An Explanation of VoIP Statistics

Summary

The following application note discusses the methods used for calculating VoIP quality using the JDSU DA-3400, DA-3600A, and QT-600. Throughout this document, all three of these products will be generically referred to as the test device.

The fundamental parameter for the estimation of perceived call quality, R-factor, was introduced by the ITU in recommendation G.107. The R-factor metric has a range from 0 (very bad) to 100 (perfect).

The following list contains all of the recommendations involved in R-factor calculations.

– ITU G.107 (E-model)

– ITU G.113, Appendix I (CODEC)

– ITU G.114 (Delay)

– ITU G.1020 (Packet loss/drop)

– ETSI TS 102 024-5 (Packet loss/drop)

R-Factor

R-factor is calculated as

(1)

$$R = Ro - Is - Id - Ie\text{-}eff - Irecency + A$$

The basic components of the R-factor are described in Table 1.

Component	Description
Ro	This is the signal-to-noise ratio
Is	This is the combination of all impairments that occur simultaneously with the voice signal
Id	This is the impairments caused by delay
Ie-eff	This is the impairments caused by a low bit rate CODEC. It also includes impairments due to packet loss and rejection
Irecency	This is the impairments resulting from significant packet loss. Significant packet loss is detected if there are eight or more packets lost in a row
A	This is the advantage factor, which allows for the compensation of impairment factors

table 1 The components of the R-factor formula

R-Factor Components

Ro and Is

Test devices do not execute any type of signal processing. The *Ro* and *Is* components are always the same and are calculated using the ITU default values for the different impairments involved in the calculation.

Id

The *Id* component captures the impairments introduced by delay. This delay includes one-way delay of packet delivery and also CODEC-related delay, which includes framing delay and look-ahead delay.

Id is typically small and does not significantly degrade the R-factor. If the delay becomes large enough, the degradation can be significant.

Ie-eff

Ie-eff is a larger impairment component than *Id* and includes degradations resulting from packet loss, packet rejection, and low voice CODECs.

A packet is lost if it is created by one VoIP end point and does not arrive at the far-end VoIP end point.

A packet is rejected if it arrives at the far end, but it arrives too early or too late to be successfully played out by the sound generation hardware. In this case, the packet arrives but is not converted into the voice stream. Usually, rejected packets are the result of excessive jitter or significant clock skew between two VoIP terminals, a VoIP phone and a media gateway, for example.

The sum of the packet loss and packet drop counters is referred to as the missing packet counter.

The missing packet counter is important for *Ie-eff* calculations only. If the packet is not played out by the voice generation hardware, it does not really matter if it was lost or rejected.

The missing packet counter is broken down into the number of lost packets and the number of rejected packets. This break down gives additional useful information about the traffic on the network.

The R-factor, calculated using formula (1), is also known as R_{CQE}, where CQE represents the conversational quality estimate. R_{CQE} captures all impairments.

The CODEC used by the voice stream also degrades stream quality because the voice compression provided by the CODEC results in some loss of voice quality. The CODEC also participates in a delay degradation, which is captured by the *Id* component.

Irecency

Using an example of a ten-minute VoIP call, there are two possible scenarios.

1. There can be significant distortion at the beginning of the call, but the rest of the call is fine.
2. There can be nine minutes of perfect quality, and, in the last minute, there can be exactly the same distortion as in the first scenario.

Experiments have shown that the perceived quality score will be significantly less in the second scenario than in the first scenario. The *Irecency* component captures these kinds of effects.

A

The advantage factor *A* component is always 0 because the test device has no prior knowledge of this component. If it is known, the tester can just add the value to the R-factor measured by the test device. The ITU recommended values for the advantage factor *A* component (as per recommendation G.107) are shown in Table 2.

