INTERNATIONAL STANDARD

ISO/IEC 23008-10

First edition
2015-04-15

AMENDMENT 1
2021-12

Information technology — High efficiency coding and media delivery in heterogeneous environments —

Part 10: MPEG media transport forward error correction (FEC) codes

AMENDMENT 1: Window-based FEC code

Technologies de l'information — Codage à haute efficacité et livraison des medias dans des environnements hétérogènes —

Partie 10: Codes de correction d'erreur anticipée pour le transport des medias MPEG

AMENDEMENT 1

Reference number
ISO/IEC 23008-10:2015/Amd. 1:2021(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

A list of all parts in the ISO/IEC 23008 series can be found on the ISO website and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Information technology — High efficiency coding and media delivery in heterogeneous environments —

Part 10: MPEG media transport forward error correction (FEC) codes

AMENDMENT 1: Window-based FEC code

Clause 5, Table 1

Replace the first paragraph and Table 1 with the following:

Table 1 specifies the code points for the FEC code algorithms specified in this document. The FEC code algorithms themselves are specified in Clauses 6 through 11.

Table 1 — FEC code algorithms and its code point

Code point	FEC code algorithm
0	Reserved for ISO/IEC use
1	RS code (Clause 6)
2	S_LDPC code (Clause 7)
3	6330 code (subclause 8.2)
4	RaptorQ LA code (subclause 8.3)
5	FireFort-LDGM code (Clause 9)
6	FEC code algorithm in SMPTE 2022-1 (Clause 10)
7	RaptorQ AD code (Clause 11)
8 ~ 255	Reserved for ISO/IEC use

Clause 10

At the end of Clause 10, add a new Clause 11 as follows:

11 Specification for RaptorQ AD code

11.1 General

This clause specifies FEC code point 7, RaptorQ AD code.

The RaptorQ AD code extends the 6330 code to support the adaptive FEC protection in one layer as defined in ISO/IEC 23008-1.

According to the different priorities defined in the DU header, the source symbol can be divided into different classes. All the classified symbols can be protected by one FEC coding matrix and generate source symbols and repair symbols in one stream.

11.2 Encoding method

Step 1: Each block is divided into classes of input symbols according to different priorities defined in the DU header. The numbers of each class are D_1, D_2, D_3, . . . , such that the first input D_1 symbols form the first class, the next D_2 input symbols form the second class, and it can be assumed that the importance of classes decreases with the class index.

Step 2: According to the source symbols, Matrix A can be designed to generate the intermediate symbols. Figure 10 shows an example Matrix A using source symbols with two classes.

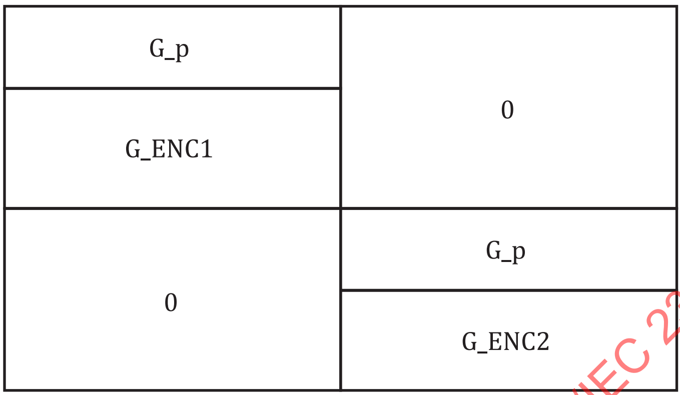

Figure 10 — Structure of Matrix A

The G_p consists of '*G_LDPC*,1', '*I_S*', '*G_LDPC*,2', '*G_HDPC*' and '*I_H*'. G_ENC_1 and G_ENC_2 are an LT-code matrix. According to Matrix A, the intermediate symbols are as follows:

$$\begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} A_1^{-1} & 0 \\ 0 & A_2^{-1} \end{bmatrix} \begin{bmatrix} 0 \\ D_1 \\ 0 \\ D_2 \end{bmatrix}$$

For the classification of more than 2 source symbols, Matrix A can be designed to generate the intermediate symbols as follows:

$$A = \begin{bmatrix} G\_p_1 & 0 & \dots & 0 \\ G\_ENC_1 & 0 & \dots & 0 \\ 0 & G\_p_2 & \dots & 0 \\ 0 & G\_ENC_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & G\_p_l \\ 0 & 0 & \dots & G\_ENC_l \end{bmatrix}$$

$$\begin{bmatrix} C_1 \\ C_2 \\ \dots \\ C_l \end{bmatrix} = \begin{bmatrix} A_1^{-1} & 0 & \dots & 0 \\ 0 & A_2^{-1} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & A_l^{-1} \end{bmatrix} \begin{bmatrix} 0 \\ D_1 \\ 0 \\ D_2 \\ \dots \\ 0 \\ D_l \end{bmatrix}$$

Step 3: As a result of overlapping windows, the probability of success in decoding intermediate symbols of more important source symbols increases, so the probability of recovering the source symbols will increase. The second encoding step can be shown as follows. If the channel is poor, the lines of G_ENC_{12} and G_ENC_{21} can be reduced, the lines of G_ENC_{11} can be increased to increase the redundancy of more important symbols and guarantee the recovery of more important source symbols.

$$\begin{bmatrix} D_1 \\ D_2 \\ R_1 \\ R_2 \end{bmatrix} = \begin{bmatrix} G\_ENC_1 & 0 \\ 0 & G\_ENC_2 \\ G\_ENC_{11} & 0 \\ G\_ENC_{12} & G\_ENC_2 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix}$$

where

R_1 is the repair symbols generated from C_1;

R_2 is the repair symbols generated from C_1 and C_2.

