Concurrent OFDM Demodulation and Turbo Decoding for Ultra Reliable Low Latency Communication

Luping Xiang, Robert G. Maunder, *Senior Member, IEEE* and Lajos Hanzo, *Fellow, IEEE*

***Abstract*—The Ultra-Reliable Low Latency Communication (URLLC) applications have been proposed in recent years, targeting a round-trip end-to-end latency less than 1 ms with high reliability. Therefore, an order of magnitude improvements are needed in all layers of the wireless communication stack. This is a particular challenge for the physical layer, where typically a processing time of the order of microseconds is required for the computationally intensive demodulation and error correction processing, among other operations. Conventionally, the reception of signals, the demodulation processing and the error correction processing are performed consecutively at the receiver. However, this approach is associated with processing times on the order of hundreds of microseconds, preventing URLLC. Therefore, this paper proposes a novel processing architecture, which is capable of performing reception, Orthogonal Frequency Division Multiplexing (OFDM) demodulation and turbo decoding concurrently, rather than consecutively, hence significantly reducing the processing time. In order to achieve concurrent operation, the OFDM demodulation is performed using a novel cumulative Fast Fourier Transform (FFT), which produces successively more reliable estimates of all transmitted symbols in each successive clock cycle. At the same time, a Fully-Parallel Turbo Decoder (FPTD) is used to recover successively more reliable estimates of all bits in each successive clock cycle.**

***Index Terms*—Fast Fourier Transform (FFT), Orthogonal Frequency Division Multiplexing (OFDM), Fully-Parallel Turbo Decoder (FPTD), latency.**

I. Introduction

In addition to significantly increased throughput and reliability, a Ultra-Reliable Low Latency Communication (URLLC) paradigm has been proposed for the next generation wireless communication systems, targeting a significantly reduced end-to-end latency of less than 1ms [1, 2]. An even lower latency is required for the communications beyond 5G [3]. Achieving an increased throughput and reliability as well as a significant reduction in latency represents a significant challenge [4–6], having no simple solutions. However once realised, this URLLC paradigm will allow humans or machines to communicate with remote mobile devices and control them seamlessly, without suffering from the lag that prevents accurate control using wireless communication systems [7]. This will enable a wide variety of new applications in remote surgery, automated driving, and virtual reality, having significant economic and societal impact [8–12]. However, the end-to-end latency of a wireless communication system is fundamentally limited by its physical layer [13, 14], which performs demodulation and error correction, among other tasks.

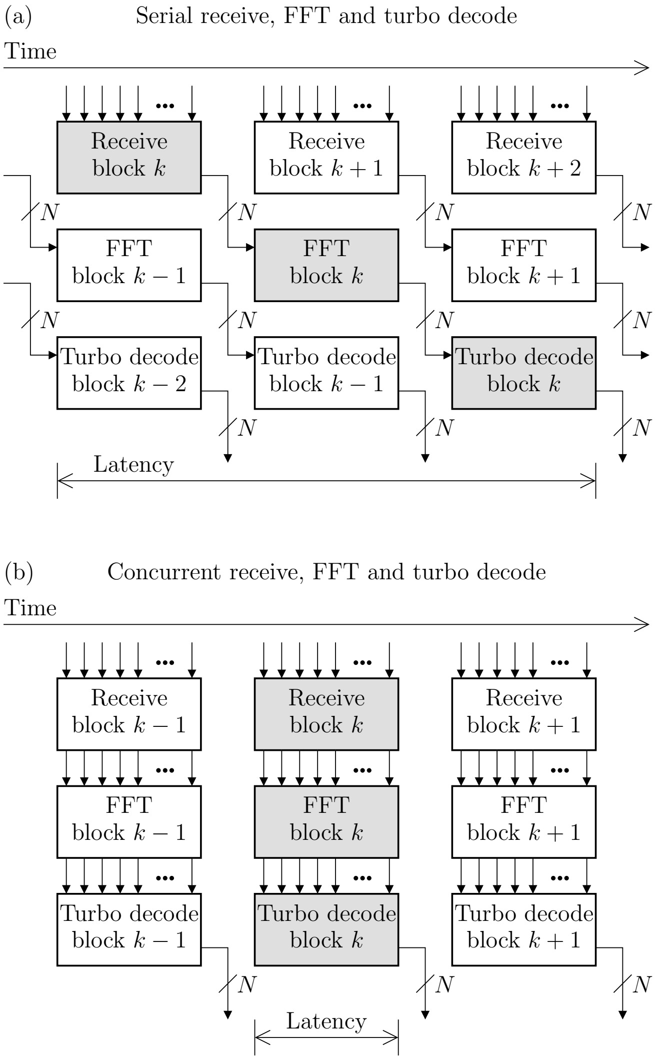

Fig. 1. Timing diagram for (a) the conventional approach and (b) the proposed ultra-low-latency approach.

Different future applications of the URLLC mode will impose different demands on the physical layer latency. For example, machine-automated low-latency capital market trad-

The authors are with Electronics and Computer Science, University of Southampton, SO17 1BJ, United Kingdom, e-mail: {lx1g15, rm, lh}@ecs.soton.ac.uk. The research data for this research is available at the University of Southampton institutional repository: 10.5258/SOTON/D1185. L. Hanzo would like to gratefully acknowledge the ERC's financial support of the Advanced Fellow Grant QuantCom.

ing relies on multi-hop wireless communication links, where financial institutions use algorithms running on their own computers for automatically buying and selling stocks, whenever they momentarily have different values on stock exchanges in different cities. In this application, each relay has in these links to have a sub-microsecond physical layer latency [15], not including the propagation delay associated with each hop. By contrast, the so-called Tactile Internet [13, 16] will allow humans to seamlessly control remote devices, provided that a physical layer latency of below 100 μs can be achieved.

