Physical Layer Time Interleaving
for the ATSC 3.0 System

Peter Klenner, Jong-Seob Baek, Nabil Sven Loghin, David Gómez-Barquero, and Woo-Suk Ko

***Abstract* — This paper presents optimized time interleaving which has been adopted for the Advanced Television System Committee (ATSC) 3.0 system as a physical layer tool to mitigate the effects of burst errors. The adopted time interleaver (TI) is very flexible and can have different configurations according to the number of physical layer pipes (PLPs) and service type, i.e., fixed, portable, and mobile. Notably, for single-PLP (S-PLP) mode a sheer convolutional TI (CTI) is used, whereas for the multiple-PLP (M-PLP) mode a hybrid TI (HTI) composed of cell interleaver, twisted block interleaver (BI), and a convolutional delay-line is used. Optionally, the CTI and the HTI can be used in conjunction with extended time interleaving and a cell interleaver (only for HTI) to further improve robustness over long burst error lengths at the expense of latency.**

***Index Terms* — ATSC 3.0, cell interleaving, twisted block interleaver, convolutional delay-line, extended time interleaving, PLP, OFDM.**

I. INTRODUCTION

IT is well-known that channels exhibiting burst error characteristic can greatly degrade the capability of forward error correction (FEC) coding in communication and especially broadcasting systems due to the lack of a return channel [1]. As an efficient means to mitigate burst error effects, various channel interleaving schemes have been widely employed [2],[3]. In particular, as one possible solution in broadcasting systems, a time interleaver (TI) has been considered as integral part of the code design on the physical layer to cope with long burst errors in time domain [4]-[10]. The TI increases time diversity and transmission robustness at the expense of end-to-end latency and zapping times. Alternatively, time interleaving can be also provided at upper layers with some additional coding, with a worse performance compared to the physical layer but with lower memory requirements at the receiver [11],[12].

Within the first generation of Digital Terrestrial Television (DTT) standards , the most powerful and flexible TI is the one adopted in ISDB-T (Integrated Services Digital Broadcasting –

Terrestrial) [4], based on a sheer convolutional interleaver of up to one second of time interleaving depth which can be independently configured for fixed and mobile (one-seg) services [5]. The DTT standard DVB-T employs a convolutional byte interleaver to randomize a burst error between outer Reed-Solomon (RS) code and inner convolutional code, with a limited time interleaving duration in the order of few ms (up to two OFDM symbols) [6]. In the ATSC 1.0 (A/53) DTT standard, a convolutional byte interleaver and trellis code interleaver were adopted providing time interleaving depths between 2 ms and 4 ms [7].

In the second generation standards for terrestrial broadcast (DVB-T2) [8],[9] and next generation handheld (DVB-NGH) [10], a TI is defined on a physical layer pipe (PLP) basis. PLPs add flexibility compared to DVB-T and ATSC 1.0 by allowing different coding, modulation, and TI parameters (or interleaving depths) [13]. The TI provides a versatile trade-off in terms of end-to-end latency, zapping time, time diversity, and power saving [14]. For DVB-T2, it is a sheer block interleaver that operates with cells (constellation symbols) [16]. The maximum time deinterleaver (TDI) memory size is $2^{19}+2^{15}$ cells. On the other hand, DVB-NGH uses a cascade of a BI and a convolutional interleaver (CI), where the CI is followed by the BI. Since NGH is aimed at portable devices, the memory size here is 2^{18} cells. While the BI is used for intra-frame interleaving (i.e., within a frame), the CI is used for inter-frame interleaving (i.e., across multiple frames). Also in NGH was introduced the concept of adaptive cell quantization to afford larger time interleaving depths with low order QPSK and 16-QAM constellations without increasing the TDI memory [10].

In contrast to DVB-T2/NGH, the physical layer TI of ATSC 3.0 is differently configured according to the number of PLPs in a subframe[1]. For the single-PLP (S-PLP) mode, a conventional convolutional TI (CTI) is used. For the multiple-PLP (M-PLP) mode, a hybrid TI (HTI) is used, which consists of a cascade of cell interleaver, twisted BI and convolutional delay-line based on First-In-First-Out (FIFO) registers. Compared to row-column block interleaving, row-twisted BI offers better minimum span properties and a slightly larger time interleaving depth (cf. Section VI).The purpose of twisted BI and convolutional delay-line is intra-subframe interleaving and

Manuscript submitted on 7th August, 2015; revised on 5th October, 2015.

P. Klenner is with the Panasonic AVC Development Centre in 63225 Langen, Germany (e-mail: peter.klenner@eu.panasonic.com).

J. Baek and W. Ko are with the Advanced Standard Research Center, CTO, LGE, Seoul, Korea (email: {jongseob.baek, woosuk.ko}@lge.com.

Nabil Sven Loghin is with the Sony European Technology Center in 70327 Stuttgart, Germany (e-mail: nabil@sony.de).

D. Gómez-Barquero is with the Institute of Telecommunications and Multimedia Applications (iTEAM), Universitat Politecnica de Valencia, Valencia 46022, Spain (e-mail: dagobar@iteam.upv.es).

[1] In ATSC 3.0, a frame can be configured with one or more subframes [15]. A subframe is a set of OFDM symbols with the same waveform attributes, such as FFT size, guard interval, and SISO (Single Input Single Output) or MIMO (Multiple Input Multiple Output) operation mode. As a result, the role of the hybrid TI is defined based on the subframe concept.

inter-subframe interleaving, respectively. The size of the TI memory is 2^{19} cells, except for QPSK modulation with time and/or frequency division multiplexing (TDM/FDM), which allows up to 2^{20} cells by using so-called extended time interleaving.

The HTI of ATSC 3.0 allows efficient single-memory usage at the receiver side, which is not easily achieved with the HTI in NGH and which was found to be one of its main benefits during the ATSC 3.0 standardization. Here, single-memory usage is the succinct description of an addressing method by which an input cell is written at the same address from which an output cell just has been read. By this approach, memory duplication for the purpose of both interleaving and deinterleaving can be avoided. Furthermore, the HTI of ATSC 3.0 exhibits a slightly larger time interleaving depth and improved minimum span properties compared to NGH.

The rest of the paper is structured as follows. Section II and Section III describe the TI configuration for S-PLP mode and M-PLP mode, respectively. Section IV briefly summarizes the TI features related to MIMO transmission in ATSC 3.0. Section V covers two important TI implementation aspects, in particular the implementation for the hybrid time deinterleaver at the receiver side based on a single memory and the handling of variable bit rate (VBR) services. Section VI analyzes the TI depth, zapping time, latency, and minimum span of different time interleaving schemes. Section VII presents some illustrative performance simulation results. Finally, the paper concludes with Section VIII.

