International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.8260

(02/2012)

SERIES G: TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

Packet over Transport aspects – Quality and availability targets

---

Definitions and terminology for synchronization in packet networks

Recommendation ITU-T G.8260

ITU-T

ITU-T G-SERIES RECOMMENDATIONS

TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TRANSMISSION MEDIA AND OPTICAL SYSTEMS CHARACTERISTICS	G.600–G.699
DIGITAL TERMINAL EQUIPMENTS	G.700–G.799
DIGITAL NETWORKS	G.800–G.899
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999
MULTIMEDIA QUALITY OF SERVICE AND PERFORMANCE – GENERIC AND USER-RELATED ASPECTS	G.1000–G.1999
TRANSMISSION MEDIA CHARACTERISTICS	G.6000–G.6999
DATA OVER TRANSPORT – GENERIC ASPECTS	G.7000–G.7999
PACKET OVER TRANSPORT ASPECTS	G.8000–G.8999
Ethernet over Transport aspects	G.8000–G.8099
MPLS over Transport aspects	G.8100–G.8199
Quality and availability targets	**G.8200–G.8299**
Service Management	G.8600–G.8699
ACCESS NETWORKS	G.9000–G.9999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T G.8260

Definitions and terminology for synchronization in packet networks

Summary

Recommendation ITU-T G.8260 provides the definitions, terminology and abbreviations used in ITU-T Recommendations on timing and synchronization in packet networks.

History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T G.8260	2010-08-12	15
2.0	ITU-T G.8260	2012-02-13	15

Keywords

Frequency, packet delay variation, phase and time, synchronization definitions.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

		Page
1	Scope	1
2	References	1
3	Definitions	1
	3.1 Terms defined in this Recommendation	1
4	Abbreviations and acronyms	5
5	Conventions	6
6	Description of packet timing concepts	6
	6.1 The nature of packet timing	6
	6.2 Differences between packet-based and physical-layer timing systems	7
	6.3 Classes of packet clocks	8
	6.4 Two-way timing protocols	8
	6.5 PDV measurement	9
	6.6 Packet timing signal equipment interface characterization	10
Appendix I – Definitions and properties of packet measurement metrics		11
	I.1 Introduction	11
	I.2 Definition of the time error sequence	13
	I.3 Packet selection	13
	I.4 PDV metrics estimating the performance of a packet slave clock	18
	I.5 PDV metrics studying floor delay packet population	30
	I.6 Summary of metric classifications	33
Bibliography		34

Recommendation ITU-T G.8260

Definitions and terminology for synchronization in packet networks

1 Scope

This Recommendation provides the definitions, terminology and abbreviations used in Recommendations on frequency, phase and time synchronization in packet networks. It includes mathematical definitions for various synchronization stability and quality metrics for packet networks, and also provides background information on the nature of packet timing systems and the impairments created by packet networks.

Ethernet physical layer methods for synchronization are based on traditional time division multiplexing (TDM) physical layer synchronization and therefore most of the definitions related to these methods are covered by [ITU-T G.810]. Additional definitions are included in this Recommendation.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T G.810] Recommendation ITU-T G.810 (1996), *Definitions and terminology for synchronization networks*.

[ITU-T G.811] Recommendation ITU-T G.811 (1997), *Timing characteristics of primary reference clocks*.

[ITU-T G.8261] Recommendation ITU-T G.8261/Y.1361 (2008), *Timing and synchronization aspects in packet networks*.

[ITU-T G.8261.1] Recommendation ITU-T G.8261.1/Y.1361.1 (2012), *Packet delay variation network limits applicable to packet based methods (Frequency synchronization)*.

[ITU-T G.8263] Recommendation ITU-T G.8263/Y.1363 (2012), *Timing characteristics of packet-based equipment clocks.*

[ITU-T Y.1413] Recommendation ITU-T Y.1413 (2004), *TDM-MPLS network interworking – User plane interworking*.

[IEEE 1588] IEEE Standard 1588-2008, *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*.

3 Definitions

3.1 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.1.1 adaptive clock recovery: Clock recovery technique that does not require the support of a network-wide synchronization signal to regenerate the timing. In this case, the timing recovery process is based on the (inter-)arrival time of the packets, e.g., timestamps or circuit emulation service (CES) packets. The information carried by the packets could be used to support this operation. Two-way or one-way protocols can be used.

3.1.2 arbitrary reference time clock (ARTC): A reference time generator that provides a reference time signal, or simply a reference phase signal, whose frequency has the accuracy of a PRC as specified in [ITU-T G.811], while the epoch does not necessarily have a relationship with an internationally recognized time standard.

3.1.3 coherent time and frequency: The condition where the timing signal carrying frequency and the timing signal carrying time-of-day or phase are traceable back to the same primary source.

3.1.4 floor delay: The notion of "floor delay" is equivalent to the notion of minimum possible transit delay of packets over a network. It may be useful to distinguish the notions of "*absolute floor delay*" and "*observed floor delay*":

- **absolute floor delay**: Absolute minimum possible transit delay of packets of a given size over a network. This may generally be described as the transit delay experienced by a packet that has experienced the minimum possible delay through each network element along a specified path. Depending on loading and other considerations, it is possible that in any given finite window of observation interval a packet with delay equal to this absolute minimum may not be observed. Full knowledge of the packet network, network elements, and routing path must be known in order to perform a theoretical analysis of the minimum transit delay.
- **observed floor delay**: Minimum transit delay of packets of a given size over a network observed over a given observation interval (for instance, during a packet delay variation (PDV) measurement).

  NOTE – As mentioned above, the observed floor delay during a PDV measurement may differ from the absolute floor delay.

3.1.5 floor delay step: The difference between the observed floor delays of two consecutive, non overlapping observation intervals, see Figure 1:

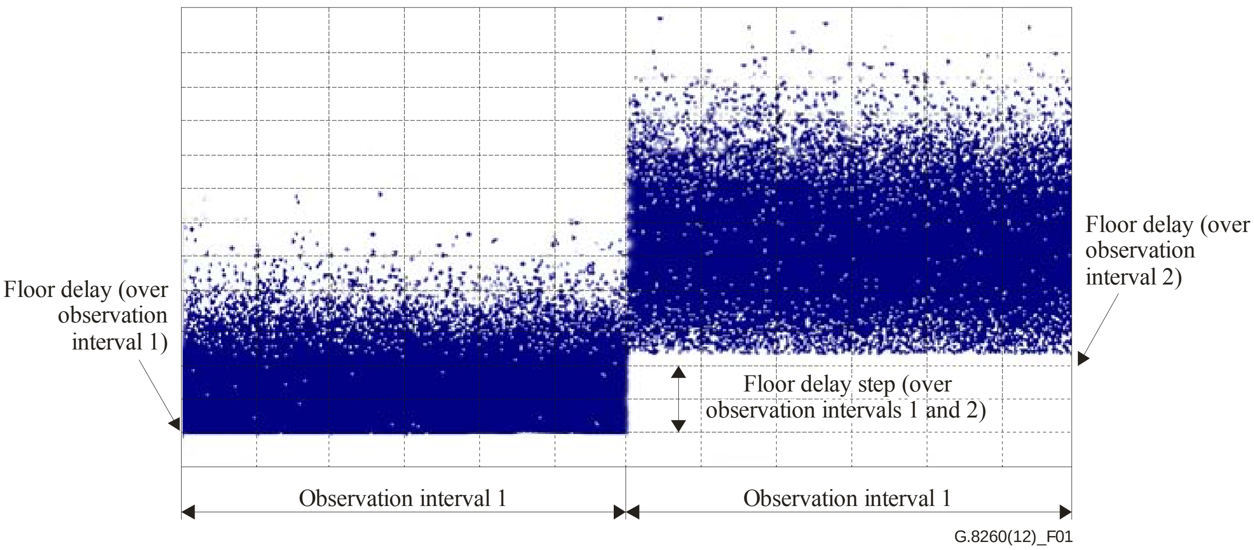

Figure 1 – Illustration of observed floor delays and floor delay step

3.1.6 packet-based method: Timing distribution method (for frequency and/or time and/or phase) where the timing information is associated with packets.

- The frequency can be recovered using two-way or one-way protocols.
- Time and phase information is recovered with a two-way protocol in order to compensate for the transfer delay from packet master clock to packet slave clock.

3.1.7 packet-based method with physical frequency support from the network: Packet based method for time and phase synchronization using frequency support from a traceable network reference clock carried by a physical layer timing trail.

NOTE – For instance, it can correspond to Telecom Boundary Clocks syntonized by a frequency reference carried at the physical layer. This type of support is expected to provide "phase/time holdover" capacities, enabling to maintain phase/time local reference during periods of failure of the phase/time distribution protocol.

3.1.8 packet-based method with timing support from the network: Packet-based method (frequency or time-phase synchronization) requiring that all the network nodes on the path of the synchronization flow implement one of the two following types of functional support:

- termination and regeneration of the timing (e.g., NTP stratum clocks, PTP boundary clock);
- a mechanism to measure the delay introduced by the network node and/or the connected links (e.g., PTP transparent clock) so that the delay variation can be compensated using this information.

3.1.9 packet-based method without timing support from the network: Packet-based method (frequency or time-phase synchronization) where the timing packets are transported over a timing transport agnostic network.

3.1.10 packet master clock: A clock that measures the precise times at which the significant instants of a packet timing signal pass the master's timing reference point (e.g., as they enter the network from the packet master clock, or as they enter the packet master clock from the network). These measurements are done relative to the master clock's local time-scale. They are forwarded to, and used to control, one or more packet slave clocks.

NOTE – In the case of a periodic packet timing signal (used for one-way frequency distribution), the event packets enter the network from the packet master clock at regular intervals, such that the master's timing information is implied from the nominal frequency of the packets.

3.1.11 packet network timing function (PNT-F): The set of functions within the inter-working function (IWF) that supports the synchronization network clock domain (see Figure B.2 of [ITU-T G.8261]). This includes the function to recover and distribute the timing carried by the synchronization network. The PNT-F clocks may be part of the IWF or may be part of any other network element in the packet network.

When the PNT-Fs are part of the IWF, they may support the CES IWF and/or change the layer over which timing is carried (i.e., from packet to physical layer and vice versa).

3.1.12 packet slave clock: A clock whose timing output is frequency locked, or phase aligned, or time aligned to one or more reference packet timing signals exchanged with a higher quality clock.

3.1.13 packet timing signal: A signal, consisting of a series of event packets or frames, that is used to convey timing information from a packet master clock to a packet slave clock.

Event packets in a packet timing signal may travel from a packet master clock to a packet slave clock or vice versa, but the flow of timing information is always in the direction from master to slave.

The significant instants of the packet timing signal are measured relative to the master's local time-scale as they pass the master's timing reference point, and these measurements are communicated to the packet slave clock.

The significant instants of the packet timing signal are also measured relative to the slave's local time-scale as they pass the slave's timing reference point.

NOTE 1 – The significant instants of the signal are the set of times that a defined location in each event packet or frame passes a given reference point in the network (e.g., the interface between the packet master clock and the network). Conventionally the defined location is the end of the start-of-frame delimiter, but it may be defined differently in any given packet timing protocol provided the definition is consistent.

NOTE 2 – In the case of a periodic packet timing signal, the master's timing information is implied from the nominal frequency of the packets.

3.1.14 phase synchronization: The term phase synchronization implies that all associated nodes have access to reference timing signals whose significant events occur at the same instant (within the relevant phase accuracy requirement). In other words, the term phase synchronization refers to the process of aligning clocks with respect to phase (phase alignment). This is shown in Figure 2.

NOTE 1 – Phase synchronization includes compensation for delay between the (common) source and the associated nodes.

NOTE 2 – This term might also include the notion of frame timing (that is, the point in time when the timeslot of an outgoing frame is to be generated).

NOTE 3 – The concept of phase synchronization (phase alignment) should not be confused with the concept of phase-locking where a fixed phase offset is allowed to be arbitrary and unknown. Phase alignment implies that this phase offset is nominally zero. Two signals which are phase-locked are implicitly frequency synchronized. Phase-alignment and phase-lock both imply that the time error between any pair of associated nodes is bounded.