However, state-of-the-art (SOTA) wireless communication systems have physical layer latencies that are significantly higher than these targets. For example, the world's fastest low-latency capital market trading links impose a physical layer latency of around 5 μs per hop, not including propagation latency. Meanwhile, SOTA implementations of the Long Term Evolution (LTE) cellular telephony standard have a physical layer latency, which significantly exceeds the 100 μs target of the Tactile Internet [13, 17]. In order to achieve a high throughput and reliability compared to predecessor schemes, 3GPP LTE employs OFDM [18–20] for mitigating echoes in the wireless channel, as well as a turbo code for correcting any remaining transmission errors [21–26] Recently, a URLLC mode of operation has been introduced in the LTE standard, which maintains the combination of OFDM and turbo coding, but aims to reduce the processing time available for these operations by 7 times [27]. Motivated by this, we adopt the combination of OFDM and turbo coding in this paper, although this combination can also be expected to have applicability to other FFT-based modulation schemes and other iterative decoding channel codes, such as low density parity check (LDPC) codes. However, these techniques impose a high signal processing complexity upon the physical layer, particularly in the receiver. As shown in Figure 1(a), the processing of the receiver's Fast Fourier Transform (FFT) [28, 29] cannot begin until the whole message block has been received, since each of its outputs is a function of the whole received block. Owing to this, the FFT produces all of its outputs simultaneously, preventing the turbo decoding process from beginning until after the FFT has been completed. In practical LTE deployments, the transmission latency incurred while receiving, the processing latency incurred while performing the FFT and the processing latency incurred while performing turbo decoding are each around 70 μs [13, 23], allowing pipelining as shown in Figure 1(a). The sum of these latencies is 210 μs, which already exceeds the above-mentioned 100 μs target, even without considering the latency associated with propagation, channel estimation, Multiple-Input Multiple-Output (MIMO) detection and transmitter processing.

This motivates our new architecture, in which the physical layer receiver components are operated concurrently, rather than consecutively. This approach is exemplified by Figure 1(b), in which the reception, FFT processing and turbo decoding of each block is performed concurrently, potentially facilitating sub-microsecond physical layer latencies in the case of low-latency capital market trading. In the case of the URLLC communications [18, 20, 30], this approach can reduce the associated latency from 210 μs to 70 μs, which is within the above-mentioned 100 μs latency target of the Tactile Internet. This leaves 30 μs for propagation and for the remaining, lower-complexity physical layer components, including channel estimation, MIMO detection and transmitter processing. Indeed, it may be expected that the proposed technique can be extended to perform some of these operations concurrently with those of Figure 1(b), within the same 70 μs. In addition, the turbo codes adopt the advantage of lower decoding complexity and better error-correction performance compared to the LDPC codes at low coding rates that are motivated in mission-critical vehicular communications for the sake of ensuring low bit error rate (BER) and ultra-high reliability [31, 32]. Our new contributions are as follows.

1) we propose a novel cumulative FFT, which is processed incrementally and concurrently with the Fully-Parallel Turbo Decoder (FPTD) of [33], throughout the process of receiving a single OFDM symbol. Since the information carried by each turbo encoded bit is spread throughout the duration of the OFDM symbol, the proposed concurrent FFT can obtain some information about each bit as soon as the reception of the OFDM symbol begins, allowing turbo decoding to start immediately. As more and more of the OFDM symbol is received with passing time, the cumulative FFT can obtain more and more information about the turbo encoded bits, which can be fed into the concurrent turbo decoding process.
2) We show that if the turbo decoder can complete a sufficient number of iterations within the duration of the OFDM symbol, then it can achieve the same error correction performance as if the turbo decoding process had only began after the reception of the OFDM symbol had been completed.

The rest of this paper is structured as follows. Section II provides a brief overview of techniques that are employed in our proposed architecture, including the FFT and the FPTD [33]. Following this, the proposed concurrent OFDM demodulation and turbo decoding architecture is proposed and detailed in Section III. The validation of this architecture is presented in Section IV, while its error correction performance and extensions to manage the trade-offs between latency, reliability and complexity are presented in Section V. Finally, we offer our conclusions and avenues for future work in Section VI.

II. Background

This section provides an overview of the FFT and FPTD techniques employed in the proposed architecture, and defines the notation that is employed in the following sections. Section II-A introduces the concept of a novel cumulative FFT, which will be detailed in Section III-B. Meanwhile, Section II-B highlights the decoding FTPD process of [33].