If there are *n* classes among the source symbols, the repair symbols can be generated as follows:

$$\begin{bmatrix} D_1 \\ D_2 \\ \dots \\ D_l \\ R_1 \\ R_2 \\ \dots \\ R_l \end{bmatrix} = \begin{bmatrix} G\_ENC_1 & 0 & \dots & 0 \\ 0 & G\_ENC_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & G\_ENC_l \\ G\_ENC_{11} & 0 & \dots & 0 \\ G\_ENC_{12} & G\_ENC_{21} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ G\_ENC_{1l} & G\_ENC_{2(l-1)} & \dots & G\_ENC_{l1} \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ \dots \\ C_l \end{bmatrix}$$

11.3 Decoding method

The decoding process is performed analogously to the encoding process of a set of received encoding symbols.

Step 1: Because some packets can be lost during the transmission, Matrix A is generated according to the loss of ISI. The lines of Matrix A should be removed based on ISI. Matrix A for decoding is as follows.

The process of generating the intermediate symbols can be generated from the source symbols and the inversion of Matrix A.

$$REV\_A = \begin{bmatrix} G\_p_1 & 0 \\ REV\_G\_ENC_1 & 0 \\ 0 & G\_p_2 \\ 0 & REV\_G\_ENC_2 \\ REV\_G\_ENC_{11} & 0 \\ REV\_G\_ENC_{12} & REV\_G\_ENC_{21} \end{bmatrix}$$

Matrix A can be transformed as follows:

$$REV\_A = \begin{bmatrix} G\_p_1 & 0 \\ REV\_G\_ENC_1 & 0 \\ REV\_G\_ENC_{11} & 0 \\ 0 & G\_p_2 \\ 0 & REV\_G\_ENC_2 \\ REV\_G\_ENC_{12} & REV\_G\_ENC_{21} \end{bmatrix}$$

$$\begin{bmatrix} G\_p_1 & 0 \\ REV\_G\_ENC_1 & 0 \\ REV\_G\_ENC_{11} & 0 \\ 0 & G\_p_2 \\ 0 & REV\_G\_ENC_2 \\ REV\_G\_ENC_{12} & REV\_G\_ENC_{21} \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 0 \\ REV\_D_1 \\ REV\_R_1 \\ 0 \\ REV\_D_2 \\ REV\_R_2 \end{bmatrix}$$

Because the intermediate symbol C_1 is protected by both window 1 and window 2, it will be recovered with high possibility.

After getting the encoded symbols, the intermediate symbols can be recovered based on different situations.

— **Situation 1**: The number of the received symbols is less than the number of source symbols, meanwhile the lines of matrix REV_A_1 are less than the rows. Both C_1 and C_2 cannot be decoded successfully.

— **Situation 2**: The number of the received symbols is less than the number of source symbols, and the lines of matrix REV_A_1 are not less than the rows. C_1 can be decoded successfully but C_2 cannot be decoded successfully.

— **Situation 3**: The number of the received symbols is more than the number of source symbols, and the lines of matrix REV_A_1 and matrix REV_A_2 are both not less than the rows. Both C_1 and C_2 can be decoded successfully using the method defined by IETF RFC 6330.

— **Situation 4**: The number of the received symbols is more than the number of source symbols. The lines of matrix REV_A_1 is not less than the row, but the lines of matrix REV_A_2 is less than the row. C_1 can be decoded successfully but C_2 cannot be decoded successfully.

— **Situation 5**: The number of the received symbols is more than the number of source symbols. The lines of matrix REV_A_1 are less than the rows, but the lines of matrix REV_A_2 are not less than the rows. Both C_1 and C_2 can be decoded successfully using the Gaussian elimination.

For the source symbols which have been divided into more than two classes, they can also be decoded based on different situations.

For more than two priorities, based on the received encoding symbols, Matrix A can be generated as follows:

$$\begin{bmatrix} G\_p_1 & 0 & \dots & 0 \\ REV\_G\_ENC_1 & 0 & \dots & 0 \\ 0 & G\_p_2 & \dots & 0 \\ 0 & REV\_G\_ENC_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & G\_P_t \\ 0 & 0 & \dots & REV\_G\_ENC_l \\ REV\_G\_ENC_{11} & 0 & \dots & 0 \\ REV\_G\_ENC_{12} & REV\_G\_ENC_{21} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ REV\_G\_ENC_{1l} & REV\_G\_ENC_{2(l-1)} & \dots & REV\_G\_ENC_{l1} \end{bmatrix}$$

And it can be transformed as follows:

$$REV\_A = \begin{bmatrix} G\_p_1 & 0 & \dots & 0 \\ REV\_G\_ENC_1 & 0 & \dots & 0 \\ REV\_G\_ENC_{11} & 0 & \dots & 0 \\ 0 & G\_p_2 & \dots & 0 \\ 0 & REV\_G\_ENC_2 \dots & \dots & \dots \\ REV\_G\_ENC_{12} & REV\_G\_ENC_{21} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & G\_p_l \\ 0 & 0 & \dots & REV\_G\_ENC_l \\ REV\_G\_ENC_{1l} & REV\_G\_ENC_{2(l-1)} & \dots & REV\_G\_ENC_{l1} \end{bmatrix}$$