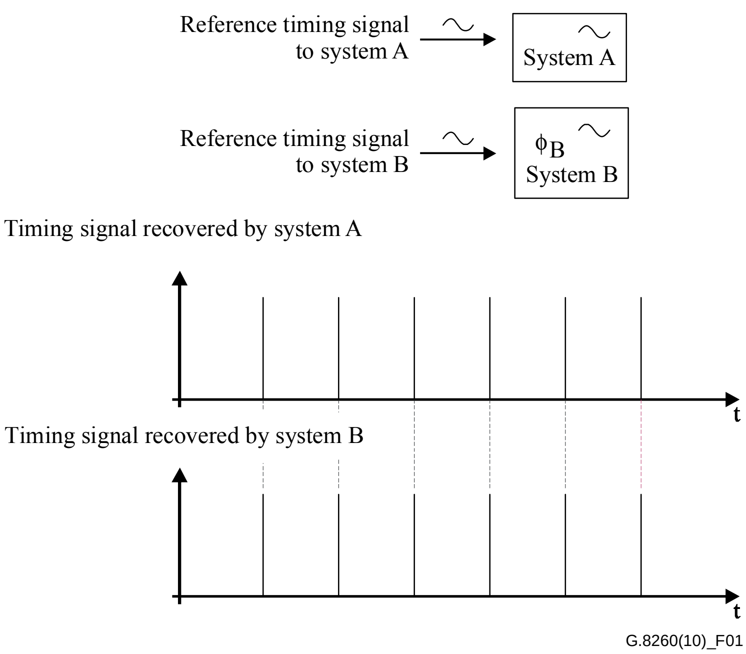

Figure 2 – Phase synchronization

3.1.15 primary reference time clock (PRTC): A reference time generator that provides a reference timing signal traceable to an internationally recognized time standard (e.g., UTC).

3.1.16 time clock: An equipment that provides the elapsed time from a reference epoch.

3.1.17 time synchronization: Time synchronization is the distribution of a time reference to the real-time clocks of a telecommunication network. All the associated nodes have access to information about time (in other words, each period of the reference timing signal is marked and dated) and share a common time-scale and related epoch (within the relevant time accuracy requirement), as shown in Figure 3.

Examples of time-scales are:

- UTC
- TAI
- UTC + offset (e.g., local time)
- GPS
- PTP
- local arbitrary time

Note that distributing time synchronization is one way of achieving phase synchronization.

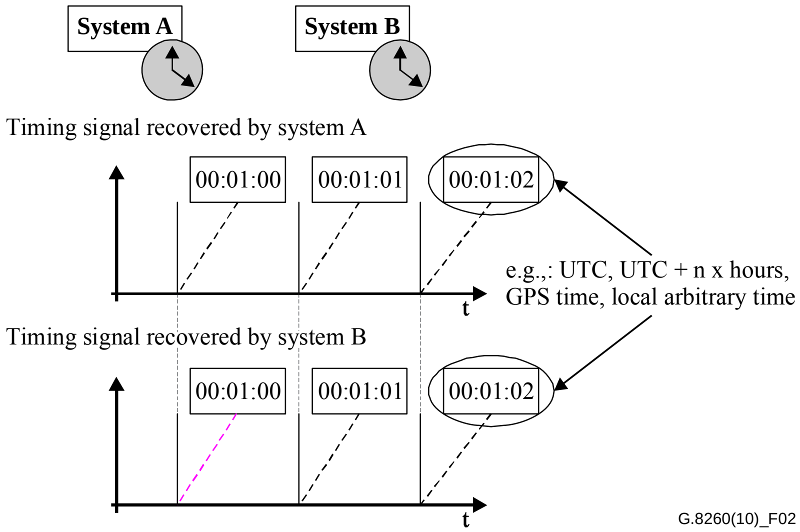

Figure 3 – Time synchronization

4 Abbreviations and acronyms

For the purposes of Recommendations on timing and synchronization in packet networks, the following abbreviations and acronyms apply:

ADEV Allan DEViation

ARTC Arbitrary Reference Time Clock

CES Circuit Emulation Services

FFO Fractional Frequency Offset

FM Frequency Modulation

GPS Global Positioning System

IWF Inter-Working Function

MAFE Maximum Average Frequency Error

MATIE Maximum Average Time Interval Error

MDEV Modified Allan DEViation

MTIE	Maximum Time Interval Error
NTP	Network Time Protocol
PDV	Packet Delay Variation
PM	Phase Modulation
PNT-F	Packet Network Timing Function
PRC	Primary Reference Clock
PRTC	Primary Reference Time Clock
PTP	Precision Time Protocol
TAI	International Atomic Time
TDEV	Time DEViation
TDM	Time Division Multiplexing
TIE	Time Interval Error
UTC	Coordinated Universal Time

5 Conventions

No conventions are used in this Recommendation.

6 Description of packet timing concepts

6.1 The nature of packet timing

A simplistic view of a generic slave clock is that it takes frequency information in, and puts frequency information out, as shown in Figure 4:

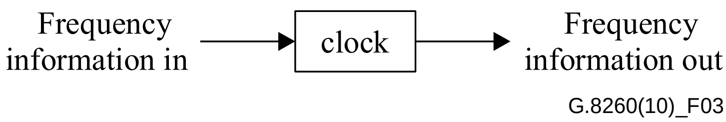

Figure 4 – Generic slave clock

Conventionally, this frequency information is encoded as a timing signal. This is typically implemented as a periodic digital signal, where the edges of the signal are reference points in time known as the "significant instants" of the signal. Timing jitter and wander causes these significant instants to vary slightly from their ideal position in time, i.e., they may not occur at precisely equally spaced points in time. A physical-layer timing signal is shown in Figure 5:

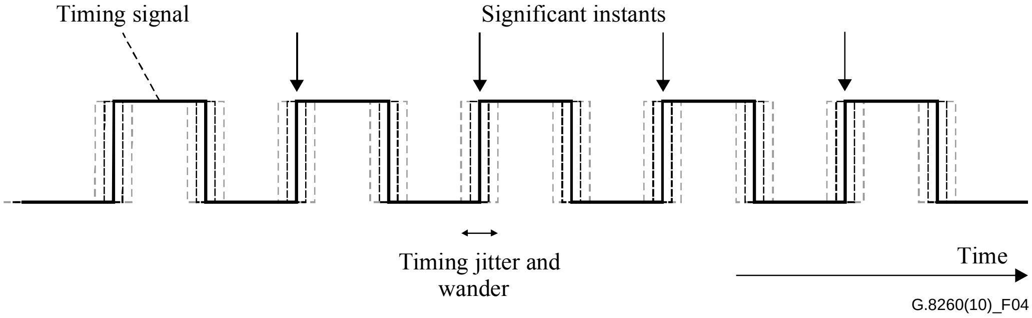

Figure 5 – Physical-layer timing signal

A packet timing signal is similar in concept. The frequency is encoded as a series of time-critical packets in a network, known as "event packets". While the transmission medium is different (packets on a network as opposed to signals on a wire), the packets still contain significant instants (normally the front edge of the packet), with a defined ideal position in time. The variation of the significant instants around their ideal position is termed "packet delay variation" (PDV). This is shown in Figure 6:

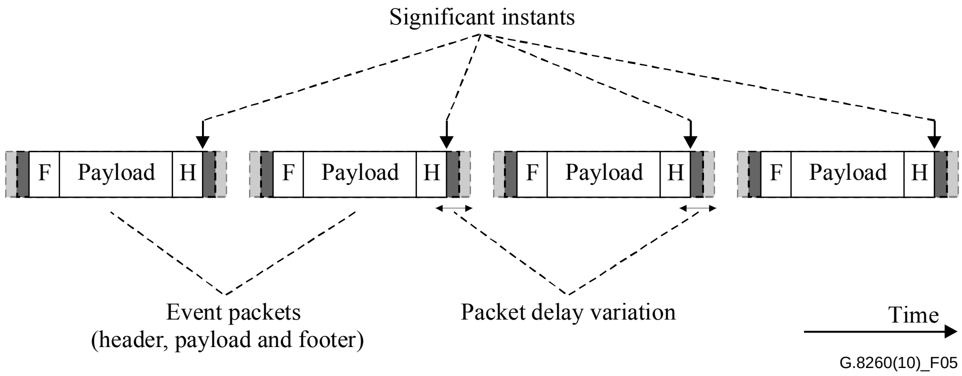

Figure 6 – Packet timing signal

Some of the causes and characteristics of packet delay variation and other impairments that may be introduced by the packet network are discussed in clause 10 of [ITU-T G.8261].

Some packet timing signals may be periodic (e.g., circuit emulation packets containing constant bit rate data), and for these the ideal position in time is implicitly given by the packet rate. Other packet timing signals are not periodic (e.g., PTP or NTP), and for these the ideal position in time is given by a timestamp embedded in the packet data. It is important to note that both periodic and non-periodic packet timing signals are still time domain signals. It is the position in time of the packets that is significant, not the contents of the packets.

6.2 Differences between packet-based and physical-layer timing systems

Packet-based timing systems are not fundamentally different from physical-layer timing systems. Conceptually, both utilize timing signals which are sequences of periodic or timed events, termed "significant instants", where there is a notion of the "ideal position in time" for each event. Similarly, after transmission of these timing signals through the network, there will be some phase noise component, corrupting this "ideal position in time". The recovery of the original timing signal is achieved by filtering the incoming timing signal to remove the transport-related phase noise and generate a clean output.

However, there are some differences which lead to packet timing signals having different characteristics to physical-layer timing signals:

• Rate of significant instants:

  The packet rate in a packet timing signal is much lower than the frequency of most physical-layer timing signals. For example, in PTP (defined in [IEEE 1588]), the sync message rate will normally be in the range 1-128 Hz, while a conventional E1 timing signal has a frequency of 2.048 MHz.

  Secondly, the packets that form the significant instants need not be sent at precisely regular intervals. While the mean rate is specified, the intervals between packets may vary. Timestamps are used to identify the precise sending time, relative to a pre-determined epoch.

- Amplitude and nature of noise processes:

  The principal cause of noise in a packet timing system is packet delay variation (PDV). The amplitude and distribution of PDV is much larger than jitter and wander in physical layer timing systems, and it may contain very low frequency components such as diurnal wander due to network loading variations.

  Unlike physical layer noise, the PDV depends not only on the physics of components but also on the architecture and implementation of network elements. Therefore, the noise is more complex and harder to model.

6.3 Classes of packet clocks

There can be several classes of packet-based clocks, depending on the combination of input and output timing signal classes. Table 1 shows the different classes, with real-world examples of each case.

Table 1 – Classes of packet-based clocks

Packet-based clock class	Input timing signal	Output timing signal	Examples
Packet master clock PEC-M	Physical-layer timing signal	Packet timing signal	PTP master, NTP server Ingress CES IWF (Note 1)
Packet slave clock PEC-S	Packet timing signal	Physical-layer timing signal	PTP slave, NTP client Egress CES IWF (Note 2)
Combined packet slave clock and packet master clock	Packet timing signal	Packet timing signal	PTP boundary clock NTP stratum *n* server (*n* > 1)
NOTE 1 – i.e., TDM to packet direction, see term "ingress IWF" in [ITU-T Y.1413]. NOTE 2 – i.e., packet to TDM direction, see term "egress IWF" in [ITU-T Y.1413].			

6.4 Two-way timing protocols

Packet timing signals may flow from packet master clock to packet slave clock or vice versa. However, in each case, the flow of timing and synchronization is always from master to slave.

In the case of a packet timing signal flowing from a packet master clock to a packet slave clock (e.g., the PTP *sync* messages), the time of exit of each event packet from the packet master clock is measured (to be precise, the time relative to the master's time-scale at which the significant instant of each event packet passes the master's timing reference point). This information is sent to the packet slave clock either in a timestamp embedded in the event packet, or in a subsequent information packet (e.g., the PTP *follow_up* message).

On reception at the packet slave clock, the arrival time of the event packet is measured (to be precise, the time relative to the slave's local time-scale at which the significant instant of each event packet passes the slave's timing reference point). The two times are compared, creating a series of time differences. These time differences are then filtered, and may be used to control the frequency of the output timing signal.

In the case of a packet timing signal flowing from a packet slave clock to a packet master clock (e.g., the PTP *delay_request* messages), the time of exit of each event packet from the packet slave clock is measured. On reception at the packet master clock, the arrival time of each event packet is measured, and this information is sent to the packet slave clock in a subsequent information packet (e.g., the PTP *delay_response* message).

The use of a two-way timing protocol (such as PTP or NTP) makes it possible to align the local time-scale to the master time-scale. The four times may be used to calculate the round-trip delay of the message exchange, and hence to calculate the time offset between the local and master time-scales.