A. Fast Fourier Transform

In Orthogonal Frequency Division Multiplexing (OFDM), a bit stream is decomposed into several parallel bit streams, each of which has much a proportionately reduced bit rate and is modulated onto a different subcarrier. In this way, rather than using a serial time-domain (TD) bit stream, OFDM uses many low-rate parallel frequency domain (FD) bit streams, which are less prone to dispersion. Typically, OFDM schemes are implemented using Discrete Fourier Transform (DFT) techniques [34–36]. To be more specific, the Inverse Discrete Fourier Transform (IDFT) is performed in the transmitter to generate a single TD OFDM symbol to represent the set of the FD bit streams, each of which typically carries a Quadrature Amplitude Modulation (QAM) symbol. Meanwhile, the corresponding DFT is performed at the receiver to recover the QAM symbols carried by the sub-carriers of the OFDM symbol. In the receiver, a N-point DFT can be defined as

$$Y_z = \sum_{n=0}^{N-1} y_n W_N^{nz}, \quad 0 \leqslant z \leqslant N-1, \tag{1}$$

where $W_N = \exp(-j2\pi/N)$. Here, y_n is the nth sample of the received TD OFDM symbol, where the set of N samples are received consecutively, spread over time. Meanwhile, Y_z is the zth FD subcarrier's QAM symbol, where each of the N FD QAM symbols is dependent on all N TD samples of the TD OFDM symbol.

Note that in practice, the demodulator's DFT is typically implemented using the FFT, which has a significantly reduced complexity if N is high [28,29]. If N is a power of 2, a Radix-2 FFT is achieved by recursively partitioning Y_z into odd- and even-indexed terms, across $v = \log_2(N)$ stages. The odd and even terms in Y_z can be expressed respectively as

$$Y_{2z} = \sum_{n=0}^{N/2-1} \left(y_n + y_{n+N/2}\right) W_{N/2}^{nz}, \quad 0 \leqslant z \leqslant \frac{N}{2} - 1, \tag{2}$$

$$Y_{2z+1} = \sum_{n=0}^{N/2-1} \left[\left(y_n - y_{n+N/2}\right) W_N^{n}\right] W_{N/2}^{nz}, \\ 0 \leqslant z \leqslant \frac{N}{2} - 1. \tag{3}$$

Here, each of (2) and (3) can be considered to be a DFT in its own right. This allows each of (2) and (3) to be further decomposed into two DFTs, comprising the odd and even elements, respectively. This may be repeated recursively, until the DFT has been fully decomposed into an individual radix-2 structure, completing the FFT.

The block diagram for the example of a $N = 16$-point radix-2 FFT with $v = 4$ computation stages is depicted in Figure 2. The inputs on the left-hand edge of Figure 2 represent TD samples. These inputs are first interleaved into a bit reversed ordering, separating the odd terms and even terms. The TD samples are passed through $v = 4$ stages of radix-2 butterflies, each of which performs a radix-2 FFT calculation, as shown in Figure 3. Following the final stage of radix-2 butterflies, the $N = 16$ FD samples are obtained.

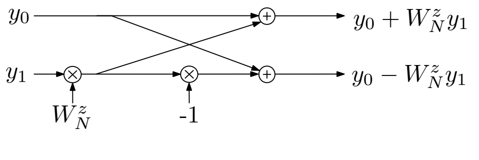

Fig. 3. The radix-2 FFT calculation.

In the conventional approach, reception and demodulation are performed serially, where the FFT operation does not begin until after all the transmitted TD samples are received. However, in our concurrent approach, the demodulation process is started promtly after a small number of samples have been received. Successively more TD samples are received in each of a series of successive clock cycles. In a naive approach, the full-length N-sample FFT may be replaced in each clock cycle, using all TD samples received so far and assuming zero values for all TD samples not yet received, as shown in Figure 2. In order to significantly reduce the complexity of repeating the N-point FFT in each clock cycle, we propose an efficient cumulative FFT in Section III-B. This eliminates all redundant calculations associated with zero-valued samples and reuses calculations from one clock cycle to the next. More specifically, an incremental part of the FFT is calculated in each clock cycle and these are accumulated across the series of clock cycles.

B. Fully-Parallel Turbo Decoder

Turbo encoders typically comprise two component encoders, which are used for encoding a sequence of information bits and an interleaved version of the sequence, respectively. Conventionally, turbo decoding is achieved using the Log-BCJR algorithm [37], which relies on forward and backward recursions along a trellis representation of the upper and lower component decoders. The component decoders are operated alternatively, over serial iterations, exchanging information via the interleaver. However, the forward and backward recursions impose data dependencies, which limit the achievable degree of parallel processing, resulting in high processing latency. Several approaches have been proposed to improve the throughput and latency of the Log-BCJR turbo decoder, most of which focus on increasing the parallelism of the conventional turbo decoder [33,38–40]. To be more specific, we have previously proposed a FPTD algorithm [33,41], which dramatically increases the parallelism of the decoding process and achieves significantly lower latency, by dispensing with the recursions of the Log-BCJR algorithm.

