2 OVERVIEW

2.1 GENERAL

This Recommended Standard defines a payload data compressor that has applicability to multispectral and hyperspectral imagers and sounders. This Recommended Standard does not attempt to explain the theory underlying the compression algorithm; that theory is partially addressed in reference [D1].

This Issue 2 revision extends the CCSDS Lossless Multispectral & Hyperspectral Image Compression standard (reference [D2]) to provide an effective method of performing either lossless or near-lossless compression of three-dimensional image data. Here, 'near-lossless' refers to the ability to perform compression in a way that the maximum error in the reconstructed image can be limited to a user-specified bound. Key changes introduced in this revision include the incorporation of a closed-loop quantization scheme to provide near-lossless compression and the extension of an entropy coding method of reference [D2] to provide better compression of low-entropy data.

The input to the compressor is an image, which for the purposes of this Recommended Standard is a three-dimensional array of integer sample values, as specified in section 3. The compressed image output from the compressor is an encoded bitstream from which the input image can be exactly or approximately reconstructed.

The compression method is capable of producing a reconstructed image meeting a fidelity constraint specified by the user during compression, including lossless compression. A user may vary fidelity settings from band to band and change these settings periodically within an image.

For a given set of compression parameters, the length of the compressed image will vary depending on image content. That is, the compressed image is variable-length. A user could exploit the ability to adaptively vary fidelity settings within an image in an effort to meet a constraint on compressed image data volume; techniques for performing this optimization are outside the scope of this document. Reference [D1] presents some examples.

A user may choose to partition the output of an imaging instrument into multiple images that are separately compressed, for example, to limit the impact of data loss or corruption on the communications channel, or to limit the maximum possible size of a compressed image. This Recommended Standard does not address such partitioning or the tradeoffs associated with selecting the size of images produced under such partitioning. Reference [D1] presents some examples.

Figure 2-1 depicts the components of the compressor defined in this Recommended Standard. The compressor consists of a predictor followed by an encoder.

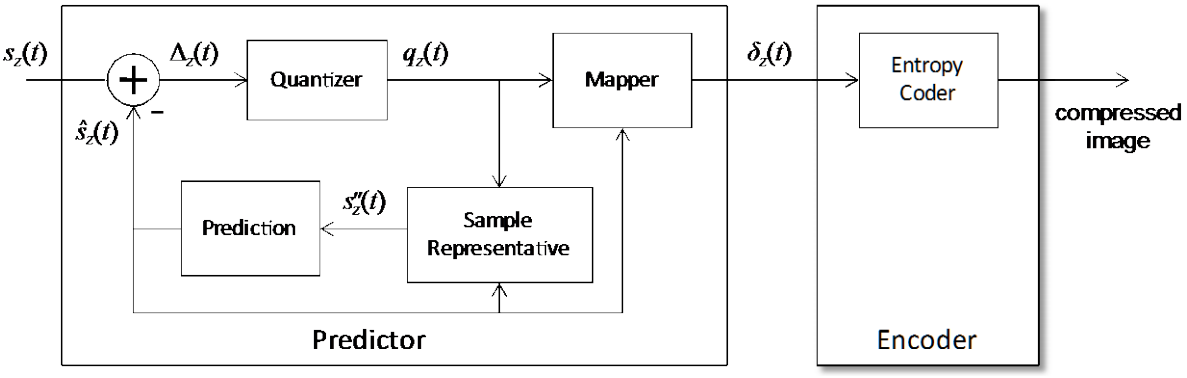

Figure 2-1: Compressor Schematic

The predictor, specified in section 4, uses an adaptive linear prediction method to predict the value of each image sample based on the values of nearby samples in a small three-dimensional neighborhood. Prediction is performed sequentially in a single pass.

The predictor makes use of an adaptively weighted prediction algorithm similar to the one in reference [D2]. Compared to reference [D2], this Recommended Standard has some minor changes in calculation of the prediction weights. More significantly, here the predictor cannot in general utilize the exact values of the original sample values $s_z(t)$ because these values will not be available to the decompressor at the time of reconstruction when compression is not lossless. Instead, prediction calculations are performed using a *sample representative* $s''_z(t)$ in place of each original sample value $s_z(t)$. The calculation of the sample representative is specified in 4.9.

The predictor in the present Recommended Standard also differs from that of reference [D2] in that each prediction residual $\Delta_z(t)$, that is, the difference between the predicted and actual sample values, is quantized using a uniform quantizer. The quantizer step size can be controlled via an *absolute error limit* (so that samples can be reconstructed with a user-specified error bound) and/or a *relative error limit* (so that samples predicted to have smaller magnitude can be reconstructed with lower error). Lossless compression in a band is obtained simply by setting the absolute error limit to zero. The quantized prediction residual $q_z(t)$ is mapped to an unsigned integer *mapped quantizer index* $\delta_z(t)$, similar to the calculation of mapped prediction residuals in reference [D2]. These mapped quantizer indices make up the output of the predictor.

The compressed image, specified in section 5, consists of a header that encodes image and compression parameters followed by a body that is produced by an entropy coder, which losslessly encodes the mapped quantizer indices. Entropy coder parameters are adaptively adjusted during this process to adapt to changes in the statistics of the mapped quantizer indices.

2.2 LOSSLESS COMPRESSION

Some simplification of the predictor arises when lossless compression is selected for the quantizer fidelity control method. Specifically, the quantization calculation (4.8) becomes trivial: $q_z(t) = \Delta_z(t)$.

In addition, under lossless compression, in the sample representative calculation (4.9), the offset parameter ψ_z has no effect and is defined as zero; and if a user chooses to set the damping parameter to zero, $\phi_z = 0$, then the sample representatives are equal to the original sample values, $s''_z(t) = s_z(t)$. This simplification can facilitate the ability to perform pipelining in a hardware implementation of the compressor.

It should be noted, however, that lossless compression performance may be improved by using a nonzero value for the damping parameter ϕ_z in the sample representative calculation.

Reference [D1] includes additional discussion.

2.3 BACKWARDS COMPATIBILITY

The features and compressed image header structure of the present Recommended Standard have been developed to ensure backwards compatibility with issue 1 of this Recommended Standard specified in reference [D2], which provided only lossless compression.

Specifically, reference [D2] can be viewed as a restricted case of the present Recommended Standard; a decompressor supporting all features of the present Recommended Standard would be able to decompress a compressed image that is compliant with issue 1. However, it should be noted that the additional features added in this issue are not limited to near-lossless compression capabilities. Thus, for example, a losslessly compressed image that is compliant with the present Recommended Standard might not be decompressible with a decompressor that is compliant with issue 1.

Table 2-1 enumerates the constraints that would need to be imposed on an implementation of this Recommended Standard to produce a compressor that is compliant with reference [D2].