Communication System	Maximum Value of A
Conventional (wirebound)	0
Mobility by cellular networks in a building	5
Mobility in a geographical area or moving in a vehicle	10
Access to hard-to-reach locations (via multi-hop satellite connections, for example)	20

table 2 Provisional examples for the advantage factor *A* as per G.107

CODEC Degradation

Each CODEC is assigned a degradation value. For instance, G.711 has a degradation value of 0, G.729 has a degradation value of 10, and G.723.1 has a degradation value of 15 (for 6.3 Kbps) or 19 (for 5.3 Kbps).

The CODEC degradation value is represented by *Ie*.

Missing Packet Degradation

Ie-eff is then calculated using the following formula:

(2)
$$Ie\text{-}eff = Ie + (95 - Ie)\,\frac{Ppl}{Ppl + Bpl}$$

Ppl is the random packet loss probability representing the packet missing rate (%).

Bpl is the packet loss robustness factor that is assigned to each CODEC. This value reflects the ability to recover (or hide) the packet loss using a method of packet loss concealment (PLC).

Most missing packets are not distributed uniformly during a call. An attempt to capture the "bursitness" of a packet loss/drop was implemented in the ITU recommendation G.1020.

Distribution of Missing Packets

The premise of capturing the "burstiness" of missing packets is based on the statistical representation of the voice stream as a sequence of *gaps* and *bursts*.

Burst is defined by the ITU in recommendation G.1020 and states:

> A burst must be a longest sequence beginning and ending with a loss during which the number of consecutive received packets is less than some value G_{min} (a suitable value for G_{min} for use with Voice over IP services would be 16 whereas for use with Video services a higher value of say 64 or 128 would be preferable).

In another words, a burst is the interval of the traffic with a relatively high rate of missing packets. The lowest value of the relatively high packet loss rate is 1 lost packet out of 16 received packets.

Gap, then, is the complement of burst. Effectively, a gap exists where there is not a burst.

Statistical analysis of the voice stream generates a set of the following parameters, capturing the "burstiness" of the traffic:

– Gap length: The average length (ms) of the gaps detected for the stream.

– Gap density: The average rate (%) of missing packets in a gap.

– Burst length: The average length (ms) of the bursts detected for the stream.

– Burst density: The average rate (%) of missing packets in a burst.

A model of a voice stream showing the bursts and gaps is shown in Figure 1.

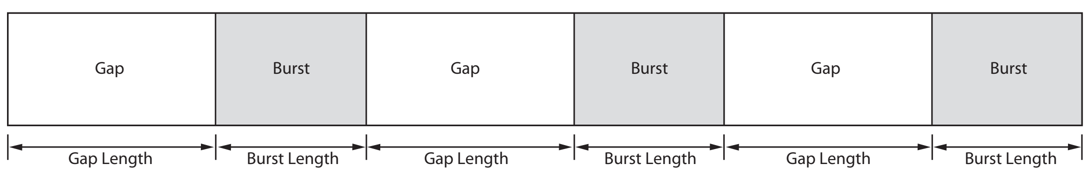

figure 1 A model of a voice stream

All of these parameters are variables in the calculation of the *Ie-eff* and *R*. These parameters in themselves are also important statistics.

Formula (2) and the gap density are used to calculate the *Ie-eff* that corresponds to the gap [*Ie(gap)*]. In the same fashion, formula (2) and the burst density are used to calculate the *Ie-eff* that corresponds to the burst [*Ie(burst)*]. Using these two values as well as the burst and gap lengths, a frustration curve is calculated. The frustration curve is a continuous exponential curve consisting of two exponents. The growing exponent has a value of 1/5 seconds, and the decreasing exponent has a value of 1/15 seconds (Figure 2).