II. TI CONFIGURATION FOR S-PLP MODE

A. Normal Convolutional Time Interleaving for S-PLP mode

Fig. 1 shows the TI configuration for S-PLP mode in ATSC 3.0 realized by a classical Forney convolutional time interleaver (CTI) [2]. The input to the CTI is a sequence of data cells g_q, i.e., modulated QAM symbols, with q being the time index. The CTI consists of N_{rows} delay lines, with the k-th line having k delay elements, $k = 0, 1, …, N_{rows}-1$. Thus, the number of columns, i.e., the maximum number of delay elements, in Fig. 1 is $N_{rows} - 1$. Input and output are controlled by two commutators, cyclically switching downwards after one cell is written in or read out, respectively. At each time, they will be located in the same position k. The total number of delay elements is given by $T = N_{rows} \cdot (N_{rows} - 1) / 2$, which is also known as triangular number. If the input commutator is located at position k, an input symbol g_q will be written to this delay line. The delay elements from this line will shift their memory content to the right neighboring delay element, respectively, and the content from the right-most delay element will be output via the output commutator. Next, the input symbol g_q can be written to the left-most delay element of this line. Both commutators will then move cyclically to the next line ($k+1$), i.e., considering "modulo N_{rows} operation". In case that the commutators have been located at the first line, $k = 0$, the input symbol is directly read out via the output commutator.

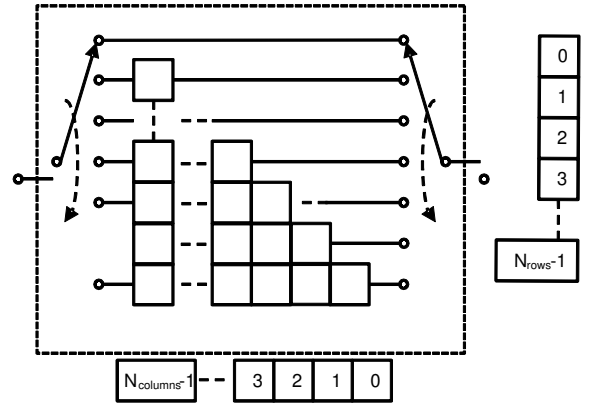ATSC 3.0 defines four different CTI structures for S-PLP,

Fig. 1. TI configuration for S-PLP mode in ATSC 3.0

given by the number of rows $N_{rows} \in \{512, 724, 887, 1024\}$. The maximum TDI memory is thus $1024/(1024-1)/2 = 523776 < 2^{19}$ cells. For a 6 MHz channel, this represents time interleaving depths of approximately 50, 100, 150, and 200 ms, respectively. No time interleaving option may also be selected. In the case of no time interleaving, cells shall be output in the same order and without any delay (corresponding to $N_{rows} = 1$).

The benefits of the CTI over a sheer BI as used in DVB-T2 are twofold. First, it is possible to achieve the same time interleaving depth with half the TDI memory. Secondly, it provides a shorter latency by about 33% [21]. Furthermore, it is worth mentioning that the CTI is not based on any fixed boundaries as occurred with block interleaving. For block interleavers, a multiple of a certain amount of cells can be transmitted within one ATSC subframe, while the remaining subframe cells have to be filled with dummy cells. In contrast, the CTI outputs cells continuously and insertion of dummy cells to achieve an integer number of FEC frames per subframe is not required, thus reducing the overhead. However, this benefit turns into a drawback of the CTI, when multiple PLPs shall be multiplexed into one subframe, especially with varying cell rates: it is not straightforward to shutdown a CTI at the end of one service transmission and to start it up again in another subframe. For that reason, the HTI was chosen for the M-PLP mode.

B. L1 Signaling parameters for the CTI

The number of rows, N_{rows}, is signaled via the parameter `L1D_CTI_depth`. For initial acquisition, the receiver faces two problems for CTI: Firstly, the start row of the commutator at the beginning of the frame is in general not the uppermost row (as is commonly enforced by other systems, which use a CTI). This row is signaled by `L1D_CTI_start_row`. Secondly, the receiver needs to know the start of a FEC codeword, since FEC coding is based on block codes in ATSC 3.0, namely BCH and LDPC codes [20]. This is solved with L1 signaling as follows: the parameter `L1D_CTI_fecframe_start` signals the index of the first cell of a codeword inside an ATSC subframe. It should be noted that the first data cell of one subframe, which enters the CTDI shall be indexed by 0, the second one by 1, the third one by 2, and so on. That is, the `L1D_CTI_fecframe_start` value should not be interpreted as the cell address inside a subframe,

which denotes the data cell index before frequency deinterleaving. In order to signal only codewords, which will appear completely in the current or upcoming subframe, the following condition needs to be fulfilled:

$$\texttt{L1D\_CTI\_fecframe\_start} \geq R_s \cdot (N_{rows} + 1), \qquad (1)$$

where $R_s = \text{mod}(\texttt{L1D\_CTI\_fecframe\_start}+\texttt{L1D\_CTI\_start\_row},$ $N_{rows})$ is the row index for the $\texttt{L1D\_CTI\_fecframe\_start}$-th cell inside the subframe. If the start position of a FEC codeword does not fulfil the condition above, that is if cells belonging to the same FEC codeword appear also in previously transmitted subframes due to the delaying nature of the convolutional interleaver, the next FEC codeword start position shall be checked and so forth.

A one bit flag, $\texttt{L1D\_plp\_TI\_extended\_interleaving}$, indicates whether normal CTI or extended CTI mode is used.

- If $\texttt{L1D\_plp\_TI\_extended\_interleaving}=1$, it is applied only to QPSK on TDM (not LDM). It provides the interleaving-depth increase corresponding to 2^{20} TI memory cells. At the receiver side, adaptive cell quantization is recommended [10] to adhere to the TDI memory constraint.
- If $\texttt{L1D\_plp\_TI\_extended\_interleaving}=0$, extended CTI not available.

III. TI CONFIGURATION FOR M-PLP MODE

The general configuration of TI for M-PLP mode in ATSC 3.0 is depicted in Fig. 2. It consists of a cascade of cell interleaver, row-twisted BI, and convolutional delay-line.

Fig. 2. TI configuration for M-PLP mode in ATSC 3.0.

Fig. 3 shows a high-level view on the core HTI structure consisting of row-twisted BI and convolutional delay-line. The twisted BI, aimed at intra-subframe interleaving, is always available. Inter-subframe interleaving is accomplished via a convolutional delay-line consisting of FIFO registers. If inter-subframe interleaving scheme is not desired, the convolutional delay-line is turned off. The detailed functionality of twisted BI and convolutional delay-line are described in the following subsections.

A. Cell Interleaver

The purpose of the cell interleaver that operates on cells is to randomize residual burst errors within an LDPC codeword. To this end, it permutes a FEC-block according to a pseudo random sequence $L_r(q)$ [8]-[10] as follows

$$d_{r,q} = g_{r,L_r(q)}, \qquad (2)$$

where $d_{r,q}$ denotes the cell interleaver output and $g_{r,q}$ its input, and the parameters r and q are defined as $0 \leq q < N_{cells}$ and $0 \leq r < N_{FEC\_TI}(n,s)$, with N_{cells} being the FEC-block size, and $N_{FEC\_TI}(n,s)$ the number of FEC-blocks belonging to TI-block n in an interleaving frame s. One TI-block
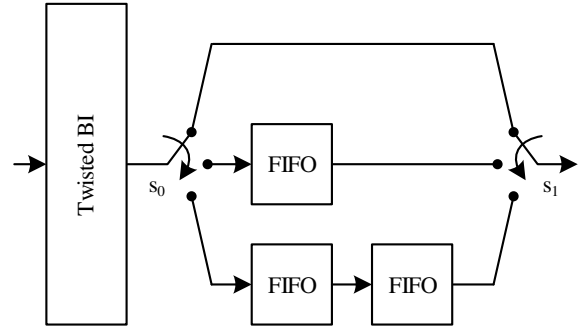

Fig. 3. The twisted BI and convolutional delay-line of the HTI structure.

corresponds to one self-contained time interleaver operation, and is the basis of operation for the cell interleaver, twisted BI, and convolutional delay-line. The cell interleaver is reset with the beginning of each new TI-block.