Table 2-1: Backwards Compatibility with CCSDS-123.0-B-1

Reference	Constraint
3.3.1	Limit dynamic range to $D \leq 16$ bits.
3.5	Do not use supplementary information tables (set $\tau = 0$).
4.4	Do not use narrow local sums.
4.8	Set the quantizer fidelity control method to be lossless (4.8.2.1.1).

Reference	Constraint
4.9, 5.3.3	Set sample representative parameters to $\phi_z = \psi_z = 0$ for all z. Each sample representative $s_z''(t)$ will be equal to $s_z(t)$. Set $\Theta = 0$ so that the Predictor Metadata header part does not include the Sample Representative subpart.
4.10.3, 5.3.3.2	Set all weight exponent offsets $\varsigma_z^{(i)}$, ς_z^* to 0. In the Primary subpart of the Predictor Metadata header part, set the Weight Exponent Offset Flag to '0'.
5.4.3.2.3.4	If using the sample-adaptive entropy coder, do not use a rescaling counter size parameter γ^* value larger than 9.
5.4.3.3	Do not use the hybrid entropy coder.

2.4 DATA TRANSMISSION

The effects of a small error or data loss event can propagate to corrupt an entire compressed image (see reference [D1] for an example). Therefore, measures should be taken to minimize errors and data loss in the compressed image.

This Recommended Standard does not incorporate sync markers or other mechanisms to flag the header of an image; it is assumed that the transport mechanism used for the delivery of the encoded bitstream will provide the ability to locate the beginning and end of a compressed image and, in the event of data corruption, the header of the next image.

In case the encoded bitstream is to be transmitted over a CCSDS space link, several protocols can be used to transfer a compressed image, including but not limited to

- Space Packet Protocol (reference [D3]);
- CCSDS File Delivery Protocol (CFDP) (reference [D4]); and
- packet service as provided by the AOS Space Data Link Protocol (reference [D5]), TM Space Data Link Protocol (reference [D6]), and Unified Space Data Link Protocol (reference [D7]).

When transmission over a CCSDS space link occurs, application of one of the set of Channel Coding and Synchronization Recommended Standards will significantly reduce the loss of portions of transmitted data caused by data corruption.

Limits on the maximum size data unit that can be transmitted may be imposed by the protocol used or by other practical implementation considerations. The user is expected to take such limits into account when using this Recommended Standard.

3 IMAGE

3.1 OVERVIEW

This section defines parameters and notation pertaining to an image. Quantities defined in this section are summarized in table E-1 of annex E.

3.2 DIMENSIONS

3.2.1 An *image* is a three-dimensional array of signed or unsigned integer sample values $s_{z,y,x}$, where x and y are indices in the spatial dimensions, and the index z indicates the spectral band.

NOTES

1 When spatially adjacent data samples are produced by different instrument detector elements, changing values of the x index should correspond to changing detector elements. Thus, for a typical push-broom imager, the x and y dimensions would correspond to cross-track and along-track directions, respectively.

2 The spectral bands of the image need not be arranged in order of increasing or decreasing wavelength. Rearranging the order of spectral bands can affect compression performance. This Recommended Standard does not address the tradeoffs associated with such a band reordering. Reference [D1] includes some discussion of this topic.

3.2.2 Indices x, y, and z take on integer values in the ranges $0 \le x \le N_X - 1$, $0 \le y \le N_Y - 1$, and $0 \le z \le N_Z - 1$, where each image dimension N_X, N_Y, and N_Z shall have a value of at least 1 and at most 2^{16}.

3.2.3 A *frame* F_y is defined as the sub-array of all image sample values with the same y coordinate value; that is,

$$F_y(z, x) = s_{z,y,x} \text{ for any } 0 \le x \le N_X - 1,\ 0 \le z \le N_Z - 1. \tag{8}$$

3.3 DYNAMIC RANGE

3.3.1 Data samples shall have a fixed-size dynamic range of D bits, where D shall be an integer in the range $2 \le D \le 32$.

3.3.2 The quantities $s_{\min}$, $s_{\max}$, and s_{mid} denote the lower sample value limit, the upper sample value limit, and a mid-range sample value, respectively. When samples are unsigned integers, the values of $s_{\min}$, $s_{\max}$, and s_{mid} are defined as

$$s_{\min} = 0,\ s_{\max} = 2^D - 1,\ s_{\text{mid}} = 2^{D-1}, \tag{9}$$

and when samples are signed integers, the values of $s_{\min}$, $s_{\max}$, and s_{mid} are defined as

$$s_{\min} = -2^{D-1},\ s_{\max} = 2^{D-1} - 1,\ s_{\mathrm{mid}} = 0. \tag{10}$$

3.4 SAMPLE COORDINATE INDICES

For notational simplicity, data samples and associated quantities may be identified either by reference to the three indices x, y, z (e.g., $s_{z,y,x}$, $\delta_{z,y,x}$, etc.), or by the pair of indices t, z (e.g., $s_z(t)$, $\delta_z(t)$, etc.); that is,

$$s_z(t) \equiv s_{z,y,x} \tag{11}$$

$$\delta_z(t) \equiv \delta_{z,y,x} \tag{12}$$

etc., where

$$t = y \cdot N_{\mathrm{X}} + x. \tag{13}$$

NOTES

1 The value of t corresponds to the index of a sample within its spectral band when samples in the band are arranged in raster-scan order, starting with index t=0.

2 Given t, the values of x and y can be computed as

$$x = t \bmod N_{\mathrm{X}} \tag{14}$$

$$y = \left\lfloor t / N_{\mathrm{X}} \right\rfloor. \tag{15}$$

3.5 SUPPLEMENTARY INFORMATION TABLES

3.5.1 OVERVIEW

A user can choose to include up to 15 *supplementary information tables* to be encoded as part of the compressed image. Each such table is a zero-dimensional (a single element), one-dimensional (one element for each band z), or two-dimensional (one element for each (z, x) pair, or each (y, x) pair) table of floating-point, signed integer, or unsigned integer value(s). Such tables can be used to provide auxiliary image information to an end user, for example, the wavelength associated with each spectral band, a band-dependent scaling factor to convert reconstructed sample values to meaningful physical units, or a table identifying defective elements of a detector array. When used, such tables are encoded in the image header, as specified in 5.3.2.3.

3.5.2 SPECIFICATION

3.5.2.1 If supplementary information tables are used, the number of such tables, τ, shall be at most 15.

3.5.2.2 For each supplementary information table, the user shall identify a *purpose* for the table according to table 3-1. The 'reserved' purpose values are reserved for future use and shall not be used.