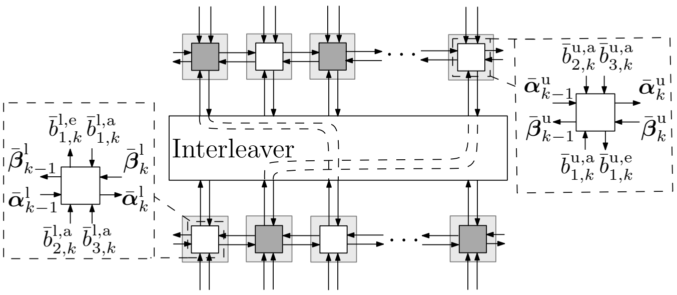
$\bar{b}^{\mathrm{u,a}}_{2,k}$ $\bar{b}^{\mathrm{u,a}}_{3,k}$
$\bar{\alpha}^{\mathrm{u}}_{k-1}$
$\bar{\alpha}^{\mathrm{u}}_{k}$
$\bar{\beta}^{\mathrm{u}}_{k-1}$
$\bar{\beta}^{\mathrm{u}}_{k}$
$\bar{b}^{\mathrm{u,a}}_{1,k}$ $\bar{b}^{\mathrm{u,e}}_{1,k}$
Interleaver
$\bar{b}^{\mathrm{l,e}}_{1,k}$ $\bar{b}^{\mathrm{l,a}}_{1,k}$
$\bar{\beta}^{\mathrm{l}}_{k-1}$
$\bar{\beta}^{\mathrm{l}}_{k}$
$\bar{\alpha}^{\mathrm{l}}_{k-1}$
$\bar{\alpha}^{\mathrm{l}}_{k}$
$\bar{b}^{\mathrm{l,a}}_{2,k}$ $\bar{b}^{\mathrm{l,a}}_{3,k}$

Fig. 4. Systematic of fully-parallel turbo decoder.

The FPTD operates on the basis of soft information in the form of Logarithmic Likelihood Ratios (LLRs). Here, we define the LLR $\bar{b}$ pertaining to a bit $b \in \{0,1\}$ as

$$\bar{b} = \ln \frac{\Pr(b=1|\mathbf{Y})}{\Pr(b=0|\mathbf{Y})}, \tag{4}$$

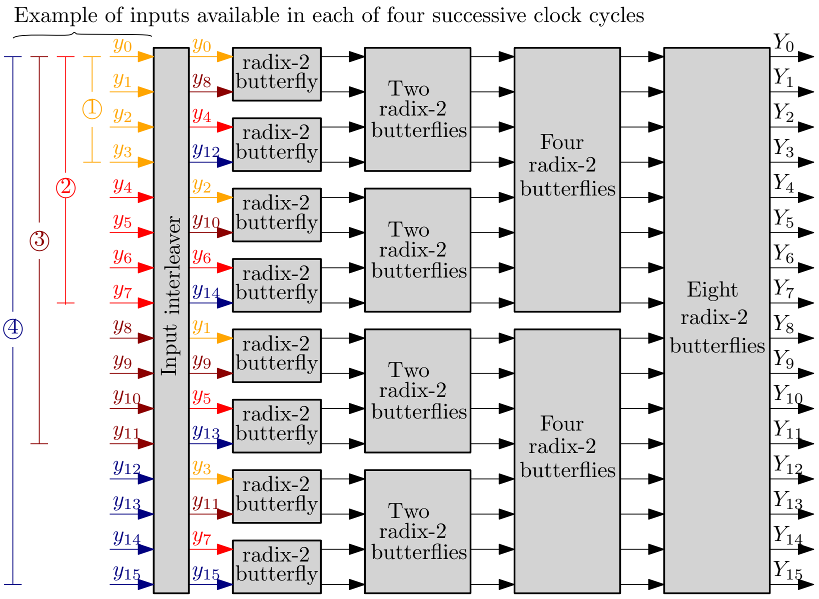
Example of inputs available in each of four successive clock cycles
①
②
③
④
y_0 y_1 y_2 y_3 y_4 y_5 y_6 y_7 y_8 y_9 y_{10} y_{11} y_{12} y_{13} y_{14} y_{15}
Input interleaver
y_0 y_8 y_4 y_{12} y_2 y_{10} y_6 y_{14} y_1 y_9 y_5 y_{13} y_3 y_{11} y_7 y_{15}
radix-2 butterfly
Two radix-2 butterflies
Four radix-2 butterflies
Eight radix-2 butterflies
Y_0 Y_1 Y_2 Y_3 Y_4 Y_5 Y_6 Y_7 Y_8 Y_9 Y_{10} Y_{11} Y_{12} Y_{13} Y_{14} Y_{15}

Fig. 2. The block diagram of the conventional FFT and the cumulative FFT approach.

where $\mathbf{Y}$ is the received signal. As depicted in Figure 4, the FPTD is comprised of an upper and a lower decoder, each of which comprises K algorithmic blocks, corresponding to the K message bits. The kth algorithmic block in the upper decoder is provided with the corresponding one of the parity *a priori* LLRs $[\bar{b}^{\mathrm{u,a}}_{2,k}]_{k=0}^{K-1}$ and the corresponding one of the systematic *a priori* LLRs $[\bar{b}^{\mathrm{u,a}}_{3,k}]_{k=0}^{K-1}$ by the demodulator, as well as the corresponding one of the message *a priori* LLRs $[\bar{b}^{\mathrm{u,a}}_{1,k}]_{k=0}^{K-1}$. Likewise, the parity and the systematic *a priori* LLRs provided for the blocks in the lower decoder can be expressed as $[\bar{b}^{\mathrm{l,a}}_{2,k}]_{k=0}^{K-1}$ and $[\bar{b}^{\mathrm{l,a}}_{3,k}]_{k=0}^{K-1}$, respectively, while the corresponding message *a priori* LLRs can be expressed as $[\bar{b}^{\mathrm{l,a}}_{1,k}]_{k=0}^{K-1}$. Note that in the case of the LTE turbo code, the systematic LLRs $[\bar{b}^{\mathrm{l,a}}_{3,k}]_{k=0}^{K-1}$ may be obtained by replicating and interleaving those $[\bar{b}^{\mathrm{u,a}}_{3,k}]_{k=0}^{K-1}$ provided to the upper decoder. Also note that for the sake of simplicity we omit the discussion of the LLRs pertaining to the twelve termination bits in the LTE standard [42].