The timing message exchange is shown in Figure 7:

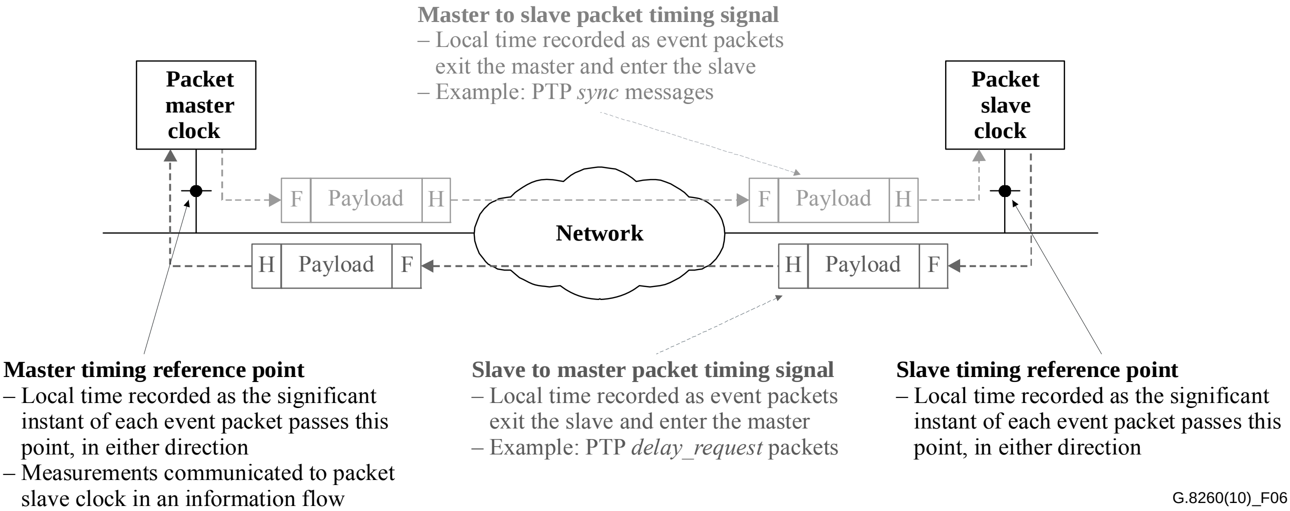

Figure 7 – Packet timing signal flow and timing reference points

6.5 PDV measurement

In general, a packet delay variation (PDV) measurement involves comparing time instants on a sequence of packets, such as those of a packet timing signal, as they pass two points in the network. A configuration for performing such a measurement is shown in Figure 8 below. For each packet, a difference is computed between the time instant taken at the point of origin and the time instant taken at the point of destination.

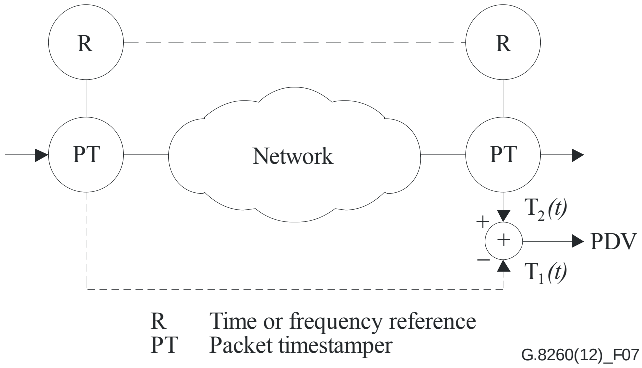

Figure 8 – Configuration for PDV measurement

An ideal configuration for making this measurement places two references traceable to a common time standard at each of the two measurement points. Such a configuration assesses not only the variation of packet delay, but also the packet transit time.

In many circumstances, such as packet-based frequency synchronization, the focus is on variation of packet delay rather than absolute packet delay. In such a case, frequency references can be employed for the references *R* and a common time reference is not required.

The use of unstable or inaccurate references directly impacts the PDV measurement, and may lead to limitations regarding the length of the PDV measurement. If the references are frequency standards, packet delay variation can be studied with the same precision as for the case where the references are time standards. If practical, a common frequency standard should be used for both *R* references. In other cases, separate primary reference clocks could be used.

The probe function could be implemented as separate equipment, or in the case where the first measurement point is at the source of the packet timing signal of interest, integrated into that equipment. In this case, the time instant could be delivered within the packet in the form of a timestamp. Similarly, in the case where the second measurement point is at the destination equipment, the probe function may be integrated into that equipment. Any inaccuracy of the timestamping function in the probes directly impacts the precision of the PDV measurement.

In the case where packets are sent according to a schedule that is known in advance, such as packets spaced by a uniform interval of time (e.g., CES), the relative origin timestamps are implicit, and the packet delay variation measurement can be performed with time stamping at the destination node.

6.6 Packet timing signal equipment interface characterization

The configuration described in clause 6.5 for measuring packet delay variation can be extended to measurement of the packet timing signal at an equipment interface. In this case, the packet timestamper with reference is connected directly to the packet timing signal interface with no intervening network.

A configuration for performing such a measurement is shown in Figure 9 below. For each packet, a difference is computed between the timestamp from the device and the timestamp taken on that same packet from the packet timestamper (PT) with reference (R).

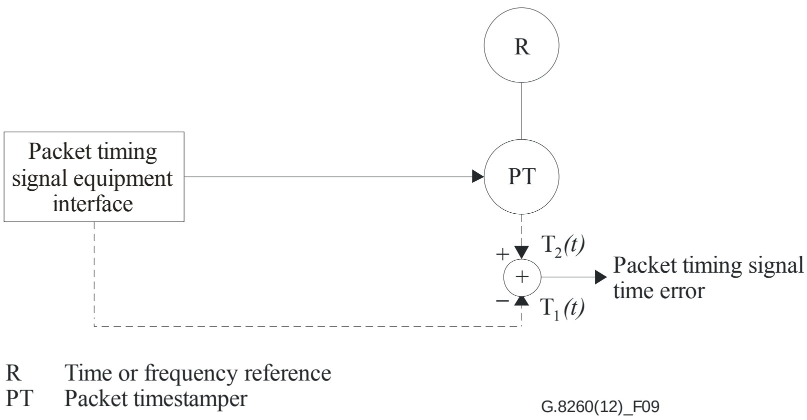

Figure 9 – Configuration for packet timing signal equipment interface measurement

Requirements on the accuracy of the reference (R) are driven by the characteristics of the packet timing signal, and in many cases might exceed those for studying network PDV. If the packet timing signal is derived directly from a primary reference, reference (R) would need to be a primary reference, ideally one with greater stability. Further, in cases where the device under study takes an external reference directly or is traceable to an external reference, the optimal configuration is for both the device and the packet timestamper (PT) in Figure 9 above to share the same reference (R).

Appendix I

Definitions and properties of packet measurement metrics

(This appendix does not form an integral part of this Recommendation.)

NOTE – This appendix contains information related to ongoing studies on the definition of suitable PDV metrics. The text below is for information only, and may be revised in a future version of the Recommendation. It must not be used as normative text, nor as an implied specification of a packet slave clock.

I.1 Introduction

With the telecommunications industry evolving and rapidly adopting packet technology, much emphasis has been placed on addressing packet synchronization and timing, including the use of measurement data to assist in specifying the performance of packet-based clocks.

Physical-layer timing signal stability quantities, including metrics such as MTIE and TDEV, have been used extensively and are central to synchronization measurement analysis. For a packet clock, the level of stability at the clock's packet network input has a direct bearing on the stability of the clock output.

In terms of the packet metrics, the goal of a first category of PDV metrics, introduced in clause I.4, is to formulate packet-based stability quantities (metrics) that will provide a means of estimating the physical-based stability quantities for the packet clock output. This is illustrated in Figure I.1:

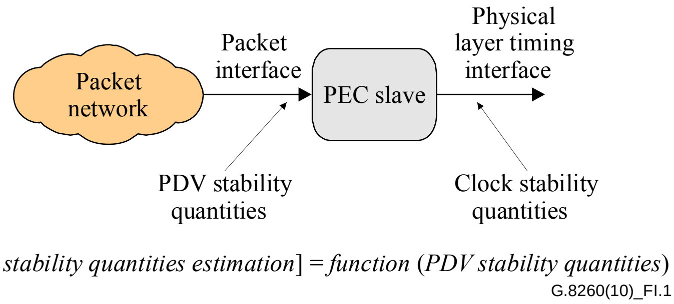

Figure I.1 – Packet equipment clock interfaces

A second category of PDV metrics is also introduced in this appendix, in clause I.5. The goal of this second category is not directly to provide an estimation of the physical-based stability quantities for the packet clock output, but simply to study the population of timing packets within a certain delay window range.

PDV measurement guidelines are provided in clause 6.5.

For packet measurement data analysis, packet selection is added as an important component to the analysis. Indeed, in order to reduce the input PDV noise, the packet slave clock implementations are generally using only a subset of the received timing packets.

Therefore, a first simple approach to analyse the PDV as received by a packet slave clock can be to display the measured PDV in the form of a histogram. It generally provides useful information about the population of packets in different delay regions, and is in some cases sufficient to analyse the network conditions. Figure I.2 shows an example plot of the measured PDV and the corresponding histogram.

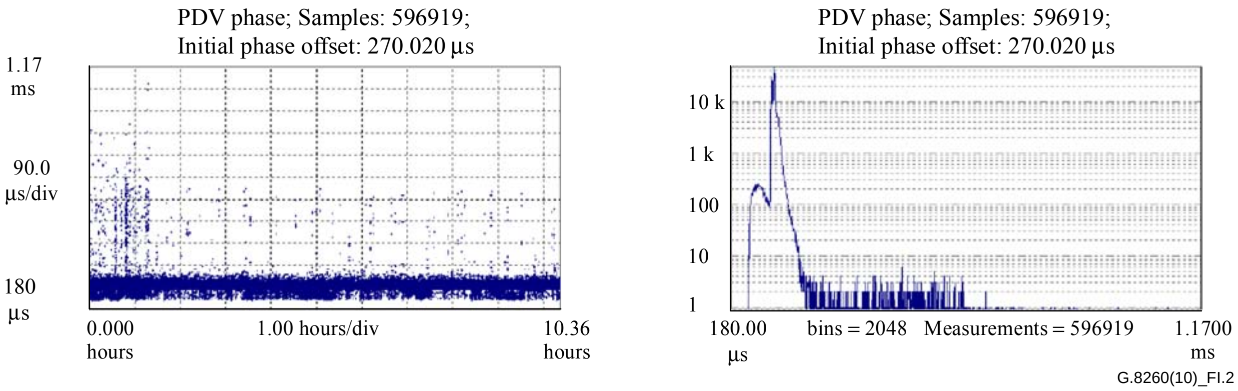

Figure I.2 – Measured packet delay and corresponding PDV histogram

In a second approach, mathematical tools (called "metrics" in this appendix) can be applied on a given PDV measurement to analyse it more in detail. Those metrics generally use only a subset of the packets. The packet selection can be either integrated into the calculation or performed as a pre-processing step. For example, the packet selection can focus on the minimum packet delay floor or more generally on some other region of packet delays.

With regard to the packet selection just discussed, it is important to point out the link between the methods of packet measurement data analysis described here and packet clock algorithms as they exist in actual equipment. For both, packet selection is important for optimization given the realities of packet delay variation.

However, it is important to mention that due to the proprietary nature of most of the packet slave clock implementations today, especially regarding the packet selection criteria, the packet selection used by a given PDV metric may not correspond to the criteria used in the packet slave clock of interest. Therefore, there can be some discrepancies between the information provided by a given PDV metric and the real performances achieved by a packet slave clock.

How to align the results provided by the PDV metrics and the performance of the packet slave clock is still under study. It may imply the specification of some minimum common behaviour in the packet selection criteria in the packet slave clock implementations.

Moreover, it is important to mention that PDV metrics compute an estimate of achievable performance through the use of PDV sample information only, and do not consider the effects of internal oscillator noise in a packet slave clock. Non-negligible differences between the estimate and the actual performance of a packet slave clock may sometimes be observed because of this effect. In order to take oscillator noise into account, the noise generation components of a packet slave clock are considered in Recommendation [ITU-T G.8263].

While metrics can provide the basis for setting equipment requirements and network limits, their value as general analysis tools leading to insight into particular sets of measurement data should not be overlooked. For TDM synchronization measurements, normative limits have been applied to the MTIE and TDEV calculations, but other metrics such as ADEV and MDEV, while not associated with normative telecom limits, have great utility as analysis tools.