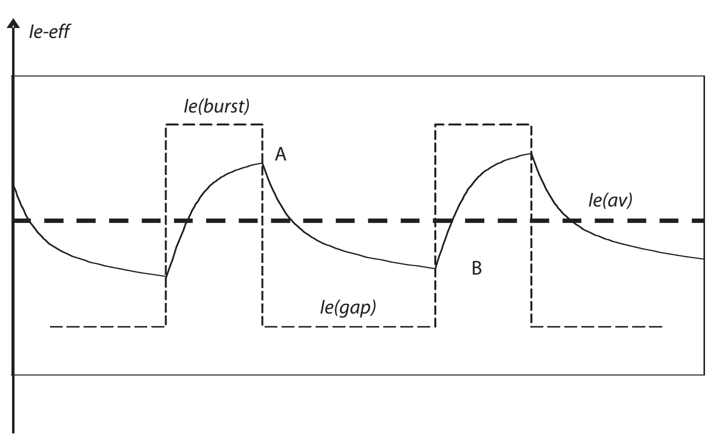

figure 2 The frustration curve

Once the frustration curve is known, the *Ie-eff* corresponding to the average value [*Ie(av)*] is calculated. This average value is the value of *Ie-eff* that is used in formula (1) for the calculation of *R*.

As part of the statistical analysis of a voice stream, the most recent position of the significant packet loss/drop, which is defined as eight packets lost or rejected in a row, is identified. This position is used to calculate the recency degradation factor.

R-Factor Summary

R-factor values include many different types of impairments. The most important types of impairments include:

– Missing packets

– CODECs

– Delay

– Recency

R-factor values range from 0 to 100, with an R-factor of 100 being perfect. In reality, there is always some unavoidable degradation, such as the impairments introduced by the microphone and the speaker in your handset. If the default parameters for these degradations are used, the highest possible R-factor for a call is 93.2. The quality scale for R-factor values is shown in Figure 3.

R-Factor Range	100-90	90-80	80-70	70-60	60-0
Speech Transmission Quality Category	Best	High	Medium	Low	(Very) Poor

PSTN Quality

figure 3 The quality scale for R-factor values

Table 3 contains of all of the input parameters that affect the R-factor value (as per recommendation G.107).

Parameter	Abbreviation	Unit	Default Value	Permitted Range
Send Loudness Rating[1]	SLR	dB	8	0 to +18
Receive Loudness Rating[1]	RLR	dB	2	-5 to +14
Sidetone Masking Rating[2]	STMTR	dB	15	10 to 20
Listener Sidetone Rating[2]	LSTR	dB	18	13 to 23
D-Value of the Telephone, Send Side[2]	Ds	–	3	-3 to +3
D-Value of the Telephone, Receive Side[2]	Dr	–	3	-3 to +3
Talker Echo Loudness Rating	TELR	dB	65	5 to 65
Weighted Echo Path Loss	WEPL	dB	110	5 to 110
Mean One-Way Delay of the Echo Path	T	ms	0	0 to 500
Round-Trip Delay in a 4-wire Loop	Tr	ms	0	0 to 1000
Absolute Delay in Echo-free Connections	Ta	ms	0	0 to 500
Number of Quantization Distortion Units	qdu	–	1	1 to 14
Equipment Impairment Factor	Ie	–	0	0 to 40
Packet Loss Robustness Factor[3]	Bpl	–	1	1 to 40
Random Packet Loss Probability[3]	Ppl	%	0	0 to 20
Circuit Noise (referred to as the 0 dBr point)	Nc	dBm0p	-70	-80 to -40
Noise Floor, Receive Side[3]	Nfor	dBmp	-64	–
Room Noise, Send Side	Ps	dB(A)	35	35 to 85
Room Noise, Receive Side	Pr	dB(A)	35	35 to 85
Advantage Factor	A	–	0	0 to 20

[1]*Total values between the microphone or receiver and the 0 dBr point*

[2]*Fixed relation: LSTR = STMR + D*

[3]*Currently under evaluation*

table 3 Default values and permitted ranges for the R-factor parameters as per G.107

For the calculation of R-factor, all of the parameters that are grayed in Table 3 will use their default values. It is important to note that this is not a disadvantage because these values are not generally known anyway. The only case where these values are known is in a lab environment.

R-Factor Derivatives

Once R-factor is calculated, many other statistics are available.

