****

ATM Forum Document Number: BTD-TEST-TM-PERF.00.03 (96-0810R6)

****

Title: ATM Forum Performance Testing Specification - Baseline Text

****

Abstract: This baseline document includes all text related to performance testing that has been agreed so far by the ATM Forum Testing Working Group.

****

Source:
Raj Jain, Gojko Babic, Arjan Durresi, Justin Dolske.
The Ohio State University

Raj Jain is now at Washington University in Saint Louis, jain@cse.wustl.edu http://www.cse.wustl.edu/~jain/

The presentation of this contribution at the ATM Forum is sponsored by NASA.

****

Date: September 1997

****

Distribution: ATM Forum Technical Working Group Members (AF-TEST, AF-TM)

****

****
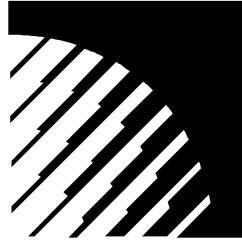
The ATM Forum

Technical Committee

ATM Forum Performance Testing Specification

7/07/97 4:58PM

BTD-TEST-TM-PERF.00.02 (96-0810R5) **September 1997**

BTD-TEST-TM-PERF.00.03 (96-0810R6)

ATM Forum Performance Testing Specifications

Version 1.0, ~~July~~September 1997

Table of Contents

1. ~~Introduction~~

1. Introduction

Performance testing in ATM deals with the measurement of the level of quality of a system under test (SUT) or an interface under test (IUT) under well-known conditions. The level of quality can be expressed in the form of metrics such as latency, end-to-end delay, effective throughput. Performance testing can be carried at the end-user application level (e.g., ~~ftp, nfs),~~FTP, NFS), at or above the ATM layers (e.g., cell switching, signaling, etc.). Performance testing also describes in details the procedures for testing the IUTs in the form of test suites. These procedures are intended to test the SUT or IUT and do not assume or imply any specific implementation or architecture of these systems.

This document highlights the objectives of performance testing and suggests an approach for the development of the test suites.

1.1. Scope

Asynchronous Transfer Mode, as an enabling technology for the integration of services, is gaining an increasing interest and popularity. ATM networks are being progressively deployed and in most cases a smooth migration to ATM is prescribed. This means that most of the existing applications can still operate over ATM via service emulation or service interworking along with the proper adaptation of data formats. At the same time, several new applications are being developed to take full advantage of the capabilities of the ATM technology through an Application Protocol Interface (API).

While ATM provides an elegant solution to the integration of services and allows for high levels of scalability, the performance of a given application may vary substantially with the IUT or the SUT utilized. The variation in the performance is due to the complexity of the dynamic interaction between the different layers. For example, an application running with TCP/IP stacks will yield different levels of performance depending on the interaction of the TCP window flow control mechanism and the ATM network congestion control mechanism used. Hence, the following points and recommendations are made. First, ATM adopters need guidelines on the measurement of the performance of user applications over different systems. Second, some functions above the ATM layer, e.g., adaptation, signaling, constitute applications (i.e. IUTs) and as such should be considered for performance testing. Also, it is essential that these layers be implemented in compliance with the ATM Forum specifications. Third, performance testing can be executed at the ATM layer in relation to the QoS provided by the different service categories. Finally, because of the extensive list of available applications, it is preferable to group applications in generic classes. Each class of applications requires different testing environment such as metrics, test suites and traffic test patterns. It is noted that the same application, e.g., ftp, can yield different performance results depending on the underlying layers used (TCP/IP to ATM versus TCP/IP to MAC layer to ATM). Thus performance results should be compared based on the utilization of the same protocol stack.

Performance testing is related to user perceived performance of ATM technology. In other words, goodness of ATM will be measured not only by cell level performance but also by frame-level performance and performance perceived at higher layers.

Most of the quality of Service (QoS) metrics, such as cell transfer delay (CTD), cell delay variation (CDV), cell loss ratio (CLR), and so on, may or may not be reflected directly in the performance perceived by the user. For example, while comparing two switches if one gives a CLR of 0.1% and a frame loss ratio of 0.1% while the other gives a CLR 1% but a frame loss of 0.05%, the second switch will be considered superior by many ~~users,~~users.

ATM Forum and ITU have standardized the definitions of ATM layer QoS metrics. We need to do the same for higher level performance metrics. Without a standard definition, each vendor will use their own definition of common metrics such as throughput and latency resulting in a confusion in the market place. Avoiding such a confusion will help buyers eventually leading to better sales resulting in the success of the ATM technology.

The initial work at the ATM Forum will be restricted to the native ATM layer and the adaptation layer. Any work on the performance of the higher layers is being deferred for further study.

1.2. Goals of Performance Testing

The goal of this effort is to enhance the marketability of ATM technology and equipment. Any additional criteria that helps in achieving that goal can be added later to this list.

a. The ATM Forum shall define metrics that will help compare various ATM equipment in terms of performance.
b. The metrics shall be such that they are independent of switch or NIC architecture.
   (i)   The same metrics shall apply to all architectures.
c. The metrics can be used to help predict the performance of an application or to design a network configuration to meet specific performance objectives.
d. The ATM Forum will develop a precise methodology for measuring these metrics.
   (i)   The methodology will include a set of configurations and traffic patterns that will allow vendors as well as users to conduct their own measurements.
e. The testing shall cover all classes of service including CBR, ~~VBRRT, VBRNRT,~~ rt-VBR, nrt-VBR, ABR, and UBR.
f. The metrics and methodology for different service classes may be different.
g. The testing shall cover as many protocol stacks and ATM services as possible.
   (i)   As an example, measurements for verifying the performance of services such as IP, Frame Relay and SMDS over ATM may be included.
h. The testing shall include metrics to measure performance of network management, connection setup, and normal data transfer.
i. The following objectives are set for ATM performance testing:
   (i)   Definition of criteria to be used to distinguish classes of applications.
   (ii)   Definition of classes of applications, at or above the ATM Layer, for which performance metrics are to be provided.

(iii)   Identification of the functions at or above the ATM Layer which influence the perceived performance of a given class of applications. Example of such functions include traffic shaping, quality of service, adaptation, etc. These functions need to be measured in order to assess the performance of the applications within that class.

(iv)   Definition of common performance metrics for the assessment of the performance of all applications within a class. The metrics should reflect the effect of the functions identified in (iii).

(v)   Provision of detailed test cases for the measurement of the defined performance metrics.

1.3. Non-Goals of Performance Testing

a. The ATM Forum is not responsible for conducting any measurements.

b. The ATM Forum will not certify measurements.

c. The ATM Forum will not set thresholds such that equipment performing below those thresholds are called "unsatisfactory."

d. The ATM Forum will not establish any requirement that dictates a cost versus performance ratio.

e. The following areas are excluded from the scope of ATM performance testing:

(i)   Applications whose performance cannot be assessed by common implementation independent metrics. In this case the performance is tightly related to the implementation. An example of such applications is network management, whose performance behavior depends on whether it is a centralized or a distributed implementation.

(ii)   Performance metrics which depend on the type of implementation or architecture of the SUT or the IUT.

(iii)   Test configurations and methodologies which assume or imply a specific implementation or architecture of the SUT or the IUT.

(iv)   Evaluation or assessment of results obtained by companies or other bodies.

(v)   Certification of conducted measurements or of bodies conducting the measurements.

1.51.4. Terminology

The following definitions are used in this ~~document.~~document:

- *Implementation Under Test (IUT)*: The part of the system that is to be tested.
- *Metric*: a variable or a function that can be measured or evaluated and which reflects quantitatively the response or the behavior of an IUT or an SUT.
- *System Under Test (SUT)*: The system in which the IUT resides.
- *Test Case*: A series of test steps needed to put an IUT into a given state to observe and describe its behavior.
- *Test Suite*: A complete set of test cases, possibly combined into nested test groups, that is necessary to perform testing for an IUT or a protocol within an IUT.

~~*1.6*~~1.5. Abbreviations

ISO	International Organization for Standardization
IUT	Implementation Under Test
NP	Network Performance
NPC	Network Parameter Control
PDU	Protocol Data Unit
PVC	Permanent Virtual Circuit
QoS	Quality of Service
SUT	System Under Test
SVC	Switched Virtual Circuit
WG	Working Group

2. Classes of Application

Developing a test suite for each existing and new application can prove to be a difficult task. Instead, applications should be grouped into categories or classes. Applications in a given class have similar performance requirements and can be characterized by common performance metrics. This way, the defined performance metrics and test suites will be valid for a range of applications. Classes of application can be defined based on one or a combination of criteria. The following criteria can be used in the definition of the classes:

(i)   ~~(i)~~   Time or delay requirements: real-time versus non real-time applications.
(ii)   ~~(ii)~~   Distance requirements: LAN versus WAN applications.
(iii)   ~~(iii)~~   Media type: voice, video, data, or multimedia application.
(iv)   ~~(iv)~~   Quality level: for example desktop video versus broadcast quality video.
(v)   ~~(v)~~   ATM service category used: some applications have stringent performance requirements and can only run over a given service category. Others can run on several service categories. An ATM service category relates application aspects to network functionalities.
(vi)   ~~(vi)~~   Others to be determined.

2.1. Performance Testing Above the ATM Layer

Performance metrics can be measured at the user application layer, and sometimes at the transport layer and the network layer, and can give an accurate assessment of the perceived performance. Since it is difficult to cover all the existing applications and all the possible combinations of applications and underlying protocol stacks, it is desirable to classify the applications into classes. Performance metrics and performance test suites can be provided for each class of applications.

The perceived performance of a user application running over an ATM network is dependent on many parameters. It can vary substantially by changing an underlying protocol stack, the ATM

service category it uses, the congestion control mechanism used in the ATM network, etc. Furthermore, there is no direct and unique relationship between the ATM Layer Quality of Service (QoS) parameters and the perceived application performance. For example, in an ATM network implementing a packet level discard congestion mechanism, applications using TCP as the transport protocol may see their effective throughput improved while the measured cell loss ratio may be relatively high. In practice, it is difficult to carry out measurements in all the layers that span the region between the ATM Layer and the user application layer given the inaccessibility of testing points. More effort needs to be invested to define the performance at these layers. These layers include adaptation, signaling, etc.

2.2. Performance Testing at the ATM Layer

The notion of application at the ATM Layer is related to the service categories provided by the ATM service architecture. The Traffic Management Specification, version 4.0, specifies five service categories: CBR, rt-VBR, nrt-VBR, UBR, and ABR. Each service category defines a relation of the traffic characteristics and the Quality of Service (QoS) requirements to network behavior. There is an assessment criteria of the QoS associated with each of these parameters. These are summarized below.

QoS PERFORMANCE PARAMETER	QoS ASSESSMENT CRITERIA
Cell Error Ratio	Accuracy
Severely-Errored Cell Block Ratio	Accuracy
Cell Misinsertion Ratio	Accuracy
Cell Loss Rate	Dependability
Cell Transfer Delay	Speed
Cell Delay Variation	Speed

A few methods for the measurement of the QoS parameters are defined in [2]. However, detailed test cases and procedures, as well as test configurations are needed for both in-service and out-of-service measurement of QoS parameters. An example of test configuration for the out-of-service measurement of QoS parameters is given in [1].

Performance testing at the ATM Layer covers the following categories:

(i)   In-service and out-of-service measurement of the QoS performance parameters for all five service categories (or application classes in the context of performance testing): CBR, rt-VBR, nrt-VBR, UBR, and ABR. The test configurations assume a non-overloaded SUT.

(ii)   (ii) Performance of the SUT under overload conditions. In this case, the efficiency of the congestion avoidance and congestion control mechanisms of the SUT are tested.

In order to provide common performance metrics that are applicable to a wide range of SUT's and that can be uniquely interpreted, the following requirements must be satisfied:

(i)   Reference load models for the five service categories CBR, rt-VBR, nrt-VBR, UBR, and ABR, are required. Reference load models are to be defined by the Traffic Management Working Group.

(ii)   ~~(ii)~~   Test cases and configurations must not assume or imply any specific implementation or architecture of the SUT.

3. Performance Metrics

In the following description System Under Test (SUT) refers to an ATM switch. However, the definitions and measurement procedures are general and may be used for other devices or a network consisting of multiple switches as well.

~~3.1 THROUGHPUT~~

3.1. Throughput

3.1.1. Definitions

There are three frame-level throughput metrics that are of interest to a ~~user.~~user:

- ~~i. Lossless~~**Loss-less throughput** - It is the maximum rate at which none of the offered frames is dropped by the SUT.
- ~~ii.~~**Peak throughput** - It is the maximum rate <u>at which the SUT operates</u> regardless of frames dropped ~~at which the SUT operates~~. The maximum rate can actually occur when the loss is not zero.
- ~~iii.~~**Full-load throughput** - It <u>is</u> the rate at which <u>the</u> SUT operates when the input links are loaded at 100% of their capacity.