Besides the LLRs, the kth algorithmic block in each decoder is provided with a vector of A forward state metrics $\bar{\boldsymbol{\alpha}}^{\mathrm{u}}_{k-1}$ or $\bar{\boldsymbol{\alpha}}^{\mathrm{l}}_{k-1}$, as well as a vector of A backward state metrics $\bar{\boldsymbol{\beta}}^{\mathrm{u}}_{k}$ or $\bar{\boldsymbol{\beta}}^{\mathrm{l}}_{k}$, where we have $A = 8$ in the LTE standard. When the kth algorithmic block in each decoder is activated, these state metrics are combined with the *a priori* LLRs in order to generate a vector of A forward state metrics $\bar{\boldsymbol{\alpha}}^{\mathrm{u}}_{k-1}$ or $\bar{\boldsymbol{\alpha}}^{\mathrm{l}}_{k-1}$, as well as a vector of A backward state metrics $\bar{\boldsymbol{\beta}}^{\mathrm{u}}_{k}$ or $\bar{\boldsymbol{\beta}}^{\mathrm{l}}_{k}$, which are provided to the neighbouring algorithmic blocks on either side in the same decoder. Furthermore, the kth algorithmic block generates a message extrinsic LLR $\bar{b}^{\mathrm{u,e}}_{1,k}$ or $\bar{b}^{\mathrm{l,e}}_{1,k}$, which is passed by the interleaver to one of the algorithmic blocks in the other decoder, where it is used as a message *a priori* LLR $\bar{b}^{\mathrm{l,a}}_{1,k}$ or $\bar{b}^{\mathrm{u,a}}_{1,k}$, respectively.

Rather than employing the alternated activation of the upper and lower decoders as in the SOTA turbo decoder, the FPTD exploits the odd-even property of the LTE interleaver to enable an odd-even processing schedule [43, 44]. To be more specific, the interleaver corresponding to each of the 188 legitimate frame lengths K supported in the LTE standard only connects algorithmic blocks of the upper decoder having an odd index to algorithmic blocks in the lower decoder that have an odd index as well. Likewise, even-index algorithmic blocks in the upper decoder are only connected to algorithmic blocks with even indices in the lower decoder. As a result, the algorithmic blocks can be grouped into two sets, where no two blocks in the same set have connections to each other. To be more explicit, the first set consists of all the odd-indexed blocks in the upper decoder, along with all even-indexed blocks in the lower decoder. Likewise, the second set comprises the remaining blocks, namely the even-indexed blocks in the upper decoder, together with the odd-indexed blocks in the lower decoder.

In the FPTD's odd-even processing schedule, it is the two sets of algorithmic blocks that iteratively exchange extrinsic LLRs and state metrics. In the first-half iteration of the FPTD operation, the odd-indexed algorithmic blocks in the upper decoder and the even-index blocks in the lower decoder operate simultaneously during a first clock cycle. The second half iteration is performed during the second clock cycle and involves the operation of all other algorithmic blocks. This process is repeated during successive iterations, with each iteration comprising only two clock cycles. Although the FPTD requires more iterations to achieve the same BER as the SOTA Log-BCJR decoder, its benefit is that the total number of clock cycles required is reduced from hundreds or thousands to tens. A detailed comparison of the FPTD with the SOTA LTE turbo decoder in terms of its complexity per iteration, the number of decoding iterations required and the overall latency, is presented in [33].

III. Proposed turbo-coded OFDM scheme

The proposed concurrent turbo-coded OFDM scheme is discussed in this section. Section III-A introduces our notation and details the proposed scheme's transmitter, which is the same as in a conventional turbo-coded OFDM scheme. Following this, the proposed concurrent detection, FFT and turbo decoding approach of the proposed receiver is detailed in Section III-B.

A. Transmitter

In the turbo-coded OFDM transmitter of Figure 5, the K number of message bits $\mathbf{b}_1^{\mathrm{u}} = [b_{1,k}]_{k=0}^{K-1}$ are encoded by an LTE turbo encoder. To be more specific, the vector of message bits $\mathbf{b}_1^{\mathrm{u}}$ is interleaved to obtain the vector of interleaved message bits $\mathbf{b}_1^{\mathrm{l}}$. These two vectors are encoded by two identical convolutional encoders (CEs), referred to as the upper and lower encoders, respectively. The resultant parity bit vectors $\mathbf{b}_2^{\mathrm{u}}$ and $\mathbf{b}_2^{\mathrm{l}}$ are interleaved with the systematic bit vector $\mathbf{b}_3^{\mathrm{u}}$, which is a replica of the message bit vector $\mathbf{b}_1^{\mathrm{u}}$. The resultant bit vector is punctured or repeated depending on the code length and then output as the turbo-encoded bit vector $\mathbf{b}_4 = [b_{3,k}]_{t=0}^{T-1}$, which comprises T bits. The T turbo-encoded bits are then converted into $N = T/\log_2 M$ symbols $\mathbf{X} = [X_n]_{n=0}^{N-1}$, using an M-ary Quadrature Amplitude Modulation (MQAM) mapper. Here, the MQAM symbols are selected from a set of M complex constellation points $\mathcal{S} = \{s_0, s_1, \cdots s_{M-1}\}$, which satisfy $\sum_{i=0}^{M-1} |s_i|^2 / M = 1$, where each QAM symbol carries $b = \log_2 M$ bits. Following this, a Serial to Parallel Converter (SPC) is used to convert the series of N symbols $\mathbf{X} = [X_n]_{n=0}^{N-1}$ into the input of the Inverse Fast Fourier Transform (IFFT), which obtains a corresponding set of N complex TD samples of the OFDM symbol $\mathbf{x} = [x_n]_{n=0}^{N-1}$. Next, the TD OFDM symbol $\mathbf{x}$ is concatenated with L samples provided by the corresponding Cyclic Prefix (CP) $[x_n]_{n=N}^{N+L-1}$, in order to avoid the inter-symbol interference associated with dispersive channels [19]. Finally, the resultant turbo-encoded, OFDM-modulated symbol is passed through a Parallel to Serial Converter (PSC) and transmitted using a Digital to Analogue Converter (DAC) and a Radio Frequency (RF) front end.