Table 3-1: Supplementary Information Table Purpose

Purpose	Interpretation
0	scale
1	offset
2	wavelength
3	full width at half maximum
4	defect indicator
5–9	reserved
10–15	user-defined

NOTE – The purpose is intended to indicate how a decompressor or end user might interpret the information in a supplementary information table. This does not impose any requirements on post-processing operations to be performed following decompression of an image.

3.5.2.3 Each supplementary information table *type* shall be unsigned integer, signed integer, or float.

3.5.2.3.1 For an unsigned integer table, the user-specified table bit depth D_{I} shall be an integer in the range $1 \leq D_{\mathrm{I}} \leq 32$, and each element of the table shall be an integer i in the range $0 \leq i \leq 2^{D_{\mathrm{I}}} - 1$.

3.5.2.3.2 For a signed integer table, the user-specified table bit depth D_{I} shall be an integer in the range $1 \leq D_{\mathrm{I}} \leq 32$, and each element of the table shall be an integer i in the range $-2^{D_{\mathrm{I}}-1} \leq i \leq 2^{D_{\mathrm{I}}-1} - 1$.

3.5.2.3.3 For a float table, user-specified significand and exponent bit depths D_{F} and D_{E} shall be integers in the range $1 \leq D_{\mathrm{F}} \leq 23$, $2 \leq D_{\mathrm{E}} \leq 8$, and the user-specified exponent bias β shall be an integer in the range $0 \leq \beta \leq 2^{D_{\mathrm{E}}} - 1$. Each element of the table shall consist of a sign bit b that is either 0 or 1, an exponent α that is an integer in the range $0 \leq \alpha \leq 2^{D_{\mathrm{E}}} - 1$, and a significand j that is an integer in the range $0 \leq j \leq 2^{D_{\mathrm{F}}} - 1$. If the exponent α is 0, the value represented is

$$(-1)^b \cdot j \cdot 2^{1-\beta-D_{\mathrm{F}}}. \tag{16}$$

If the exponent α is $2^{D_E}-1$, the value represented is non-numeric:

- $+\infty$ if $b = 0$ and $j = 0$;
- $-\infty$ if $b = 1$ and $j = 0$;
- NaN ('not a number', an undefined or unrepresentable value) if $j \neq 0$.

Otherwise, $0 < \alpha < 2^{D_E}-1$, and the value represented is

$$(-1)^b \left(2^{D_F} + j\right) 2^{\alpha-\beta-D_F}. \tag{17}$$

NOTE – When $D_E = 8$, $D_F = 23$, and $\beta = 127$, the float table representation of values is the same as the IEEE 754 single-precision binary floating-point format (binary32). When $D_E = 5$, $D_F = 10$, and $\beta = 15$, the float table representation of values is the same as the IEEE 754 half-precision binary floating-point format (binary16).

3.5.2.4 Each supplementary information table *structure* shall be zero-dimensional, one-dimensional, two-dimensional-*zx*, or two-dimensional-*yx*.

3.5.2.4.1 A zero-dimensional signed or unsigned integer supplementary information table consists of a single integer i.

3.5.2.4.2 A one-dimensional signed or unsigned integer supplementary information table consists of N_Z integers i_z, for $z = 0,\ldots, N_Z - 1$.

3.5.2.4.3 A two-dimensional-*zx* signed or unsigned integer supplementary information table consists of $N_Z \cdot N_X$ integers $i_{z,x}$, for $z = 0,\ldots, N_Z - 1$, $x = 0,\ldots, N_X - 1$.

3.5.2.4.4 A two-dimensional-*yx* signed or unsigned integer supplementary information table consists of $N_Y \cdot N_X$ integers $i_{y,x}$, for $y = 0,\ldots, N_Y - 1$, $x = 0,\ldots, N_X - 1$.

3.5.2.4.5 A zero-dimensional float supplementary information table consists of a single element, defined by sign bit b, significand j, and exponent α.

3.5.2.4.6 A one-dimensional float supplementary information table consists of N_Z elements, each defined by sign bit b_z, significand j_z, and exponent α_z, for $z = 0,\ldots, N_Z - 1$.

3.5.2.4.7 A two-dimensional-*zx* float supplementary information table consists of $N_Z \cdot N_X$ elements, each defined by sign bit $b_{z,x}$, significand $j_{z,x}$, and exponent $\alpha_{z,x}$, for $z = 0,\ldots, N_Z - 1$, $x = 0,\ldots, N_X - 1$.

3.5.2.4.8 A two-dimensional-*yx* float supplementary information table consists of $N_Y \cdot N_X$ elements, each defined by sign bit $b_{y,x}$, significand $j_{y,x}$, and exponent $\alpha_{y,x}$, for $y = 0,\ldots, N_Y - 1$, $x = 0,\ldots, N_X - 1$.

4 PREDICTOR

4.1 OVERVIEW

This section specifies the calculation of the *predicted sample values* $\hat{s}_{z,y,x}$ and *mapped quantizer indices* $\delta_{z,y,x}$ from the input image samples $s_{z,y,x}$. Quantities defined in this section are summarized in table E-2 of annex E.

This Recommended Standard makes use of the same adaptively weighted predictor as reference [D2], but to accommodate near-lossless compression, prediction calculations are performed using *sample representatives* $s''_{z,y,x}$, defined in 4.9, in place of the original sample values $s_{z,y,x}$. This is necessary so that the decompressor can duplicate the prediction calculation.

Prediction can be performed causally in a single pass through the image. Prediction at sample $s_{z,y,x}$, that is, the calculation of $\hat{s}_{z,y,x}$ and $\delta_{z,y,x}$, generally depends on the values of sample representatives for nearby samples in the current spectral band and P preceding (i.e., lower-indexed) spectral bands, where P is a user-specified parameter (see 4.2). Figure 4-1 illustrates the typical neighborhood used for prediction; this neighborhood is suitably truncated when $y = 0$, $x = 0$, $x = N_X - 1$, or $z < P$.