In the following discussion two main classes of PDV metrics can be identified:

a) *PDV metrics useful for specifying PDV network limits*: PDV metrics providing a quantitative indication of whether characteristics of the originating clock are sufficiently present to allow recovery of the originating clock. To achieve this, the PDV metric must provide an indication of whether a given performance, expressed in terms of traditional performance metrics (metrics for physical based timing signals), is achievable at the output of a packet slave clock. This can be achieved by means of the specification of a mask defining the PDV tolerance of a packet clock. This class of metrics is required by network operators in order to understand whether the packet network is suitable for carrying timing via packets, that is, to check whether the PDV noise produced by a packet network is within the minimum tolerance of packet slave clocks as specified in [ITU-T G.8263].

b) *PDV metrics useful for studying the characteristics of the network, but not for specifying PDV network limits*: PDV metrics characterizing specific aspects of a packet network's typical behaviour (e.g., in terms of PDV statistics). This class of metrics can be useful for the purpose of getting a better understanding on how packet networks behave, for instance indicating which method of timing packet pre-selection is more suitable for timing recovery. This information can be used, for instance, by packet slave clock vendors for optimizing the design of clock recovery algorithms (e.g., how to select timing packets) beyond the minimum specification of a packet slave clock in [ITU-T G.8263]. However, this class of metrics is not suitable for specifying PDV network limits.

The metrics related to class A are used in [ITU-T G.8261.1] to specify the PDV network limits, while the metrics related to class B are for information only.

I.2 Definition of the time error sequence

For packet timing signals the packet time-error sequence can be established in the following way. For specificity, consider the transfer of timing packets originating at the packet master clock and terminating at the packet slave clock. In the case of PTP (see [IEEE 1588]) the rate of packets, f_0, is determined via negotiation between master and slave and can be as high as f_0 = 128 packets/s. PTP packets may not be equally spaced, but will meet this nominal rate over the long term. The ideal position in time for these packets is given by a timestamp embedded in the packet data.

Packets leave the master with a long-term mean spacing of $\tau_0 = 1/f_0$. From a signal processing perspective, the sampling rate is f_0 and an arbitrary mathematical-time origin for describing the times of departure from the master can be chosen. With this choice of time origin, the kth packet departs the master at time $t = k{\cdot}\tau_0$. In practice the kth packet will depart at time T_k which is but approximately equal to $k{\cdot}\tau_0$. Note that in the case of circuit emulation the times of departure are considered to be exactly spaced by τ_0. The kth packet then arrives at the slave at time S_k, where:

$$\begin{aligned} S_k &= T_k + \Delta + \varepsilon_k \\ e_k &= S_k - T_k \end{aligned} \qquad \text{(I-1)}$$

where Δ is the reference transit delay time and ε_k is the transit delay time variation (i.e., packet delay variation or PDV). For the calculation of some PDV metrics, the operation may involve differencing and consequently the reference transit delay time, Δ, is moot since it is part of every term. Consequently, for purposes of calculating these PDV metrics, e_k can be used as the packet delay variation and be used interchangeably with ε_k. The same principle applies for packets that traverse the network from the slave to the master.

If the (hypothetical) packet time-error signal $x(t)$ is considered, then the sample of $x(t)$ taken at the sampling instant T_k is none other than e_k. That is, the sequence $\{e_k\}$ is equivalent to the packet time-error sequence but on a non-uniform time grid. The normal packet time-error sequence, $\{x_k\}$, is actually the sequence generated by sampling the packet time-error signal $x(t)$ on a uniform grid with sampling interval τ_0.

I.3 Packet selection

Physical layer timing signals are stationary and Gaussian in nature. Therefore, the relevant applied stability quantities (i.e., MTIE and TDEV) will usually use every noise sample point (significant instant) in the stability quantification process in order to filter out as much noise as possible and achieve the best stability quantification possible.

Packet-based timing signals, on the other hand, are not always stationary or Gaussian in nature. Hence, methods of quantifying them (thus attaining a better estimation of their ability to carry timing information) would usually require selecting only a subset of their entire population or in general performing some pre-filtering before applying the specific stability quantification analysis. The following discussion focuses on the approach that involves a selection of packets.

I.3.1 Packet selection types

As mentioned above in the introduction, when applying some PDV metrics, packet selection can be incorporated into the calculation or into a pre-processing step.

- The packet selection techniques integrated into the calculation are useful in metrics that are intended to determine the characteristics of the packet network in terms of the packet delay variation behaviour. The main benefit of this approach is to provide a generic tool independent of the characteristics of a specific packet slave clock implementation (e.g., time interval used to select packets). The main purpose of this approach is therefore to support vendors with progressing packet timing recovery techniques (class B metrics).
- On the other hand, pre-processing selects packets from suitable, pre-defined time window lengths. Therefore, the selection process resembles that of a practical packet clock in steady state operation. This approach is therefore more suitable for the definition of metrics used to specify network limits (i.e., class A metrics) as some assumption is made on a "minimum" implementation of a packet slave clock as specified in [ITU-T G.8263].

I.3.1.1 Pre-processed packet selection

With pre-processed packet selection, quantifying packet timing signals is carried in two steps:

1) Applying a specific packet selection procedure to select a specific subset of PDV noise samples, having similar delay properties, among the entire population of PDV noise samples.
2) Applying the required stability quantification algorithm (metric) over the selected group of samples to get an estimation of the achievable output clock quality estimation.

NOTE – As mentioned earlier, there can be some discrepancies between the information provided by a given PDV metric and the real performances achieved by a packet slave clock.

This is shown in Figure I.3:

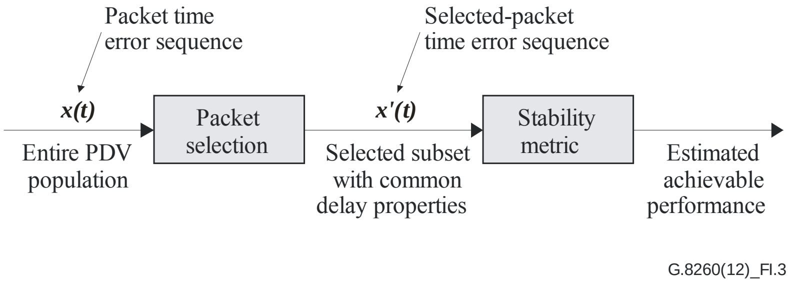

Figure I.3 – Pre-processed packet selection

In essence, an input packet time-error sequence $x(t)$ is subject to packet selection which produces a new packet time-error sequence $x'(t)$. The input packet time-error sequence is divided into time windows of equal length. A fraction of packets are selected from each window in a similar manner and the information is combined so that each window produces a single value to the new packet time-error sequence. Unless otherwise specified, the time errors of the selected packets in each selection window are averaged to produce a single delay value.

When a metric is computed using pre-processed packet selection, the name of the metric is prepended by the term *pktselected*, e.g., *pktselectedTDEV* (see clause I.4.1.1).

In the case of pre-processed packet selection, the preliminary packet selection process is independent of the applied stability quantification. Thus, different combinations of the two might yield interesting properties and need to be looked into. Both need to be fully defined as each has significant influence on the resulting performance measurement.

I.3.1.2 Integrated packet selection

With integrated packet selection, the packet selection is integrated into the metric calculation, as shown in Figure I.4. Generally, this involves replacing a full population averaging calculation with a selection process that may or may not include averaging.

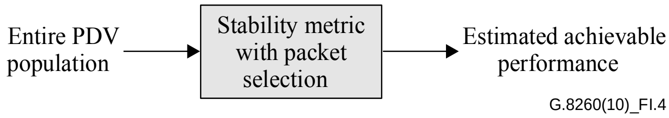

Figure I.4 – Integrated packet selection

NOTE – As mentioned earlier, there can be some discrepancies between the information provided by a given PDV metric and the real performances achieved by a packet slave clock.

I.3.2 Packet selection methods

Four examples of packet selection methods are described in the clauses that follow. The first two, minimum packet selection and percentile packet selection, focus on packet data at the floor. The second two, band packet selection and cluster range packet selection, can be applied either at the floor or at some other region.

I.3.2.1 Minimum packet selection method

The minimum packet selection method involves selecting a minimum within a section of data. This can be represented as follows:

$$x_{\min}(i) = \min[x_j] \text{ for } (i \le j \le i+n-1) \qquad \text{(I-2)}$$

I.3.2.2 Percentile packet selection method

The percentile packet selection method is related to the minimum packet selection method, except that instead of selecting the minimum, some number (or some percentage) of minimum values are chosen and averaged together. It is a special case of the band packet selection method described below with the lower index set to zero.

I.3.2.3 Band packet selection method

The band packet selection method can be used to select a section of packet data at the floor or from some other region such as the ceiling or somewhere else above the floor. To perform the band packet selection, it is first necessary to represent the sorted packet time-error sequence. Let x' represent this sorted phase sequence from minimum to maximum over the range $i \le j \le i + n - 1$. Next, it is necessary to represent the indices which are themselves set based on the selection of two percentile levels.

Let a and b represent indices for the two selected percentile levels. The averaging is then applied to the x' variable indexed by a and b. The number of averaged points m is related to a and b: $m = b - a + 1$.

$$x'_{band\_mean}(i) = \frac{1}{m}\sum_{j=a}^{b} x'_{j+i} \qquad \text{(I-3)}$$

A percentile level is selected by using rounding to find the closest index from the desired percentile value. The additional constraint is that the index value has a minimum of the first index and a maximum of the last index. Thus, for example, a set of ten points with a percentile set to 2% (0.02) would be set to the minimum index so that at least a single point would be selected.

I.3.2.4 Cluster range packet selection method

The cluster range packet selection method uses a time/phase-bounded range rather than indices based on percentiles (probabilities) to perform the packet selection. This selection method involves the selection of a group of one or more packets which are closely related with respect to their transit time. The location of the cluster may be made based on various criteria, for example, packets at the floor or from some other region observed in the window interval, or the location of the cluster may be based on other criteria or information outside the interval. The cluster of packets could then be processed in a variety of ways to generate a single value for that interval, such as the mean transit time of all packets within the cluster.

Figure I.5 shows an example packet delay sequence, zooming in on an example of a packet cluster for a single window interval.

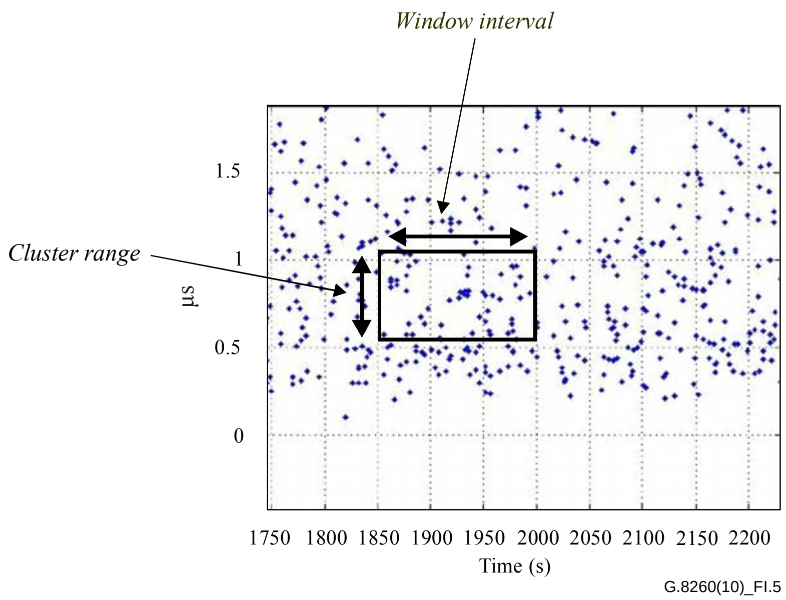

Figure I.5 – Example of concept of cluster range packet selection

The cluster selection method involves the following pre-determined choices:

1) The range of the packet transit times accepted within a cluster is set and is related to the target performance. That is, the range can be chosen to best serve the application for which the clock is intended. This is the *cluster range,* δ, and is specified in units of time.

2) The selection window interval for the cluster is set.

3) The cluster location or cluster anchor value, $a(n)$, within the overall distribution of packet delays is variable and can be programmed to best characterize the type(s) of noise introduced by the packet network. That is, the optimal time error is obtained when the cluster selection method identifies the packets that represent the most stable transit delay. The specific cluster anchor type may be considered as the cluster rule, denoted as $clusterType = rule$ or $clusterType_{rule}$.

Denote the time error sequence of the packet timing signal at the slave clock packet interface by $\{x(k\tau_P)\}$. That is, the underlying sampling interval (nominal packet interval) is τ_P. The cluster selection method considers K samples (packets) using a fixed-window processing architecture and generates a new time error sequence $\{x'(n\tau_0)\}$ where $\tau_0 = K \cdot \tau_P$. Note that the sample value $x(n\tau_0)$ is based on the K input samples $\{x(m\tau_P);\ m = nK, (nK+1), \ldots, ((n+1)K-1)\}$. This sample value can be expressed as:

$$x'(n\tau_0) = \frac{\sum_{i=0}^{(K-1)} x\big((nK+i)\tau_P\big)\cdot\phi(n,i)}{\sum_{i=0}^{(K-1)} \phi(n,i)} \qquad \text{(I-4)}$$

where in equation (I-4), $\phi(\ )$ is the indicator function that expresses the selection mechanism in a mathematical manner and is given by:

$$\phi(n,i) = \begin{cases} 1 & \text{for } \left|x\big((nK+i)\tau_P\big) - a(n)\right| \le \frac{\delta}{2} \\ 0 & \text{otherwise} \end{cases} \qquad \text{(I-5)}$$

In equation (I-5), δ is the cluster range and $a(n)$ is the *anchor value* for the particular window of K samples. Note that equation (I-4) generates the new time error sequence by computing the average over those packets that satisfy the selection rule.