One such statistic is the Mean Opinion Score (MOS). MOS values are derived from a different set of ITU recommendations and are based on listening experiments. The same voice stream is presented for listening to a statistically large number of people. The people are then asked to score the quality of the call using the following categories:

- Excellent = 5
- Good = 4
- Fair = 3
- Poor = 2
- Bad = 1

The ITU provides the formula to calculate the MOS from the R-factor. The only parameter needed in order to calculate MOS is R.

R-Factor Variations

All of the previously discussed statistics (R-factor and MOS) have focused on how people perceive call quality.

Other variations of R-factor and MOS are also useful.

R_{LQE}

One variation of R-factor is listening quality estimate (LQE). Significant delay can potentially generate cross-talk. If a user is just listening to a call, delay is not an issue.

Therefore, R_{LQE} is calculated using the following formula:

$$R_{LQE} = Ro - Is - Ie\text{-}eff + A \qquad (3)$$

In this case, the delay and recency impairments are removed. Therefore, it is clear that

$$R_{LQE} \geq R$$

Once R_{LQE} is calculated, the other LQE variations (MOS_{LQE}) are calculated using the same formula.

Calls are rarely “one-way”, where R_{LQE} is the best metric. Therefore, R_{LQE} is rarely used as the primary quality metric.

Additional Statistics

Table 4 shows additional statistics that are reported by the test device:

Statistic	Description
Average packet loss	This is the rate (%) of missing packets. It includes both lost and rejected packets
Total packet loss	This is the total number of packets that were lost or rejected
Total packet drop	This is the total number of rejected packets
Total number of packets received	This is the total number of received packets

table 4 Additional statistics reported by the test device

Jitter

In this section, the various jitter metrics in the test device are discussed. It is important to understand that network jitter does not directly affect the values for R-factor, MOS, and others. Network jitter affects the values indirectly thru the number of rejected packets. The greater the jitter and the clock skew, the more packets will be rejected. The relationship between jitter and the number of rejected packets is indirect due to network jitter buffers. The purpose of a jitter buffer is to eliminate as much of the jitter as possible.

Jitter Notations

If all of the packets in the stream are enumerated with an index number ($i = 0,1,\ldots N$), each packet of the RTP stream has two timestamps associated with it. They include:

- t_i is the time when a packet was received by the test device (hardware timestamp).
- r_i is the RTP timestamp taken from the RTP header of the packet.

The RTP timestamp (r_i) is essentially the expected arrival time of the packet. In general, the RTP time stamp is different from the actual packet arrival time (t_i). Jitter, then, is the measure of the deviation between r_i and t_i.

Instantaneous Jitter

Inter-packet delay between the packet just received and the previous packet is calculated using the following formula:

(4) $$\Delta t_i = t_i - t_{i-1}, \text{where } i = 1, 2, \ldots N$$

Also, using the RTP timestamps, the expected inter-packet delay is calculated by:

(5) $$\Delta r_i = r_i - r_{i-1}, \text{where } i = 1, 2, \ldots N$$

The instantaneous jitter, then, is defined as the deviation between Δt_i and Δr_i :

(6) $$J_i = \Delta t_i - \Delta r_i$$

Figure 4 shows a jitter graph of an example of instantaneous jitter.