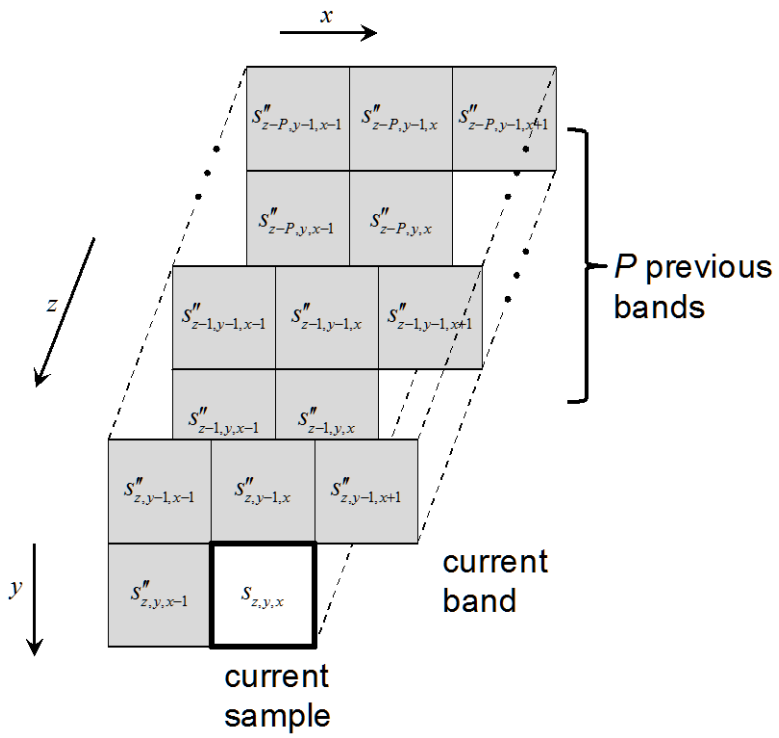

Figure 4-1: Typical Prediction Neighborhood

Within each spectral band, the predictor computes a *local sum* of neighboring sample representative values (see 4.4). Each such local sum is used to compute a *local difference* (see 4.5). Predicted sample values are calculated using the local sum in the current spectral band and a weighted sum of local difference values from the current and previous spectral bands (see 4.7). The *weights* (see 4.6) used in this calculation are adaptively updated (see 4.10) following the calculation of each predicted sample value. Each prediction residual, that is, the difference between a given sample value $s_{z,y,x}$ and the corresponding predicted sample value $\hat{s}_{z,y,x}$, is quantized (see 4.8) and then mapped to an unsigned integer $\delta_{z,y,x}$, the mapped quantizer index (see 4.11). The quantized value of sample $s_{z,y,x}$ is used to calculate a corresponding *sample representative* value $s''_{z,y,x}$ (see 4.9).

The local sum $\sigma_{z,y,x}$ (see 4.4) is a weighted sum of sample representatives in spectral band z that are adjacent to sample $s_{z,y,x}$. Figure 4-2 illustrates the sample representatives used to calculate the local sum. A user may choose to perform prediction using *neighbor-oriented* or *column-oriented* local sums for an image, and local sums may be *wide* or *narrow*. When neighbor-oriented local sums are used, the local sum is equal to a combination of up to four neighboring sample representative values in the spectral band (except when $y = 0$, $x = 0$, or $x = N_X - 1$, in which case these four values are not all available, and the local sum calculation is suitably modified, as detailed in 4.4). When column-oriented local sums are used, the local sum is equal to four times the neighboring sample representative value in the previous row (except when $y = 0$, in which case this value is not available and the local sum calculation is suitably modified as detailed in 4.4). Narrow local sums are defined to eliminate the dependency on sample representative $s''_{z,y,x-1}$ when calculating $\sigma_{z,y,x}$, which may facilitate pipelining in a hardware implementation.

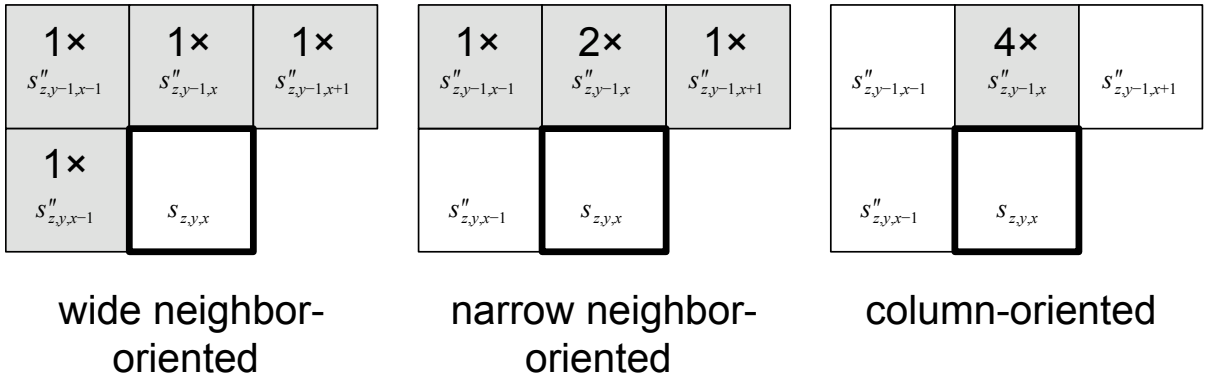

Figure 4-2: Samples Used to Calculate Local Sums

The local sums are used to calculate local difference values. In each spectral band, the *central local difference*, $d_{z,y,x}$, is equal to the difference between the local sum $\sigma_{z,y,x}$ and four times the sample representative value $s''_{z,y,x}$ (see 4.5.1). The three *directional local differences*, $d^{\mathrm{N}}_{z,y,x}$, $d^{\mathrm{W}}_{z,y,x}$, and $d^{\mathrm{NW}}_{z,y,x}$, are each equal to the difference between $\sigma_{z,y,x}$ and four times a sample value labeled as 'N', 'W', or 'NW' in figure 4-3 (except when this sample value is not available, that is, at image edges, as detailed in 4.5.2).

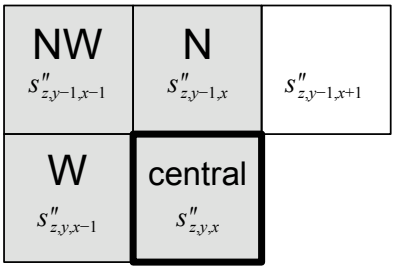

Figure 4-3: Computing Local Differences in a Spectral Band

A user may choose to perform prediction for an image in *full* or *reduced* mode (see 4.3). Under reduced mode, prediction depends on a weighted sum of the central local differences computed in preceding bands; the directional local differences are not used, and thus need not be calculated, under reduced mode. Under full mode, prediction depends on a weighted sum of the central local differences computed in preceding bands and the three directional local differences computed in the current band.

As described in reference [D1], the use of reduced mode in combination with column-oriented local sums tends to yield smaller compressed image data volumes for raw (uncalibrated) input images from push-broom imagers that exhibit significant along-track streaking artifacts. The use of full mode in combination with neighbor-oriented local sums tends to yield smaller compressed image data volumes for whiskbroom imagers, frame imagers, and calibrated imagery.

The prediction residual, the difference between the sample value $s_{z,y,x}$ and the predicted sample value $\hat{s}_{z,y,x}$, is quantized (see 4.8), and the quantizer index is mapped to an unsigned integer $\delta_{z,y,x}$ (see 4.11). This mapping is invertible, so that the decompressor can exactly reconstruct the quantizer index. User-specified *absolute* and/or *relative error limit* values (see 4.8) control the *maximum error* value $m_z(t)$ for each sample. Reconstruction of sample $s_{z,y,x}$ with at most $m_z(t)$ units of error can be achieved by a decompressor. However, the Recommended Standard imposes no specific requirements on reconstructing a sample by a decompressor, and minimizing the maximum reconstruction error of each sample may not minimize other distortion metrics (see reference [D1] for an example).

