY SWITCH

In Fig. 1 is depicted FILO (first-bit in to the last-bit out), which is one of the metrics used to measure frame latency. Other alternative metrics such as FIFO (first-bit in to the first-bit out), LILO (last-bit in to the last-bit out), and LIFO (last-bit in to the first-bit out) latencies can be easily obtained from Fig.1. A complete analysis of these metrics is given in [1, 2], where it is shown that unfortunately none of the four above metrics is appropriate for ATM networks. So LIFO may result in negative values, FIFO does not reflect the expansion and compression of gaps on output, and FILO is strongly influenced by the frame gap pattern.

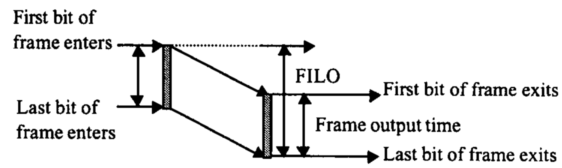

Fig. 1. Frame latencies.

We designed and introduced a new metric to measure frame latency, called MIMO [1, 2, 5], which is better suited for ATM networks. MIMO latency is the difference between the measured LILO latency of the frame through the system under test and that through an ideal system [5, 6]:

$$MIMO = LILO - LILO_0. \quad (1)$$

Where $LILO_0$ is the latency the frame experiences when passed through an ideal switch. In other words MIMO latency is the extra delay introduced by the switch under test, compared to its corresponding zero-delay switch. Fig. 2 shows the output pattern when a three ($n = 3$) back-to-back cell frame is passed through an ideal switch, which input speed is twice ($m = 2$) the output speed. The ideal switch sends out as soon possible the first bit as well as the other bits. Fig. 3 shows that the same input frame has experienced some delay by a non-ideal switch.

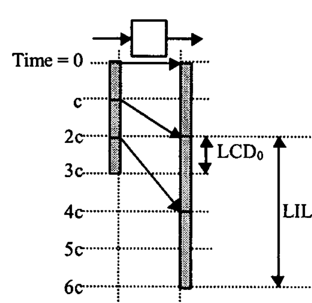

Fig. 2. Zero delay or ideal switch behavior

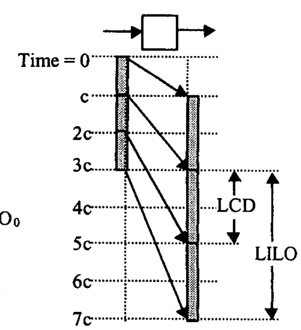

Fig. 3. LCD and LILO

MIMO latency is the difference between the measured LILO latency of the frame through the system under test and that through an ideal system [5, 6]. So for the case of Fig. 3:

$LILO_0 = 3c \quad \text{where } c = \text{cell time}$
$LILO = 4c$
$MIMO = LILO - LILO_0 = c$

For other values of *n* and *m*:

$LILO_0 = n(m-1)c$
$LILO = n(m-1)c + delay$
$MIMO = LILO - LILO_0 = delay = c$

An important point, which needs to be clarified is the influence of Input/Output speeds of the switch. When we change the in/out link speed, we are operating with a different switch. Its measured delay may be very different from that of another switch with almost the same hardware but different I/O speeds. The corresponding zero delay switch is also different. For instance, in the examples given above, LILO with $m=2$ and $m=100$ are different. So are $LILO_0$s. The difference between LILO and $LILO_0$ is MIMO. MIMO definition shows that MIMO metric doesn't depend upon the measurement loads. This doesn't mean that "delay" and MIMO value do not depend for internal implementation reasons upon the measurement loads. If there is a dependence of the delay upon the measurement loads this is an issue for the designer and manufactures of the switch.

Our goal by subtracting $LILO_0$ is to subtract out the workload dependent part from the metric definition so that when comparing multiple switches the results are not overshadowed by the workload dependent part (for example, the part that depends upon n – the frame size). An equivalent definition of MIMO is [1]:

$MIMO = FILO\text{-}FILO_0$

Where $FILO_0$ is the latency the frame would experience if it passes through the ideal switch.