A model graph of throughput vs<u>.</u> input rate is shown in Figure 3.1. Level X defines the loss-less throughput, level Y defines the peak throughput and level Z defines the full-load throughput.
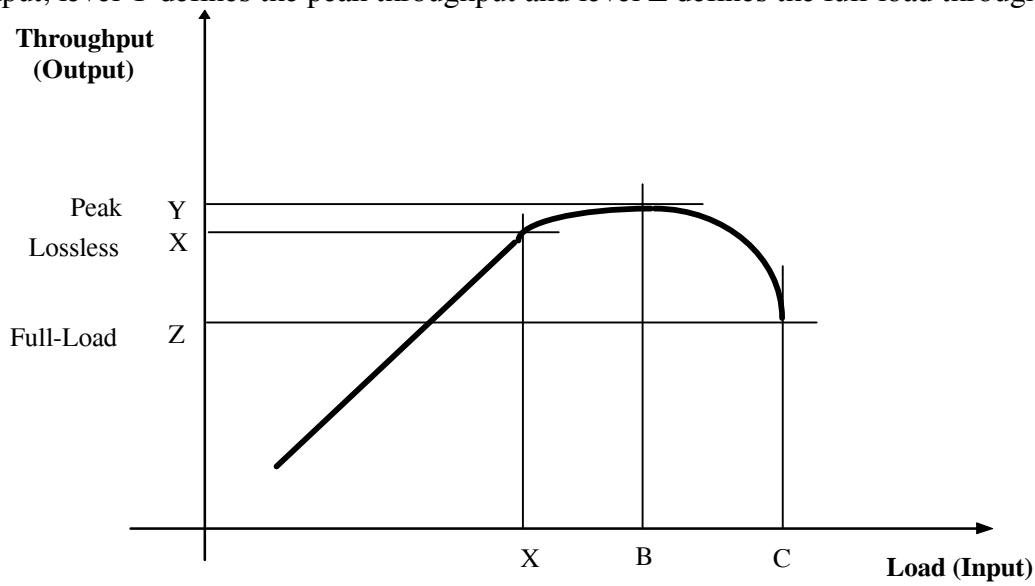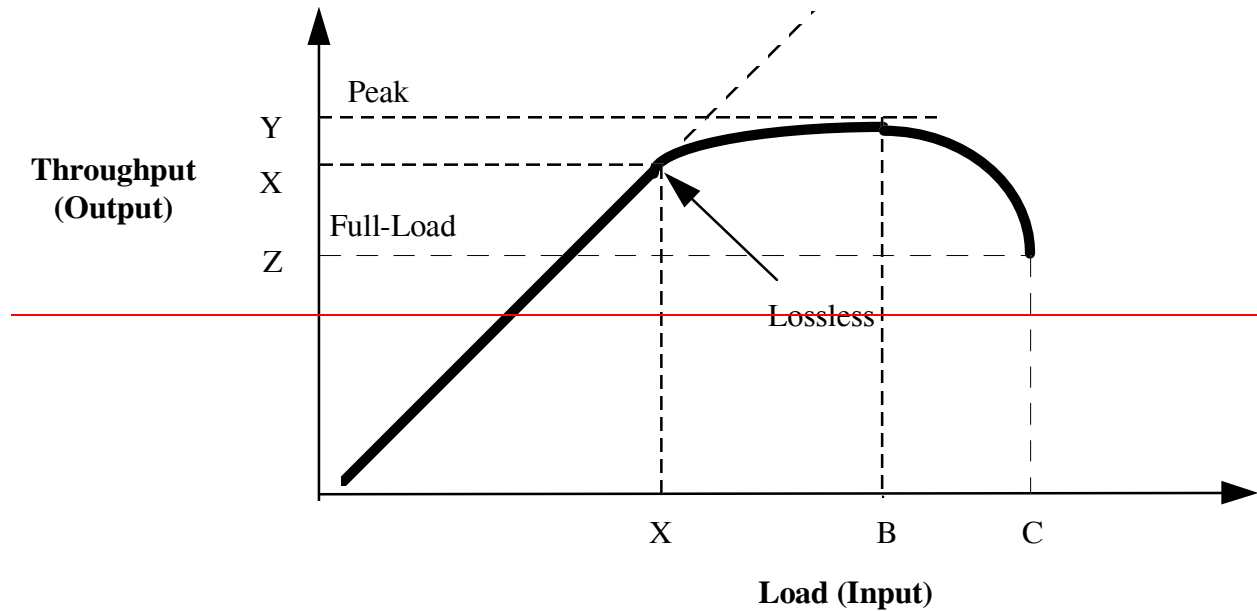
Figure 3.1: Peak, loss-less and full-load throughput

The loss-less throughput is the highest load at which the count of the output frames equals the count of input frames. ~~Peak~~The peak throughput is the maximum throughput that can be achieved in spite of the losses. ~~Full-load~~The full-load throughput is the throughput of the system at 100% load on input links. Note that the peak throughput may equal the loss-less throughput in some cases.

Only frames that are received completely without errors are included in frame-level throughput computation. Partial frames and frames with CRC errors are not included.

3.1.2. Units

Throughput should be expressed in the effective bits/sec, counting only bits from frames excluding the overhead introduced by the ATM technology and transmission systems. ~~bits/sec.~~
This is preferred over specifying it in frames/sec or cells/sec. Frames/sec requires specifying the frame size. The throughput values in frames/sec at various frame sizes cannot be compared without first being converted into bits/sec. Cells/sec is not a good unit for frame-level performance since the cells aren't seen by the user.
~~3.1.3 Statistical Variations~~
~~The tests should be run NRT times for TRT seconds each. Here NRT (number of repetitions for throughput tests)~~
~~and TRT (time per repetition for throughput tests) are parameters. These and other such parameters and their default~~
~~values are listed later in Table 3.2.~~
~~If T_i is the throughput in ith run, The mean and standard errors of the measurement should be computed as follows:~~
~~Mean throughput = (Σ T_i)/n~~
~~Standard deviation of throughput = (Σ (T_i-Mean throughput)^2)/(n-1)~~
~~Standard error = Standard deviation of throughput/$\sqrt{n}$~~
~~Given mean and standard errors, the users can compute an 100(1-α) percent confidence interval as follows:~~
~~100(1-α)-percent confidence interval = (mean - z × std error, mean + z × std error)~~

Here, z is the (1-α/2) quantile of the unit normal variate. For commonly used confidence levels, the quantile values are as follows:

Confidence	α	Quantile
90%	0.1	1.615
99%	0.01	2.346
99.9%	0.001	3.291

3.1.4 Traffic Pattern

The input traffic will consist of frames of length FSA bytes each. Before starting the throughput measurements, all required VCs will be set up (for an n-port SUT) in one of the following four configurations (see Figure 3.2):.
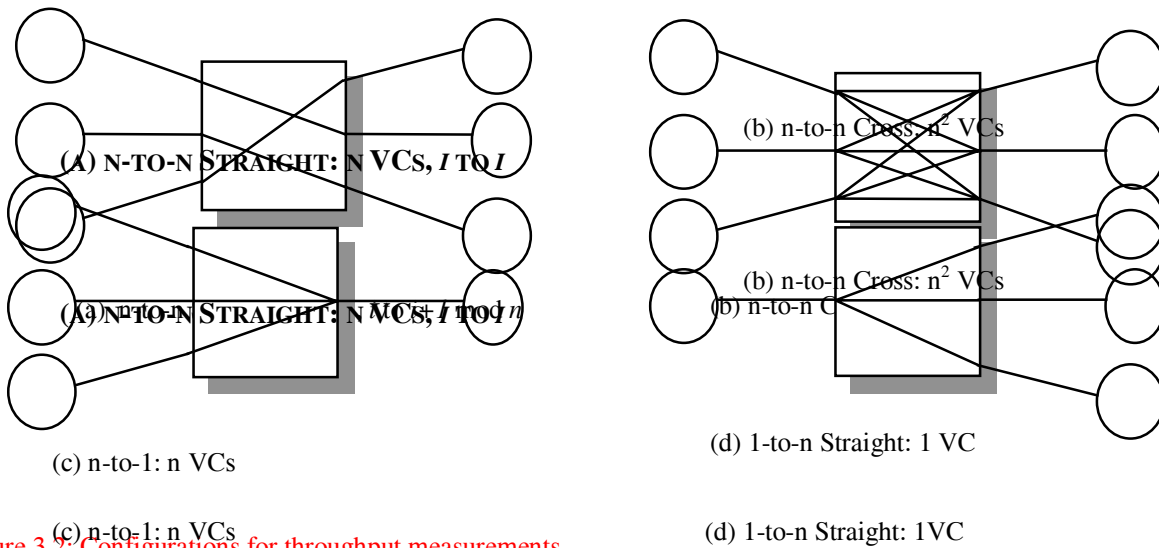

(A) N-TO-N STRAIGHT: N VCS, *I* TO *I*

(A) N-TO-N STRAIGHT: N VCS, *I* mod *n*

(c) n-to-1: n VCs

(b) n-to-n Cross: n² VCs

(b) n-to-n Cross: n² VCs

(b) n-to-n C

(d) 1-to-n Straight: 1 VC

(c) n-to-1: n VCs   (d) 1-to-n Straight: 1VC

Figure 3.2: Configurations for throughput measurements

1. **n-to-n straight**: All frames input from port i exit to port i+1 modulo n. This represents almost no path interference among the VCs. Total n VCs.
2. **n-to-n cross**: Input from port each port is divided equally to exit on each of the n output ports. Total n² VCs.
3. **n-to-1**: Input from all ports is destined to one output port. Total n VCs.
4. **1-to-n**: Input from one port is multicast to all output ports. Total 1 VC.
5. **n-to-n multicast**: Input from all ports is multicast to all output ports. Total n VCs.

The frames will be delivered to the layer under test equally spaced at a given input rate. The rate at which the cells reach SUT may vary depending on the service used. For example, for ABR traffic, the allowed cell rate may be less than the link rate for some configurations.

At each value of the input rate to the layer under test, the total number of frames sent to SUT and received from SUT are recorded. The input rate is computed based on the time from the first bit of first frame enters the SUT to the last bit of the last frame enters SUT. The throughput (output rate) is computed based on the time from the first bit of the first frame exits to the last bit of the last frame exits SUT.

If the input frame count and the output frame count are the same then the input rate is increased and the test is conducted again. The lossless throughput is the highest throughput at which the count of the output frames equals the count of the input frames. If the input rate is increased even further, although some frames will be lost, the throughput may increase till it reaches the peak throughput value after which the further increase in input rate will result in a decrease in the throughput. The input rate is increased further till it reaches 100% of the link rate. The full-load throughput is then recorded.

(b) n-to-n Cross: n² VCs

(e) n-to-n Multi to multipoint VCs

3.1.5 Background Traffic

The tests can be conducted under two conditions - with background traffic and without background traffic. Higher priority traffic like VBR can act as background traffic for the experiment. Further details of measurements with background traffic (multiple service classes simultaneously) are for further study. Until then all testing will be done without any background traffic.

3.1.6 Guidelines For Using This Metric
To be specified.

3.2 FRAME LATENCY
3.2.1 Definition

3.1.3. Statistical Variations

There is no need for obtaining more than one sample for any of the three frame-level throughput metrics. Consequently, there is no need for calculation of the means and/or standard deviations of throughputs.

3.1.4. Measurement Procedures

Before starting measurements, a number of VCCs (or VPCs), henceforth referred to as "foreground VCCs", are established through the SUT. Foreground VCCs are used to transfer only the traffic whose performance is measured. That traffic is referred as the foreground traffic. Characteristics of foreground traffic are specified in 3.1.5.

The tests can be conducted under two conditions:
- without background traffic;
- with background traffic;

Procedure without background traffic

The procedure to measure throughput in this case includes a number of test runs. A test run starts with the traffic being sent at a given input rate over the foreground VCCs with early packet discard disabled (if this feature is available in the SUT and can be turned off). The average cell transfer delay is constantly monitored. A test run ends and the foreground traffic is stopped when the average cell transfer delay has not significantly changed (not more than 5%) during a period of at least 5 minutes.

During the test run period, the total number of frames sent to the SUT and the total number of frames received from the SUT are recorded. The throughput (output rate) is computed based on the duration of a test run and the number of received frames.

If the input frame count and the output frame count are the same then the input rate is increased and the test is conducted again.

The loss-less throughput is the highest throughput at which the count of the output frames equals the count of the input frames.

The input rate is then increased even further (with early packet discard enabled, if available). Although some frames will be lost, the throughput may increase till it reaches the peak throughput value. After this point, any further increase in the input rate will result in a decrease in the throughput.

The input rate is finally increased to 100% of the link input rates and the full-load throughput is recorded.

Procedure with background traffic

Measurements of throughput with background traffic are under study.

3.1.5. Foreground Traffic

Foreground traffic is specified by the type of foreground VCCs, connection configuration, service class, arrival patterns, frame length and input rate.

Foreground VCCs can be permanent or switched, virtual path or virtual channel connections, established between ports on the same network module on the switch, or between ports on different network modules, or between ports on different switching fabrics.

A system with n ports can be tested for the following connection configurations:
- n-to-n straight,
- n-to-$(n-1)$ full cross,
- n-to-m partial cross, $1 \leq m \leq n-1$,
- k-to-1, 1<k<n,
- 1-to-$(n-1)$ multicast,
- n-to-$(n-1)$ multicast.