4.2 NUMBER OF BANDS FOR PREDICTION

The user-specified parameter P, which shall be an integer in the range $0 \leq P \leq 15$, determines the number of preceding spectral bands used for prediction. Specifically, prediction in spectral band z depends on central local differences, defined in 4.5.1, computed in bands $z-1, z-2, \ldots, z-P_z^*$, where

$$P_z^* = \min\{z, P\}. \tag{18}$$

4.3 FULL AND REDUCED PREDICTION MODES

4.3.1 A user may choose to perform prediction using *full* or *reduced* mode for an image, except when the image has width of one (i.e., $N_X = 1$), in which case reduced mode shall be used.

4.3.2 Under both full and reduced modes, prediction in spectral band z makes use of central local differences from the preceding P_z^* spectral bands. Under full prediction mode, prediction in spectral band z additionally makes use of three directional local differences, defined in 4.5.2, computed in the current spectral band z. Thus the number of local difference values used for prediction at each sample in band z, denoted C_z, is

$$C_z = \begin{cases} P_z^*, & \text{reduced prediction mode} \\ P_z^* + 3, & \text{full prediction mode} \end{cases}. \tag{19}$$

4.4 LOCAL SUM

4.4.1 The *local sum* $\sigma_{z,y,x}$ is an integer equal to a weighted sum of previous sample representative values in band z that are neighbors of sample $s_{z,y,x}$. A user may choose to perform prediction using *neighbor-oriented* or *column-oriented* local sums for an image, except when the image has width 1 (i.e., $N_X = 1$), in which case column-oriented local sums shall be used. In either case, a user may choose to use *wide* or *narrow* local sums.

NOTE – Column-oriented local sums are not suggested under full prediction mode.

4.4.2 When wide neighbor-oriented local sums are used, $\sigma_{z,y,x}$ is defined as

$$\sigma_{z,y,x} = \begin{cases} s''_{z,y,x-1} + s''_{z,y-1,x-1} + s''_{z,y-1,x} + s''_{z,y-1,x+1}, & y > 0,\ 0 < x < N_X - 1 \\ 4s''_{z,y,x-1}, & y = 0,\ x > 0 \\ 2\left(s''_{z,y-1,x} + s''_{z,y-1,x+1}\right), & y > 0,\ x = 0 \\ s''_{z,y,x-1} + s''_{z,y-1,x-1} + 2s''_{z,y-1,x}, & y > 0,\ x = N_X - 1 \end{cases}; \tag{20}$$

when narrow neighbor-oriented local sums are used, $\sigma_{z,y,x}$ is defined as

$$\sigma_{z,y,x} = \begin{cases} s''_{z,y-1,x-1} + 2s''_{z,y-1,x} + s''_{z,y-1,x+1}, & y > 0,\ 0 < x < N_X - 1 \\ 4s''_{z-1,y,x-1}, & y = 0,\ x > 0,\ z > 0 \\ 2\left(s''_{z,y-1,x} + s''_{z,y-1,x+1}\right), & y > 0,\ x = 0 \\ 2\left(s''_{z,y-1,x-1} + s''_{z,y-1,x}\right), & y > 0,\ x = N_X - 1 \\ 4s_{\text{mid}}, & y = 0,\ x > 0,\ z = 0 \end{cases}; \tag{21}$$

when wide column-oriented local sums are used, $\sigma_{z,y,x}$ is defined as

$$\sigma_{z,y,x} = \begin{cases} 4s''_{z,y-1,x}, & y > 0 \\ 4s''_{z,y,x-1}, & y = 0,\ x > 0 \end{cases}; \tag{22}$$

and when narrow column-oriented local sums are used, $\sigma_{z,y,x}$ is defined as

$$\sigma_{z,y,x} = \begin{cases} 4s''_{z,y-1,x}, & y > 0 \\ 4s''_{z-1,y,x-1}, & y = 0,\ x > 0,\ z > 0 \\ 4s_{mid}, & y = 0,\ x > 0,\ z = 0 \end{cases}, \tag{23}$$

where sample representative values $s''_{z,y,x}$ are defined in 4.9.

NOTE – The value of $\sigma_{z,0,0}$ is not defined, as it is not needed.

4.5 LOCAL DIFFERENCES

4.5.1 CENTRAL LOCAL DIFFERENCE

When x and y are not both zero (i.e., when $t > 0$), the central local difference $d_{z,y,x}$ is defined as

$$d_{z,y,x} = 4\, s''_{z,y,x} - \sigma_{z,y,x}\,. \tag{24}$$

4.5.2 DIRECTIONAL LOCAL DIFFERENCES

When x and y are not both zero (i.e., when $t > 0$), the three directional local differences are defined as

$$d^{\mathrm{N}}_{z,y,x} = \begin{cases} 4s''_{z,y-1,x} - \sigma_{z,y,x}, & y > 0 \\ 0, & y = 0 \end{cases}, \tag{25}$$

$$d^{\mathrm{W}}_{z,y,x} = \begin{cases} 4s''_{z,y,x-1} - \sigma_{z,y,x}, & x > 0,\ y > 0 \\ 4s''_{z,y-1,x} - \sigma_{z,y,x}, & x = 0,\ y > 0 \\ 0, & y = 0 \end{cases}, \text{ and} \tag{26}$$

$$d^{\mathrm{NW}}_{z,y,x} = \begin{cases} 4s''_{z,y-1,x-1} - \sigma_{z,y,x}, & x > 0,\ y > 0 \\ 4s''_{z,y-1,x} - \sigma_{z,y,x}, & x = 0,\ y > 0 \\ 0, & y = 0 \end{cases}. \tag{27}$$

NOTE – Directional local differences are not used under reduced prediction mode.

4.5.3 LOCAL DIFFERENCE VECTOR

For $t > 0$, the local difference vector $\mathbf{U}_z(t)$ is a vector of the C_z local difference values used to calculate the predicted sample value $\hat{s}_z(t)$. Under full prediction mode, $\mathbf{U}_z(t)$ is defined as

$$\mathbf{U}_z(t) = \begin{bmatrix} d_z^{\mathrm{N}}(t) \\ d_z^{\mathrm{W}}(t) \\ d_z^{\mathrm{NW}}(t) \\ d_{z-1}(t) \\ d_{z-2}(t) \\ \vdots \\ d_{z-P_z^*}(t) \end{bmatrix}, \tag{28}$$

and under reduced prediction mode, for $z > 0$, $\mathbf{U}_z(t)$ is defined as

$$\mathbf{U}_z(t) = \begin{bmatrix} d_{z-1}(t) \\ d_{z-2}(t) \\ \vdots \\ d_{z-P_z^*}(t) \end{bmatrix}. \tag{29}$$

NOTE – Under reduced mode, $\mathbf{U}_0(t)$ is not defined, as it is not needed.