B. Receiver

The proposed receiver schematic is depicted in Figure 6, where the signal is received using a RF front end, an Analogue to Digital Converter (ADC), a CP remover, and a SPC. These components have naturally low latencies compared to the rest of the schematic, which comprises a novel cumulative FFT, a bank of N novel Quadrature Amplitude Modulation (QAM) demappers and the FPTD [33]. The operations of the cumulative FFT, the N QAM demappers and the FPTD are spread over a total of C clock cycles. Figure 6 illustrates a 'toy' example, in which $C = 4$ clock cycles are used to recover $K = 8$ bits from $N = 16$ samples of the 4QAM-modulated OFDM symbol.

After the removal of the CP, the received samples of the OFDM symbol can be expressed as

$$\mathbf{y} = \mathbf{h} * \mathbf{x} + \mathbf{n}, \tag{5}$$

where $\mathbf{h} = [h_n]_{n=0}^{N-1}$ is the Channel Impulse Response (CIR) and $\mathbf{n} = [n_n]_{n=0}^{N-1}$ is the Additive White Gaussian Noise (AWGN), which has a zero mean and a variance of $\sigma^2 = 1/(2\gamma)$, where γ denotes the Signal-to-Noise Ratio (SNR). The corresponding FD signal obtained after the FFT operation can be expressed as

$$\begin{aligned}\mathbf{Y} =& \boldsymbol{\mathcal{F}}\{\mathbf{h} * \mathbf{x} + \mathbf{n}\}\\ =& \mathbf{HX} + \mathbf{N}.\end{aligned} \tag{6}$$

Here, for the kth element in $\mathbf{Y}$, we have $Y_k = H_k X_k + N_k$, where H_k is the single tap channel gain of the corresponding QAM symbol X_k. However, instead of waiting to receive all N samples of the OFDM symbol before performing the FFT operation, the novel approach of Figure 6 performs the cumulative FFT during each clock cycle, while the TD samples are still being received. The N/C samples of the OFDM-modulated symbol received in each clock cycle comprises the fraction $1/C$ of the total number of samples N. Within the same clock cycle, these N/C TD samples are immediately forwarded to the cumulative FFT, which updates its output symbols $\mathbf{Y}$ by incorporating these samples. More specifically, the cumulative FFT effectively calculates an FFT of all samples received so far, while assuming zero values for the remaining samples in the OFDM symbol that have not yet been received. This is exemplified in Figure 2, where only $N/C = 4$ of the received symbols are forwarded to the culmulative FFT in the first clock cycle, with the remaining symbols that have not yet been received being set to zeros. After the interleaver, these $N/C = 4$ input symbols are evenly distributed among the inputs of the radix-2 FFT calculations, as demonstrated in Figure 3, with the other inputs taking zeros.

The operation of the cumulative FFT may be demonstrated by decomposing (1) in terms of the TD samples received in each clock cycle, as follows.

$$\begin{aligned}Y_z &= \sum_{n=0}^{N-1} y_n e^{-j2\pi nz/N}\\ &= \sum_{c=0}^{C-1} \sum_{m=0}^{N/C-1} y_{\frac{N}{C}c+m} e^{-j2\pi z(\frac{N}{C}c+m)/N}\\ &= \sum_{c=0}^{C-1} e^{-j2\pi zc/C} \sum_{m=0}^{N/C-1} y_{\frac{N}{C}c+m} e^{-j2\pi zm/N}\end{aligned} \tag{7}$$

where $y_{\frac{N}{C}c+m}$ is the mth TD sample received in the cth clock cycle. This decomposition reveals that the operation of the FFT over all the N received symbols is equivalent to performing an N/C-point FFT over each set of N/C TD samples received in each clock cycle and performing a weighted sum of the results across the C clock cycles.