III. LAST CELL DELAY (LCD) METRIC

LCD measures the latency of a frame through a switch by measuring the last-bit-in-to-first-bit-out (LIFO) latency of the last cell of the frame. For example, consider an n-cell frame passing through a switch with input speed of x and output speed of x/m, shown in Fig. 1 for $n=3$ and $m=2$. At input, each cell time is c, while at the output it is mc. In this case, the last bit of the last cell enters the switch at nc and the first bit of that cell exits at $(n-1)mc + delay$. The last cell's LIFO latency or LCD of the switch is $(n-1)mc + delay - nc$ or $c(nm-m-n) + delay$, where "*delay*" is the real delay introduced by the switch and in our example is equal to c. As shown in Fig. 3, for $n=3$ and $m=2$, LCD is 2c. For $n=100$, and $m=100$, this delay will be 9800c. So in general for this example we have:

$LCD = c(nm - m - n) + delay$

As shown in the above relation, LCD definition has an explicit dependency from the measurement load, which are number of frames ***n*** and ratio between output speed and input speed ***m***.

If the switch would introduce no delay then this is the case of an ideal switch shown in Fig. 2. In this case we have:

$delay = 0 \text{ and } LCD = c(nm - m - n)$

Let name the LCD of the ideal switch LCD_0, so we have:

$LCD_0 = c(nm - m - n)$

Now we can rewrite the expression for LCD as:

$LCD = LCD_0 + delay$

LCD_0 is a constant factor that depends on the measurement load and has nothing to do with the delay introduced by the switch. Thus it is desirable not to have LCD_0 term in the definition of the metric used to measure the switch delay. This is accomplished by MIMO latency, which definition does contain no terms explicitly depending on the measurement loads. So, MIMO latency can be obtained from LCD:

$MIMO = LCD - LCD_0$

IV. MIMO VERSUS LCD

A. Accountability

Generally, the measured performance of a system depends upon the system as well as the workload. Some metrics are highly workload dependent while others are less dependent. A metric, which depends more on the system and less on the workload, is generally preferred particularly if the users are interested in comparing the systems and not the workloads. It turns out that the LCD frame latency as defined in [3] has the undesirable property that it depends highly on the workload. This is obvious from the example shown in Fig. 2 and 3. A vendor trying to sell the switch would use small frames, say, $n=2$, and claim its LCD is zero while a competing vendor will use large frames, say, $n=100$ and show that the same switch has large delay. In this example, $LCD = c(nm - m - n) + delay$, where $c(nm - m - n)$ is the workload dependent part and $delay = c$ is the workload-independent (or switch dependent) part.

The dependency of LCD metric on the input frame configuration is not a desirable feature in comparing different switches' performances. However, if the workload is given and a user is interested in knowing the total delay introduced for that workload, then any of the measured latencies, including FILO, LILO, or LCD can be used. For example, LCD can be used as an indication of delay in an "in-service" measurement when the user is more interested in computing total delay between the entry and exit from a given network. This is exactly why ATM forum performance testing specification allows both FILO and MIMO latencies to be specified. LCD can be used in place of FILO [9].

On the other hand, for out-of-service performance testing, when the user wants to compare multiple networks or switches, MIMO is a better indicator of the switch performance since the workload part has been taken out.

B. Additivity

In [5, 6], it is shown that MIMO latency of a series of components can be computed as follows:

$$MIMO_{\Sigma} = \sum MIMO_i + \sum LILO_{0i} - LILO_{0\Sigma} \quad (2)$$

Here, $MIMO_i$ is the MIMO latency of the i^{th} component, $LILO_{0i}$ is the LILO latency of the i^{th} component if it were an ideal switch, and $LILO_{0\Sigma}$ is the LILO latency of the entire series if replaced by a black box consisting of an ideal switch. Note that computing LILO latency of an ideal switch requires knowledge of only the i/o speeds and is trivial in most cases. If the input speed is same or slower than the output speed, the LILO latency of the ideal switch is zero.

LCD is additive, so:

$$LCD_{\Sigma} = \sum LCD_i \quad (3)$$

But to ensure LCD additivity two different definitions of LCD are required, depending upon whether the component is a switch or a wire. For switches, LCD is defined as the LIFO latency of the last cell of the frame. For wires, LCD is defined as the FILO latency of the last cell of the frame. Also a wire must always follow a switch and vice versa.