At the transmitter side, a dedicated buffer is being written to linearly with the cells of the r-th FEC-block $g_{r,q}$. Afterwards the buffer is (pseudo-)randomly read according to $L_r(q)$ in order to output the cell interleaved FEC-block $d_{r,q}$. The benefits of this approach emerge at the receiver side:

- The same permutation function as at the transmitter can be reused, i.e., there is no need to determine the inverse permutation.
- A dedicated cell deinterleaving buffer is not needed. It is possible to read cells linearly from the TDI and to effect cell deinterleaving on the fly.

In contrast to DVB-T2/NGH [8]-[10], the cell interleaving can be optionally switched on or off by L1 signaling, i.e., $\texttt{L1D\_plp\_HTI\_cell\_interleaver}$.

B. Row-Twisted Block Interleaver

For row-twisted BI, the input FEC-blocks are written column-wise and linearly to a given memory and read diagonally. Fig. 4a and Fig. 4b illustrate the column-wise writing and diagonal-wise reading operations of the twisted BI, respectively. Fig. 4c shows the output after the diagonal-wise reading. Here, it is noted that the same interleaving operation can be equivalently expressed as a row-twisted block interleaver since its functionality can be also achieved by writing FEC-blocks column-wise, then applying an increasing circular shift of each row to the left, followed by reading the block interleaver column by column.

Based on a single block of linear addressable RAM and after column-wise writing, the diagonal-wise reading can be performed by calculating the position for cells with a coordinate $(R_i, C_{i,j})$ (for $i = 0, \cdots, N_{cells} N_c - 1$ and $j = 0, 1, \cdots$) as follows:

$$\begin{aligned} R_i &= \text{mod}(i, N_{cells}), \\ T_{i,j} &= \text{mod}(j \cdot R_i, N_c), \\ C_{i,j} &= \text{mod}\left(T_{i,j} + \left\lfloor \frac{i}{N_{cells}} \right\rfloor, N_c\right), \\ a_{i,j} &= N_{cells} \cdot C_{i,j} + R_i, \end{aligned} \qquad (3)$$

where R_i and $C_{i,j}$ indicate the row and column indices, respectively, and $T_{i,j}$ is a twisting parameter. $a_{i,j}$ denotes an

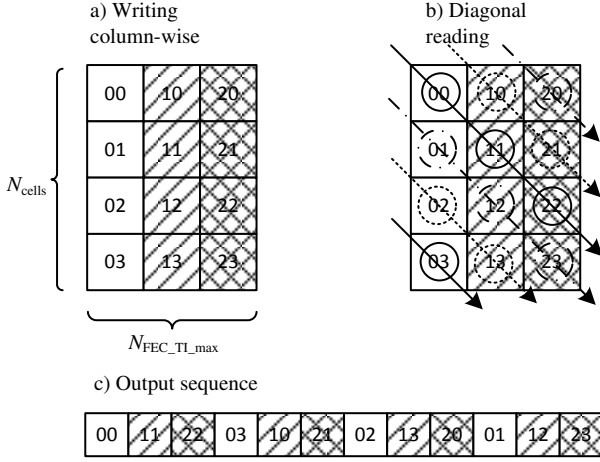a) Writing column-wise   b) Diagonal reading

c) Output sequence

Fig. 4. An example of the twisted BI operation; a) FEC-blocks are written column-wise. b) Cells are read diagonal-wise. c) The resulting output cell sequence after diagonal-wise reading

actual memory address to read cells in the linear RAM. Index j in (3) is meant for the twisted BI operation with single-memory usage. For example, the memory is initially filled with the first TI-block ($j = 0$). Then a cell is read from the first TI-block while a cell from the second TI-block ($j = 1$) is written at the exact same position and so forth. $\lfloor x \rfloor$ and mod denote floor and modulo operation, respectively.

The number of rows of the twisted BI, N_{cells}, is equal to the number of cells in a FEC block while the number of columns N_c can be set maximally to $N_{FEC\_TI\_MAX}$, where $N_{FEC\_TI\_MAX}$ denotes the maximum number of FEC-blocks per TI-block. For variable bit rate (VBR) services, the actual number of FEC-blocks per TI-block can be smaller than $N_{FEC\_TI\_MAX}$ and can also vary between TI-blocks. The details of handling VBR are described in Section V-B.

C. Convolutional Delay-Line

The detailed diagram of the convolutional delay-line is provided in Fig. 5. It consists of two commutators or switches s_0 and s_1 and N_{IU} branches connected to FIFO registers. The switches move cyclically from one branch to the next whenever a certain number of cells have been read from the block interleaver. This number is identical to the number of FEC-blocks currently contained in the block interleaver. The VBR case is described in Section V-B.

In order to determine the sizes of the FIFO registers, the following intermediate parameters are defined for the HTI:

$$L_{IU} = \left\lfloor \frac{N_{cells}}{N_{IU}} \right\rfloor,$$
$$N_{large} = \text{mod}\,(N_{cells}, N_{IU})\,, \quad (4)$$
$$N_{small} = N_{IU} - N_{large}.$$

The rationale underlying these parameters is to distribute pieces of a FEC-block to the FIFO registers. These pieces can contain different numbers of cells in cases where a FEC-block does not contain an integer multiple of the number of branches, N_{IU}. Ultimately, equations (4) separate a FEC-block into N_{large} interleaving units containing $L_{IU}+1$ cells and N_{small} interleaving units containing L_{IU} cells.
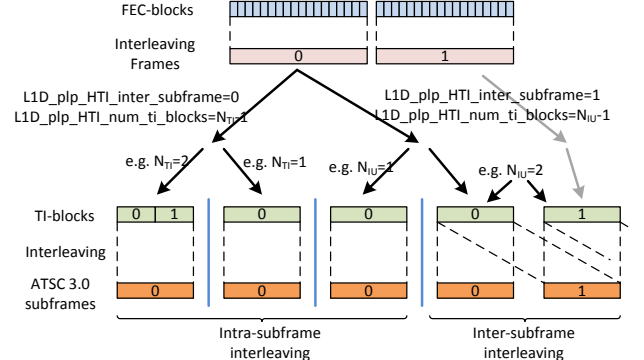

Fig. 6. Time interleaving for L1D_plp_HTI_inter_subframe = 0 and 1, and for L1D_plp_HTI_num_ti_blocks = 1 and 2

The sizes of the FIFO registers can now be derived in a straightforward manner as follows

$$M_{i,j} = \begin{cases} (L_{IU} + 1) \cdot N_{FEC\_TI\_MAX}, & i < N_{large}, j \le i, \\ L_{IU} \cdot N_{FEC\_TI\_MAX}, & N_{large} \le i < N_{IU}, j \le i. \end{cases} \quad (5)$$