Figure 3 shows that if either of the two inputs of a radix-2 butterfly adopts a value of zero, then its two applied outputs adopt the value of the other input, but with different phase shifts. This observation is exploited in the proposed cumulative FFT, where the N/C TD samples received in each clock cycle are evenly distributed by the input interleaver to provide only one non-zero value among each set of C adjacent nodes. Therefore, the output of the first $\log_2 C$ stages would simply

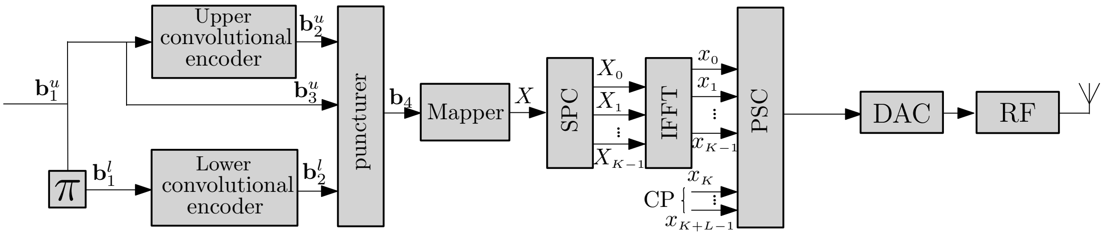

Fig. 5. Transmitter schematic of a turbo-coded OFDM communication system.

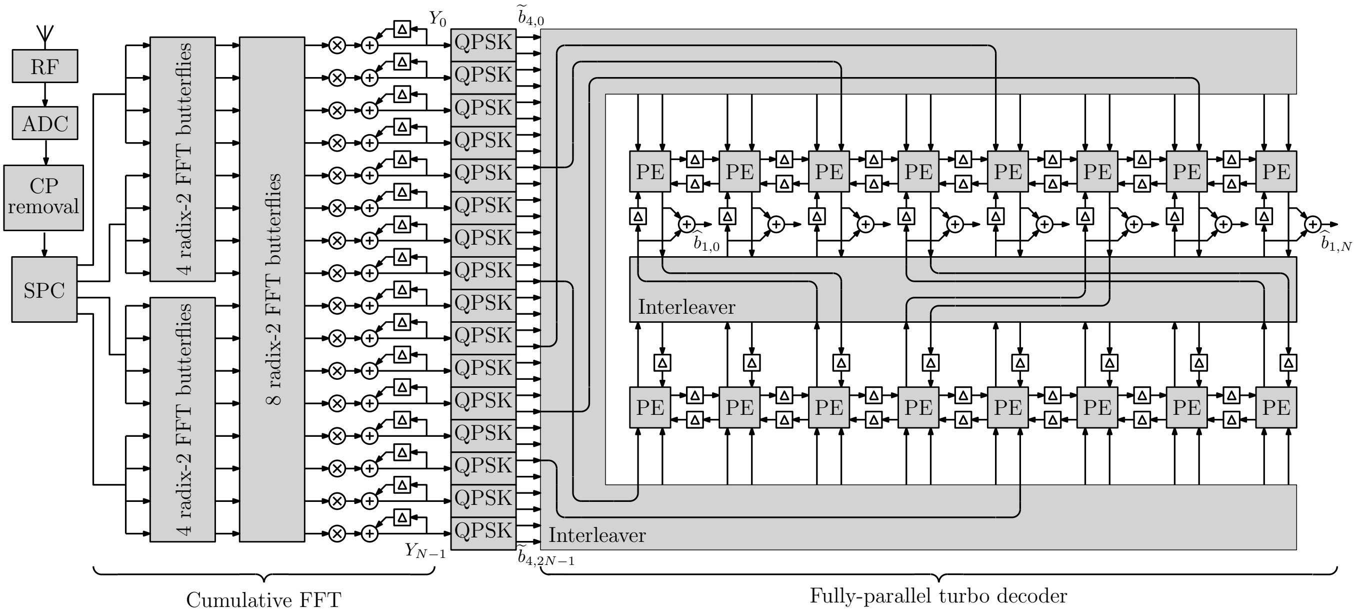

Fig. 6. A toy example of the proposed ultra-low-latency architecture for the concurrent receive, FFT and turbo decode approach of Figure 1(b).

be the replicas of the non-zero values, but with different phase shifts applied. By exploiting this, only $\log_2(N/C)$ layers of radix-2 butterflies are required in order to perform the N-point FFT operation, where multipliers $\otimes$ are employed to shift the phase $e^{-j2\pi zc/N}$ of the N resultant symbols. Following this, adders $\oplus$ and registers $\boxed{\triangle}$ are used to accumulate the results obtained in each successive clock cycle. The proposed cumulative FFT behaves as an SPC, with each successive clock cycle providing N QAM-modulated symbols of progressively higher quality, containing a diminishing level of Inter-Carrier Interference (ICI).

Each of the N QAM demappers of Figure 6 processes the corresponding $M = 4$QAM-modulated symbol by approximately modeling the ICI as additional Gaussian-distributed noise [45]. More specifically, the detection of symbols modulated using a set $\mathcal{S}$ of M constellation points begins by equalising the symbol that it is provided with in each clock cycle, converting it into soft LLRs. The *a priori* probability of the ith bit $b_{4,i}$ in symbol $X_k \in \mathcal{S}$ being 0, given the kth received symbol Y_k, is

$$P(\tilde{b}_{4,i} = 0|Y_k) = \sum_{X_k \in \mathcal{S}|X_{k,i}=0} P(X_k|Y_k) \tag{8}$$

According to Bayes' rule, we have

$$P(X_k|Y_k) = \frac{P(X_k)P(Y_k|X_k)}{P(Y_k)} \tag{9}$$

For the case of the single tap channel characterised in (6), we have

$$\begin{aligned} P(Y_k|X_k) &= \frac{1}{\sqrt{2\pi}\sigma_r} \exp\left(-\frac{1}{2\sigma_r^2}\left\|Y_k - H_k X_k\right\|^2\right) \\ &\propto \exp\left(-\frac{1}{2\sigma_r^2}\left\|Y_k - H_k X_k\right\|^2\right), \end{aligned} \tag{10}$$

where $\sigma_r^2 = \frac{N+(N-D)\gamma\sum_{k=0}^{N-1}\|H_k\|^2}{2D\gamma}$, as will be demonstrated as follows. Here, D denotes the number of QAM symbols received in each clock cycle.