Different connection configurations are illustrated in Figure 3.2, where each configuration includes one ATM switch with four ports, with their input components shown on the left and their output components shown the right.
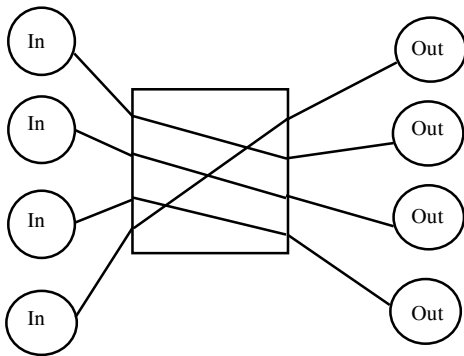
a. n-to-n straight: *n* VCCs; *n*=4
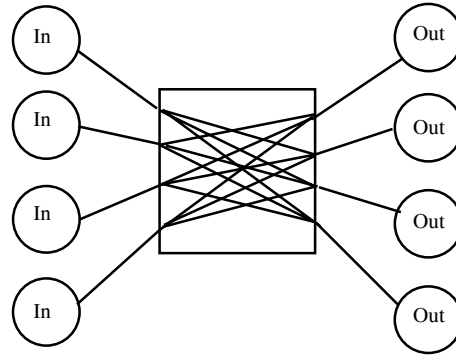
b. n-to-(n−1) full cross: *n*× (*n*−1) VCCs; *n*=4
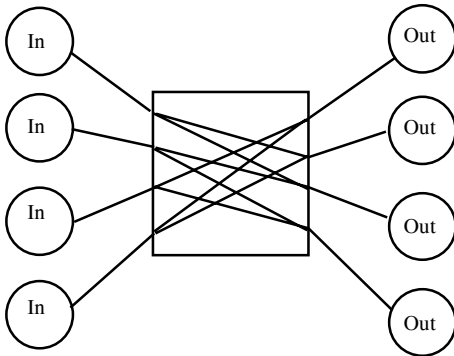
c. n-to-m partial cross: *n*×*m* VCCs; *n*=4, *m*=2
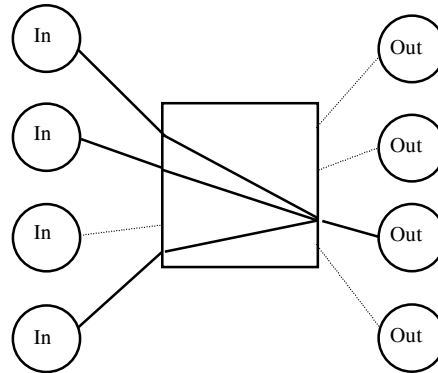
d. k-to-1: *k* VCCs; *k*=3
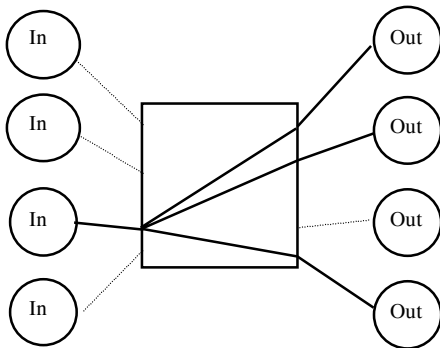
e. 1-to-(n-1) multicast: one (multicast) VCC
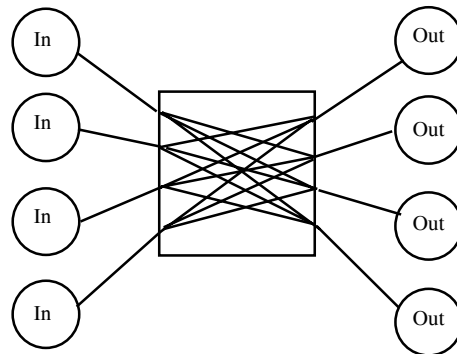
f. n-to-(n−1) multicast: *n* (*multicast*) VCCs; *n*=4

Figure 3.2: Connection configurations for foreground traffic

In the case of n-to-n straight, input from one port exits to another port. This represents almost no path interference among the foreground VCCs. There are n foreground VCCs. See Figure 3.2a.

In the case of n-to-(n−1) full cross, input from each port is divided equally to exit on each of the other $(n-1)$ ports. This represents intense competition for the switching fabric by the foreground VCCs. There are $n \times (n-1)$ foreground VCCs. See Figure 3.2b.

In the case of n-to-m partial cross, input from each port is divided equally to exit on the other m ports $(1 \le m \le n-1)$. This represents partial competition for the switching fabrics by the foreground VCCs. There are $n \times m$ foreground VCCs as shown in Figure 3.2c. Note that n-to-n straight and n-to-(n−1) full cross are special cases of n-to-m partial cross with $m=1$ and $m=n-1$, respectively.

In the case of k-to-1, input from k $(1 < k < n)$ ports is destined to one output port. This stresses the output port logic. There are k foreground VCCs as shown in Figure 3.2d.

In the case of 1-to-(n−1) multicast, all foreground frames input on the one designated port are multicast to all other (n−1) ports. This tests single multicast performance of the switch. There is only one (multicast) foreground VCC as shown in Figure 3.2e.

Use of the 1-to-(n−1) multicast connection configuration for the foreground traffic is under study.

In the case of n-to-(n−1) multicast, input from each port is multicast to all other $(n-1)$ ports. This tests multiple multicast performance of the switch.. There are n (multicast) foreground VCCs. See Figure 3.2f.

Use of the n-to-(n−1) multicast connection configuration for the foreground traffic is under study.

The following service classes, arrival patterns and frame lengths for foreground traffic are used for testing:
- UBR service class: Traffic consists of equally spaced frames of fixed length. Measurements are performed at AAL payload size of 64 B, 1518 B, 9188 B and 64 kB. Variable length frames and other arrival patterns (e.g. self-similar) are under study.
- ABR and VBR service classes are under study.

The required input rate of foreground traffic is obtained by loading each link by the same fraction of its input rate. In this way, the input rate of foreground traffic can also be referred to as a fraction (percentage) of input link rates. The maximum foreground load (MFL) is defined as the sum of rates of all links in the maximum possible switch configuration. Input rate of the foreground traffic is expressed in the effective bits/sec, counting only bits from frames, excluding the overhead introduced by the ATM technology and transmission systems.

3.1.6. Background Traffic

Higher priority traffic (like VBR or CBR) can act as background traffic for experiments. Further details of measurements with background traffic using multiple service classes simultaneously are under study. Until then, all testing will be done without any background traffic.

3.1.7. Guidelines For Scaleable Test Configurations

It is obvious that testing larger systems, e.g., switches with larger number of ports, could require very extensive (and expensive) measurement equipment. Hence, we introduce scaleable test configurations for throughput measurements that require only one ATM monitor with one generator/analyzer pair. Figure 3.3 presents a simple test configuration for an ATM switch with eight ports in a 8-to-8 straight connection configuration. Figure 3.4 presents a test configuration with the same switch in an 8-to-2 partial cross connection configuration. The former configuration emulates 8 foreground VCCs, while the later emulates 16 foreground VCCs.

In both test configurations, there is one link between the ATM monitor and the switch. The other seven ports have external loopbacks. A loopback on a given port causes the frames transmitted over the output of the port to be received by the input of the same port.

The test configurations in Figure 3.3 and Figure 3.4 assume two network modules in the switch, with switch ports P0-P3 in one network module and switch ports P4-P7 in the another network module. Foreground VCCs are always established from a port in one network module to a port in the another network module. These connection configurations could be more demanding on the SUT than the cases where each VCC uses ports in the same network module. An even more demanding case could be when foreground VCCs use different fabrics of a multi-fabric switch.

Approaches similar to those in Figure 3.3 and Figure 3.4 can be used for n-to-(n−1) full cross and other types of n-to-m partial cross connection configurations, as well as for larger switches. Guidelines to set up scaleable test configurations for the k-to-1 connection configuration are under study.

It should be noted that in the proposed test configurations, because of loopbacks, only permanent VCCs or VPCs can be established.

It should also be realized that in the test configurations with loopbacks, if all link rates are not identical, it is not possible to generate foreground traffic equal to the MFL. The maximum foreground traffic load for a n-port switch in those cases equals $n \times$ lowest link rate. Only in the case when all link rates are identical is it possible to obtain MFL level. If all link rates are not identical, and the MFL level needs to be reached, it is necessary to have more than one analyzer/generator pair.
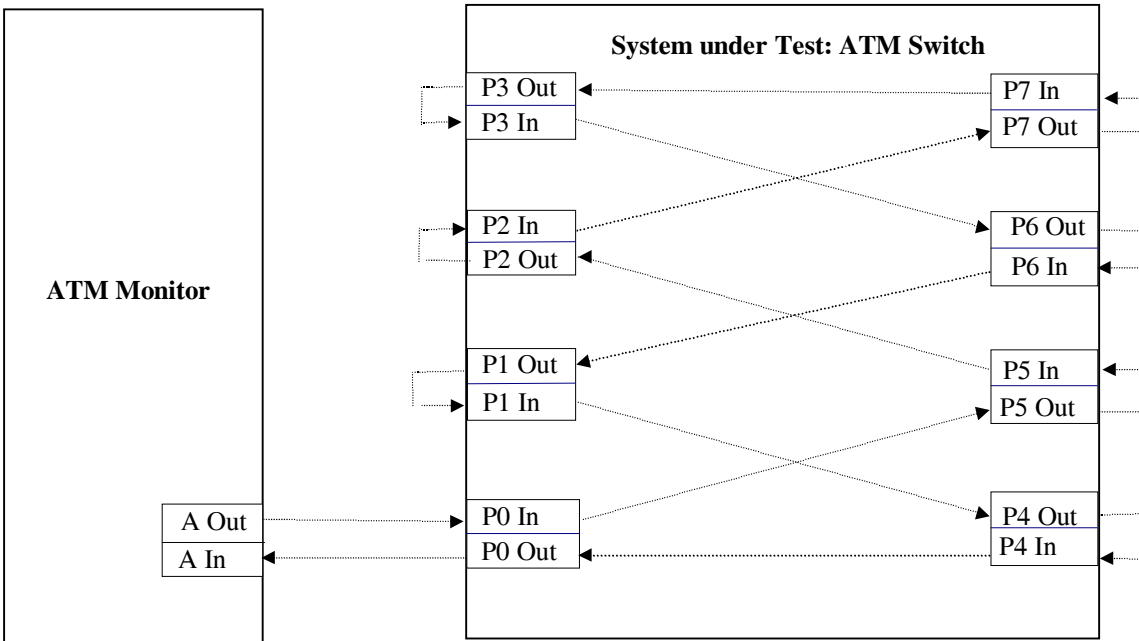
Figure 3.3: A scaleable test configuration for throughput measurements using only one generator/analyzer pair with 8-port switch and a 8-to-8 straight connection configuration
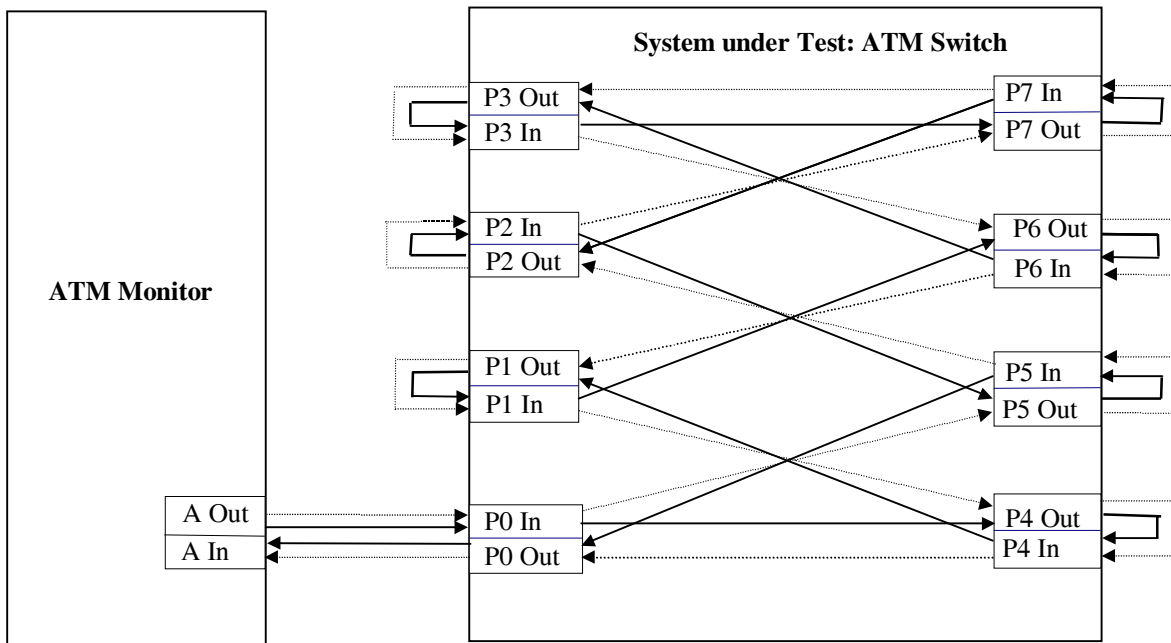
Figure 3.4: A scaleable test configuration for throughput measurements using only one generator/analyzer pair with 8-port switch and a 8-to-2 partial cross connection configuration

3.1.8. Reporting results

Results should include a detailed description of the SUT, such as the number of ports, rate of each port, number of ports per network module, number of network modules, number of network modules per fabric, number of fabrics, maximum foreground load (MFL), software version, and any other relevant information.

Values for the loss-less throughput, the peak throughput with corresponding input load, and the full-load throughput with corresponding input load (if different from MFL) are reported along with foreground (and background, if any) traffic characteristics.

The list of foreground traffic characteristics and their possible values are now provided:
- type of foreground VCCs: permanent virtual path connections, switched virtual path connections, **permanent virtual channel connections**, switch virtual channel connections;
- foreground VCCs established: between ports inside a network module, **between ports on different network modules**, between ports on different fabrics, some combination of previous cases;
- connection configuration: n-to-n straight, n-to-(n–1) full cross, **n-to-m partial cross** with m **= 2, 3**, 4, …, n–1, **k-to-1** with k**=**2, 3, **4**, 5, 6, …, 1-to-(n-1) multicast, n-to-(n-1) multicast;
- service class: **UBR**, ABR, VBR;
- arrival patterns: **equally spaced frames**, self-similar, random;
- frame length: 64 B, **1518 B**, **9188 B** or 64 kB, variable;

Values in bold indicate traffic characteristics for which measurement tests must be performed and for which throughput values must be reported.

3.2. Frame Latency

3.2.1. Definition

The frame latency for a system under test is measured using a "Message-in Message-out (MIMO)" definition. Succinctly, MIMO latency is defined as follows:

~~MIMO Latency = Min{First bit in to last bit out latency – nominal frame output time, last bit in to last bit out latency}~~

$$MIMO\ latency = FILO\ latency – NFOT$$

where
- FILO latency = Time between the first-bit entry and the last-bit exit
- NFOT = Nominal Frame Output Time, defined as the time a frame needs to pass through *the zero-delay switch*, that can be calculated using the following procedure:

Initially NFOT = 0 and time t is measured from the arrival of the first bit of the first cell. For each cell with its first bit arriving at time t $\Rightarrow$ NFOT = max{t, NFOT} + CT.

Here CT is the larger of the cell input time or cell output time. Cell times are computed as the cell size of 424 bits divided by the respective link rates in bits per sec.

An equivalent MIMO latency definition is:

$$\text{MIMO latency} = \begin{cases} \text{LILO latency} & \text{if Input Link Rate} \leq \text{Output Link Rate} \\ \text{FILO latency} - \text{NFOT} & \text{otherwise} \end{cases}$$

where
- LILO latency = Time between the last-bit entry and the last-bit exit

Frame Latency Measurements and Calculation

To obtain MIMO latency for a given frame, the time of occurrence for the following two events need to be recorded:
- First-bit of the frame enters into the SUT,
- Last-bit of the frame exits from the SUT.