4.6 WEIGHTS

4.6.1 WEIGHT VALUES AND WEIGHT RESOLUTION

4.6.1.1 In the prediction calculation (see 4.7), for $t > 0$, each component of the local difference vector $\mathbf{U}_z(t)$ is multiplied by a corresponding integer *weight value*.

4.6.1.2 The resolution of the weight values is controlled by the user-specified parameter Ω, which shall be an integer in the range $4 \leq \Omega \leq 19$.

4.6.1.3 Each weight value is a signed integer quantity that can be represented using $\Omega + 3$ bits. Thus each weight value has minimum and maximum possible values $\omega_{\min}$ and $\omega_{\max}$, respectively, where

$$\omega_{\min} = -2^{\Omega+2}, \; \omega_{\max} = 2^{\Omega+2} - 1. \tag{30}$$

NOTE – Increasing the number of bits used to represent weight values (i.e., using a larger value of Ω) provides increased resolution in the prediction calculation. This Recommended Standard does not address the tradeoffs associated with selecting the value of Ω. Reference [D1] presents some examples.

4.6.2 WEIGHT VECTOR

The weight vector $\mathbf{W}_z(t)$ is a vector of the C_z weight values used in prediction. Under full prediction mode,

$$\mathbf{W}_z(t) = \begin{bmatrix} \omega_z^{\mathrm{N}}(t) \\ \omega_z^{\mathrm{W}}(t) \\ \omega_z^{\mathrm{NW}}(t) \\ \omega_z^{(1)}(t) \\ \omega_z^{(2)}(t) \\ \vdots \\ \omega_z^{(P_z^*)}(t) \end{bmatrix}, \tag{31}$$

and under reduced prediction mode, for $z > 0$,

$$\mathbf{W}_z(t) = \begin{bmatrix} \omega_z^{(1)}(t) \\ \omega_z^{(2)}(t) \\ \vdots \\ \omega_z^{(P_z^*)}(t) \end{bmatrix}, \tag{32}$$

where the weight values are calculated as specified in 4.6.3 and 4.10.

NOTE – Under reduced mode, $\mathbf{W}_0(t)$ is not defined as it is not needed.

4.6.3 INITIALIZATION

4.6.3.1 General

A user may choose to use either *default* or *custom* weight initialization, defined below, to select the initial weight vector $\mathbf{W}_z(1)$ for each spectral band z. The same weight initialization method shall be used for all spectral bands.

4.6.3.2 Default Weight Initialization

4.6.3.2.1 When default weight initialization is used, for each spectral band z, initial weight vector components $\omega_z^{(1)}(1)$, $\omega_z^{(2)}(1)$, …, $\omega_z^{(P_z^*)}(1)$, shall be assigned values

$$\omega_z^{(1)}(1) = \frac{7}{8}2^{\Omega}, \quad \omega_z^{(i)}(1) = \left\lfloor \frac{1}{8}\omega_z^{(i-1)}(1) \right\rfloor, i = 2,3,\ldots,P_z^* . \tag{33}$$

4.6.3.2.2 With this option, under full prediction mode the remaining components of $\mathbf{W}_z(1)$ shall be assigned values

$$\omega_z^{\mathrm{N}}(1) = \omega_z^{\mathrm{W}}(1) = \omega_z^{\mathrm{NW}}(1) = 0 . \tag{34}$$

4.6.3.3 Custom Weight Initialization

4.6.3.3.1 When custom weight initialization is used, for each spectral band z, the initial weight vector $\mathbf{W}_z(1)$ shall be assigned using a user-specified *weight initialization vector* Λ_z, consisting of C_z signed Q-bit integer components.

NOTES

1 The weight initialization vector Λ_z may be encoded in the header as described in 5.3.

2 A weight initialization vector Λ_z might be selected based on instrument characteristics or training data, or might be selected based on a weight vector from a previous compressed image.

4.6.3.3.2 The weight initialization resolution Q shall be a user-specified integer in the range $3 \le Q \le \Omega + 3$ bits.

4.6.3.3.3 The initial weight vector $\mathbf{W}_z(1)$ shall be calculated from Λ_z by

$$\mathbf{W}_z(1) = 2^{\Omega+3-Q}\Lambda_z + \left\lceil 2^{\Omega+2-Q} - 1 \right\rceil \mathbf{1}, \tag{35}$$

where **1** denotes a vector of all ‘ones’.

NOTE – In the $(\Omega + 3)$-bit two’s complement representation of each component of $\mathbf{W}_z(1)$, the Q MSBs are equal to the binary representation of the corresponding component of Λ_z. The remaining bits, if any, are made up of a ‘0’ bit followed by ‘1’ bits in the remaining positions.

4.7 PREDICTION CALCULATION

4.7.1 For $t > 0$, the predicted central local difference $\hat{d}_z(t)$ is equal to the inner product of vectors $\mathbf{W}_z(t)$ and $\mathbf{U}_z(t)$:

$$\hat{d}_z(t) = \mathbf{W}_z^{\mathrm{T}}(t)\mathbf{U}_z(t), \tag{36}$$

except for $z = 0$ under reduced mode, in which case $\hat{d}_z(t) = 0$.

4.7.2 The high-resolution predicted sample value, $\breve{s}_z(t)$, is calculated as

$$\breve{s}_z(t) = \mathrm{clip}\left(\mathrm{mod}_R^*\left[\hat{d}_z(t) + 2^{\Omega}\left(\sigma_z(t) - 4s_{\mathrm{mid}}\right)\right] + 2^{\Omega+2}s_{\mathrm{mid}} + 2^{\Omega+1}, \left\{2^{\Omega+2}s_{\min}, 2^{\Omega+2}s_{\max} + 2^{\Omega+1}\right\}\right), \tag{37}$$

where the user-selected register size parameter R shall be an integer in the range $\max\{32, D + \Omega + 2\} \leq R \leq 64$.

NOTE – Increasing the register size R reduces the chance of an overflow occurring in the calculation of a high-resolution predicted sample value. This Recommended Standard does not address the tradeoffs associated with selecting the value of R. Reference [D1] provides some discussion.