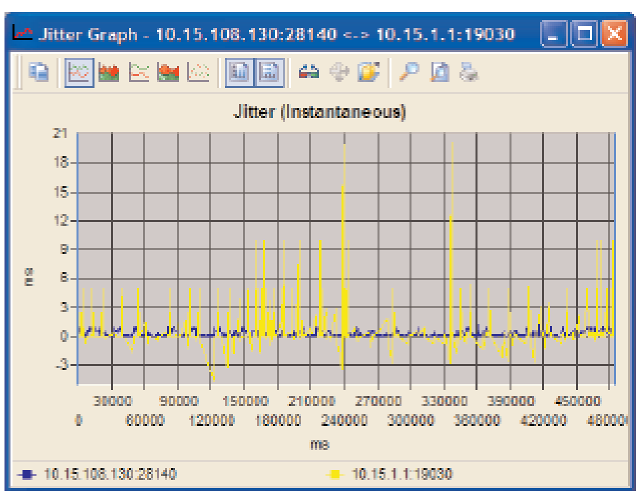

figure 4 A jitter graph showing instantaneous jitter

Jitter by RFC 3550

The RTP specification (RFC 3550) describes the recommended algorithm for calculating jitter. This algorithm takes into consideration that the instantaneous jitter described in the previous section is a jerky result. The RFC specification proposes to apply a low-pass filter to the instantaneous jitter.

More specifically, RFC jitter (or smoothed jitter) is calculated from the instantaneous jitter according to the formula below:

$$j_1 = J_1$$
$$j_i = \frac{15}{16} j_{i-1} + \frac{1}{16} \left|J_i\right|, \text{ where } i = 2, \ldots N \tag{7}$$

If the smoothed jitter is calculated using the instantaneous jitter values from the previous section, the resulting jitter graph is shown in Figure 5.

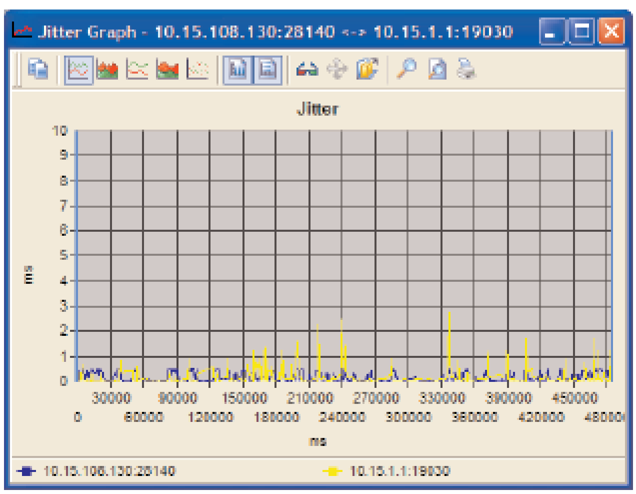

figure 5 A jitter graph showing RFC or smoothed jitter

In comparing the graph in Figure 5 with the graph in Figure 4, it is clear that instantaneous jitter has a maximum value of approximately 20 ms and smoothed jitter has a maximum value that never goes above 3 ms. Smoothed jitter eliminates the “jerkiness” of instantaneous jitter.

Absolute Jitter

Instantaneous and RFC jitter are based on inter-packet delays. If the very first packet is taken as a reference packet, the following can be calculated:

(8) $$\nabla t_i = t_i - t_0$$

(9) $$\nabla r_i = r_i - r_0, \text{ where } i = 2, \ldots N$$

∇t_i shows the actual arrival time of the packet calculated relative to the very first (reference) packet. ∇r_i is the expected arrival time calculated relative to the very first (reference) packet.

Absolute jitter, then, is defined as the deviation between these two times:

(10) $$G_i = \nabla t_i - \nabla r_i$$

Therefore, instantaneous jitter can be thought of as absolute jitter with a moving reference point, where each packet is the reference packet for the previous one.

When does absolute jitter provide more information than instantaneous jitter?

To answer this question, the same traffic used in the previous section is considered.

Figure 6 shows that the absolute jitter catches the clock skew. In this specific case, the clock skew between different VoIP stations is virtually 0. The clock skew shown is the skew between the VoIP stations' clocks and the clock of the capturing device.