The time between the second and the first events is FILO latency. If measurement data are available at cell level, what is usually the case with contemporary ATM monitors, it can be shown that:

FILO latency = First cell's transfer delay + First cell to last cell inter-arrival time

where
- cell transfer delay (CTD) is the time between the first bit of the cell entering the switch and the last bit of the cell leaving the switch,
- cell inter-arrival time is the time between arrival from the switch of the last bit of the first cell and arrival from the switch of the last bit of the second cell.

Given the cell pattern of a frame on input, NFOT can be obtained using the procedure from its definition. Then, substituting FILO latency and NFOT in the MIMO latency formula would give the SUT delay for the given frame.

In the cases when Input Link Rate ≤ Output Link Rate, MIMO latency can be obtained easier. In those cases, the time of occurrence for the following two events need to be recorded:
- Last-bit of the frame enters into the SUT,
- Last-bit of the frame exits from the SUT.

The time between the second and the first events is LILO latency. When measurement data are available at cell level, it can be shown that:

LILO latency = Last cell's transfer delay – Cell input time

and in these cases, LILO latency would give the SUT delay for the given frame.

An explanation of MIMO latency and its justification is presented in Appendix A.

To measure MIMO latency, a sequence of equally spaced frames are sent at a particular rate. After the flow has been established, one of the frames in the flow is marked and the time of the following four events is recorded for the marked frame while the flow continues unperturbed:

1. First-bit of the frame enters into the SUT
2. Last-bit of the frame enters into the SUT
3. First-bit of the frame exits from the SUT
4. Last-bit of the frame exits from the SUT

The time between the first-bit entry and the last-bit exit (events 1 and 4 above) is called first-bit in to last-bit out (FILO) latency. The time between the last-bit entry to the last-bit exit (events 2 and 4 above) is called last-bit in to last-bit out (LILO) latency. Given the frame size and the nominal output link rate, the nominal frame output time is computed as follows:

Nominal frame output time = Frame size/Nominal output link rate

Substituting the FILO latency, LILO latency, and Nominal frame output time in the MIMO latency formula gives the frame level latency of the SUT.

3.2.2 Units

The latency should be specified in micro-seconds.

3.2.3 Statistical Variations

NML samples of the latency are obtained by sending NML marked frames at TTL/(NML + 1) intervals for a total test duration of TTL seconds. Here, NML and TTL are parameters. Their default values are specified in Table 3.2. The mean and standard errors computed (in a manner similar to that explained in Section 3.1 for Throughput) from these samples are reported as the test results.

3.2.4 Traffic Pattern

The input traffic will consist of frames of length FSA bytes. Here, FSA is a parameter. Its default value is specified in Table 3.2.

Before starting the throughput measurements, all required VCs will be set up (for an n-port SUT) in one of the following configurations (see Figure 3.2):

1. **n-to-n straight**: All frames input from port i exit to port i+1 modulo n. This represents almost no path interference among the VCs.
2. **n-to-n cross**: Input from port each port is divided equally to exit on each of the n output ports.
3. **n-to-1** : Input from all ports is destined to one output port.
4. **1-to-n**: Input from a port is multicast to all output ports. Total 1 VC.

The frames will be delivered to the layer under test equally spaced at a given input rate. For latency measurement, the input rate will be set at the input rate corresponding to the lossless throughput. This avoids the problem of lost marked cells and missing samples.

3.2.5 Background Traffic

The tests can be conducted under two conditions - with background traffic and without background traffic. Higher priority traffic like VBR can act as background traffic for the experiment. Further details of measurements with background traffic (multiple service classes simultaneously) are for further study. Initially all tests will be conducted without the background traffic.

3.2.6 Guidelines For Using This Metric

To be specified.

3.3. THROUGHPUT FAIRNESS

3.3.1 Definition

Given n contenders for the resources, throughput fairness indicates how far the actual individual allocations are from the ideal allocations. In the most general case of a network, ideal allocation is defined by max-min allocation to various contending virtual circuits. For the simplest case of n VCs sharing a link with a total throughput T, the throughput of each VC should be T/n.

If the actual measured throughputs of n VCs sharing a system (a single switch or a network of switches) are found to be $\{T_1, T_2, ..., T_n\}$, where the optimal max-min throughputs[1] should be $\{\hat{T}_1, \hat{T}_2, ..., \hat{T}_n\}$, then the fairness of the system under test is quantified by the "fairness index" computed as follows:

---

[1] Other policies could be used but must be specified.

$$\text{Fairness Index} = \frac{\left(\Sigma x_i\right)^2}{n\Sigma x_i^2}$$

Where, $x_i = T_i / \hat{T_i}$ is the relative allocation to ith VC.

This Fairness Index has the following desirable properties:

1. It is dimensionless. The units used to measure the throughput (bits/sec, cells/sec, frames/sec) do not affect its value.

3.2.2. Units

The latency should be specified in μsec.

3.2.3. Statistical Variations

For the given foreground traffic and background traffic, the required times and/or delays, needed for MIMO latency calculation, are recorded for p frames, according to the procedures described in 3.2.4. Here p is a parameter and its default (and the minimal value) is 100.

Let M_i be the MIMO latency of the ith frame. Note that MIMO latency is considered to be infinite for lost or corrupted frames. The mean and standard errors of the measurement are computed as follows:

Mean MIMO latency = $(\Sigma M_i) / p$

Standard deviation of MIMO latency = $(\Sigma(M_i - \text{mean MIMO latency})^2) / (p-1)$

Standard error = standard deviation of MIMO latency / $p^{1/2}$

Given the mean and the standard error, the users can compute a $100(1-\alpha)$-percent confidence interval as follows:

$100(1-\alpha)$-percent confidence interval = (mean $- z \times$ standard error, mean $+ z \times$ standard error)

Here, z is the $(1-\alpha/2)$-quantile of the unit normal variate. For commonly used confidence levels, the quantile values are as follows:

Confidence	α	Quantile
90%	0.1	1.615
99%	0.01	2.346
99.9%	0.001	3.291

The value of p can be chosen differently from its default value to obtain the desired confidence level.

3.2.4. Measurement Procedures

For MIMO latency measurements, it is first necessary to establish one VCC (or VPC) used only by foreground traffic, and a number of VCCs or VPCs used only by background traffic. Then, the background traffic is generated. Characteristics of background traffic are described in section 3.2.6. When flow of the background traffic has been established, the foreground traffic is generated. Characteristics of foreground traffic are specified in section 3.2.5. After the steady state flow of foreground traffic has been reached the required times and/or delays needed for MIMO latency calculation are recorded for p consecutive frames from the foreground traffic, while the flow of background traffic continue uninterrupted. The entire procedure is referred to as one measurement run.

3.2.5. Foreground traffic

MIMO latency depends upon several characteristics of foreground traffic. These include the type of foreground VCC, service class, arrival patterns, frame length, and input rate.

The foreground VCC can be a permanent or switched, virtual path or virtual channel connection established between ports on the same network module of the switch, or between ports on different network modules, or between ports on different switching fabrics.

For the UBR service class, the foreground traffic consists of equally spaced frames of fixed length. Measurements are performed on AAL payload sizes of 64 B, 1518 B, 9188 B and 64 kB. Variable length frames and other arrival patterns (e.g. self-similar) are under study. ABR service class is also under study.

Input rate of foreground traffic is expressed in the effective bits/sec, counting only bits from AAL payload excluding the overhead introduced by the ATM technology and transmission systems.

The first measurement run is performed at the lowest possible foreground input rate (for the given test equipment). For later measurement runs, the foreground load is increased up to the point when losses in the traffic occur or up to the full foreground load (FFL). FFL is equal to the lesser of the input and the output link rates used by the foreground VCC. Suggested input rates for the foreground traffic are: 0.5, 0.75, 0.875, 0.9375, 0.9687, ..., i.e. $1 - 2^{-k}$, k = 1, 2, 3, 4, 5, …, of FFL.

3.2.6. Background Traffic

Background traffic characteristics that affect frame latency are the type of background VCCs, connection configuration, service class, arrival patterns (if applicable), frame length (if applicable) and input rate.

Like the foreground VCC, background VCCs can be permanent or switched, virtual path or channel connections, established between ports on the same network module on the switch, or between ports on different network modules, or between ports on different switching fabrics. To avoid interference on the traffic generator/analyzer equipment, background VCCs are established in such way that they do not use the input link or the output link of the foreground VCC in the same direction.

For a SUT with w ports, the background traffic can use (w–2) ports, not used by the foreground traffic, for both input and output. The port with the input link of the foreground traffic can be used as an output port for the background traffic. Similarly, the output port of the foreground traffic can be used as an input port for the background traffic. Overall, background traffic can use an equivalent of $n=w$–1 ports. The maximum background load (MBL) is defined as the sum of rates of all links, except the one used as the input link for the foreground traffic, in the maximum possible switch configuration.

A SUT with w (=n+1) ports is measured for the following background traffic connection configurations:
- n-to-n straight, with n background VCCs, (Figure 3.2.a);
- n-to-(n–1) full cross, with $n \times (n$–1) background VCCs. (Figure 3.2.b);
- n-to-m partial cross, $1 \le m \le n$–1, with $n \times m$ background VCCs. (Figure 3.2.c);
- 1-to-(n–1) multicast, with one (multicast) background VCC. (Figure 3.2.e);
- n-to-(n–1) multicast, with n (multicast) background VCC. (Figure 3.2.d);

Use of the 1-to-(n–1)multicast and n-to-(n–1) multicast connection configurations for the background traffic is under study.

The following service classes, arrival patterns (if applicable) and frame lengths (if applicable) are used for the background traffic:
- UBR service class: Traffic consists of equally spaced frames of fixed length. Measurements are performed at AAL payload size of 64 B, 1518 B, 9188 B and 64 kB. This is a case of bursty background traffic with priority equal to or lower than that of the foreground traffic. Variable length frames and other arrival patterns (e.g. self-similar) are for further study.
- CBR service class: Traffic consists of a contiguous stream of cells at a given rate. This is a case of non-bursty background traffic with priority higher than that of the foreground traffic.
- VBR and ABR service classes are under study.

Input rate of the background traffic is expressed in the effective bits/sec, counting only bits from frames excluding the overhead introduced by the ATM technology and transmission systems.

In the cases of n-to-n straight, n-to-(n–1) full cross and n-to-m partial cross connection configurations, measurement are performed at input rates of 0, 0.5, 0.75, 0.875, 0.9375, 0.9687, … ($1 - 2^{-k}$, k = 0, 1, 2, 3, 4, 5,…) of MBL. The required traffic load is obtained by loading each input link by the same fraction of its input rate. In this way, the input rate of background traffic can also be expressed as a fraction (percentage) of input link rates.

### 3.2.7. Guidelines For Scaleable Test Configurations

Scaleable test configurations for MIMO latency measurements require only one ATM test system with two generator/analyzer pairs. Figure 3.5 presents the test configuration with an ATM switch with eight ports (w=8). There are two links between the ATM monitor and the switch, and they are used in one direction by the background traffic and in the another direction by the foreground traffic, as indicated. The other six (w–2) ports of the switch are used only by the background traffic and they have external loopbacks. A loopback on a given port causes the frames transmitted over the output of the port to be received by the input of the same port.

Figure 3.5 shows a 7-to-7 straight connection configuration for the background traffic. The n-to-(n–1) full cross configuration and the n-to-m partial cross configurations can also be similarly implemented.

The test configuration shown assumes two network modules in the switch with ports P0-P3 in one network module and ports P4-P7 in the another network module. Here, the foreground VCC and background VCCs are established between ports in different network modules.

It should be noted that in the proposed test configurations, because of loopbacks, only permanent VCCs or VPCs can be established.

It should also be realized that in test configurations, if all link rates are not identical, it is not possible to generate background traffic (without losses) equal to MBL. The maximum background traffic input rate in those cases equals (n–1) × lowest link rate. Only in the case where all link rates are identical is it possible to obtain MBL level without losses in background traffic.
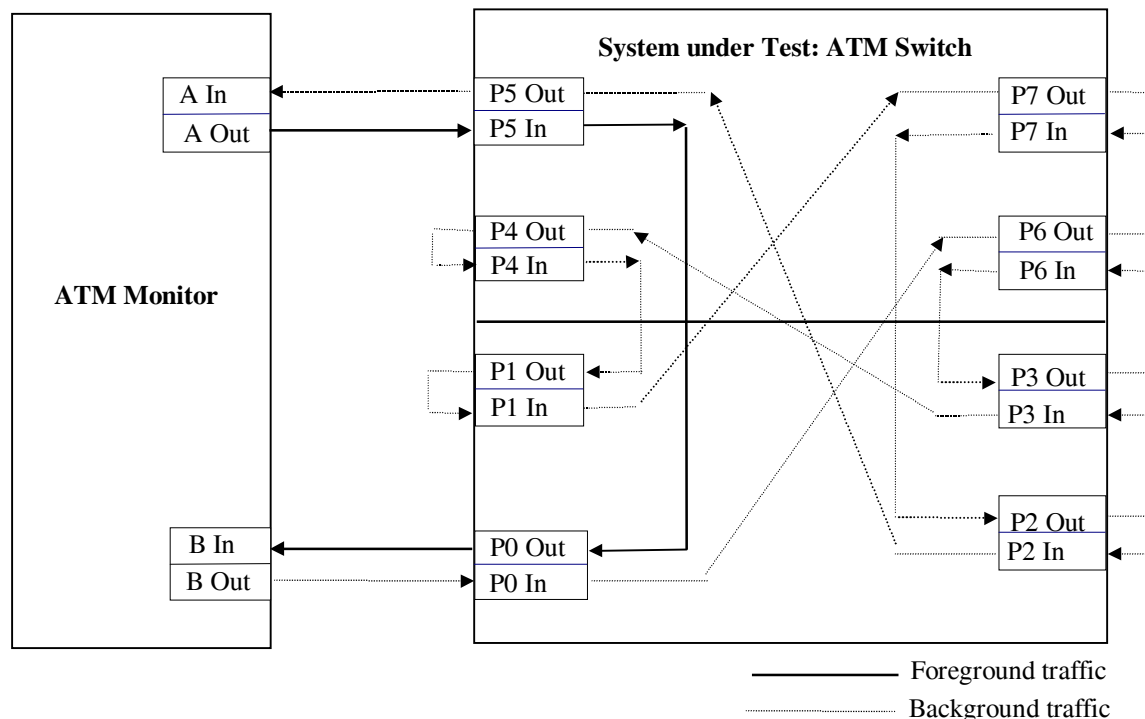
Figure 3.5: A scaleable test configuration for measurements of MIMO latency using only two generator analyzer pairs with 8-port switch and 7-to7 straight configuration for background traffic

If the link rates are different, it is possible to obtain MBL in the n-to-n straight case, but background traffic will have losses. In this case, the foreground traffic should use the lowest rate port in the switch as the input, while the highest rate port in the switch should be used as the output. The background traffic enters the SUT through the highest rate port and passes successively through ports of decreasing speeds. At the end, the background traffic exits the switch through the lowest rate port.