The total number of cells, M_{cells}, contained in the convolutional delay-line is given by

$$M_{cells} = \sum_{i=0}^{N_{large}-1}(L_{IU} + 1) \cdot N_{FEC\_TI\_MAX} \cdot i$$
$$+ \sum_{i=N_{large}}^{N_{IU}-1} L_{IU} \cdot N_{FEC\_TI\_MAX} \cdot i. \quad (6)$$

D. Time Interleaving Modes

There are two basic options of time interleaving for each PLP, intra-subframe interleaving and inter-subframe interleaving. The FEC-blocks output from the cell interleaver (if used) or the constellation mapper are grouped into so-called interleaving frames (IFs). The number of FEC-blocks contained in an IF is signalled by L1D_plp_HTI_num_fec_blocks and may vary from a minimum value of 1 to a maximum value of L1D_plp_HTI_num_fec_blocks_max. Each IF can contain a variable number of FEC-blocks. The number of FEC-blocks, N_{blocks}, contained in the current IF is signalled by L1D_HTI_num_fec_blocks. N_{blocks} may vary from a minimum value of 1 to a maximum value of $N_{blocks\_if\_max}$. $N_{blocks\_if\_max}$ is signaled as L1D_HTI_num_fec_blocks_max. Each IF is either mapped directly onto one subframe or spread over several (N_{IU}) subframes. Each IF is also divided into one or multiple TI-blocks (N_{TI}). The TI-blocks within an IF can contain a slightly different number of FEC blocks per TI-block.

- Intra-subframe interleaving option: Each IF is mapped directly to one subframe and the IF is composed of one or several TI-blocks (N_{TI}) as shown in Fig. 6. Each of the TI-blocks may be deinterleaved and decoded immediately after its complete reception in the receiver. This allows the maximum bit-rate for the PLP to be increased. This option is signalled as L1D_plp_HTI_inter_subframe = 0. For this option, the number of TI-blocks per IF is set to N_{TI} = L1D_plp_HTI_num_ti_blocks.

- Inter-subframe interleaving option: Each IF contains one TI-block and is mapped to one or more than one subframes (N_{IU}). Fig. 6 shows an example in which one IF is mapped onto two subframes. This gives greater time diversity for low data rate services. This option is signalled by
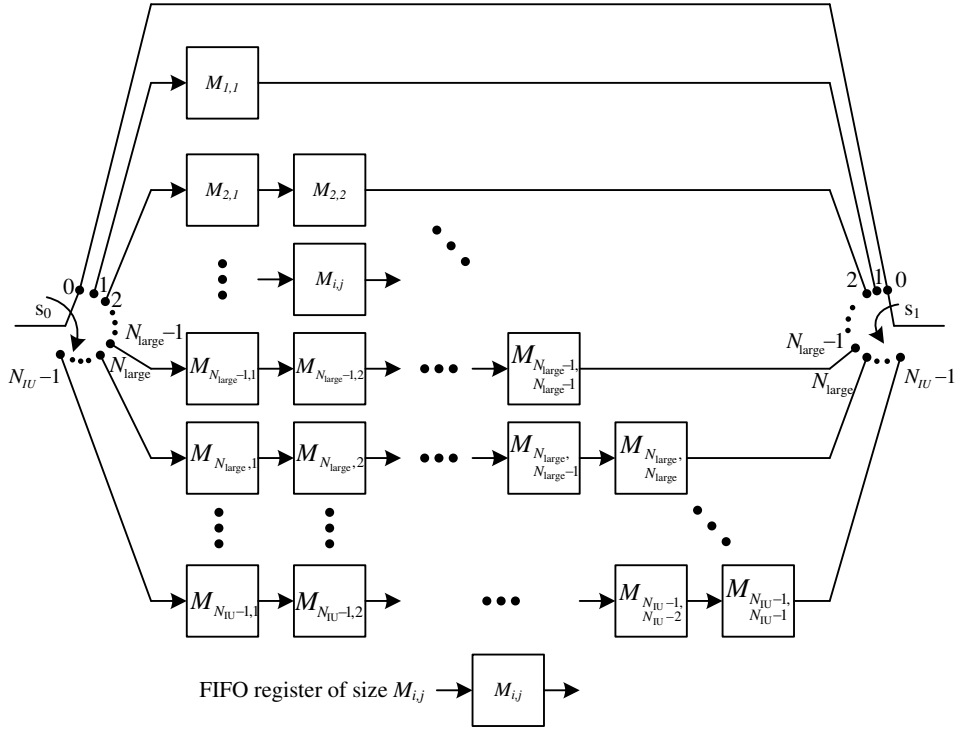
Fig. 5. The detailed diagram of the convolutional delay-line

L1D_plp_HTI_inter_subframe = 1. For this option, N_{IU} = L1D_plp_HTI_num_ti_blocks.

E. L1-Signaling parameters for the HTI

The following list of parameters comprises all L1 signalling fields related to the TI for M-PLP mode in ATSC 3.0:

- L1D_plp_HTI_inter_subframe: A 1-bit field to determine the hybrid time interleaving mode. When L1D_plp_HTI_inter_subframe=0, inter-subframe interleaving is not used, only intra-subframe interleaving is used. When L1D_TI_inter_subframe=1, inter-subframe interleaving is used with one TI-block per interleaving frame spread over multiple ATSC 3.0 subframes.
- L1D_plp_HTI_num_ti_blocks: A 4-bit field to determine the number of TI-blocks per interleaving frame, N_{TI}, when L1D_plp_HTI_inter_subframe=0 and the number of frames, N_{IU}, over which cells from one TI-block are carried when L1D_plp_HTI_inter_subframe=1. In case its value is set to 0, each IF contains one TI block and is mapped directly to one subframe, irrespective of the value of L1D_plp_HTI_inter_subframe (cf. middle of Fig. 6).
- L1D_plp_HTI_num_fec_blocks_max: A 12-bit field to determine the maximum number of FEC blocks per interleaving frame for the current PLP.
- L1D_plp_HTI_num_fec_blocks: A 12-bit field to indicate the number of FEC blocks contained in the current interleaving frame for the current PLP.
- L1D_plp_HTI_cell_interleaver: A 1-bit field to determine whether the cell interleaver is used or not.

F. Extended Time Interleaving for S-PLP mode

Typically, a time deinterleaver (TDI) operates on bit tuples representing the in-phase component (I), quadrature-phase (Q), and channel state information (CSI). For a given TDI memory size in bits, the number of cells held in the TDI depends on the quantization of I, Q, and CSI. For low-order constellations, the I/Q components can be quantized with less resolution than for high-order constellations without significant impact on performance. Consequently, quantizing cells differently depending on the robustness of the constellation can provide a longer TI duration for a low-order constellation without increasing the physical memory at the receiver side. This is known as adaptive quantization, and it is a receiver implementation issue.

DVB-NGH defines a TDI memory size for QPSK and 16-QAM of twice the size than 64-QAM and 256-QAM [10], and operates with two-cell memory units (so-called pairwise interleaving), such that those two cells always remain adjacent cells. In contrast to DVB-NGH, ATSC 3.0 allows extended time interleaving only for QPSK in conjunction with doubling the TI memory size up to 2^{20} cells. Extended time interleaving can only be used with TDM and/or FDM, but not with Layer Division Multiplexing (LDM) [17]. For S-PLP, it is introduced by doubling the number of rows of the CTI, and for M-PLP by increasing the number of FEC-blocks per TI-Block (up to roughly twice the number of FEC-blocks per TI-block).