4.7.3 The double-resolution predicted sample value is

$$\tilde{s}_z(t) = \begin{cases} \left\lfloor \dfrac{\breve{s}_z(t)}{2^{\Omega+1}} \right\rfloor, & t > 0 \\ 2s_{z-1}(t), & t = 0, P > 0, z > 0 \\ 2s_{\mathrm{mid}}, & t = 0 \text{ and } (P = 0 \text{ or } z = 0) \end{cases}. \tag{38}$$

4.7.4 The predicted sample value $\hat{s}_z(t)$ is defined as

$$\hat{s}_z(t) = \left\lfloor \frac{\tilde{s}_z(t)}{2} \right\rfloor. \tag{39}$$

4.8 QUANTIZATION

4.8.1 QUANTIZER OUTPUT

The prediction residual $\Delta_z(t)$ is the difference between the predicted and actual sample values,

$$\Delta_z(t) = s_z(t) - \hat{s}_z(t)\,. \tag{40}$$

The prediction residual shall be quantized using a uniform quantizer with step size $2m_z(t) + 1$, producing as quantizer output the signed integer *quantizer index* $q_z(t)$, defined as

$$q_z(t) = \begin{cases} \Delta_z(0), & t = 0 \\ \mathrm{sgn}(\Delta_z(t))\left\lfloor \dfrac{|\Delta_z(t)| + m_z(t)}{2m_z(t)+1} \right\rfloor, & t > 0 \end{cases}, \tag{41}$$

where the *maximum error* value $m_z(t)$ is determined via user-specified quantizer fidelity settings as specified in 4.8.2.

NOTE – Given $q_z(t)$, reconstruction of sample $s_z(t)$ with no more than $m_z(t)$ units of error is possible. Thus lossless compression is achieved for this sample when $m_z(t) = 0$.

4.8.2 FIDELITY CONTROL

4.8.2.1 Controlling Maximum Error

4.8.2.1.1 For a given image, a user may choose the quantizer fidelity control method to be *lossless*, in which case

$$m_z(t) = 0 \tag{42}$$

for all z and t. Otherwise, the user may control the maximum error value $m_z(t)$ by specifying an *absolute error limit* a_z for each z, a *relative error limit* r_z for each z, or both.

NOTE – Restrictions on allowed error limit values are specified in 4.8.2.2.

4.8.2.1.2 When only absolute error limits are used, the maximum error shall be computed as

$$m_z(t) = a_z \tag{43}$$

for all z and t; when only relative error limits are used,

$$m_z(t) = \left\lfloor \frac{r_z |\hat{s}_z(t)|}{2^D} \right\rfloor \tag{44}$$

for all z and t; and when both absolute and relative error limits are used,

$$m_z(t) = \min\left(a_z, \left\lfloor \frac{r_z \left| \hat{s}_z(t) \right|}{2^D} \right\rfloor \right) \tag{45}$$

for all z and t.

4.8.2.2 Allowed Error Limit Values

4.8.2.2.1 If absolute error limits are used, then for each spectral band z, the value of a_z shall be an integer in the range $0 \le a_z \le 2^{D_A} - 1$, where the user-specified *absolute error limit bit depth* D_A shall be an integer in the range $1 \le D_A \le \min\{D-1, 16\}$.

4.8.2.2.2 If relative error limits are used, then for each spectral band z, the value of r_z shall be an integer in the range $0 \le r_z \le 2^{D_R} - 1$, where the user-specified *relative error limit bit depth* D_R shall be an integer in the range $1 \le D_R \le \min\{D-1, 16\}$.

4.8.2.3 Error Limit Assignment Methods

4.8.2.3.1 If used, absolute error limits shall be either (a) *band-dependent*, in which case the user shall specify a set of absolute error limit values $\{a_z\}_{z=0}^{N_Z-1}$, or (b) *band-independent*, in which case $a_z = A^*$ for each spectral band z, where A^* shall be the user-specified integer *absolute error limit constant*, satisfying $0 \le A^* \le 2^{D_A} - 1$.

4.8.2.3.2 If used, relative error limits shall be either (a) *band-dependent*, in which case the user shall specify a set of relative error limit values $\{r_z\}_{z=0}^{N_Z-1}$, or (b) *band-independent*, in which case $r_z = R^*$ for each spectral band z, where R^* shall be the user-specified integer *relative error limit constant*, satisfying $0 \le R^* \le 2^{D_R} - 1$.

NOTE – When both absolute and relative error limits are used for an image, the choice of assignment methods for relative and absolute error limits need not be the same. That is, band-independent absolute error limits may be used in combination with band-dependent relative error limits, and vice-versa.

4.8.2.4 Periodic Error Limit Updating

4.8.2.4.1 When used, error limit values may be fixed for an entire image, or the user may choose to use *periodic error limit updating*, in which case error limit values are periodically updated.

4.8.2.4.2 When periodic error limit updating is used, the user shall provide error limit values every 2^u frames, where the user-specified *error limit update period exponent u* shall be an integer in the range $0 \leq u \leq 9$.

4.8.2.4.3 All other quantizer fidelity settings (choice to use absolute and/or relative error limits, choice between band-dependent and band-independent assignment methods for the error limit method[s] in use, and error limit bit depth[s]) shall be fixed for the entire image.

4.8.2.4.4 Periodic error limit updating shall not be used with Band-SeQuential (BSQ) input order (defined in 5.4.2.3).

4.9 SAMPLE REPRESENTATIVES

4.9.1 Sample representatives are calculated using user-specified *resolution* parameter Θ, which shall be an integer in the range $0 \leq \Theta \leq 4$, and for each spectral band z, parameters *damping*, ϕ_z, and *offset*, ψ_z.

4.9.1.1 Each ϕ_z shall be a user-specified integer in the range $0 \leq \phi_z \leq 2^{\Theta} - 1$.

4.9.1.2 Each ψ_z shall be a user-specified integer in the range $0 \leq \psi_z \leq 2^{\Theta} - 1$, unless lossless fidelity control is used, in which case $\psi_z = 0$.

4.9.2 The sample representative $s''_z(t)$, which has the same resolution as the original samples, shall be calculated as

$$s''_z(t) = \begin{cases} s_z(0), & t = 0 \\ \left\lfloor \dfrac{\tilde{s}''_z(t) + 1}{2} \right\rfloor, & t > 0 \end{cases} \tag{46}$$

from the double-resolution sample representative

$$\tilde{s}''_z(t) = \left\lfloor \frac{4\left(2^{\Theta} - \phi_z\right) \cdot \left(s'_z(t) \cdot 2^{\Omega} - \mathrm{sgn}(q_z(t)) \cdot m_z(t) \cdot \psi_z \cdot 2^{\Omega - \Theta}\right) + \phi_z \cdot \breve{s}_z(t) - \phi_z \cdot 2^{\Omega+1}}{2^{\Omega+\Theta+1}} \right\rfloor, \tag{47}$$

where $\breve{s}_z(t)$ is the high-resolution predicted sample value defined in 4.7.2, and

$$s'_z(t) = \text{clip}\left(\hat{s}_z(t) + q_z(t)\left(2m_z(t)+1\right), \{s_{\min}, s_{\max}\}\right) \tag{48}$$

is a clipped version of the quantizer bin center.