There are two problems with differing definitions of LCD for switches and wires. First, the LCD latency of a zero-length (or very short length) wire cannot be by definition less than c (one cell time). So the latency of a 1 km of fiber would be c+5μs and not 5μs. Second, if we put two switches back-to-back the total LCD is not the sum of individual LCDs. Fig. 4 shows a concatenation of two switches, with each switch having an LCD of 0, but the LCD of the two switches combined is 2c. In this case, in order to arrive to the right result the user should add the LCD of the intermediate wire, which is defined as the FILO latency of the last cell. So the definition of LCD of wire is different from the definition of LCD of a switch. The conclusion is that the additivity of LCD is not simple, it requires the use of two different definitions for LCD, depending on the delay element.

C. Simplicity

Both LCD and MIMO are simple. For MIMO, the user has to know the input/output speeds of the SUT and the cell arrival pattern to subtract out the workload dependent part. For LCD, the user has to properly classify each blackbox either as a switch or a wire and to ensure that wires and switches alternate.

In cases, where the user is interested in comparing multiple systems, it is important to subtract out the workload-dependent part and so the knowledge of workload is required. However, if the user is interested only in in-service allocation of delays among various components for a given workload, LCD may be used.

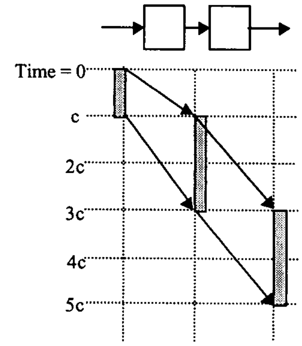

Fig. 4. Two Switches back to back

D. Non Negativity

MIMO by definition gives always non-negative values. LCD can be negative if the first bit of the cell comes out of a component before the last bit goes in. For example, a simple digital amplifier that takes in distorted waveform and outputs noise-free square bit pattern will have a negative LCD.

E. MIMO versus LCD, further Examples

In Fig. 5a [7] is shown a switch with the simple property that it takes 2c time to process every cell. The input speed is 20 times faster than the output speed. For the 3-cell sequence shown, the MIMO latency is 2c while LCD is 39c. Of course, LCD does not reflect the true behavior of the switch and can be made arbitrarily large by increasing the number of cells in the sequence.

Fig. 5b [7] shows another switch that arbitrarily delays cells by large amounts. The input speed is the same as the output one. An ideal switch in this case will be able to output each cell as it enters the switch. However, the system under test introduces delays. For a 3-cell frame, the third cell is unnecessarily delayed by 38c. If a customer wants to buy a 20x I/O speed switch, this is not a very good one because of its delay. MIMO latency correctly reflects this fact by being at 38c.

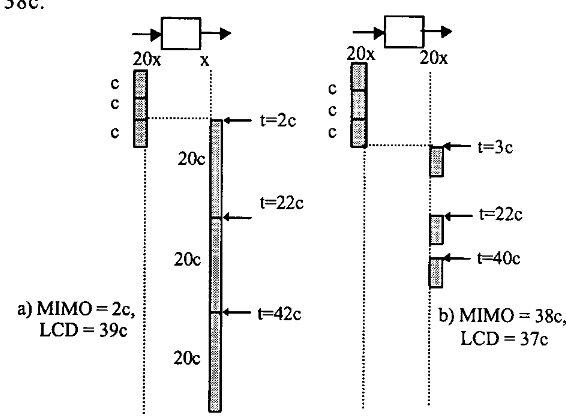

Fig.5. Comparing LCD versus MIMO

After comparing the examples of Fig. 5a and 5b, it is found that switch in Fig. 5b unnecessarily delays the cells and so is not a good one. Increased value of MIMO reflects this fact. The LCD of Fig. 5a and 5b switches are 39c and 37c respectively, making the customer believe that the second switch is even better than the first one when the contrary is true. In fact, the value of LCD can be arbitrarily increased (decreased) by simply changing the number of cells in the frames.

Fig. 6 [7] shows a concatenation of two switches. The first switch has the input speed 100 times the output speed. The second switch has the output speed 100 times the input speed. Both switches have the property that they unnecessarily delay the cell by 2c-times.

As shown, each frame has 100 cells. MIMO latency for both switches is 2c regardless of the number of cells in the frame. LCD latency for the first switch is 9802c and that for the second switch is only c, leading one to believe that the first switch is really bad. In fact, it can be made to look arbitrarily worse by simply changing the workload, that is the number of cells in the frame.