IV. TIME INTERLEAVING FOR MIMO IN ATSC 3.0

The MIMO transmission chain of ATSC 3.0 re-uses as many blocks as possible from the single input single output (SISO) antenna baseline, including FEC codes, bit interleavers, constellations and frequency and time interleavers [18].
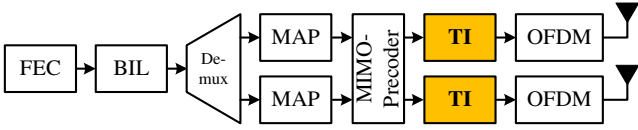
Fig 7. 2x2 MIMO-transmission chain in ATSC 3.0

Fig. 7 shows the 2x2 MIMO transmission chain of ATSC 3.0, where it can be seen that the TI is carried out after the generation of two MIMO streams. Hence, there are two parallel TIs. MIMO uses the same TI as for SISO, with the restriction that the time interleaving applied to both MIMO streams should be identical. The TDI memory requirement applies for each TI of each antenna, and thus MIMO requires twice the memory as for SISO. In Fig. 7, it can be seen that the TI is also placed after the so-called MIMO precoder, which provides the transmit signal with spatial diversity [19]. This enables an iterative receiver design, where the time deinterleaver can operate independently in front of an efficient turbo-loop between the MIMO demapper and the FEC decoder.

V. TIME INTERLEAVING IMPLEMENTATION

A. Single memory usage for the HTI

In this section, implementation guidelines for the time interleaving based on a single memory (at the transmitter side) are provided. The focus lies on the hybrid time interleaving, since the implementation of convolutional interleaving (cf. Fig. 1) is well-known [2]. Hybrid time interleaving using a single-memory can be achieved through the separate operation of a twisted block interleaver and processing the convolutional delay-line.

The T2-implementation guidelines [6] describe an addressing scheme which allows a memory efficient implementation of row-column block deinterleaving. Whereas a naïve approach would require two separate deinterleavers operated alternatively, it is possible with this addressing scheme, to use a single piece of linear RAM to accomplish time deinterleaving.

$$k_0 = 1$$
$$k_j = mod\big(k_{j-1}N_c, N_cN_{cells}\big) + \left\lfloor \frac{k_{j-1}}{N_{cells}} \right\rfloor \qquad (7)$$
$$a_{i,j} = \begin{cases} mod\big(i \cdot k_j, N_cN_{cells} - 1\big) & \text{if } i < N_cN_{cells} - 1 \\ N_cN_{cells} - 1 & \text{if } i = N_cN_{cells} - 1 \end{cases}$$

Fortunately, the formulation for the twisted block deinterleaving can be readily defined by slightly changing the second-line equation of (6) as

$$k_j = mod\big(k_{j-1}N_{cells}, N_cN_{cells}\big) + \left\lfloor \frac{k_{j-1}}{N_c} \right\rfloor .$$

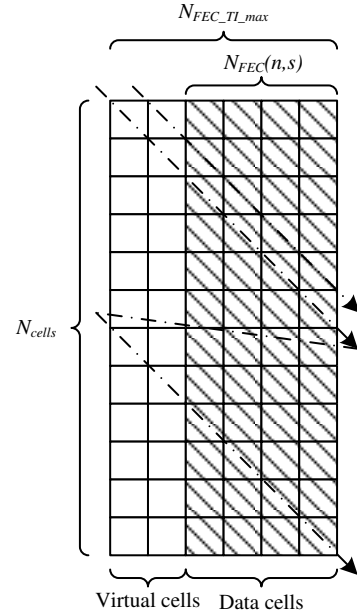The subsequent convolutional delay-line is operated as described in Section III-B.

Fig 8. Block interleaver containing virtual cells and data cells. Virtual cells are introduced to handle a variable number of FEC-blocks per interleaving frame.

B. Handling Variable Cell Rate

For variable bit rate transmission, the number of FEC-blocks can change between TI-blocks. For this case the concept of a virtual cell was introduced in ATSC 3.0. It is of note that virtual cells are not necessary in the case of constant cell rate, when the number of FEC-blocks in all TI-blocks is constant.

Fig. 8 illustrates twisted block interleaving in case of VBR with an example, in which two columns out of six are filled with virtual cells. During the diagonal-wise reading process for intra-subframe interleaving (in isolation of the convolutional delay-line) virtual cells along the diagonals are skipped when cells are output.

For inter-subframe interleaving, using the convolutional delay-line, virtual cells serve as padding. Virtual cells are never passed to the time interleaver output; they are however passed to the FIFO-registers of the delay-line.

Fig. 9 shows an example of the aforementioned process. The HTI structure in Fig. 9a uses a twisted blockinterleaver with N_{cells}=4 rows and a maximum of $N_{FEC\_TI\_MAX}$=3 columns. The switches s_0 and s_1 switch to next branch after having passed $N_{FEC\_TI\_MAX}$=3 cells. The number of subframes, over which the convolutional delay-line spans, is chosen to N_{IU}=2; hence, there is a single delay-line branch with a FIFO-register having space for six cells. In Fig. 9b three consecutive TI-blocks are shown with three, one, and two FEC-blocks. Please note, that in TI-block 1 and 2, virtual cells fill the first empty columns of the respective TI-blocks. Fig. 9c shows the output from the HTI, and Fig. 9d the state of the FIFO-register after the respective TI-block has been passed into the HTI.
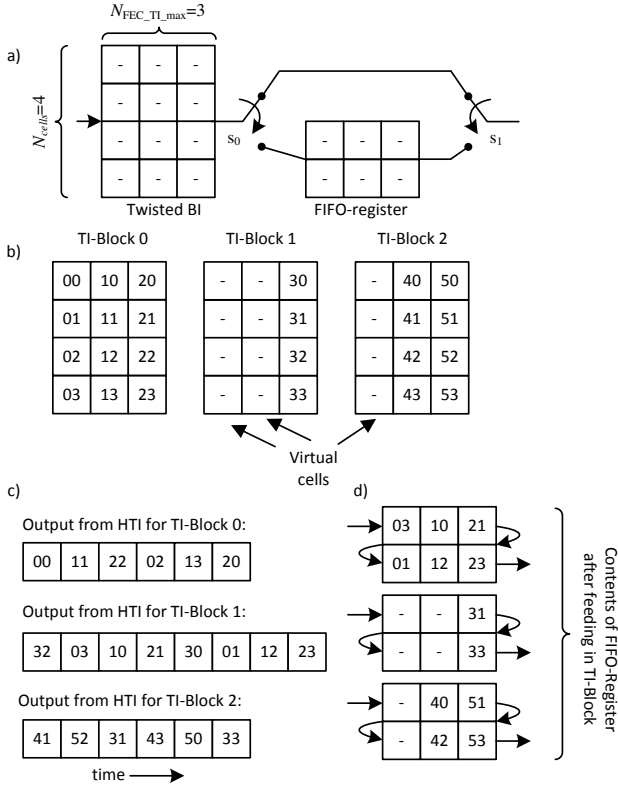
Fig 9. Example for handling VBR in the case of inter-subframe interleaving: a) HTI-structure for N_{IU}=2, N_{cells}=4, $N_{FEC\_TI\_MAX}$=3, b) A sequence of TI-blocks with variable number of codewords, c) Output generated from the HTI upon feeding in each TI-block, d) State of the FIFO-Register after each TI-block

After TI-block 0 has been passed to HTI, the first cells being read from the Twisted BI are 00, 11, and 22. These are directed to the HTI-output, and the switches s_0 and s_1 move to the next branch. The next cells read from the BI are 03, 10, and 21; they are placed into the FIFO-register. The next chunk of cells (02, 13, and 20) is passed to the HTI-output and the remaining chunk (01, 12, and 23) again to the FIFO-register.