NOTES

1 Reconstructing sample $s_z(t)$ with value $s'_z(t)$ by the decompressor ensures that reconstruction error will be at most $m_z(t)$. If $m_z(t) = 0$ then $s'_z(t) = s_z(t)$.

2 Setting $\phi_z = \psi_z = 0$ causes the sample representative $s''_z(t)$ to be equal to $s'_z(t)$.

3 The difference between the sample representative $s''_z(t)$ and the predicted sample value $\hat{s}_z(t)$ may exceed $m_z(t)$.

4.10 WEIGHT UPDATE

4.10.1 The double-resolution prediction error $e_z(t)$ is an integer defined as

$$e_z(t) = 2s'_z(t) - \tilde{s}_z(t). \tag{49}$$

4.10.2 For $t > 0$, the weight update scaling exponent $\rho(t)$ is an integer defined as

$$\rho(t) = \text{clip}\left(v_{\min} + \left\lfloor \frac{t - N_X}{t_{\text{inc}}} \right\rfloor, \{v_{\min}, v_{\max}\}\right) + D - \Omega, \tag{50}$$

where user-specified integer parameters $v_{\min}$, $v_{\max}$, and t_{inc} are constrained as follows:

a) The values of $v_{\min}$ and $v_{\max}$ shall be integers in the range $-6 \le v_{\min} \le v_{\max} \le 9$.

b) The weight update scaling exponent change interval t_{inc} shall be a power of 2 in the range $2^4 \le t_{\text{inc}} \le 2^{11}$.

NOTE – These parameters control the rate at which weights adapt to image data statistics. The initial weight update scaling exponent is $\rho(1) = v_{\min} + D - \Omega$, and at regular intervals determined by the value of t_{inc}, $\rho(t)$ is incremented by one until reaching a final value $v_{\max} + D - \Omega$. Smaller values of $\rho(t)$ produce larger weight increments, yielding faster adaptation to source statistics but worse steady-state compression performance.

4.10.3 For $t > 0$, following the calculation of $\tilde{s}_z(t)$, components of the next weight vector in the spectral band, $\mathbf{W}_z(t+1)$, are defined as

$$\omega_z^{(i)}(t+1) = \mathrm{clip}\left(\omega_z^{(i)}(t) + \left\lfloor \frac{1}{2}\left(\mathrm{sgn}^+\left[e_z(t)\right] \cdot 2^{-\left(\rho(t)+\varsigma_z^{(i)}\right)} \cdot d_{z-i}(t) + 1 \right) \right\rfloor, \left\{ \omega_{\min}, \omega_{\max} \right\} \right), \tag{51}$$

and, when full prediction mode is used, for the directional components,

$$\omega_z^{\mathrm{N}}(t+1) = \mathrm{clip}\left(\omega_z^{\mathrm{N}}(t) + \left\lfloor \frac{1}{2}\left(\mathrm{sgn}^+\left[e_z(t)\right] \cdot 2^{-\left(\rho(t)+\varsigma_z^{*}\right)} \cdot d_z^{\mathrm{N}}(t) + 1 \right) \right\rfloor, \left\{ \omega_{\min}, \omega_{\max} \right\} \right), \tag{52}$$

$$\omega_z^{\mathrm{W}}(t+1) = \mathrm{clip}\left(\omega_z^{\mathrm{W}}(t) + \left\lfloor \frac{1}{2}\left(\mathrm{sgn}^+\left[e_z(t)\right] \cdot 2^{-\left(\rho(t)+\varsigma_z^{*}\right)} \cdot d_z^{\mathrm{W}}(t) + 1 \right) \right\rfloor, \left\{ \omega_{\min}, \omega_{\max} \right\} \right), \tag{53}$$

$$\omega_z^{\mathrm{NW}}(t+1) = \mathrm{clip}\left(\omega_z^{\mathrm{NW}}(t) + \left\lfloor \frac{1}{2}\left(\mathrm{sgn}^+\left[e_z(t)\right] \cdot 2^{-\left(\rho(t)+\varsigma_z^{*}\right)} \cdot d_z^{\mathrm{NW}}(t) + 1 \right) \right\rfloor, \left\{ \omega_{\min}, \omega_{\max} \right\} \right). \tag{54}$$

4.10.4 The inter-band weight exponent offsets $\varsigma_z^{(i)}$, for $z = 0,\ldots, N_{\mathrm{Z}}-1$ and $i = 1,...,P_z^*$, and intra-band weight exponent offsets ς_z^* shall be user-specified integers in the range $-6 \le \varsigma_z^{(i)} \le 5$ and $-6 \le \varsigma_z^* \le 5$, respectively.

NOTE – The quantity $\left\lfloor \frac{1}{2}\left(\mathrm{sgn}^+\left[e_z(t)\right] \cdot 2^{-(\rho(t)+\zeta)} \cdot d + 1 \right) \right\rfloor$ is equivalent to $\left\lfloor \frac{1}{2}\left(\left\lfloor \mathrm{sgn}^+\left[e_z(t)\right] \cdot 2^{-(\rho(t)+\zeta)} \cdot d \right\rfloor + 1 \right) \right\rfloor$ but is not in general equivalent to $\left\lfloor \frac{1}{2}\left(\mathrm{sgn}^+\left[e_z(t)\right] \cdot \left\lfloor 2^{-(\rho(t)+\zeta)} \cdot d \right\rfloor + 1 \right) \right\rfloor$.

4.11 MAPPED QUANTIZER INDEX

The signed quantizer index $q_z(t)$ is converted to an unsigned *mapped quantizer index* $\delta_z(t)$ defined as

$$\delta_z(t) = \begin{cases} \left|q_z(t)\right| + \theta_z(t), & \left|q_z(t)\right| > \theta_z(t) \\ 2\left|q_z(t)\right|, & 0 \le (-1)^{\tilde{s}_z(t)} q_z(t) \le \theta_z(t) \\ 2\left|q_z(t)\right| - 1, & \text{otherwise} \end{cases}, \tag{55}$$

where

$$\theta_z(t) = \begin{cases} \min\left\{ \hat{s}_z(0) - s_{\min}, s_{\max} - \hat{s}_z(0) \right\} & t = 0 \\ \min\left\{ \left\lfloor \frac{\hat{s}_z(t) - s_{\min} + m_z(t)}{2m_z(t)+1} \right\rfloor, \left\lfloor \frac{s_{\max} - \hat{s}_z(t) + m_z(t)}{2m_z(t)+1} \right\rfloor \right\}, & t > 0 \end{cases}. \tag{56}$$

NOTE – Each mapped quantizer index $\delta_z(t)$ can be represented as a D-bit unsigned integer.