Another point to consider from Fig. 6 is that while each switch is good, the system consisting of the two switches together has a bottleneck in the middle. This bottleneck is the consequence of the mismatch among the switches' link speeds. If there is no special justification, then this bottleneck is simply the result of bad engineering. This is reflected by the combined MIMO of 9904c. If the bottleneck is improved the MIMO improves too. The combined LCD in this case is 9903c. When there is a link between the switches we should apply the additivity of LCD using different definitions for LCD of the switches and that of the wire.

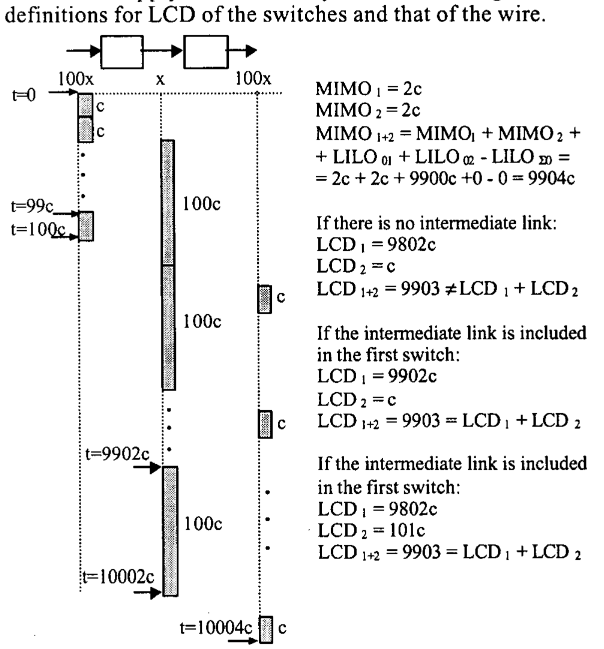

Fig. 6. Comparing LCD vs. MIMO

In [7] is suggested that the first switch includes the edge links and the second switch includes what remains of end to end connection. This may rise another problem, that of arbitrary assignment of the delay among the network elements. For example when the intermediate link is included in the first switch we get: $LCD_1 = 9902c$ and $LCD_2 = c$. When the intermediate link is included in the second switch the results are: $LCD_1 = 9802c$ and $LCD_2 = 101c$.

V. Test Measurements

In Fig. 7 [1, 8] is shown the test configuration to measure LCD and MIMO latencies when the input link rate is higher than the output link rate. The switch tested has 18 ports of 25 Mbps and four ports of 155 Mbps. We used a 155 Mbps UTP-5 link between the monitor port 1 and the switch port A1 and a 25 Mbps link between the monitor port 2 and the switch port D1. In this configurations:

- Cell Input Time = 2.83 µs
- Cell Output Time = 424[bits] / Output Link Rate = 424[bits] / 25.6 [Mbps] = 16.56 µs

In Table 1 are presented the measurement results of LCD, FILO and MIMO for eight test runs. All tests are performed with 32-cell frames. The first test uses a contiguous test frame on input, i.e. cells of the test frame were transmitted back-to-back. All other tests use discontinuous frames on input, with gaps between cells of the test frame, as indicated in the second column. Our tests do not show any significant difference if gaps include unassigned cells or cells of other frames, which leave the switch through output links other than the one used by the test frames. The third, fourth and fifth columns present measurement results for LCD, FILO and MIMO, respectively.

Analyzing the results shown in Table 1, it is clear that both LCD and FILO are heavily dependent on the input frame configuration. The results confirm the main advantage of MIMO, that it reflects only the delay introduced by the switch itself. In this case the variation of LCD and FILO could be misleading about the performance of the switch under test.

The switch latency is higher in the first 5 tests due to cell queuing. In the last three tests, the gap between the cells is large and there is no queuing. MIMO latency clearly reflects this effect.

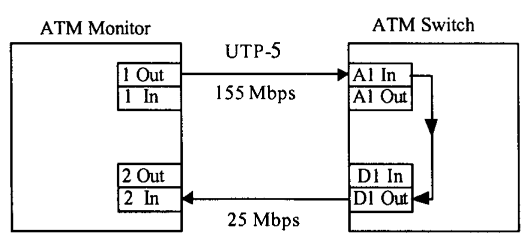

Fig. 7. Test configuration to measure LCD and MIMO

Table 1: Measurement results. (All times are in μs)