3.2.8. Reporting results

Reported results should include detailed description of the SUT, such as the number of ports, rate of each port, number of ports per network module, number of network modules, number of network modules per fabric, number of fabrics, the software version and any other relevant information.

Values of the mean and the standard error of MIMO latency are reported along with values of foreground and background traffic characteristics for each measurement run.

The list of foreground and background traffic characteristics and their possible values are now provided:

Foreground traffic:
- type of foreground VCC: permanent virtual path connection, switched virtual path connection, **permanent virtual channel connection**, switch virtual channel connection;
- foreground VCC established: between ports inside a network module, **between ports on different network modules**, between ports on different switching fabrics;
- service class: **UBR**, ABR;
- arrival patterns: **equally spaced frames**, self-similar, random;
- frame length: 64 B, **1518 B**, **9188 B** or 64 kB, variable;
- full foreground load (FFL);
- input rate: **the lowest rate possible for the given test equipment**, and 0.5, **0.75**, 0.875, 0.9375, 0.9687, ..., (i.e., $1 - 2^{-k}$, k = 1, 2, 3, 4, 5, …,) of FFL.

Background traffic:
- type of background VCC's: permanent virtual path connections, switched virtual path connections, **permanent virtual channel connections**, switch virtual channel connections;
- foreground VCCs established: between ports inside a network module, **between ports on different network modules**, between ports on different switching fabrics, some combination of previous cases;
- connection configuration: n-to-n straight, n-to-(n–1) full cross**, n-to-m partial cross with *m* = 2, 3**, 4, …, *n*–1, 1-to-(n-1) multicast, n-to-(n-1) multicast;
- service class: **UBR, CBR**, ABR, VBR;

- arrival patterns (when applicable): **equally spaced frames**, self-similar, random;
- frame length (when applicable): 64 B, 1518 B, **9188 B**, 64 kB, variable;
- maximum background load (MBL);
- input rate: **0**, 0.5, 0.75, **0.875**, 0.9375, 0.9687, … (i.e., $1 - 2^{-k}$, k = 0, 1, 2, 3, 4, 5,…) of MBL.

Values in bold indicate traffic characteristics for which measurement tests must be performed and for which MIMO latency values must be reported.

3.3. Throughput Fairness

3.3.1. Definition

There are two throughput fairness metrics that are of interest to users:

- *Peak throughput fairness*: this is the fairness at a frame load for the peak throughput.
- *Full-load throughput fairness*: This is the fairness at a frame load for the full-load throughput.

Given *n* virtual circuits sharing a system (a single switch or a network of switches) and contending for the resources, throughput fairness indicates how far the actual individual allocations are from the ideal allocations. In the simplest case for a total throughput T, the ideal allocation should be T/n. We consider that in the most general case, the ideal allocation is defined by max-min allocation and that allocation is to be used.[2]

If the actual measured throughputs of *n* virtual circuits are found to be $\{T_1, T_2, ..., T_n\}$, where the ideal throughputs should be $\{\hat{T}_1, \hat{T}_2, ..., \hat{T}_n\}$, then the throughput fairness of the system under test is quantified by the "fairness index" computed as follows:

$$\text{Fairness index} = (\textstyle\sum x_i)^2 / (n \times \textstyle\sum x_i^2)$$

where:
- $x_i = T_i / \hat{T}_i$ is the relative allocation to *i*th VC.

Note that fairness index is not limited to throughput. It can be applied to other metrics, such as latency. However, extreme unfairness in latency is expected show up as unfairness in throughput and vice versa. Therefore, it is not required to quantify fairness of latency.

3.3.2. Units

This fairness index is dimension-less. The units used to measure the throughput (bits/sec, cells/sec, or frames/sec) do not affect its value. In addition, the fairness index has the following desirable properties:

---

[2] Other policies could be used but must be specified.

- 2. It is a normalized measure that ranges between zero and one. The maximum fairness is 100% and the minimum 0%. This makes it intuitive to interpret and present.
- 3. If all x_i's are equal, the allocation is fair and the fairness index is one.
- 4. If n-k of n x_i's are zero, while the remaining k x_i's are equal and non-zero, the fairness index is k/n. Thus, a system which allocates all its capacity to 80% of VCs has a fairness index of 0.8 and so on.

3.3.2 Load Level and Traffic Pattern

Throughput fairness is quantified via the fairness index for each of the throughput experiments in which there are either multiple VCs or multiple input or output ports. Thus, it applies to all three throughput measures (lossless, peak, and full-load) and all four traffic patterns (n-to-n straight, n-to-n cross, n-to-1, and 1-to-n) described in Section 3.1.4.

Note that in the case of n-to-n cross, there are n^2 VCs and, therefore, n^2 should be substituted in place of n in the fairness index.

In the case of a 1-to-n pattern, there is only one VC and all input is expected to be multicast to n output ports. The fairness will measure the equality of throughput to the output ports.

No additional experiments are required for throughput fairness. The detailed results obtained for the throughput tests are analyzed to compute the fairness.

3.3.3 Statistical Variation

3.3.3. Measurement procedures

To measure a peak throughput fairness, the peak throughput for the given SUT has to be first obtained as described in 3.1.4. An experiment for peak throughput fairness is performed by generating the input load corresponding to the peak throughput and recording throughput for each foreground virtual circuit. The experiment is repeated p times. Here p is a parameter and its default value is 30.

To measure a full throughput fairness, the full-load throughput for the given SUT has to be first obtained as described in 3.1.4. Then experiments for full-load throughput fairness are performed similarly to peak throughput fairness experiments.

3.3.4. Statistical Variations

The throughput tests are run NRT times for TRT seconds each. Recall that NRT and TRT are parameters. The fairness is computed for each individual run. Let F_i be the fairness for the ith run, throughput experiment, then the mean fairness is computed as follows:

$$\text{Mean Fairness} = \text{sum}(F_i)/NRT$$

3.3.4 Background Traffic

The throughput tests are conducted with and without background traffic. Higher priority VBR traffic can act as background traffic. Further details for measurements with background traffic (multiple service classes simultaneously) are for further study. Until then all performance testing will be done without any background traffic.

3.3.5 Reporting Results

The fairness index values are reported for each of the throughput experiments in the tabular format specified in Table 3.1.

Note that fairness index is not limited to throughput. It can be applied to other metrics, such as latency. However, extreme unfairness in latency is expected show up as unfairness in throughput and vice versa. Therefore, it is not required to quantify fairness of latency.

3.3.6 Guidelines For Using This Metric

To be specified.

3.4. FRAME LOSS RATIO
3.4.1 Definition

$$\text{Mean Fairness} = (\Sigma F_i) / p$$

3.3.5. Reporting Results

Values of the mean fairness for peak and lossless throughput (with indication of a number of experiments) are reported along with a detailed description of the SUT, foreground traffic characteristics, and background traffic characteristics (if any), as defined in 3.1.8.

3.4. Frame Loss Ratio

3.4.1. Definition

Frame loss ratio is defined as the fraction of frames that are not forwarded by a system under ~~test~~ test
 (SUT) due to lack of resources. Partially delivered frames are considered lost.

Frame loss ratio = (Input frame count - output frame count)/(input frame count)

There are two frame loss ratio metrics that are of interest to a ~~user~~user:

- ~~i. **Peak throughput frame loss ratio** - It~~*Peak throughput frame loss ratio*: This is the frame loss ratio at a frame load for the peak throughput.
- ~~ii.~~*Full-load throughput frame loss* ~~ratio   It~~*ratio*: This is the frame loss ratio at a frame load for the full-load throughput.

3.4.2. Units

The frame loss ration is expressed as a fraction of input frames.

3.4.3. Measurement Procedures

~~These metrics are~~The frame loss ratio metric is related to the throughput:

Frame Loss Ratio = (Input Rate - Throughput)/Input Rate

Thus, no additional experiments are required for frame loss ratios. These can be derived from tests performed for throughput ~~measurements provided the input rates are recorded.~~measurements.
~~3.4.2 Unit~~
~~The frame loss ratio is expressed as a fraction of input frames.~~

3.4.4. Statistical Variations

Since there is only one sample for any of the three frame-level throughput metrics, there is no need for calculation of the means and/or standard deviations of frame loss ratio.

3.4.5. Reporting Results

Values of the frame loss ratios for peak and lossless are reported along with a detailed description of the SUT, foreground traffic characteristics, and background traffic characteristics (if any), as defined in 3.1.8.

3.5. Maximum Frame Burst Size (MFBS)

3.5.1. Definition

Maximum Frame Burst Size (MFBS) is the maximum number of frames that source end systems can send at the peak rate through a system under test without incurring any ~~loss.~~
loss. MFBS measures the data buffering capability of the SUT and its ability to handle back-to-back frames.

Many applications and transport layer protocol drivers often present a burst of frames to AAL for transmission. For such applications, Maximum Frame Burst Size provides an useful indication.

This metric is particularly relevant to UBR service category since the UBR sources are always allowed to send a burst at peak rate. ABR sources may be throttled down to a lower rate if a switch runs out of buffer.

3.5.2. Units

MFBS should be expressed in octets of AAL payload field. This is preferred over number of frames or cells. The former requires specifying the frame size and the latter is not very meaningful for a frame-level metric. Also, number of cells has to be converted to octets for use by AAL users.

It may be useful to indicate the frame size for which MFBS has been measured. If MFBS is found to be highly variable with frame size, a number of common AAL payload field sizes such as 64 octets, 536 octets, 1518 octets, and 9188 octets may be used (exact sizes are for further study).

3.5.3. Statistical Variations

The number of frames sent in the burst is increased successively until a loss is observed on any VC. The maximum number of frames that can be sent without loss are reported as MFBS. The tests should be repeated NRT times. The average of NRT repetitions is reported as the MFBS for the system under test.

3.5.4. Traffic Patterns

The MFBS is measured for n-to-1 traffic pattern specified in Section 3.1.4. Optionally, it can be measured for other traffic patterns also. The value obtained for n-to-1 pattern is expected to be smaller than that for other patterns.

3.5.5. Guidelines For Using This Metric

To be specified.
3.6. CALL ESTABLISHMENT LATENCY

3.6. Call Establishment Latency

3.6.1. Definition

For short duration VCs, call establishment latency is an important part of the user perceived performance. Informally, the time between submission of a call setup request to a network and the receipt of the connect message from the network is defined as the call establishment latency. The time lost at the destination while the destination was deciding whether to accept the call is not under network control and is, therefore, not included in call setup latency (See Figure 3.1).3.6).
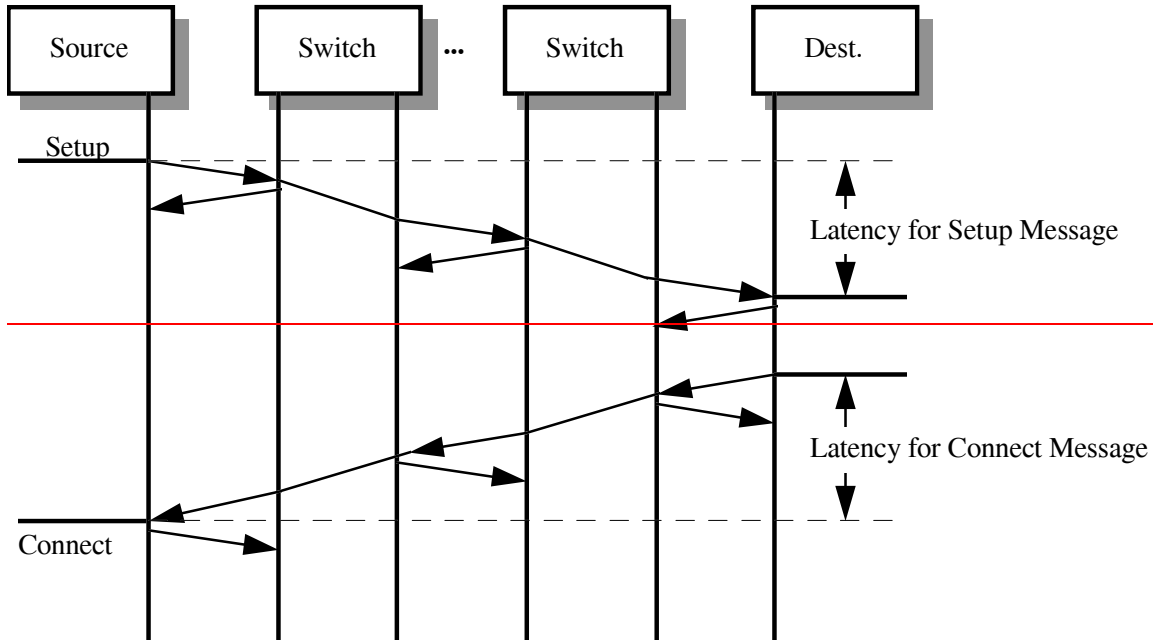
Figure 3.1: Call establishment