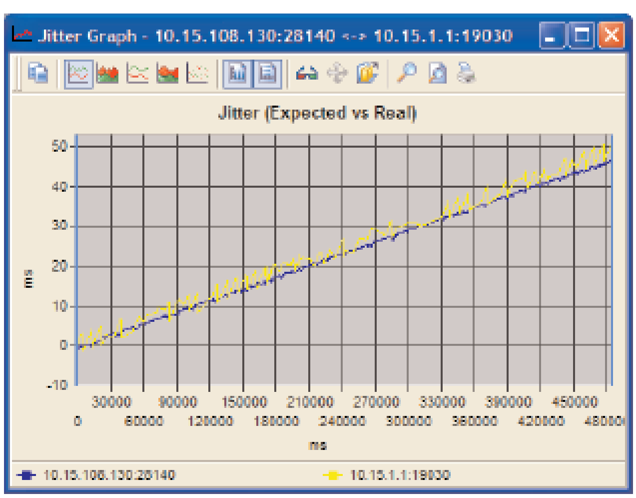

figure 6 An example of a jitter graph showing expected vs. real jitter

However, the situation can be different. Figure 7 shows a jitter graph from another call.

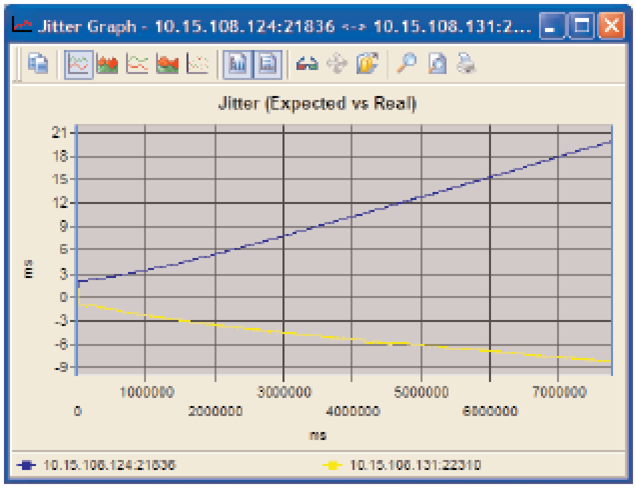

figure 7 Another example of a jitter graph showing expected vs. real jitter

In this example, the clock skew is approximately 20 ms for the call duration of 130 minutes. The relative clock skew between VoIP stations is calculated using the following formula:

$$\frac{20}{130 \times 60 \times 1000} \approx 2.6 \times 10^{-6} \tag{11}$$

Although some clock skew exists, it is within the acceptable range. The tolerance for the level of clock skew specified by the ITU is 50×10^{-6}.

Comparing Jitter Statistics

In order to compare the usefulness of the different types of jitter statistics previously described, a call with a two-minute duration is considered.

Figure 8 shows the absolute jitter graph.

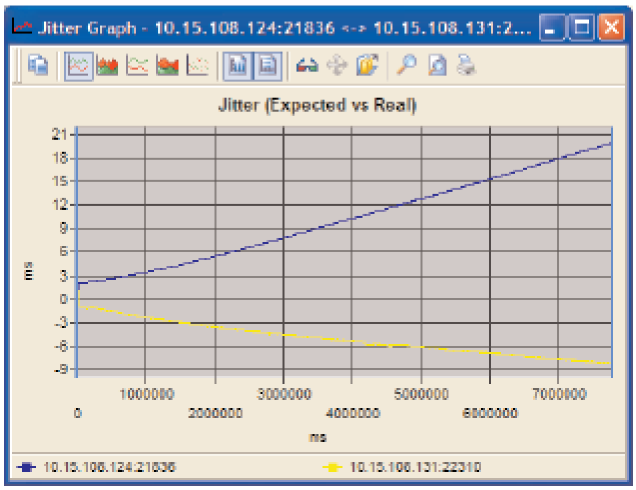

figure 8 The absolute jitter graph of a call

In this case, the clock skew is calculated as:

$$\frac{30}{2 \times 60 \times 1000} \approx 250 \times 10^{-6}$$

This value is outside of the acceptable tolerance level. Figure 9 shows the instantaneous jitter graph for the same call.