The *anchor value* can be interpreted as a nominal value for the window and is established according to a pre-determined cluster type. For example, the selection rule for this *anchor value* could be:

- Minimum transit delay over the K packets in the window, represented as *clusterType = min* or $clusterType_{min}$
- Average (mean) transit delay over the K packets in the window, represented as *clusterType = mean* or $clusterType_{mean}$
- An absolute minimum value that may be determined before, during or after the sample window, represented as *clusterType = min_absolute* or $clusterType_{min\_absolute}$

  When using an absolute value it is possible that no packets may be selected within the window (similar to a total packet loss situation). Note that the determination of an absolute minimum value after the sample window (as opposed to before or during) would only be used in post-analysis situations, as the information regarding the future packet delay transit times is not available to the client in a real-time system.

It is common to refer to the *anchor value* as the transit delay of a particular packet that is then called the *anchor packet*. This is generally true except when the *anchor value* is not associated with a particular packet (e.g., mean value or absolute minimum value).

It may be helpful to use the representation *cluster* (δ, *clusterType*) where δ is the cluster range, and the *clusterType* is an indication of the rule used to generate the *anchor value*.

I.3.3 Consideration of non-stationary network conditions

As the packet selection can focus on a particular statistical region, it is important to consider the case where network packet delay statistics are not stationary, but rather change over time. For example, if a floor-based metric is applied to packet measurement data where the floor shifts, the application of the floor-based metric would perhaps be best applied to sections of the data separately (see Figures I.6 and I.7). In many cases, segregating the data into sections might not be so straightforward, such as the case of an increasing load ramp. Such a situation is for further study.

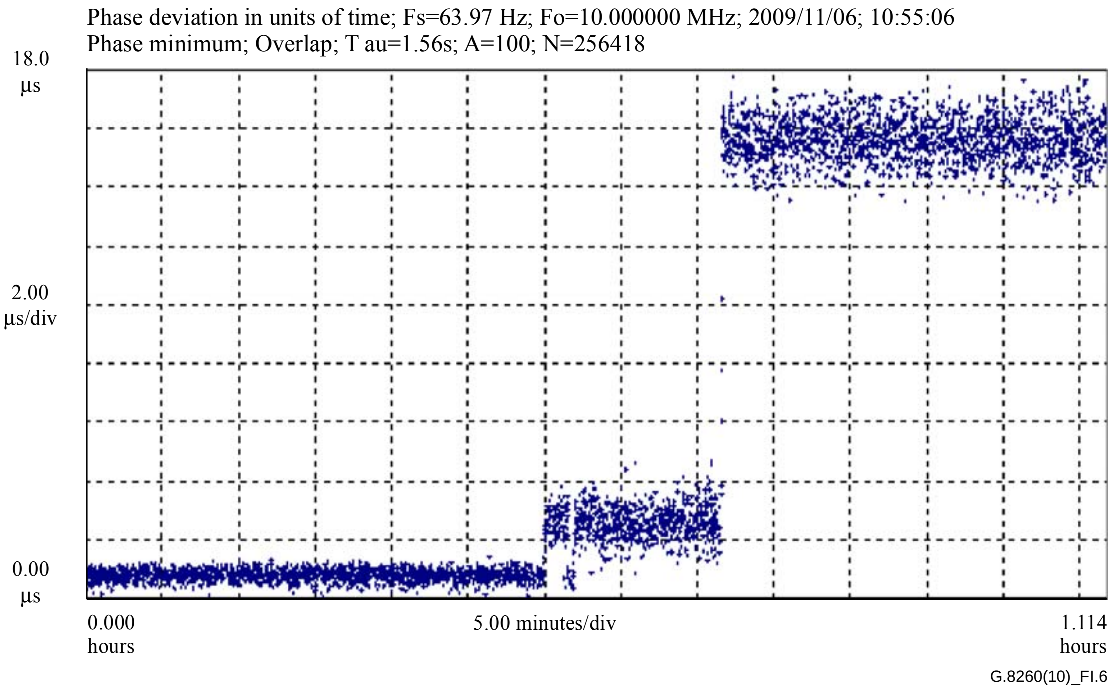

Figure I.6 – Minimum tracking statistic shows three distinct sections

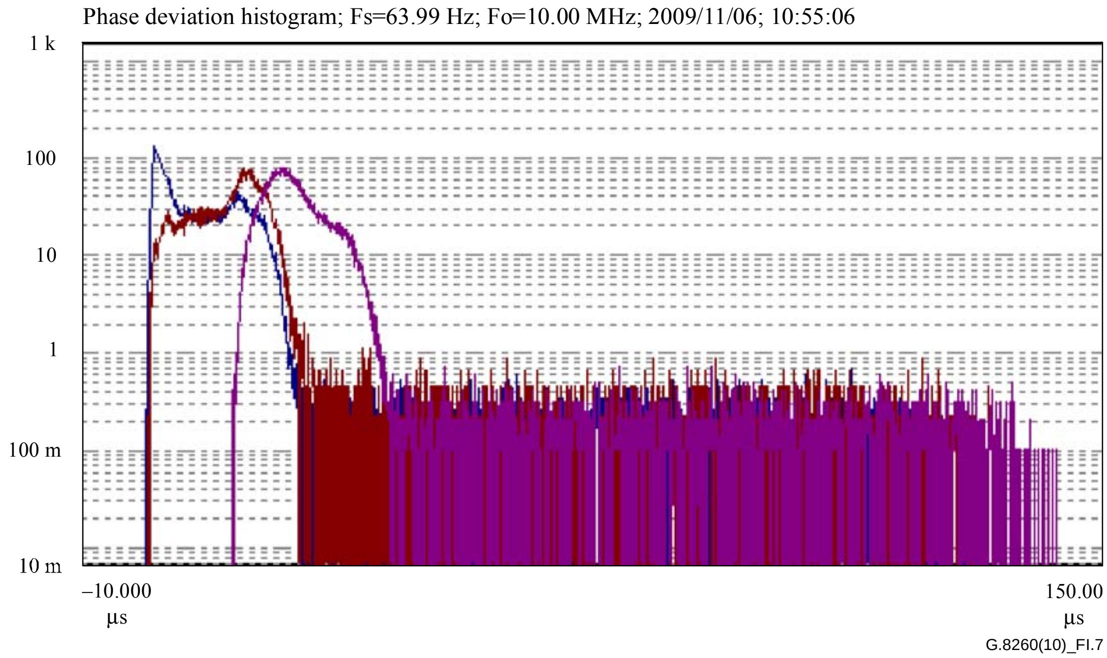

Figure I.7 – Histograms (PDFs) for the three sections

I.4 PDV metrics estimating the performance of a packet slave clock

The following clauses describe the stability metrics and a few specific associations with packet selection that have been studied for quantifying packet network timing signals. Two main approaches can be defined when applying a PDV metric:

- Metrics without pre-filtering

  Metrics without pre-filtering are intended to determine the characteristics of the packet network in terms of the packet delay variation behaviour. One of the advantages of this approach is to provide a generic tool for analysis of the packet delay variation independent of the filtering bandwidth of a specific packet slave clock implementation. One of the main purposes of this approach is therefore to support vendors with progressing packet timing recovery techniques.

- Metrics including pre-filtering

  These metrics include a pre-filtering process prior to the calculation of the actual metric (see box "Bandwidth Filtering" in Figure I.11). The use of a metric that includes pre-filtering resembles that of a simplified model of a packet clock in steady state operation. This approach is therefore suitable for the definition of metrics used to specify network limits as some assumption is made on a "minimum" implementation of a packet slave clock as specified in [ITU-T G.8263].

I.4.1 Metrics without pre-filtering

I.4.1.1 TDEV

TDEV has been specified in [ITU-T G.810] and used in other Recommendations to specify network wander limits for physical timing signals. TDEV is also applicable to packet timing data. In relation to packet timing data, TDEV can be applied to pre-processed PDV data or integrated into the calculation. The case where TDEV is applied to pre-processed PDV data, which can be referred to as *pktselectedTDEV*, is depicted by Figure I.3 with TDEV as the stability metric.

The integrated methods based on TDEV include minTDEV, percentileTDEV, and bandTDEV. The minTDEV and percentileTDEV metrics focus the packet selection on the minimum packet delay floor and the more general bandTDEV metric can select packet delays from any region, for example, the floor, just above the floor, in the middle, or the ceiling. The integrated methods can be applied for MATIE and MAFE metrics described further below but are described in depth only in the TDEV clause.

Like the TDEV metric, the TDEV-based packet measurement metrics study the noise processes in the packet measurement data – white PM, flicker PM, random walk PM, flicker FM, and random walk FM. With the incorporation of packet selection, it is often possible that one or more of these noise processes can be reduced as compared to analysis incorporating no selection.

I.4.1.1.1 MinTDEV

Definition

The minTDEV operator has been defined based on the TDEV metric. The TDEV metric is shown below in equation (I-6):

$$\sigma_x(\tau) = \mathrm{TDEV}(\tau) = \sqrt{\frac{1}{6n^2}\left\langle\left[\sum_{i=1}^{n}(x_{i+2n} - 2x_{i+n} + x_i)\right]^2\right\rangle} \qquad \text{(I-6)}$$

The TDEV operator is based on the mean of the sample window (equation I-7):

$$x_{mean}(i) = \frac{1}{n}\sum_{j=0}^{n-1} x_{j+i} \qquad \text{(I-7)}$$

Compared with the TDEV operator, in the minTDEV operation the mean of the sample window is replaced with the minimum of the sample window as shown in equation (I-8):

$$x_{\min}(i) = \min\,[x_j] \text{ for } (i \le j \le i+n-1) \qquad \text{(I-8)}$$

Substituting equation (I-8) back into original TDEV definition yields the definition of minTDEV(τ) (with $\tau = n\tau_0$):

$$\sigma_{x\_\min}(n\tau_0) = \text{minTDEV}(n\tau_0) = \sqrt{\frac{1}{6}\left\langle \left[x_{\min}(i+2n) - 2x_{\min}(i+n) + x_{\min}(i)\right]^2 \right\rangle}\,, \quad \text{(I-9)}$$

for $n = 1, 2, ...,$ integer part $\left(\frac{N}{3}\right)$

where the angle brackets denote ensemble average.

Estimator formula

minTDEV($n\tau_0$) may be estimated by:

$$\text{minTDEV}(n\tau_0) \cong \sqrt{\frac{1}{6(N-3n+1)} \sum_{i=1}^{N-3n+1} \left[x_{\min}(i+2n) - 2x_{\min}(i+n) + x_{\min}(i)\right]^2} \quad \text{(I-10)}$$

for $n = 1, 2, ...,$ integer part $\left(\frac{N}{3}\right)$

Usage

The minTDEV operator has been indicated as a useful tool when in combination with packet networks that exhibit a PDV behaviour, where it is possible to identify a suitable set of packets with packet delay variation close to a minimum delay.

In fact, these packets are less impacted by the queuing delays, and therefore are more representative of the original timing. Because of its definition, the minTDEV may not fully address all network scenarios, (e.g., those with two-sided PDV distributions for which minimum selection can show large variations and hence increased TDEV noise), and further study is needed.

Pros and cons

The minTDEV calculation gives information on network packet delay noise processes but is not suitable for frequency offset characterization.

Like TDEV, minTDEV is sensitive to systematic effects which could mask noise components.

Unlike TDEV, minTDEV is sensitive to a small number of outliers (low-lying in this case).

The definition of the precise aspects that create the potential sensitivities listed above and the subsequent method of handling these when applying this metric are for further study.

I.4.1.1.2 PercentileTDEV

Definition and estimator formula

The percentileTDEV calculation is a special case of bandTDEV where the lower index *a* is assigned to 0 (see the bandTDEV definition below). Therefore, the definition and estimator formula are given by the definition and estimator formula, respectively, for bandTDEV (see I.4.1.1.3) with $a = 0$. Like the minTDEV metric, percentileTDEV focuses on the minimum packet delay floor. Instead of selecting a single minimum point, a (typically) small set of points at the floor are averaged together.