Thus, the sum of the latency experienced by the setup message and the resulting connect message is the call setup latency.
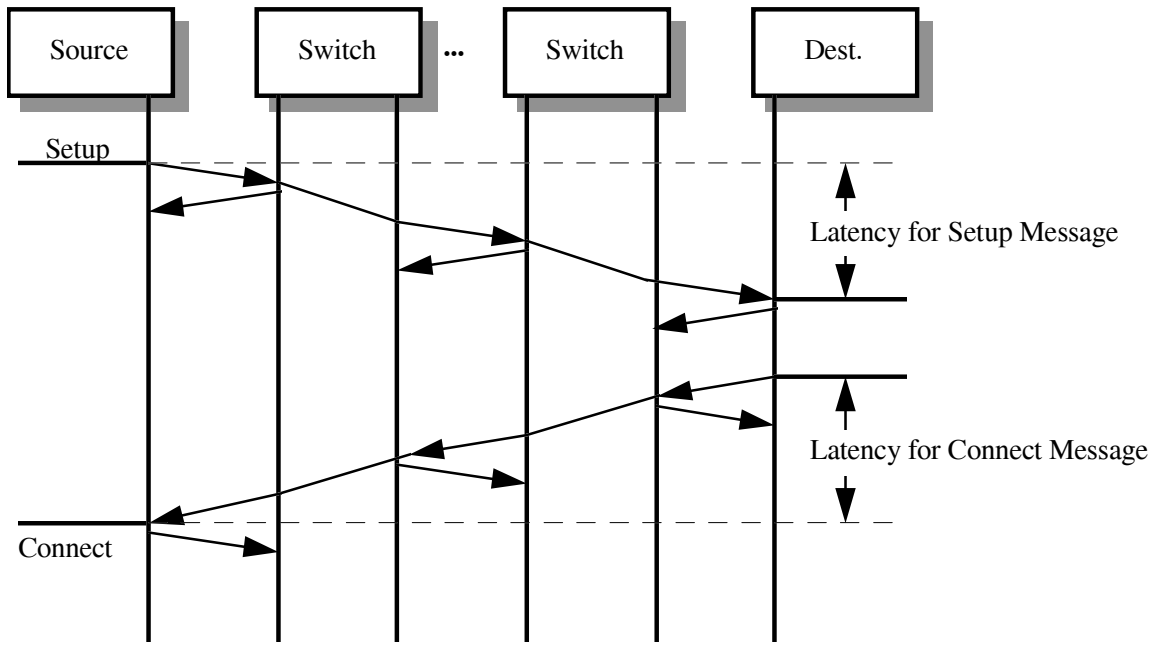

Figure 3.6: Call establishment

The main problem in measuring these latencies is that both these messages span multiple cells with intervening idle/unassigned cells. Unlike X.25, frame relay, and ISDN networks, the

messages in ATM networks are not contiguous. Therefore, the MIMO latency metric defined in Section 3.2 is used[3]. Thus,

Call Establishment Latency = MIMO Latency for SETUP message
+ MIMO latency for the corresponding Connect message

Recall that the MIMO latency for a frame is defined as the minimum of last-bit-in-to-last-bit-out (LILO) and the difference of first-bit-in-to-last-bit-out (FILO) and normal frame output time (NFOT).

$$\text{MIMO Latency} = \text{Min}\{\text{LILO, FILO-NFOT}\}$$

3.6.2. Units

Call establishment latency is measured in units of time.

3.6.3. Configurations

The call establishment latency as defined above applies to any network of switches. In practice, it has been found that the latency depends upon the number of switches and the number of PNNI group hierarchies traversed by the call. It is expected that measurements will be conducted on multiple switches connected in a variety of ways. In all cases, the number of switches and number of PNNI group hierarchies traversed should be indicated.
The simplest configuration is that of a single switch connecting both the source and the destination end systems. Further configurations are for further study.

3.6.4. Statistical Variations

The latency measurement is repeated NRT times. Each time a different node pair is selected randomly as the source and destination end system. The average and standard error of NRT such measurements is reported. For a single n-port switch it is expected that all n ports are equally probable candidates to be source and destination end-system.

3.6.5. Guidelines For Using This Metric

To be specified.

3.7. Application Goodput

---

[3] Applies only if cells of setup and connect messages are contiguous at the input port.

Application-goodput captures the notion of what an application sees as useful data transmission in the long term. Application-goodput is the ratio of packets(frames) received to packets(frames) transmitted over a measurement interval.

The application-goodput (AG) is defined as:

$$AG = \frac{\text{Frames Received in Measurement Interval}}{\text{Frames Transmitted in Measurement Interval}}$$

where Measurement Interval is defined as the time interval from when a frame was successfully received to when the frame sequence number has advanced by n.

Note that traditionally goodput is measured in bits per sec. However, we are interested in a non-dimensional metric and are primarily interested in characterizing the useful work derived from the expended effort rather than the actual rate of transmission. While the application-goodput is intended to be used in a single-hop mode, it does have meaningful end-to-end semantics over multiple hops.

Notes:

- 1. This metric is useful when measured at the peak load which is characterized by varying the number of transmitted frames must be varied over a useful range from 2000 frames per second (fps) through 10000 fps at a nominal frame size of 64 bytes. Frame sizes are also varied through 64 bytes, 1518 bytes, and 9188 bytes to represent small, medium, and large frames respectively. Note that the frame sizes specified do not account for the overhead of accommodating the desired frame transmission rates over the ATM medium.
- 2. Choose the measurement interval to be large enough to accommodate the transmission of the largest packet (frame) over the connection and small enough to track short-term excursions of the average goodput.
- 3. It is important not to include network management frames and/or keep alive frames in the count of received frames.
- 4. There should be no changes of frame handling buffers during the measurement.
- 5. The results are to be reported as a table for the three different frame sizes.

3.7.1. Guidelines For Using This Metric
To be specified.

3.8 REPORTING RESULTS
The throughput and latency results will be reported in a tabular format as follows:

Table 3.1: Tabular format for reporting performance testing results

Traffic Pattern	Throughput											Latency		MFBS
	Lossless			Peak				Full load						
	Mean	Std. Err.	Fairness	Mean	Std. Err	Fairness	FLR	Mean	Std. Err.	Fairness	FLR	Mean	Std. Err.	
n-to-n														

Straight													
n to n Cross													
n to 1													
1 to n													

3.9 DEFAULT PARAMETER VALUES

The default values of the parameters used in performance testing are listed in Table 3.2.

Table 3.2: List of Parameters and their default values

Parameter	Meaning	Default
NRT	Number of repetitions of throughput experiments	30
TRT	Time of each repetition of throughput experiment	60 sec
FSA	Frame Size for AAL performance experiments	9188 Bytes
NML	Number of marked frames sent in latency experiments	30
TTL	Total time of latency experiments	31 sec

To be specified.

4. References

[1] ATM Forum Technical Committee, "Introduction to ATM Forum Test Specifications", at-test-0022.00, December 1994.

[2] S. Sathaye (editor), "ATM Forum Traffic Management Specification Version 4.0", AF-TM 95-0013R10 Straw Vote, February 1996.

Editorial note: The current text of Appendix A is outdated, because it has not been adjusted to reflect the new MIMO latency definition given in 3.2.1. It is expected that the new Appendix A will be presented at September '97 ATMF meeting.

Appendix A: MIMO Latency

A.1. Definition

MIMO latency (Message-In Message-Out) is a general definition of the latency that applies to an ATM switch or a group of ATM switches and it is defined as follows:

$$\text{MIMO latency} = \min \{\text{LILO latency, FILO latency} - \text{NFOT}\}$$

where:
- LILO latency = Time between the last-bit entry and the last-bit exit
- FILO latency = Time between the first-bit entry and the last-bit exit
- NFOT = Nominal Frame Output Time = FIT × Input Rate/Output Rate
- FIT = Frame Input Time = Time between the first-bit entry and the last-bit entry

Note that for contiguous frames on input:

$$\text{Frame Input Time} = \text{Frame Size} / \text{Input rate}$$

and then it follows:

$$\text{NFOT} = \text{Frame Size/Input Rate} \times \text{Input Rate/Output Rate}$$
$$= \text{Frame Size/Output rate}$$

The following is an equivalent definition for MIMO Latency:

$$\text{MIMO latency} = \begin{cases} \text{LILO latency} & \text{if input rate} \leq \text{output rate} \\ \\ \text{FILO latency} - \text{NFOT} & \text{if input rate} \geq \text{output rate} \end{cases}$$

Note that for input rate = output rate:

$$\text{MIMO latency} = \text{LILO latency} = \text{FILO latency} - \text{NFOT}$$

A.2. Introduction

In the rest of the Appendix we justify the MIMO latency definition. In this section, we start with a single bit case and a simple contiguous frame case. Then we systematically consider contiguous frame cases and then discontiguous frame cases in the ATM environment in Section A.3 and Section A.4, respectively.

For a single bit case (see Figure A.1), the latency is generally defined as the time between the instant the bit enters the system to the instant the bit exits from the system.
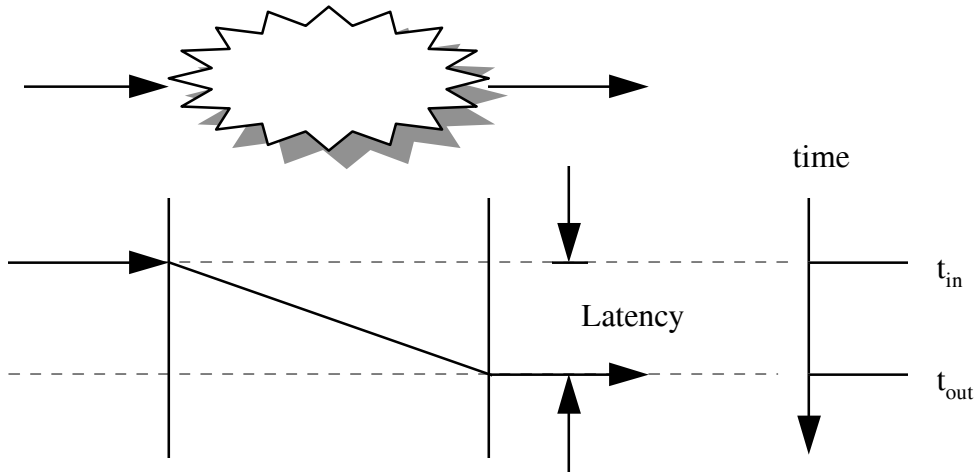
Figure A.1: Latency for a single bit

For multi-bit frames, there are several possible definitions. Consider the case of contiguous frames, i.e. all bits of the frames are sent (on input) and delivered (on output) contiguously without any gap between bits. In this case, latency can be defined in one of the following four ways:

1. FIFO latency: Time between the first-bit entry and the first-bit exit
2. LILO latency: Time between the last-bit entry and the last-bit exit
3. FILO latency: Time between the first-bit entry and the last-bit exit
4. LIFO latency: Time between the last-bit entry and the first-bit exit

If the input link and the output link are of the same speed and frames are contiguous (see Figure A.2), FIFO and LILO latencies are identical. In this case FILO and LIFO latencies can be computed from FIFO (or LILO) latency given the frame length and input rate or output rate:

$$FILO = FIFO + \text{Frame Size/Input rate} = FIFO + \text{Frame Size/Output rate}$$
$$LIFO = FIFO - \text{Frame Size/Input rate} = FIFO - \text{Frame Size/Output rate}$$

It is clear that FIFO (or LILO) is a preferred metric in this case since it is independent of the frame length while FILO and LIFO would be different for each frame size. That is one of reasons why we shall not further consider FILO and LIFO.
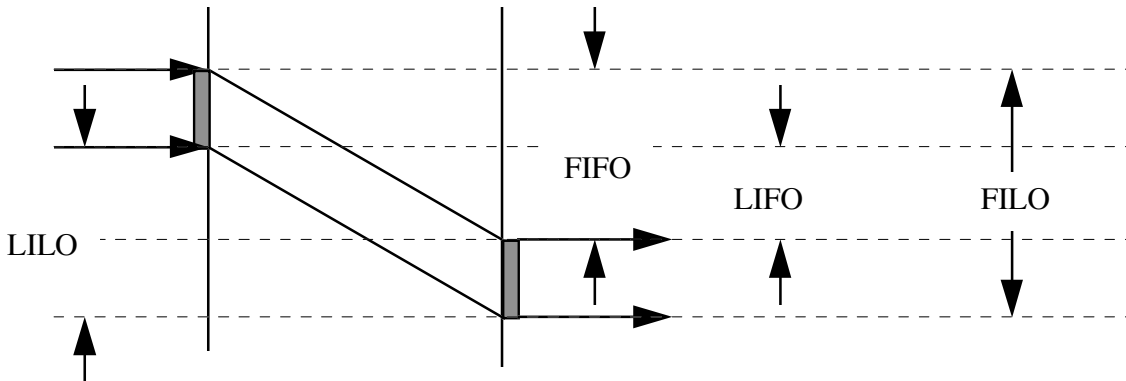
Figure A.2: Latency for multi-bit frames, input rate = output rate

Unfortunately, none of the above four metrics apply to an ATM network (or switch) latency since:

-    the input and output link may be of different speeds, and

-    - the frames are not always sent in (on input) or delivered out (on output) contiguously, i.e., there may be idle times between cells of a frame either on input and/or output.

In the following, we consider first contiguous frames and then discontiguous frames in an ATM network. We compare FIFO, LILO and MIMO metrics and show that MIMO is the correct metric in all cases while other metrics apply to some cases but give incorrect results in others.

A.3. Contiguous Frames

In this section we consider cases where frames on input as well as on output are contiguous, i.e., without any gaps between their cells. Depending upon the relative magnitude of input and output rates and the delay through the switch, there are six possible cases. These cases and the applicability of the three metrics are shown in Table A.1.