Test No.	Frame Pattern	LCD	FILO	MIMO
1	No gap	368.45	563.3	33.3
2	1-cell gaps	279.22	561.8	31.8
3	2-cell gaps	192.49	562.8	32.8
4	3-cell gaps	103.26	561.3	31.3
5	4-cell gaps	27.44	560.3	30.3
6	5-cell gaps	17.01	562.8	19.9
7	6-cell gaps	19.31	652.8	22.2
8	7-cell gaps	19.01	740.3	21.9

VI. MIMO VERSUS LCD APPLICABILITY

Based on previous sections' analysis we can conclude that MIMO and LCD have advantages and disadvantages. These metrics can be seen as complimentary because they are more appropriate in different applications. MIMO metric is better suited to compare performances of different switches in out-of-service measurements, where MIMO can precisely measure the delay introduced by the switch independent of the input frame. LCD can be used for in-service measurements and for user perceived delay. In this sense, LCD takes the place of FILO or LILO as currently specified in the ATM Forum testing document [9].

If LCD is used in place of LILO, the following relationship may be useful for aggregation:

For any one component:

$$MIMO = LCD - LCD_0 \tag{4}$$

This relationship applies for both switches and wires, that is, LCD can be FILO or LIFO.

For a network of switches we will have the following expressions:

$$LCD_\Sigma = \Sigma LCD_i \tag{5}$$

$$MIMO_\Sigma = \Sigma LCD_i - LCD_{0\Sigma} \tag{6}$$

Where $LCD_{0\Sigma}$ is the LCD of the entire series if it is replaced by a black box consisting of an ideal switch.

Furthermore the expression:

$$MIMO_\Sigma = \Sigma MIMO_i + \Sigma LCD_{0i} - LCD_{0\Sigma} \tag{7}$$

can be used to relate the delay of the network to the delays of individual switches measured in performance testing by $MIMO_i$ and the mismatch between input and output link speeds given by LCD_{0i}.

VII. CONCLUSIONS

We analyzed and compared the use of MIMO and LCD for measuring ATM frame latency. MIMO is less affected by test workload and is a better indicator of the delay introduced by the system itself. Consequently, MIMO is better suited for out-of-service measurements, when the user is interested in comparing multiple switches, independently of the workload. LCD can be used (in place of FILO or LILO) for in-service measurement where the user is interested in total delay for a given workload.

REFERENCES

[1] Arjan Durresi, Raj Jain, Gojko Babic, "Experience with ATM Switch Performance Testing", *Proceedings IEEE International Conference on Networks (ICON'99)*, Brisbane, Australia, September 28 - October 1, 1999, pp.143-150, http://www.cis.ohio-state.edu/~durresi/papers/perfexp.html

[2] Gojko Babic, Raj Jain, Arjan Durresi, "ATM Performance Testing and QoS Management" in F. Golshani, Ed., *The IEC ATM Handbook*, International Engineering Consortium, Chicago, IL, 2000, in press. http://www.cis.ohio-state.edu/~jain/papers/exp_book.html

[3] K. Glossbrenner, F. Kaudel, "ATM Layer Reference Events for Monitoring Frames," *ATM Forum contribution* 99-0282, April 1999.

[4] K. Glossbrenner, F. Kaudel, "ATM Layer Reference Events for Monitoring Frames," *Contribution to ANSI Committee T1A1.3*, May 3,1999

[5] Arjan Durresi, Raj Jain, and Gojko Babic, "Frame Delay Through ATM Switches and networks: MIMO Latency and its aggregation," *Contribution to ANSI Committee T1A1.3/99-019*, May 3,1999, http://www.cis.ohio-state.edu/~durresi/ansi/0a13019.html/

[6] Arjan Durresi, Raj Jain, and Gojko Babic, "Aggregation of MIMO Latency", *ATM Forum contribution 99-0243*, April1999, http: //www.cis.ohio-state.edu/ ~durresi/atmf/a99-243.htm.

[7] K. Glossbrenner, F. Kaudel, "Analysis of MIMO," *Contribution to ANSI Committee T1A1-3/99-035*, July 14, 1999

[8] Arjan Durresi, Raj Jain, Gojko Babic, and Bruce Northcote "Methodology for Implementing Scalable Test Configurations in ATM Switches, " *Proceedings IEEE International Conference on Computer Communication and Networks (ICCCN'99)*, Boston, MA, October 11-13, 1999, pp. 628-633, http://www.cis.ohio-state.edu/~durresi/papers/scalab.html/

[9] ATM Forum Performance Testing Specification, *AF-TEST-TM-0131.000*, October, 1999