When TI-block 1 is processed, the first chunk of cells are two virtual cells followed by the cell with index 32; since virtual cells are skipped at the HTI output only cell 32 appears at the HTI output. The next chunk of cells from the twisted BI are two virtual cells followed by cell 31, which are passed to the FIFO-register. TI-block 2 is processed similarly.

VI. ANALYSIS OF TIME INTERLEAVING-DEPTH, LATENCY, ZAPPING TIME, AND MINIMUM SPAN

A. Time Interleaving-Depth

In this paper, the time interleaving-depth T_D is defined as the time span between the first and the last cells of a time interleaved FEC-block. In general, it is a function of the FEC code-rate, R_C, the cardinality of the modulation alphabet, η, the information rate, R_b, and the size of the TDI memory. It is an indicator of the coherence time of the channel the system can cope with and reliably counter the effects of burst-errors. For five types of interleaver, the time interleaving depths are

summarized here:

- $(M \times N)$ - row-column block interleaver

$$T_{D,RC-BI} = \frac{R_C \cdot \log_2(\eta) \cdot TDI}{R_b} \left(1 + \frac{1 - N\_FEC\_TI}{TDI}\right) \qquad (8a)$$

- Row-twisted block interleaver

$$T_{D,RT-BI} = \frac{R_C \cdot \log_2(\eta) \cdot TDI}{R_b} \qquad (8b)$$

- Classical convolutional interleaver with N_{rows} branches

$$T_{D,CC} = \frac{R_C \cdot \log_2(\eta) \cdot TDI}{R_b} \cdot \frac{2N_{rows}}{N_{rows} - 1} \qquad (8c)$$

- NGH-hybrid interleaver

$$T_{D,NGH} = \frac{R_C \cdot \log_2(\eta) \cdot TDI}{R_b} \cdot \frac{2\left(N_{IU} + \frac{1 - N\_FEC\_TI}{N_{cells} \cdot N\_FEC\_TI}\right)}{N_{IU} + 1} \qquad (8d)$$

- ATSC 3.0-hybrid time interleaver

$$T_{D,ATSC} = \frac{R_C \cdot \log_2(\eta) \cdot TDI}{R_b} \cdot \frac{2N_{IU}}{N_{IU} + 1} \qquad (8e)$$

In Fig. 10, the time interleaving depths of convolutional interleaving, row-twisted BI, and both HTIs of ATSC 3.0 and NGH are compared, assuming that all TIs have storage for 2^{19} cells available. The BI provides the smallest time interleaving depth, while the CTI provides twice the largest time interleaving depth and thus the largest. The two HTIs are providing TI depths in between these two extremes. In theory, with the same TDI memory the HTI in ATSC 3.0 can afford a slightly larger TI depth than in NGH due to the row-twisted block-interleaver (cf. (8d) and (8e)); in practice, both TI depths can be considered the same.

B. Latency

In this paper, latency is defined as the elapsed time between the first cell of a FEC-block entering the time interleaver and the same cell output from the time deinterleaver. For simplicity's sake, other contributing factors to the delay (namely, OFDM and the framing structure) are not considered.

In this sense, the resulting values represent a lower bound on the latency (and also zapping time, later) observed in an actual implementation.

- Row-twisted and row-column block interleaver

$$T_{L,BI} = 2 \cdot \frac{R_C \cdot \log_2(\eta) \cdot TDI}{R_b} \qquad (9a)$$

- Classical convolutional interleaver with N_{rows} branches

$$T_{L,CC} = \frac{R_C \cdot \log_2(\eta)}{R_b} \cdot (N_{rows} - 1) \cdot N_{rows} = 2 \cdot \frac{R_C \cdot \log_2(\eta) \cdot TDI}{R_b} \qquad (9b)$$

- Hybrid time interleaver

$$T_{L,HTI} = \frac{R_C \cdot \log_2(\eta)}{R_b} \cdot (N_{IU} + 1) \cdot N_{cells} \cdot N_{FEC\_TI}$$
$$= 2 \cdot \frac{R_C \cdot \log_2(\eta) \cdot TDI}{R_b} \qquad (9c)$$

It is apparent from (9a)-(9c) that in all cases the latency for all TIs is identical if the TDI-memory has the same size.
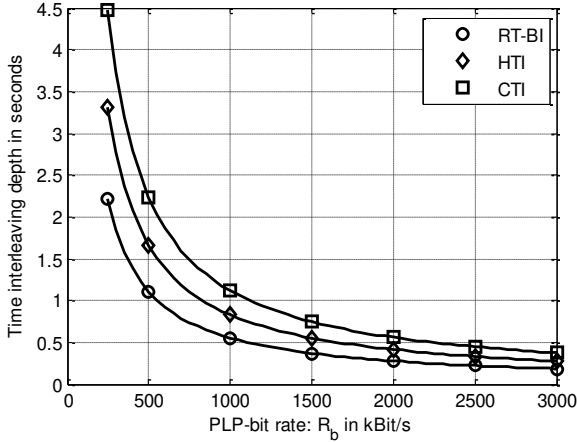
Fig 10. Time interleaving depths for QPSK, CR=8/15, N_{ldpc}=64800, N_{IU}=4, and N_{rows}=1024

Fig 11. Zapping times depending on PLP bit-rate for QPSK, CR=8/15, N_{ldpc}=64800, N_{IU}={2,4}, and N_{rows}=1024. TDI-memory 2^{19} in all cases.

C. Zapping Time

Zapping time, here, is understood as the time that it takes for a *time interleaved* cell to pass through the time deinterleaver and appear at its output.

- Row-twisted and row-column block interleaver

$$T_{Z,BI} = \frac{R_C \cdot \log_2(\eta) \cdot TDI}{R_b} \qquad (10a)$$

- Classical convolutional interleaver with N_{rows} branches

$$T_{Z,CC} = 2 \cdot \frac{R_C \cdot \log_2(\eta) \cdot TDI}{R_b} \qquad (10b)$$

- Hybrid time interleaver

$$T_{Z,HTI} = \frac{R_C \cdot \log_2(\eta)}{R_b} \cdot N_{IU} \cdot N_{cells} \cdot N_{FEC\_TI} \qquad (10c)$$

Fig. 11 compares the zapping times for CTI, BI, and HTI. The resemblance to the time interleaving depth in Fig. 10 is obvious. With the same TDI-memory, block deinterleaving reduces the zapping time by half compared to CTI. The HTI yields zapping times in between BI and CTI depending on the number of interleaving units.