Table A.1: Applicability of Various Latency Definitions For Contiguous Frames

No.	Case		FIFO	LILO	MIMO
1aC	Input rate = Output rate, zero-delay switch		√	√	√
~~1bC~~	~~Input rate = Output rate, nonzero-delay switch~~	~~√~~	~~√~~	~~√~~	
1bC	Input rate = Output rate, nonzero-delay switch		√	√	√
2aC	Input rate < Output rate, zero-delay switch		×	√	√
~~2bC~~	~~Input rate < Output rate, nonzero-delay switch~~	~~×~~	~~√~~	~~√~~	
2bC	Input rate < Output rate, nonzero-delay switch		×	√	√
3aC	Input rate > Output rate, zero-delay switch		√	×	√
~~3bC~~	~~Input rate > Output rate, nonzero-delay switch~~	~~√~~	~~×~~	~~√~~	
3bC	Input rate > Output rate, nonzero-delay switch		√	×	√

√ ⇒ The metric gives a valid result.
× ⇒ The metric gives an invalid result.

As indicated above, we consider a zero-delay switch and a nonzero-delay switch. The cases with a zero-delay switch are especially useful to verify the validity of a latency definition, because the switch delay is known in advance (equal to zero).
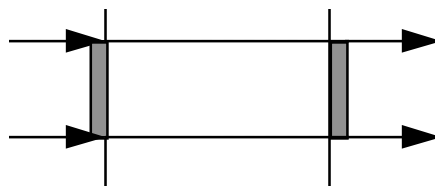
It should be noted that in all cases for contiguous frames on input and on output, the following relation always holds:

FIFO = FILO − Frame size/Output rate,

and this relation will be used in this section.

Case 1aC: Contiguous Frames, Input rate = Output rate, Zero-Delay Switch
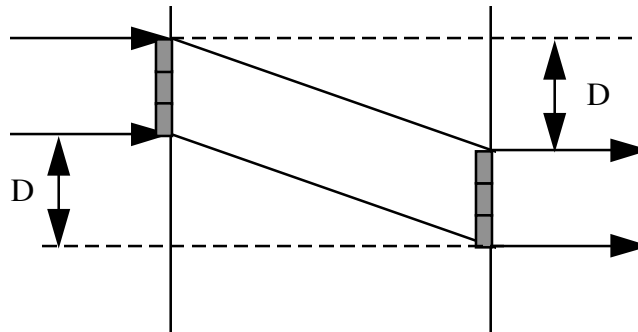
Figure A.1aC shows the flow in this case.

Figure A.1aC: Contiguous frames, Input rate = Output rate, Zero-delay switch

In this case, the bits appear on the output as soon as they enter on the input. Here we have:

- FIFO = 0, correct

- LILO = 0, correct

- MIMO = min {LILO, FILO – Frame Size/Output rate} = min {0, FIFO) = 0, correct

Case 1bC: Contiguous Frames, Input rate = Output rate, Nonzero-Delay Switch

Figure A.1bC shows the flow in this case.

Figure A.1bC: Contiguous frames, Input rate = Output rate, Nonzero-delay switch

In this case, the switch latency D is determined by a delay of the first bit (or the last bit). Here we have:

- FIFO = D, correct

- LILO = D, correct

- MIMO = min {LILO, FILO – Frame Size/Output Rate} = min {D, FIFO} = D, correct

Case 2aC: Contiguous Frames, Input rate < Output rate, Zero-Delay Switch

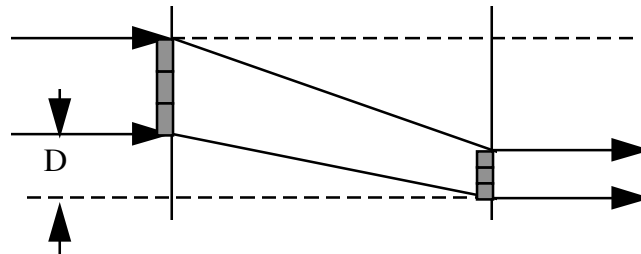Figure A.2aC shows the flow in this case.
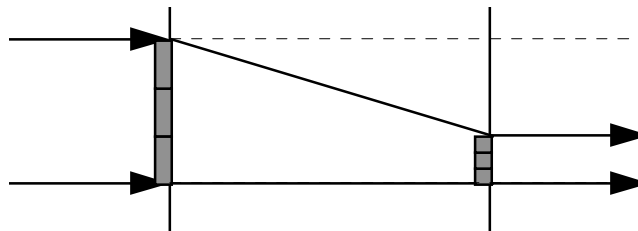
Figure A.2aC: Contiguous frames, Input rate < Output rate, Zero-delay switch

In this case, a contiguous frame on the output is possible only if the transmission of incoming bits is scheduled such that there will not be any buffer underflow until the last bit. Here we have:

- FIFO > 0, incorrect; Note that FIFO may change with changing output rate (while not changing the switch latency). So, FIFO does not correctly represent the switch latency.

- LILO = 0, correct

- MIMO = min {LILO, FILO – Frame Size/Output Rate} = min {0, FIFO} = 0, correct

Case 2bC: Contiguous Frames, Input rate < Output rate, Nonzero-Delay Switch

Figure A.2bC shows the flow in this case.

Figure A.2bC: Contiguous frames, Input rate < Output rate, Nonzero-delay switch

In this case, the switch latency D is determined by a delay of the last bit. Here we have:

- FIFO > D, incorrect; As in Case 2aC, FIFO may change with changing output rate (without changing the switch latency). So, FIFO does not correctly represent the switch latency.

- LILO = D, correct

- MIMO = min {LILO, FILO – Frame Size/Output Rate} = min {D, FIFO} = D, correct

Case 3aC: Contiguous Frames, Input rate > Output rate, Zero-Delay Switch

Figure A.3aC shows the flow in this case.
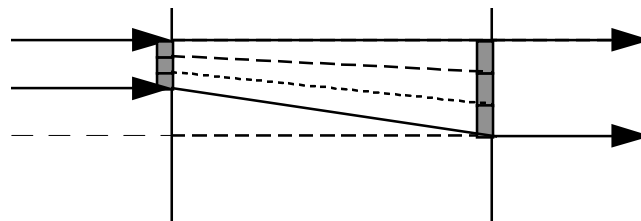
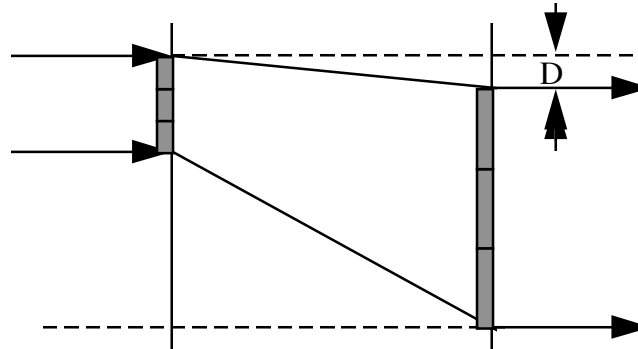**Figure A.3aC**: Contiguous frames, Input rate > Output rate, Zero-delay switch

In this case, only the first bit on the input appears immediately on the output, and other bits have to be buffered, because the input rate is larger (more bits are input) than the output rate (fewer bits are output). Here we have:

- FIFO = 0, correct

- LILO > 0, incorrect; Note that LILO may change with changing the output rate and not changing the switch otherwise. So, LILO does not correctly represent the switch latency.

- MIMO = min {LILO, FILO − Frame Size/Output Rate} = min {LILO, FIFO} = 0, correct

Case 3bC: Contiguous Frames, Input rate > Output rate, Nonzero-Delay Switch

Figure A.3bC shows the flow in this case.

Figure A.3bC: Contiguous frames, Input rate > Output rate, nonzero-delay switch

In this case, the switch latency D is determined by a delay of the first bit. Here we have:

- FIFO = D, correct

- LILO > D, incorrect; As in Case 3aC, LILO may change with changing the output rate and not changing the switch otherwise. So, LILO does not correctly represent the switch latency.

- MIMO = min {LILO, FILO − Frame Size/Output Rate} = min {LILO, FIFO} = D, correct

A.4. Discontiguous Frames

In this section we consider cases where frames on input as well as on output are discontiguous, i.e. there are gaps between cells of frames. Depending upon the number of gaps on input and output, we have three possibilities:

- The number of gaps on output is same as that on input. This is the case of *no change in gaps*.

- The number of gaps on output is more than that on input. This is the case of *expansion of gaps*.

- The number of gaps on output is less than that on input. This is the case of *compression of gaps*.

It should be noted that cases with contiguous frames on input and/or output are special cases of discontiguous frames with no gaps.

The nine cases and the applicability of the three metrics (FIFO, LILO and MIMO) to those cases are shown in Table A.2. Each case includes a case with a nonzero delay switch and (if possible) a case with a zero-delay switch.

Table A.2: Applicability of Various Latency Definitions For Discontiguous Frames
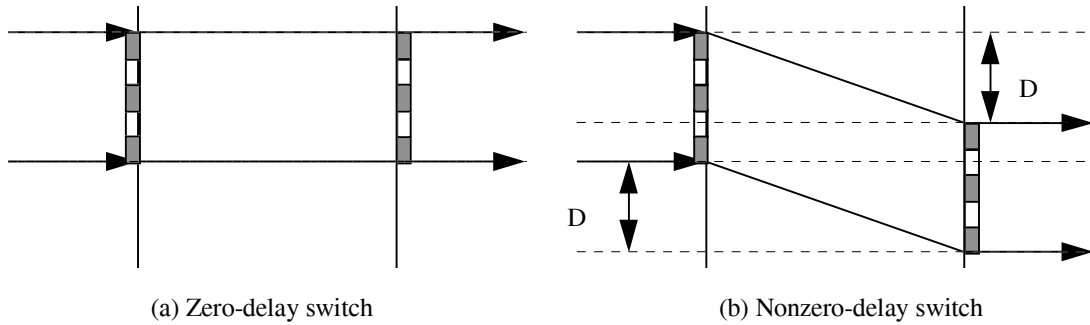
No.	Case	FIFO	LILO	MIMO
1aD	Input rate = Output rate, no change in gaps	√	√	√
1bD	Input rate = Output rate, expansion of gaps	×	√	√
1cD	Input rate = Output rate, compression of gaps	×	√	√
2aD	Input rate < Output rate, no change in gaps	×	√	√
2bD	Input rate < Output rate, expansion of gaps	×	√	√
2cD	Input rate < Output rate, compression of gaps	×	√	√
3aD	Input rate > Output rate, no change in gaps	√	×	√
3bD	Input rate > Output rate, expansion of gaps	×	×	√
3cD	Input rate > Output rate, compression of gaps	×	×	√

√ ⇒ The metric gives a valid result.
× ⇒ The metric gives an invalid result.

Case 1aD: Discontiguous Frames, Input rate = Output rate, No Changes in Gaps

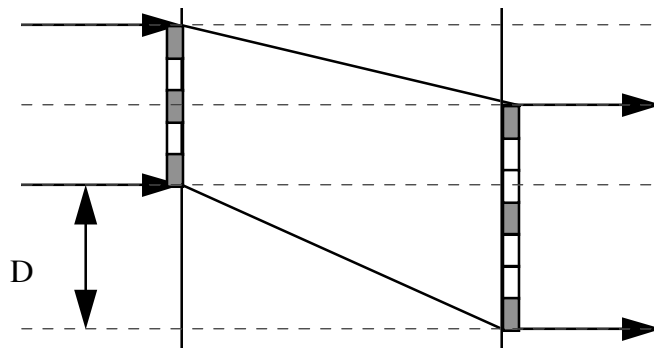Figure A.1aD shows the flow for a zero-delay switch and a nonzero- delay switch.

(a) Zero-delay switch   (b) Nonzero-delay switch

Figure A.1aD: Discontiguous frames, Input rate = Output rate, No change in gaps

This case is similar to cases 1aC and 1bC. The switch latency is determined by a delay of the first bit (or the last bit). Here we have:

- FIFO = D, correct

- LILO = D, correct

- Input rate = Output rate
• MIMO = min {LILO, FILO – FIT} = {D, D} = D, correct

Case 1bD: Discontiguous Frames, Input Rate = Output Rate, Expansion of Gaps

Figure A.1bD shows the flow for a nonzero-delay switch, while a zero-delay switch with expansion of gaps is an impossible scenario.

Figure A.1bD: Discontiguous frames, Input rate = Output rate, Expansion of gaps

In this case, the switch latency D is given by:

$$D = \text{first bit delay} + \text{time of additional gaps on output}$$

Here we have:

- FIFO < D, incorrect; FIFO is incorrect because it does not reflect expansion of gaps. Note that for a nonzero-delay switch, FIFO may be zero (the case of zero delay for the first bit)
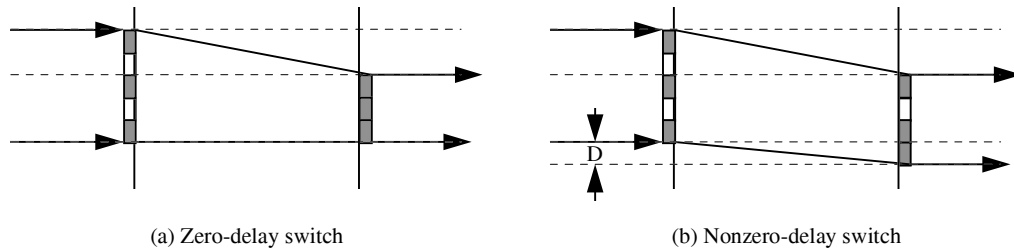
- LILO = D, correct

- Input rate = Output rate
• MIMO = min {LILO, FILO − FIT} = min {D, D} = D, correct

Case 1cD: Discontiguous Frames, Input Rate = Output Rate, Compression of Gaps

Figure A.1cD shows the flow for a zero-delay and a nonzero-delay switch with compression of gaps.