Usage

The percentileTDEV metric is applied much like the minTDEV metric. See the clause on minTDEV usage above. The percentile TDEV metric has the advantage that, in some circumstances, noise is reduced when a number floor packet delay measurement is selected and averaged together as opposed to the selection of a single point (see Figure I.8 below).

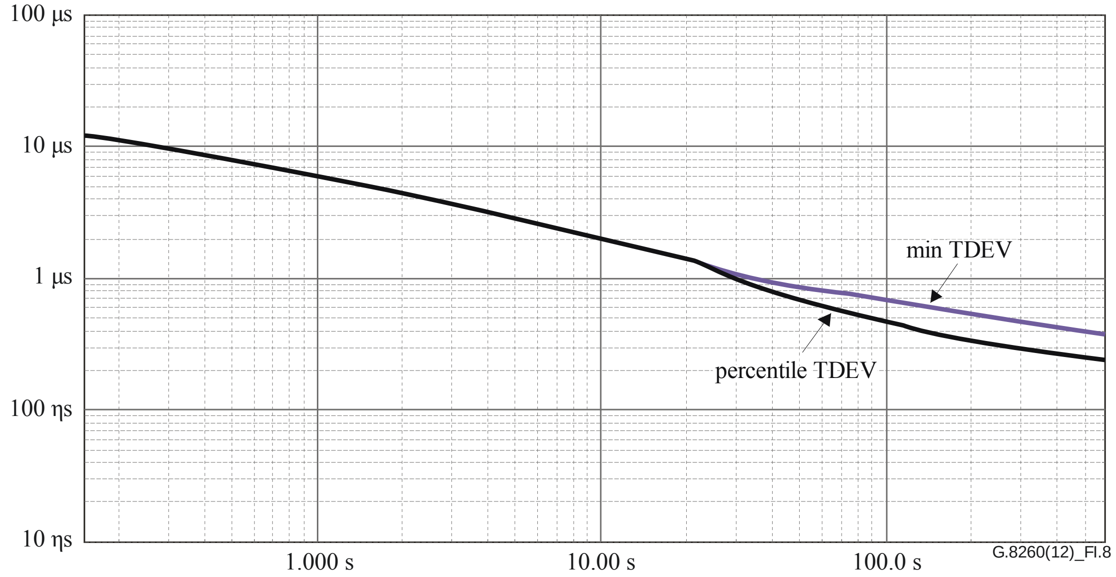

Figure I.8 – minTDEV vs. percentileTDEV (1%)

Pros and cons

Like minTDEV, percentileTDEV gives information on network packet delay noise processes but is not optimal for frequency offset characterization.

Like TDEV and minTDEV, percentileTDEV is sensitive to systematic effects which could mask noise components.

Unlike TDEV and like minTDEV, percentileTDEV is sensitive to a small number of low-lying outliers (though less sensitive than minTDEV).

An additional parameter, the percentile index, must be selected for percentileTDEV.

The definition of the precise aspects that create the potential sensitivities listed above and the subsequent method of handling these when applying this metric are for further study.

I.4.1.1.3 BandTDEV

Definition

To define bandTDEV, it is first necessary to represent the sorted phase data. Let x' represent this sorted phase sequence from minimum to maximum over the range $i \le j \le i+n-1$. Next it is necessary to represent the indices which are themselves set based on the selection of two percentile levels. Let a and b represent indices for the two selected percentile levels. The averaging is then applied to the x' variable indexed by a and b. The number of averaged points m is related to a and b: $m = b - a + 1$.

$$x'_{band\_mean}(i) = \frac{1}{m}\sum_{j=a}^{b} x'_{j+i} \tag{I-11}$$

bandTDEV can then be defined as:

$$\sigma_{x\_band}(\tau) = \text{bandTDEV}(\tau) = \sqrt{\frac{1}{6}\left\langle \left[x'_{band\_mean}(i+2n) - 2x'_{band\_mean}(i+n) + x'_{band\_mean}(i) \right]^2 \right\rangle} \tag{I-12}$$

where the angle brackets denote ensemble average.

$x'_{band\_mean}(i) = \frac{1}{m}\sum_{j=a}^{b} x'_{j+i}$, where x' represents a phase sequence sorted from minimum to maximum over the range $i \le j \le i + n - 1$, and a and b represent indices for two selected percentile levels. A percentile level is selected by using rounding to find the closest index from the desired percentile value. The additional constraint is that the index value has a minimum of the first index and a maximum of the last index.

Estimator formula

bandTDEV($n\tau_0$) may be estimated by:

$$\text{bandTDEV}(n\tau_0) \cong \sqrt{\frac{1}{6(N-3n+1)}\sum_{i=1}^{N-3n+1}\left[x'_{band\_mean}(i+2n) - 2x'_{band\_mean}(i+n) + x'_{band\_mean}(i)\right]^2} \quad \text{(I-13)}$$

for n = 1, 2, ..., integer part $\left(\frac{N}{3}\right)$

Usage

The bandTDEV calculation has the flexibility, in comparison to minTDEV and percentileTDEV, of being able to select a region of packet delay values away from the floor. Thus, if the population of packet delay values at the floor is noisier than the population immediately above, bandTDEV indices could be selected to focus analysis on that region.

Some of the comments on minTDEV usage apply here, but bandTDEV can apply effectively to distributions other than one-sided distributions slanted towards minimum delay packet. It is particularly effective for packet delay distributions with a well-populated mode somewhere in the packet delay distribution.

Pros and cons

Like minTDEV and percentileTDEV, bandTDEV gives information on network packet delay noise processes but is not optimal for frequency offset characterization.

Like TDEV, minTDEV, and percentileTDEV, bandTDEV is sensitive to systematic effects which could mask noise components.

The definition of the precise aspects that create the potential sensitivities listed above and the subsequent method of handling these when applying this metric are for further study.

I.4.1.1.4 ClusterTDEV

Definition

To define clusterTDEV, it is first necessary to represent the sorted phase data. Let x' represent this sorted phase sequence from minimum to maximum over the range $i \le j \le i + n - 1$. Next it is necessary to represent the cluster type that determines the cluster anchor $a(n)$ and the cluster range δ. Let a and b represent indices for the packets that fit within cluster range δ. The averaging is then applied to the x' variable for the cluster range δ. The number of averaged points is m, where $m = b - a + 1$ for that cluster.

$$x'_{cluster}(i) = \frac{1}{m}\sum_{j=a}^{b} x'_{j+i} \quad \text{(I-14)}$$

For clusterTDEV the average of the values in the cluster range as per equation (I-14) is substituted for the mean value in the defining equation for TDEV. ClusterTDEV($n\tau_0$) can then be defined as:

$$\sigma_{x\_cluster}(\tau)=\text{clusterTDEV}(\tau)=\sqrt{\frac{1}{6}\left\langle\left[x'_{cluster}(i+2n)-2x'_{cluster}(i+n)+x'_{cluster}(i)\right]^2\right\rangle} \qquad \text{(I-15)}$$

where the angle brackets denote ensemble average.

Estimator formula

clusterTDEV($n\tau_0$) may be estimated by:

$$\text{clusterTDEV}(n\tau_0)\cong\sqrt{\frac{1}{6(N-3n+1)}\sum_{j=1}^{N-3n+1}\left[x'_{cluster}(i+2n)-2x'_{cluster}(i+n)+x'_{cluster}(i)\right]^2}, \qquad \text{(I-16)}$$

for $n = 1, 2, ...,$ integer part $\left(\frac{N}{3}\right)$

Usage

The clusterTDEV calculation has the flexibility of being able to select a region of packet delay values away from the floor. Thus, if the population of packet delay values at the floor is noisier than the population immediately above, clusterTDEV indices could be selected to focus analysis on that region. Generally speaking, clusterTDEV provides a quantitative measure of stability of transit delays that are in a pre-determined band based on a phase/time range centred at a value that is determined by a chosen selection rule.

Some of the comments on minTDEV usage apply to clusterTDEV as well, but clusterTDEV can apply effectively to distributions other than one-sided distributions slanted towards minimum delay packet. It is particularly effective for packet delay distributions with a well-populated mode somewhere in the packet delay distribution.

Pros and cons

Like minTDEV, clusterTDEV gives information on network packet delay noise processes but is not suitable for frequency offset characterization.

Like TDEV and minTDEV, clusterTDEV is sensitive to systematic effects which could mask noise components.

Unlike TDEV and like minTDEV, clusterTDEV is sensitive to frequency offsets. Frequency offsets may be more difficult to ascertain precisely when neither a well-populated floor nor ceiling exists.

Two additional parameters, the cluster range and the cluster rule, must be selected for clusterTDEV.

The definition of the precise aspects that create the potential sensitivities listed above and the subsequent method of handling these when applying this metric are for further study.

It may be helpful to use the representation clusterTDEV(τ,δ ,*clusterType*) where τ is the observation interval, δ the cluster range, and the *clusterType* provides the rule used to generate the *anchor value*. Generally the rule is available from context and in that case need not be included in the representation.

For example:

$$\text{minTDEV}(\tau)=\text{clusterTDEV}(\tau,0,clusterType_{\min})$$

I.4.1.2 Maximum average time interval error and maximum average frequency error

Maximum average time interval error (MATIE) and maximum average frequency error (MAFE) describe maximum phase or frequency deviations. MATIE and MAFE include a noise averaging function similar to TDEV.

I.4.1.2.1 Maximum average time interval error (MATIE)

Definition

Two adjacent sliding observation windows are used to analyse the time error of a clock or selected packet delay data. The width of the observation windows (τ) is used as the independent variable (x-axis of the resulting curve) like in TDEV.

The average time error value is computed in the two adjacent windows. The averaging establishes the filtering capability that resembles the one used in TDEV. The unsigned difference between two consecutive windows is determined by subtracting the average of one window with the other and calculating absolute value.

The sliding window averaging process described above is a low-pass filtering process approximately corresponding to the one applied by a PLL filter to a timing signal. The difference calculation compares the estimation of the phase of the clock output at two time instances, which are a distance of $n\tau_0$ apart, see the MATIE formula below.

The two adjacent sliding windows are swept over the whole time error data and the maximum value is taken for expressing worst-case occurrence expected from the data.

For the MATIE analysis of packet data, the same calculation is done for different values of the window size (τ), similar to TDEV.

The function applied to discrete data samples is described in equation (I-18).

MATIE($n\tau_0$) is defined as a specified percentile, β, of the random variable:

$$X_n = \max_{1 \le k \le N-2n+1} \frac{1}{n} \left| \sum_{i=k}^{n+k-1} (x_{i+n} - x_i) \right| \tag{I-17}$$

for $n = 1, 2, ...,$ integer part ($N/2$)

where x_i is the packet time error sequence (and is a random sequence), $n\tau_0$ is the observation window length, n is the number of samples in the window, τ_0 is the sample interval, N is the number of samples in the data set, and k is incremented for sliding the window. MATIE describes the maximum of average time changes between adjacent windows of length $n\tau_0$.

Estimator formula

MATIE($n\tau_0$) may be estimated by:

$$\text{MATIE}(n\tau_0) \cong \max_{1 \le k \le N-2n+1} \frac{1}{n} \left| \sum_{i=k}^{n+k-1} (x_{i+n} - x_i) \right| \tag{I-18}$$

for $n = 1, 2, ...,$ integer part ($N/2$)

The above is a point estimate, and is obtained for measurements (i.e., samples x_i of the packet time error sequence, which represent the data values) over a single measurement period (see Figure II.1 of ITU-T G.810). Estimates of MATIE (for specified N, $\tau = n\tau_0$, and β), and their respective degrees of statistical confidence, may be obtained from measured data if measurements are made for multiple measurement periods (see clause II.5 of [ITU-T G.810]).

Usage

When applied to pre-processed delay data, corresponding to the selected subset in Figure I.3, MATIE predicts the largest difference in averaged time interval error that occurs between adjacent averaging windows of width τ.