Comparing latency (9) and zapping time (10), the following is noted. For block interleaving, the latency is twice the zapping time since both interleaver and deinterleaver need to be filled completely before reading is possible. For CTI, latency and zapping time are identical, since roughly speaking for the assumption of an idealized system the deinterleaving starts immediately with the interleaving. For HTI, the zapping time, similar to the time interleaving depth, comes to lie between these two extremes

D. Minimum Span

The minimum span of an interleaver π is defined according to [2] as

$$S_{min} = \min_{i \neq j, \forall i}\{|i - j| + |\pi(i) - \pi(j)|\}. \qquad (11)$$

It is a measure for the separation of cells before and after the cell permutation. In Table I, an example is shown for block interleaving with N_{cells}=8100 rows and N_FEC_TI columns. The resulting values in the left-most table columns were found
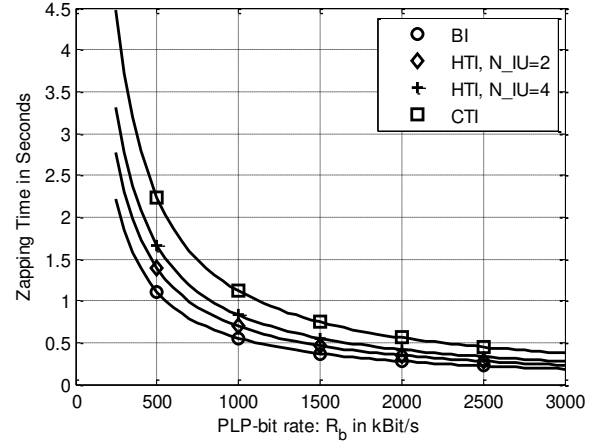
by a computer search. They illustrate that in general the row-twisted BI exhibits larger minimum span properties than row-column interleaving.

TABLE I
Minimum Span for BIs with N_{cells}=8100 rows and N_FEC_TI columns

N_FEC_TI	S_{min}	
	Row-Column BI	Row-Twist BI
3	4	6
6	7	12
9	10	18
12	13	24

VII. PERFORMANCE RESULTS

Computer simulations were performed for a TU6-channel to illustrate the performance of the hybrid time interleaver in ATSC 3.0 depending on the Doppler spread and different number of frames (N_{IU}=1, 15) over which codewords are spread. In order to simplify the simulation set-up, the HTI is slightly misused here in that a single PLP makes use of the complete TI memory. In practice, however, the respective PLP would need to share the TI memory with its peers constituting the same service.

TABLE II
Simulation Parameters

Coderate	9/15
N_{ldpc} (codeword length)	64800
Constellation	64-QAM (non-uniform)
N_{FFT}	8192
N_a (# of active carriers)	6817
N_{IU}	1 and 15
N_{FEC_TI} (# of FEC-blocks per TI block)	48 (N_{IU}=1), 6 (N_{IU}=15)
Bandwidth	6 MHz
f_D (Doppler spread)	2, 4,8, 16, and 55 Hz

The underlying BICM-parameters are summarized in Table II. The LDPC code, bit-interleaver and constellation were taken from the ATSC 3.0 baseline, a set of BICM blocks identified as best performing during the standardization process [20].
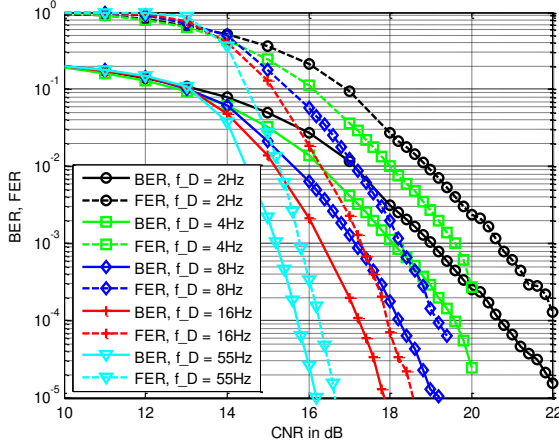
Fig 12. Simulation results for intra-subframe interleaving only (convolutional delay-line is not used: $N_{IU}=1$, $N_{FEC\_TI}=48$)

Fig 13. Simulation results for inter-subframe interleaving (convolutional delay-line is in use: $N_{IU}=15$)

Doppler frequencies were chosen between 2 Hz and 55 Hz. At a carrier frequency of 500 MHz, this corresponds to speeds between ~4 km/h and ~120 km/h and, hence, covers scenarios typical for pedestrian and vehicular propagation conditions in the UHF-band.

The ATSC 3.0 specification limits the TDI memory to a maximum of 2^{19} cells. For intra-subframe interleaving this limit is reached with 48 codewords per TI-Block, i.e., the TDI memory needs to hold 518,400 cells. The resulting performance is shown in Fig. 12. With increasing Doppler spread more time diversity is afforded, and the error rates decrease accordingly.

In order to accommodate inter-subframe interleaving (cf. Fig. 13) the number of FEC-blocks needs to be decreased. For interleaving over $N_{IU}=15$ frames, the number of codewords is reduced to $N_{FEC\_TI}=6$, which keeps the same TDI memory of 518,400 cells as before-hand. Based on (8e), it can be shown that compared to Figure 12 the time interleaving depth is larger by a factor of $2N_{IU}/(N_{IU}+1)=1.875$. In this example, the gains over intra-frame interleaving range approximately between 1 and 2 dB.

VIII. CONCLUSIONS

This paper gave an overview of the time interleaving facilities of the physical layer of ATSC 3.0. The two main modes are a classical convolutional interleaver aimed at constant cell rate transmission and a hybrid time interleaver consisting of cell-interleaving, row-twisted block interleaving and a convolutional delay-line. Both modes can operate with extended time interleaving, which extends the time interleaving depth without necessarily increasing the TDI memory size. The HTI is specifically designed to handle variable cell rate, i.e., a varying number of FEC blocks between TI-blocks. The time interleaving depth, latency, zapping time, and minimum span of each TI was discussed. Shortly addressed was also the application of time interleaving in a MIMO-setting.

IX. NOTE

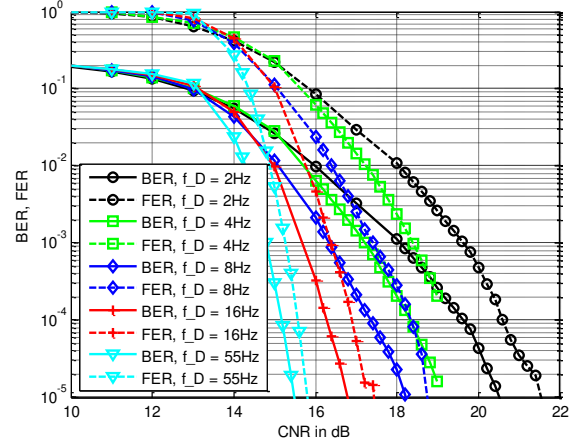It is noted that some elements of this paper could be changed

during the Candidate Standard phase of the ATSC 3.0 physical layer specification.

REFERENCES

[1] S. Lin and D. J. Costello, Error Control Coding, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2004.

[2] G. Jr. Forney, "Burst-Correcting Codes for the Classic Bursty Channel," *IEEE Trans. Commun. Tech.*, vol. 19, no. 5, pp. 772-781, Oct. 1971.