(a) Zero-delay switch   (b) Nonzero-delay switch

Figure A.1cD: Discontiguous frames, Input rate = Output rate, Compression of gaps

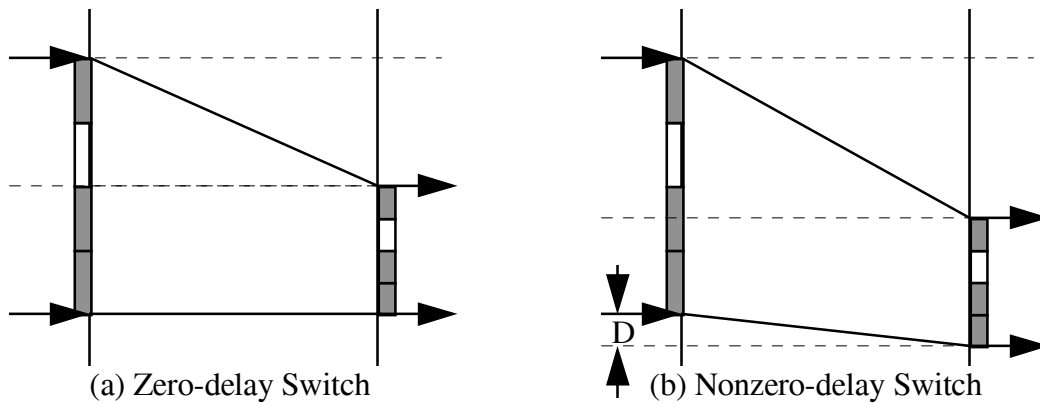In this case, the switch latency D is given by:

$$D = \text{Last bit delay} = \text{First bit delay} − \text{Time of additional gaps on input}$$

Here we have:

- FIFO > D, incorrect; FIFO is incorrect because it does not reflect compression of gaps.

- LILO = D, correct

- Input rate = Output rate
• MIMO = min {LILO, FILO − FIT} = min {D, D} = D, correct

Case 2aD: Discontiguous Frames, Input Rate < Output Rate, No change in Gaps

Figure A.2aD shows the flow for a zero-delay switch and a nonzero-delay switch.

(a) Zero-delay Switch   (b) Nonzero-delay Switch

Figure A.2aD: Discontiguous frames, Input rate < Output rate, No change in gaps

This case is similar to cases 2aC and 2bC. The switch latency D is determined by a delay of the last bit. Here we have:

- FIFO > D, incorrect; FIFO may change with changing the output rate and not changing the switch otherwise. So, FIFO does not correctly represent the switch latency.

- LILO = D, correct

- Input rate < Output rate
- FILO – FIT×Input rate/Output rate > D
- MIMO = min {LILO, FILO – FIT×Input rate/Output rate} = D, correct

Case 2bD: Discontiguous Frames, Input Rate < Output Rate, Expansion of Gaps

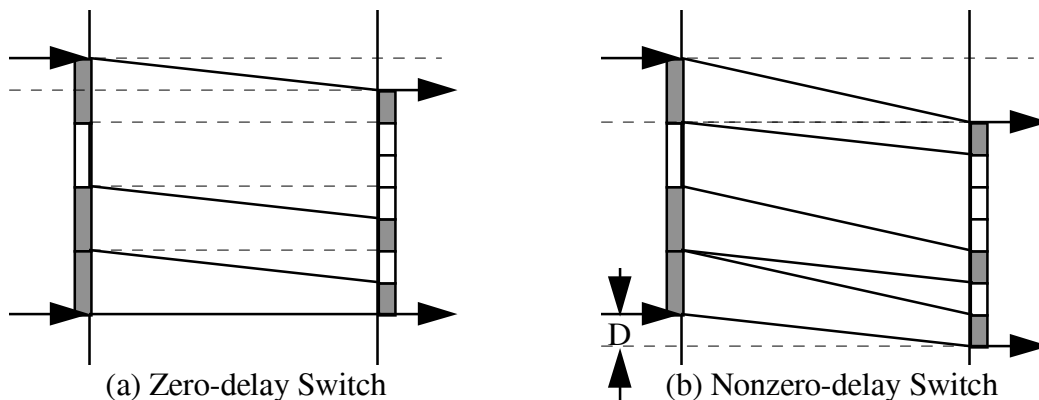Figure A.2bD shows the flow for a zero-delay switch and a nonzero-delay switch.

(a) Zero-delay Switch   (b) Nonzero-delay Switch

Figure A.2bD: Discontiguous frames, Input rate < Output rate, Expansion of gaps

In this case, the switch latency D is determined by a delay of the last bit. Here we have:

- FIFO is incorrect because:
a. FIFO may be affected by changing the output rate and not changing the switch (latency) otherwise.
b. FIFO may change by changing the number of gaps on the output while the switch (latency) is unchanged. It should be noted that for this case, with the given input rate and the given number of gaps on input, it is possible to produce cases with the appropriate output rate and the appropriate number of gaps on output such that FIFO > D or FIFO < D or even FIFO = D, all without changing the switch (latency).

- LILO = D, correct

- Input rate < Output rate
- FILO – FIT×Input rate/Output rate > D
- MIMO = min {LILO, FILO – FIT×Input rate/Output rate} = D, correct

Case 2cD: Discontiguous Frames, Input Rate < Output Rate, Compression of Gaps

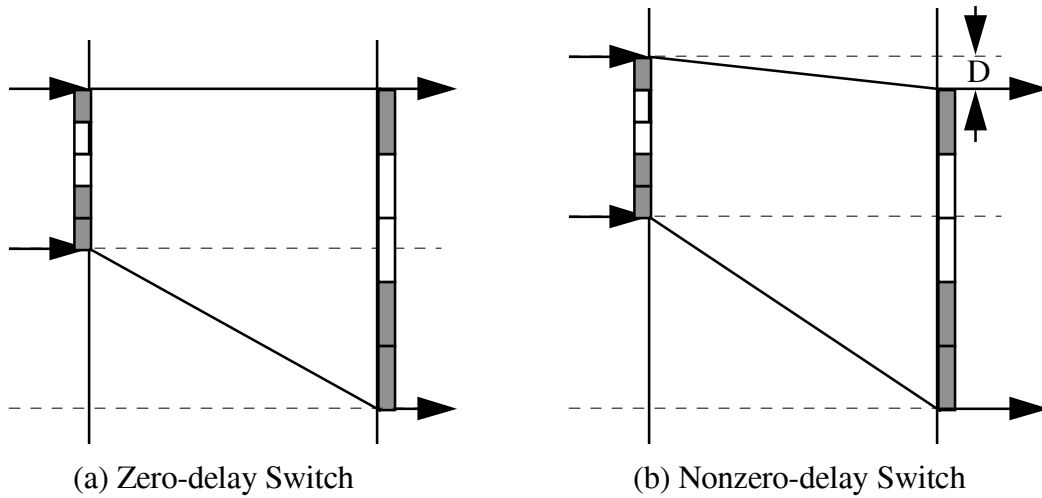Figure A.2cD shows the flow for a zero-delay switch and a nonzero-delay switch.

(a) Zero-delay Switch   (b) Nonzero-delay Switch
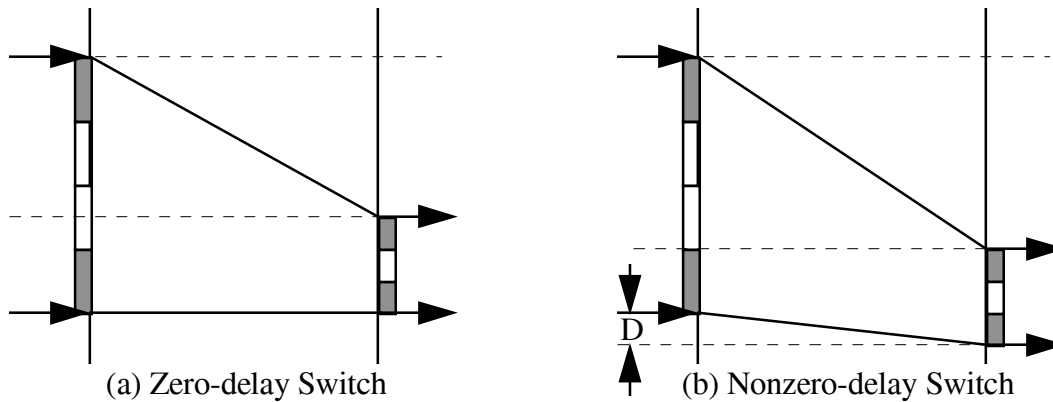
Figure A.2cD: Discontiguous frames, Input rate < Output rate, Compression of gaps

In his case, the switch latency D is determined by the last bit delay. Here we have:

- FIFO > D incorrect; FIFO may be affected by changing the output rate or/and with changing the number of gaps on the output while the switch (latency) is unchanged. So, FIFO does not correctly represent the switch latency.

- LILO = D, correct

- Input rate < Output rate
• FILO – FIT×Input rate/Output rate > D
• MIMO = min {LILO, FILO – FIT×Input rate/Output rate} = D, correct

Case 3aD: Discontiguous Frames, Input Rate > Output Rate, No Change in Gaps

Figure A.3aD shows the flow for a zero-delay switch and a nonzero-delay switch.

(a) Zero-delay Switch   (b) Nonzero-delay Switch

Figure A.3aD: Discontiguous frames, Input rate > Output rate, No Change in Gaps

This case is similar to cases 3aC and 3bC. The switch latency D is determined by a delay of the first bit. Here we have:

- FIFO = D, correct

- LILO > D, incorrect; Note that LILO may change with changing the output rate and not changing the switch otherwise. So, LILO does not correctly represent the switch latency.

- MIMO = min {LILO, FILO − FIT×Input rate/Output rate} = min {LILO, D} = D, correct

Case 3bD: Discontiguous Frames, Input Rate > Output Rate, Expansion of Gaps

Figure A.3bD shows the flow for a nonzero-delay switch, while a zero-delay switch with expansion of gaps is an impossible scenario.
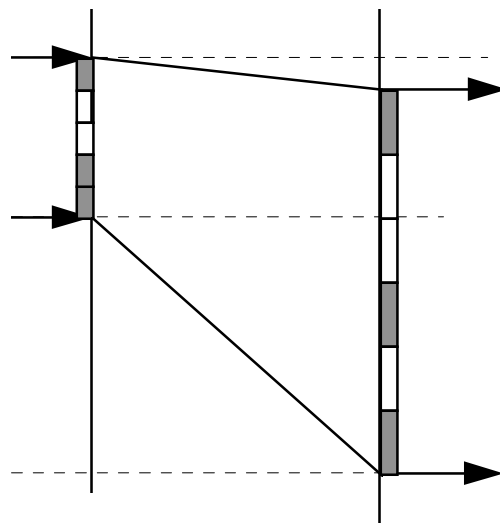
Figure A.3bD: Discontiguous frames, Input rate > Output rate, Expansion of gaps
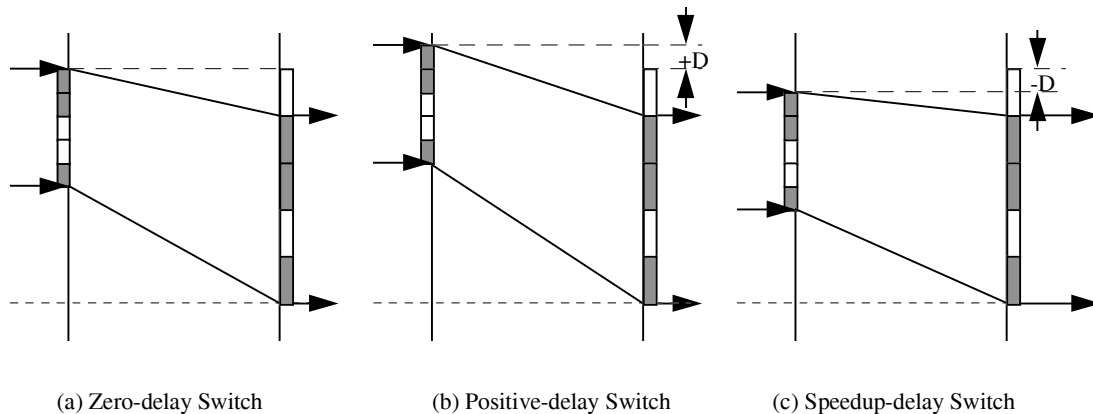
In this case, the switch latency D is given by:

D = first bit delay + time of additional gaps on output

Here we have:

- FIFO < D, incorrect; FIFO is incorrect because it does not reflect expansion of gaps. Note for a nonzero-delay switch, FIFO may be even zero (the case of a zero delay for the first bit)

- LILO > D, incorrect; Here a similar argument applies as in Case 3aD for LILO incorrectly being influenced by the output rate, but with the observation that LILO correctly accounts for a time of additional gaps.

- MIMO = min{LILO, FILO − FIT×Input rate/Output rate} = min{LILO, D} = D, correct

Case 3cD: Discontiguous Frames, Input Rate > Output Rate, Compression of Gaps

Figure A.3cD shows the flow for a zero-delay switch, the positive-delay switch and the speed-up switch.

(a) Zero-delay Switch   (b) Positive-delay Switch   (c) Speedup-delay Switch

Figure A.3cD: Discontiguous frames, Input rate > Output rate, Compression of gaps

In this case the switch latency D is given by:

D = first bit delay − time of missing gaps on output

Three cases can be distinguished:

a. the case of a zero-delay switch, where:

first bit delay = time of missing gaps on output

b. the case of a positive-delay switch, where:

first bit delay > time of missing gaps on output

c. the case of a speedup-delay switch (a negative-delay switch), where:

first bit delay < time of missing gaps on output

- FIFO > D, incorrect; FIFO is incorrect because it does not reflect compression of gaps. Note that, here FIFO may be zero (the case of zero delay for the first bit) while the switch latency is negative

- LILO > D, incorrect; Here a similar argument applies as in Case 3aD for LILO incorrectly being influenced by the output rate, but with the observation that LILO correctly accounts for a time of missing gaps.

- MIMO = min {LILO, FILO – FIT×Input rate/Output rate} = min {LILO, D} = D, correct

In summary, MIMO latency is the only metric that applies to all cases.