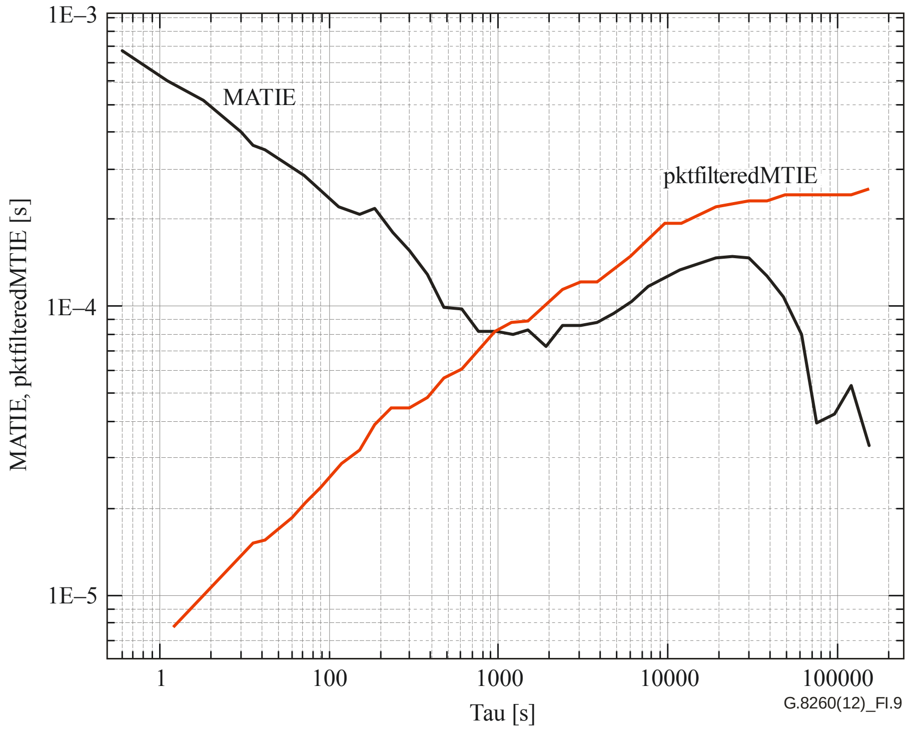

Figure I.9 – The black curve represents MATIE applied to pre-processed delay data. The red curve depicts *pktfiltered*MTIE, see I.4.2, with a 960s averaging function. These values are for example.

I.4.1.2.2 MAFE

Definition

There is a simple relationship between MAFE and the MATIE metric defined above.

$$\mathrm{MAFE}(n\tau_0) = \frac{\mathrm{MATIE}(n\tau_0)}{n\tau_0} \qquad \text{(I-19)}$$

Thus, MAFE($n\tau_0$) is defined as a specified percentile, β, of the random variable:

$$Z_n = \frac{\max\limits_{1 \le k \le N-2n+1} \frac{1}{n} \left| \sum_{i=k}^{n+k-1} (x_{i+n} - x_i) \right|}{n\tau_0} \qquad \text{(I-20)}$$

for n = 1, 2, ..., integer part (N/2)

where x_i is the packet time error sequence (and is a random sequence), $n\tau_0$ is the observation window length, n is the number of samples in the window, τ_0 is the sample interval, N is the number of samples in the data set, and k is incremented for sliding the window. MAFE is a dimensionless, normalized frequency ($\Delta f/f$).

Estimator formula

MAFE($n\tau_0$) may be estimated by:

$$\mathrm{MAFE}(n\tau_0) \cong \frac{\max\limits_{1 \le k \le N-2n+1} \frac{1}{n} \left| \sum_{i=k}^{n+k-1} (x_{i+n} - x_i) \right|}{n\tau_0} \qquad \text{(I-21)}$$

for n = 1, 2, ..., integer part (N/2)

The above is a point estimate, and is obtained for measurements (i.e., samples x_i of the packet time error sequence, which represent the data values) over a single measurement period (see Figure II.1 of ITU-T G.810). Estimates of MAFE (for specified N, $\tau = n\tau_0$, and β), and their respective degrees of statistical confidence, may be obtained from measured data if measurements are made for multiple measurement periods (see clause II.5 of [ITU-T G.810]).

Usage

When applied to a physical clock signal data corresponding to the clock output in Figure I.1, at small τ values where the MAFE calculation does not do any further filtering to the clock signal, MAFE expresses the peak frequency error of the clock, see Figure I.10. At larger τ values where MAFE represents narrower bandwidth than in the clock servo producing the signal, MAFE presents how the maximum frequency error could be reduced by further averaging of the clock signal.

When applied to pre-processed delay data, corresponding to the selected subset in Figure I.3, MAFE predicts the maximum frequency error calculated from the largest difference in averaged time interval error observed between adjacent averaging windows of width τ, see Figure I.10.

Figure I.10 shows the effect of MAFE applied to pre-processed delay data and to the time error data of one particular physical clock that uses an averaging period on the order of 1000s. The definition of averaging period is for further study, and this value is for example.

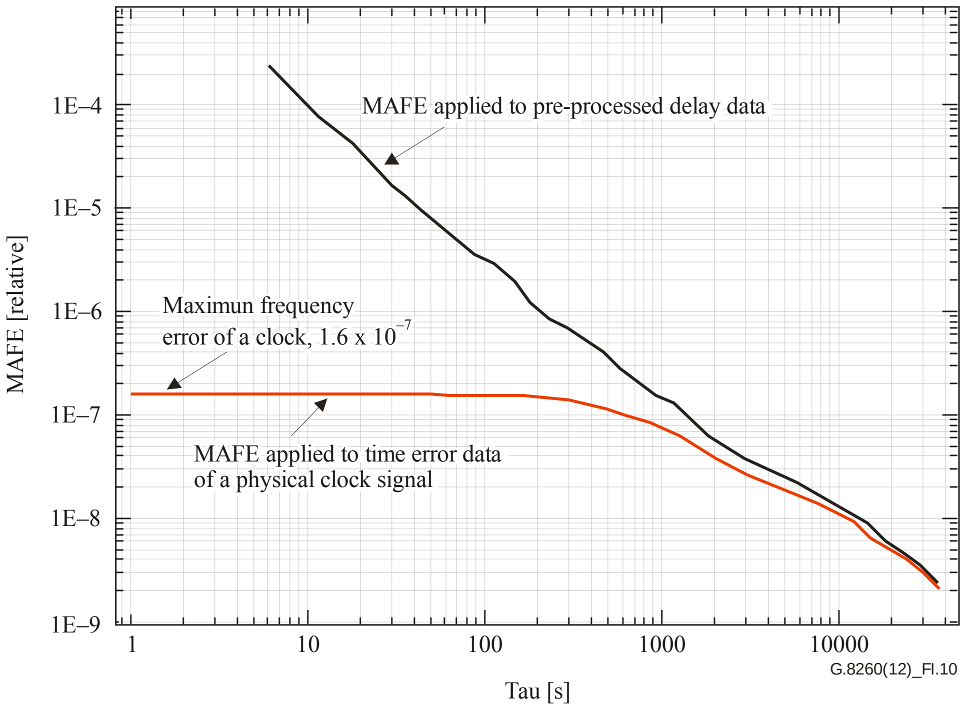

Figure I.10 – MAFE applied to pre-processed delay data and to time error data of physical clock

Pros and cons

MAFE is well-suited for the characterization of frequency error.

MAFE is not suited for the study of noise processes, unlike the complementary TDEV metrics.

Like the minTDEV and percentileTDEV metrics, MAFE with floor-based selection is sensitive to a small number of low-lying outliers.

The definition of the precise aspects that create the potential sensitivities listed above and the subsequent method of handling these when applying this metric are for further study.

I.4.1.2.3 Integrating packet selection in the MATIE or MAFE calculation

The packet selection can be integrated in the MATIE and MAFE calculations, as discussed earlier. The definitions and estimator formulas for minMATIE and minMAFE are given in the following subclauses.

I.4.1.2.3.1 minMATIE

Definition

minMATIE($n\tau_0$) is defined as a specified percentile, β, of the random variable:

$$U_n = \max_{1 \le k \le N-2n+1} \left| x_{\min}(k+n) - x_{\min}(k) \right| \qquad \text{(I-22)}$$

where $x_{\min}(k) = \min[x_j] \text{ for } (k \le j \le k+n-1)$, x_i is the packet time error sequence (and is a random sequence), $n\tau_0$ is the observation window length, n is the number of samples in the window, τ_0 is the sample interval, N is the number of samples in the data set, and k is incremented for sliding the window.

Estimator formula

minMATIE($n\tau_0$) may be estimated by:

$$\text{minMATIE}(n\tau_0) \cong \max_{1 \le k \le N-2n+1} \left| x_{\min}(k+n) - x_{\min}(k) \right| \qquad \text{(I-23)}$$

for n = 1, 2, ..., integer part (N/2)

The above is a point estimate, and is obtained for measurements (i.e., samples x_i of the packet time error sequence, which represent the data values) over a single measurement period (see Figure II.1 of [ITU-T G.810]). Estimates of minMATIE (for specified N, $\tau = n\tau_0$, and β), and their respective degrees of statistical confidence, may be obtained from measured data if measurements are made for multiple measurement periods (see clause II.5 of [ITU-T G.810]).

I.4.1.2.3.2 minMAFE

Definition

minMAFE($n\tau_0$) is defined as a specified percentile, β, of the random variable:

$$V_n = \frac{\max_{1 \le k \le N-2n+1} \left| x_{\min}(k+n) - x_{\min}(k) \right|}{n\tau_0} \qquad \text{(I-24)}$$

for n = 1, 2, ..., integer part (N/2)

where $x_{\min}(k) = \min[x_j] \text{ for } (k \le j \le k+n-1)$, where x_i is the packet time error sequence (and is a random sequence), $n\tau_0$ is the observation window length, n is the number of samples in the window, τ_0 is the sample interval, N is the number of samples in the data set, and k is incremented for sliding the window.

Estimator formula

minMAFE($n\tau_0$) may be estimated by:

$$\text{minMAFE}(n\tau_0) \cong \frac{\text{minMATIE}(n\tau_0)}{n\tau_0} = \frac{\max\limits_{1 \le k \le N-2n+1} \left| x_{\min}(k+n) - x_{\min}(k) \right|}{n\tau_0} \quad \text{(I-25)}$$

for n = 1, 2, ... integer part (N/2)

The above is a point estimate, and is obtained for measurements (i.e., samples x_i of the packet time error sequence, which represent the data values) over a single measurement period (see Figure II.1 of [ITU-T G.810]). Estimates of minMATIE (for specified N, $\tau = n\tau_0$, and β), and their respective degrees of statistical confidence, may be obtained from measured data if measurements are made for multiple measurement periods (see clause II.5 of [ITU-T G.810]).

I.4.2 Metrics including pre-filtering

As introduced in clause I.3.1.1, "Pre-processed packet selection", an input packet time-error sequence $x(t)$ is subject to packet selection, which produces a new selected-packet time-error sequence $x'(t)$. Additionally that new packet time-error sequence $x'(t)$ may be subsequently filtered to create a filtered-packet time-error sequence $y(t)$. This flow is shown below in Figure I.11.

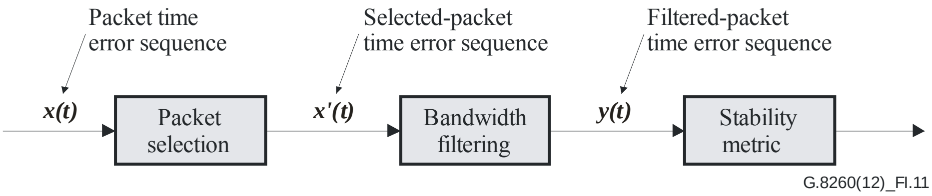

Figure I.11 – Packet selection and filtering flow

The selected-packet time-error sequence $x'(t)$ may be filtered by applying an averaging function in line with the clock bandwidth, with averaging time related to the window length of the packet selection process. In particular, 1:10 is as an example of a suitable ratio of the window length of the packet selection block to the time constant of the bandwidth filtering block in Figure I.11[1].

The parameters of the selection process must be chosen to ensure that sufficient packet information is available to allow the filtering process to operate. As an example, assuming a packet rate of 1 packet/s and a selection window of 100 s, the minimum possible selection percentage is 1%, resulting in the selection of 1 packet in every window.

In many cases a higher packet rate would be used for these measurements in order to get a higher number of samples.

The following applies a sliding window averaging function with length b (the number of windows, where each window has K samples):

$$y(n\tau_0) = \frac{1}{b} \sum_{i=n}^{n+b-1} (x'_i), i = 1,2,...N-b+1 \quad \text{(I-26)}$$

[1] The time constant of a PLL, also known as its characteristic response time, provides an indication of the duration of the effects on the output of the PLL due to a given input. This is why it is important that the selection window is properly chosen in order to get a significant number of samples during this period of time. Note that the time constant τ_c is related to the 3dB bandwidth of the PLL f_{3dB}, by the following relationship: $\tau_c = 1/(2\pi \cdot f_{3dB})$

The filtered-packet time-error sequence $y(t)$ may be used to compute TIE and applied to traditional synchronization metrics defined in [ITU-T G.810] such as MTIE and TDEV. When TIE or a metric are computed using $y(t)$, the term *pktfiltered* is prepended to TIE or the name of the metric, respectively. As illustrated in Figure I.11, the *pktfiltered* operation includes both packet selection and bandwidth filtering.