[3] R. Garzon Bohorquez, C.A. Nour, and C. Douillard, "Channel interleavers for terrestrial broadcast: analysis and design," " *IEEE Trans. Broad.* vol. 60, no. 4, pp. 679–692, Dec. 2014.

[4] M. Takada and M. Saito, "Transmission System for ISDB-T," *Proc. of the IEEE*, vol. 94, no. 1, pp. 251-256, Jan. 2006.

[5] M. Takada, *et al.*, "*An Overview of the ISDB-T One-Seg Broadcasting, ISDB-Tsb and ISDB-Tmm,*" in Next Generation Mobile Broadcasting, CRC Press, pp. 151-183, 2013.

[6] ETSI EN 300 744, "Digital Video Broadcasting (DVB); framing structure, channel coding and modulation for digital terrestrial television".

[7] ATSC A/53, "Digital Television Standard," parts 1-6, Jan. 2007.

[8] ETSI EN 302 755, "Digital Video Broadcasting (DVB); Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2)"

[9] ETSI TS 102 831, "Digital Video Broadcasting (DVB); Implementation guidelines for a second generation digital terrestrial television broadcasting system (DVB-T2)".

[10] DVB Bluebook A160, "Digital Video Broadcasting (DVB); Next Generation broadcasting system to Handheld, physical layer specification (DVB-NGH)"

[11] D. Gomez-Barquero, P. F. Gomez, D. Gozalvez, B. Sayadi, and L. Roullet, "Novel Physical Layer Split FEC Scheme for Long Time Interleaving with Fast Zapping Support," *IEEE Trans. Broadcast.*, vol. 58, no. 2, pp. 269-276, 2012.

[12] D. Gomez-Barquero, D. Gozalvez and N. Cardona, "Application Layer FEC for Mobile TV Delivery in IP Datacast over DVB-H Systems," *IEEE Trans. Broadcast.*, vol. 55, no. 2, pp. 396-406, 2009.

[13] I. Eizmendi, *et al.*, "DVB-T2: The Second Generation of Terrestrial Digital Video Broadcasting System," *IEEE Trans. Broadcast.*, vol. 60, no. 2, pp. 258-271, June 2014.

[14] D. Gozalvez, D. Gomez-Barquero, D. Vargas, and N. Cardona, "Time Diversity in Mobile DVB-T2 Systems," *IEEE Trans. on Broadcasting*, vol. 57, no. 3, pp. 617-628, 2011.

[15] M. Earnshaw, K. Shelby, H. Lee, Y, Oh, and M. Simon, "Physical Layer Framing for ATSC 3.0," *IEEE Transactions on Broadcasting*, vol. 62, no. 1, 2016.

[16] N. Loghin, *et. al*, "Non-Uniform Constellations for ATSC 3.0," IEEE Transactions on Broadcasting, vol. 62, no. 1, 2016.

[17] S.-I. Park, *et. al*, "Low Complexity Layered Division Multiplexing System for ATSC 3.0," *IEEE Transactions on Broadcasting*, vol. 62, no. 1, 2016.

[18] D. Gomez-Barquero, *et al*., "MIMO for ATSC 3.0," *IEEE Trans. Broadcast.*, vol. 62, no. 1, 2016.

[19] D. Vargas, *et al*., "MIMO for DVB-NGH, the Next Generation Mobile TV Broadcasting," *IEEE Communications Magazine*, vol. 51, no. 7, pp. 130-137, July 2013.

[20] L. Michael and D. Gomez-Barquero, "Bit Interleaved Coding and Modulation for ATSC 3.0," *IEEE Trans. Broadcast.*, vol. 62, no. 1, 2016.

[21] P. F. Gomez, D. Gomez-Barquero, D. Gozalvez, A. Añorga, and M. Breiling, "Time Interleaving in DVB-NGH," Next Generation Mobile Broadcasting, pp. 355-288, CRC Press, 2013.

Peter Klenner was born in Bremen, Germany. He received the Diploma and PhD degree in Electrical Engineering and Communication and Information Technology from the University of Bremen in 2004 and 2011, respectively.

Since 2011, he is working as a Research Engineer on DTV standardization at the Panasonic AVC Langen Development Centre, close to Frankfurt a. M., Germany. His pursuits in research comprise, without being limited to, optimization, coding theory, iterative decoding, OFDM, MIMO communication, and massively parallel programming.

Jong-Seob Baek received the Ph.D. degree in electrical and electronics engineering from Yonsei University, Seoul, Korea, in 2007.

He was a Senior Engineer with the DMC Research Center, Samsung Electronics, Suwon, from 2007 to 2009. From April 2010 to March 2011, he was with the Georgia Institute of Technology, Atlanta, US, in a reach fellow position. Since September 2012, he has been with the Advanced Standard Research Center, CTO, LG Electronics, as a Senior Engineer. He has been joining the ATSC3.0 standardization from 2012.

He has published over 20 papers in IEEE journals. His current research interests include advanced digital signal processing and analysis for future broadband wireless multimedia broadcasting and communication systems.

Nabil Sven Loghin (born Muhammad) received his diploma degree in Electrical Engineering and PhD degree from the University of Stuttgart, Germany, in 2004 and 2010, respectively, both with summa cum laude. Since 2009 he is with Sony, working on DTV standardization and localization algorithms. His research interests include channel coding, iterative decoding, QAM mapping optimization, and multiple-antenna communications.

David Gómez-Barquero received the double M.Sc. degrees in telecommunications engineering from the Universitat Politecnica de Valencia (UPV), Spain, and the University of Gävle, Sweden, in 2004, the Ph.D. degree in telecommunications from the UPV in 2009; and he carried out a 2-year post-doc at the Fraunhofer Heinrich Hertz Institute, Germany. He is a Senior Researcher (Ramon & Cajal Fellow) with the Institute of Telecommunications and Multimedia Applications, UPV, where he leads a research group working on next generation broadcasting technologies. Previously, he hold visiting research appointments at Ericsson Eurolab, Germany, the Royal Institute of Technology, Sweden, the University of Turku, Finland, the Technical University of Braunschweig, Germany, the Sergio Arboleda University of Bogota, Colombia, and the New Jersey Institute of Technology, USA.

Dr. Gómez-Barquero has been since 2008 actively participating in the European digital television standardization forum DVB in different topics such as upper layer forward error correction, DVB-T2, T2-Lite, and DVB-NGH. In 2013, he joined the U.S. digital television standardization forum ATSC to work on ATSC 3.0, acting as Vice-Chairman of the Modulation and Coding Ad-Hoc Group. He is the Editor of the book entitled Next Generation Mobile Broadcasting (CRC Press).

Woo-Suk Ko was born in In-Cheon, Korea. He received the B.S. degree in Electronic Eng. from Yonsei University in 1996 and the M.S. and Ph.D. degree in Electrical and Electronic Eng. from the same university in 1998 and 2002 respectively.

In 2002, he joined the Digital TV Laboratory of LG Electronics, Seoul, Korea. He was involved in developing the receiver chip sets for broadcasting systems such as Korean terrestrial Digital Multimedia Broadcasting (DMB), the satellite DMB and European DVB-T.

Since 2007, he has participated in the second generation DVB standardization in Europe including DVB-T2, DVB-C2 and DVB-NGH. He has been joining the ATSC3.0 standardization from 2012, which is the next generation terrestrial broadcasting standard in US.