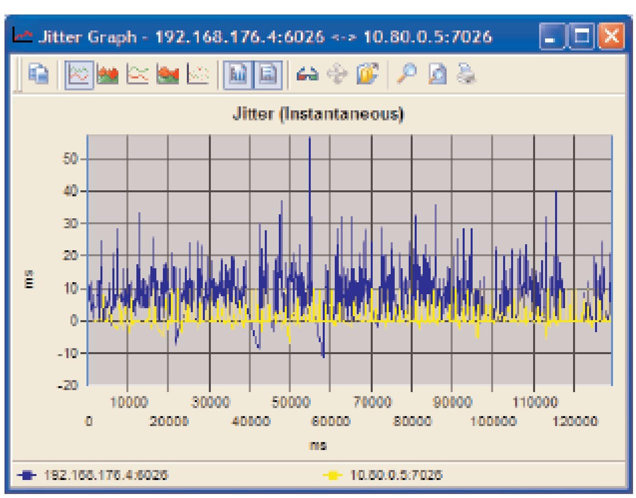

figure 9 The instantaneous jitter graph of a call

In contrast, Figure 10 shows the RFC (smoothed) jitter of the same call.

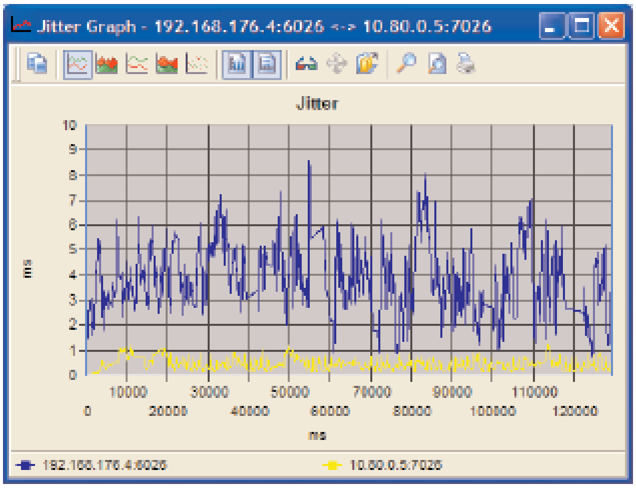

figure 10 The RFC (smoothed) jitter of a call

For this example, it is assumed that the jitter buffer has a size of 80 ms and that the late window is set at 40 ms. The jitter buffer emulation for this call shows that approximately 4 packets will be dropped.

The graph for instantaneous jitter shows that one packet was probably dropped because of the jitter spike. The other three packets were dropped because of the clock skew.

The RFC jitter graph shows that the jitter never goes above 9 ms. Therefore, if the instantaneous and absolute jitter are not known, the tester has no idea why the jitter buffer dropped the packets.

Absolute jitter is an important metric, but its importance is defined only by the fact that it catches the clock skew.

Jitter Summary

For RFC and instantaneous jitter, it makes sense to calculate the minimum, maximum, mean, and standard deviation of the values. These calculations demonstrate the "jerkiness" of the voice stream, from a jitter standpoint.

However, the same statistics for absolute jitter do not make much sense because the clock skew, if present, introduces a significant bias in these statistics and basically destroys their usefulness.

The test device provides:

- The minimum, maximum, mean, and standard deviation for RFC jitter
- The current value of the RFC jitter minimum, maximum, mean, and standard deviation for instantaneous jitter
- The clock skew

Interval Statistics and Cumulative Statistics

For some statistics, it is important to differentiate between interval and cumulative values. Cumulative statistics are calculated starting from the beginning of the stream. Interval statistics are calculated starting from the time of the previous extraction of statistics.

Table 5 provides a list of relevant VoIP statistics.

Statistic
Total number of packets received
Total number of packets lost
Total number of packets dropped
Average packet loss/drop density (%)
Instantaneous jitter minimum value
Instantaneous jitter maximum value
Instantaneous jitter mean value
Instantaneous jitter standard deviation value

table 5 VoIP Statistics

For each of these statistics, the test device reports two values. One is a cumulative value and the other one is an interval value.