I.4.2.1 pktfilteredTIE

pktfilteredTIE is the TIE of the filtered-packet time-error sequence, substituted into the formula defined in [ITU-T G.810].

$$\text{pktfilteredTIE}(t,\tau) = y(t+\tau) - y(t) \tag{I-27}$$

I.4.2.2 pktfilteredMTIE

pktfilteredMTIE is the MTIE of the filtered-packet time-error sequence, obtained from the respective formula given in [ITU-T G.810] for the definition or estimator.

Definition

pktfilteredMTIE(τ) is defined as a specified percentile, β, of the random variable:

$$Y = \max_{0 \le t_0 \le T-\tau} \left(\max_{t_0 \le t \le t_0+\tau} [y(t)] - \min_{t_0 \le t \le t_0+\tau} [y(t)] \right) \tag{I-28}$$

where T is the measurement interval and τ is the observation interval.

Estimator formula

pktfilteredMTIE($n\tau_0$) may be estimated by:

$$\text{pktfilteredMTIE}(n\tau_0) \cong \max_{1 \le k \le N-n} \left[\max_{k \le i \le k+n} y_i - \min_{k \le i \le k+n} y_i \right], \quad n = 1,2,\ldots,N-1 \tag{I-29}$$

The above is a point estimate, and is obtained for measurements over a single measurement period (see Figure II.1 of [ITU-T G.810]). Estimates of pktfilteredMTIE (for specified T, $\tau = n\tau_0$, and β), and their respective degrees of statistical confidence, may be obtained from measured data if measurements are made for multiple measurement periods (see clause II.5 of [ITU-T G.810]).

I.4.2.3 pktfilteredTDEV

pktfilteredTDEV is the TDEV of the filtered-packet time-error sequence, obtained from the respective formula given in [ITU-T G.810] for the definition or estimator.

Definition

pktfilteredTDEV($n\tau_0$) is defined as:

$$\text{pktfilteredTDEV}(n\tau_0) = \sqrt{\frac{1}{6n^2} \left\langle \left[\sum_{i=1}^{n} \left(y_{i+2n} - 2y_{i+n} + y_i \right) \right]^2 \right\rangle} \tag{I-30}$$

where the angle brackets denote an ensemble average.

Estimator formula

pktfilteredTDEV($n\tau_0$) may be estimated by:

$$\text{pktfilteredTDEV}(n\tau_0) \cong \sqrt{\frac{1}{6n^2(N-3n+1)} \sum_{j=1}^{N-3n+1} \left[\sum_{i=j}^{n+j-1} \left(y_{i+2n} - 2y_{i+n} + y_i \right) \right]^2} \qquad \text{(I-31)}$$

$$\text{for } n = 1, 2, ..., \text{integer part} \left(\frac{N}{3} \right)$$

I.4.2.4 pktfilteredFFO

pktfilteredFFO is the fractional frequency offset of the filtered-packet time-error sequence, $y(t)$, substituted into the formula defined in [GR-1244-CORE]. Refer to [ITU-T G.810], clauses 4.5.2 and I.2, for the definition and a description of fractional frequency offset.

Estimator formula

pktfilteredFFO($N\tau_o$) may be estimated by

$$\text{pktfilteredFFO}(N\tau_o) \cong \frac{6x10^{-9}}{N\tau_0} \sum_{i=1}^{N} y_i \left(\frac{2i}{(N^2-1)} - \frac{1}{N-1} \right) \qquad \text{(I-32)}$$

I.5 PDV metrics studying floor delay packet population

The objective of this category of PDV metrics is to study the population of timing packets within a certain fixed cluster range starting at the observed floor delay. The population of timing packets can then be compared with acceptance or rejection thresholds. The main idea here is to ensure that at least a minimum number of packets, or alternately a minimum percentage of packets, always remains within the specified fixed cluster range starting at the observed floor delay.

As an example, consider the diagram shown in Figure I.12. The packet delay values are shown as a function of time. Some packets arrive within a certain range of the smallest observed delay (those below the red line) and others arrive outside that range. Those packets arriving within the range are counted each window interval. This count is compared against an acceptance criterion for each window interval. If all window intervals meet the acceptance criterion then the network has met the PDV network limit.

The windows depicted in Figure I.12 are shown as non-overlapping but contiguous. This is often referred to as a "jumping window" approach. The "sliding window" approach considers windows that are shifted by 1 packet (sample). Intermediate approaches consider different levels of overlap. Using sliding windows will detect all non-stationary and short transient failure events. Jumping and overlapping windows are a subset of the sliding window approach. Implementations may choose to use a jumping window or an overlapping window approach.

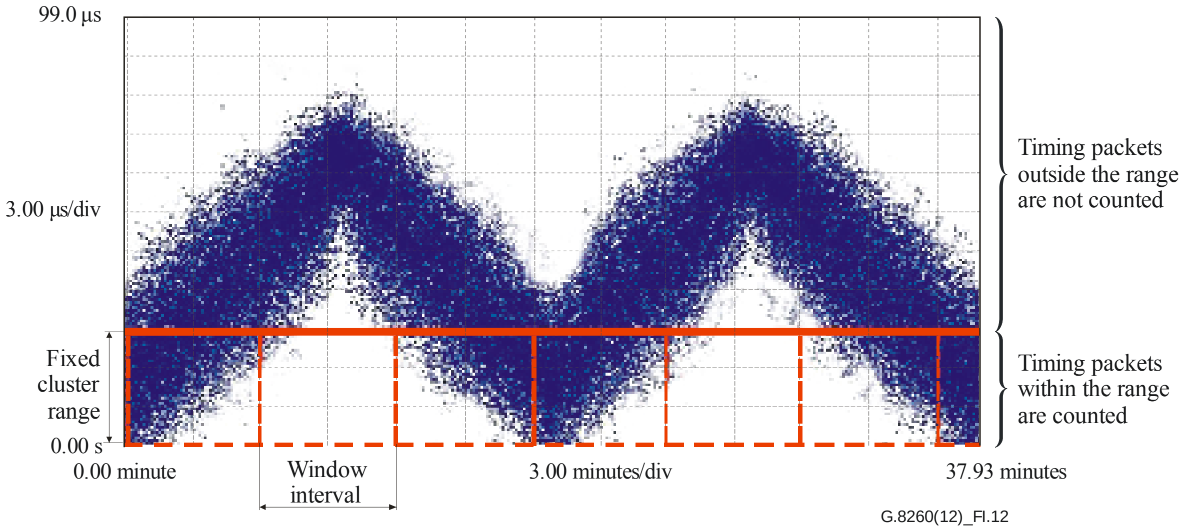

Figure I.12 – Example of PDV metric studying the population of packets within a fixed cluster range starting at the observed floor delay

PDV limits specified in terms of these metrics are considered as met if at least *m* packets, or alternately at least *p%* of packets, are observed for any window interval of *t* seconds within a fixed cluster range starting at the observed floor delay and having a size δ. If less than *m* packets are observed, or alternately less than *p%* of packets, then the PDV limit is considered as not met.

This process can be described in the following way:

Let $x[i]$ represent the measured latency of timing packet *i*, where $0 \leq i < N$. That is, there are *N* packets in the measurement data set. Let the nominal time between timing packets be represented by τ_P. Let δ represent the cluster range and let *W* represent the window interval in units of time, which can also be expressed as *K* samples, where $K = W/\tau_P$. *K* represents the (nominal) number of packets transmitted in the window interval.

NOTE – It is assumed that the packet rate of the timing flow is nominally constant. The case for a variable rate of packet transmission is for further study.

Define the minimum observed delay as:

$$d_{\min} = \min_{0 \leq i < N} x[i] \qquad \text{(I-33)}$$

The observed d_{min} given by equation (I-33) is an estimate of the absolute minimum latency that a packet may experience. If a better estimate of the absolute minimum latency is available, for example from previous measurement data, that alternate value may be used. In all cases, the equations below are valid for choices of minimum delay less than or equal to the observed d_{min}.

Then, define the indicator function which performs floor packet selection:

$$\phi_F(i,\delta) = \begin{cases} 1; & \text{if } x[i] \leq d_{\min} + \delta \\ 0; & \text{otherwise} \end{cases} \quad \text{for } 0 \leq i < N \qquad \text{(I-34)}$$

Note that equation (I-34) assumes that packet delay is always greater than d_{min}.

The convention followed in equations (I-35), (I-36) and (I-37) is that sample index "*n*" is associated with the end of the window. That is, the floor packet metrics are based on complete windows and consequently values of *n* less than $(K - 1)$ are not defined.

Then define the *Floor Packet Count* (*FPC*) sequence with parameters n, W ($W = K \cdot \tau_P$) and δ:

$$FPC(n,W,\delta) = \sum_{j=n-(K-1)}^{n} \phi_F(j,\delta) \quad \text{for } (K-1) \le n < N \tag{I-35}$$

Define the *Floor Packet Rate* (*FPR*) sequence with parameters n, W and δ:

$$FPR(n,W,\delta) = \frac{FPC(n,W,\delta)}{W} \quad \text{for } (K-1) \le n < N \tag{I-36}$$

Define the *Floor Packet Percent* (*FPP*) sequence with parameters n, W and δ:

$$FPP(n,W,\delta) = \left(\frac{\tau_P}{W}\right) \times FPC(n,W,\delta) \times 100\ \% \quad \text{for } (K-1) \le n < N \tag{I-37}$$

The *floor packet percent* is applicable to defining network limits. That is, the network performance is acceptable if

$$\min_{(K-1) \le n < N} \{FPP(n,W,\delta)\} \ge p\% \tag{I-38}$$

where the network acceptance criterion is p%, and the parameters W and δ are provided in the appropriate Recommendation, for example [ITU-T G.8261.1].

The *floor packet rate* (equivalently *floor packet count*) are suitable metrics for identifying the slave clock tolerance limit. That is, the slave clock must meet its specified output performance if

$$\min_{(K-1) \le n < N} \{FPC(n,W,\delta)\} \ge m \tag{I-39}$$

where the parameters m, W and δ are provided in the appropriate recommendation as applicable.

Equations (I-36), (I-37), (I-38) and (I-39) are general and appropriate for sliding window approaches. Jumping and overlapping window calculations can be obtained by sub-sampling the sliding window samples.

For the jumping window case, estimates are derived every K samples. That is, the jumping-window samples are simply the sliding-window estimates under-sampled by a factor of K. Over the full measurement interval, there are $M = (N/K)$ jumping window samples and consequently the index for the jumping-window sequence ranges from 0 through (M – 1).

The jumping window approach is suitable when network conditions are stationary and spectral and probability density parameters do not change rapidly. The sliding windows may be more appropriate, for example, for short term transient or rapidly changing events.

NOTE 1 – This category of PDV metrics requires a long enough measurement period such that the observed floor delay would give a good enough estimation of the absolute floor delay. The minimum measurement period depends on the type of network considered. Long measurement periods, for instance over one or several days, should be favoured in order to study diurnal PDV effects.

NOTE 2 – Like minTDEV and MAFE, these metrics may be sensitive to a small number of low-lying outliers. The definition of the precise aspect that creates the potential sensitivity, and the subsequent method of handling this when applying this metric, is for further study.

NOTE 3 – This category of PDV metrics is sensitive to non-stationary network conditions, as described in clause I.3.3, that produce floor delay steps of significant amplitude, which may occur for instance during network re-routing events. The handling of floor delay steps is for further study.

NOTE 4 – These metrics are mainly intended to be used as post-processing metrics. The use of these metrics for real-time processing is for further study.

NOTE 5 – These metrics can be used to study the PDV noise produced independently by the forward or the reverse direction of a packet timing flow. Consideration of the combined effect of both directions is for further study.

I.6 Summary of metric classifications

Clause I.1 provides definitions of two classes of PDV metrics: class A and class B.

Among the PDV metrics listed in clauses I.4 and I.5, the following are currently considered as suitable for defining PDV network limits (class A), and are used in [ITU-T G.8261.1]:

– The PDV metrics studying minimum floor delay packet population (clause I.5)

It should be noted that the MAFE and pktfilteredMTIE metrics are considered as candidates for defining network limits (class A).

The possible use of other PDV metrics (clause I.4) for defining network limits is for further study.

Bibliography

[GR-1244-CORE] GR-1244-CORE (2009), *Clocks for the Synchronized Network: Common Generic Criteria,* Telcordia Technologies Generic Requirements, (Issue 4, October).

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	**Transmission systems and media, digital systems and networks**
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Terminals and subjective and objective assessment methods
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects and next-generation networks
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2012