After performing the culmulative FFT on the received symbol in the cth clock cycle, we have

$$\begin{aligned} Y_{z,c} &= \mathcal{F}\left\{\boldsymbol{h} * (\boldsymbol{x}\boldsymbol{I}_D) + \boldsymbol{n}\right\}_z \\ &= H_z \sum_{k=0}^{N} (\frac{1}{N}\sum_{n=0}^{N-1} X_n W_N^{nk}) d_k W_N^{-kz} + N_z, \end{aligned} \tag{11}$$

where $\boldsymbol{I}_D$ is first D columns of a diagonal matrix $\boldsymbol{I}_N$, with a size of $N \times N$ and

$$d_k = \begin{cases} 1, & \text{if } \frac{N}{C}c \geqslant k; \\ 0, & \text{otherwise.} \end{cases}. \tag{12}$$

Using the derivations of Appendix A, the ICI and noise can be separated in $Y_{z,c}$, which can be expressed as

$$Y_{z,c} = H_z \frac{1}{N} \sum_{k=0}^{N-1} \sum_{n=0}^{N-1} X_n W_N^{nk} d_k W_N^{-kz} + N_z$$

$$= \underbrace{H_z \frac{D}{N} X_z}_{\text{what we expect}} + \underbrace{H_z \frac{1}{N} \sum_{n=0,n\neq z}^{N-1} X_n \sum_{k=0}^{D-1} \left[W_N^{(n-z)k} \right]}_{\text{ICI}} + \underbrace{N_z}_{\text{noise}} \tag{13a}$$

$$= \frac{D}{N} \left(H_z X_z - \frac{H_z}{D} \sum_{n=0,n\neq z}^{N-1} \sum_{k=D}^{N-1} X_n W_N^{(n-z)k} + \frac{N}{D} N_z \right), \tag{13b}$$

where the derivation of (13b) is also provided in Appendix A. Then the signal to noise power ratio can be expressed as

$$\frac{P_{Noise}}{P_{signal}} = \frac{D \cdot T_s}{N \cdot T_s} \sum_{z=0}^{N-1} r_{noise}^2 = \frac{D \cdot N^2}{N \cdot D^2} \sum_{z=0}^{N-1} N_z^2 = \frac{N}{D} N_0, \tag{14}$$

whereas the ICI to signal power ratio can be expressed as

$$\begin{aligned} \frac{P_{ICI}}{P_{signal}} &= \frac{D \cdot T_s}{N \cdot T_s} \sum_{z=0}^{N-1} r_{ici}^2 \\ &= \frac{D}{N} \sum_{z=0}^{N-1} \left\| \frac{H_z}{D} \sum_{n=0,n\neq z}^{N-1} \sum_{k=D}^{N-1} X_n W_N^{(n-z)k} \right\|^2 \\ &= \frac{D}{N} \sum_{z=0}^{N-1} \frac{\|H_z\|^2}{D^2} N(N-D) \\ &= \frac{N-D}{D} \sum_{k=0}^{N-1} \|H_k\|^2. \end{aligned} \tag{15}$$

Now, we obtain the relationship between the signal power and the ICI power together with the noise, as

$$\frac{P_{ICI+Noise}}{P_{signal}} = \frac{N}{D} N_0 + \frac{N-D}{D} \sum_{k=0}^{N-1} \|H_k\|^2. \tag{16}$$

Therefore, given the unit signal power, the variance of the ICI and noise can be expressed as

$$\begin{aligned} \sigma_r &= \frac{\frac{N}{D} N_0 + \frac{N-D}{D} \sum_{k=0}^{N-1} \|H_k\|^2}{2} \\ &= \frac{N + (N-D)\gamma \sum_{k=0}^{N-1} \|H_k\|^2}{2D\gamma}. \end{aligned} \tag{17}$$

The LLR of the ith bit in symbol X_k can now be expressed as

$$\begin{aligned} \text{LLR}(\tilde{b}_{4,i}) &= \log \frac{P(\tilde{b}_{4,i} = 0|Y_k)}{P(\tilde{b}_{4,i} = 1|Y_k)} \\ &= \log \frac{\sum_{X_{k,0} \in \mathcal{S}_0} \frac{1}{M\sigma_r} \exp(-\frac{1}{2\sigma_r^2} \|Y_k - H_k X_{k,0}\|)}{\sum_{X_{k,1} \in \mathcal{S}_1} \frac{1}{M\sigma_r} \exp(-\frac{1}{2\sigma_r^2} \|Y_k - H_k X_{k,1}\|)}, \end{aligned} \tag{18}$$

where the symbol set $\mathcal{S}_0$ comprises all constellation points in $\mathcal{S}$ that imply 0 values for the ith bit $\tilde{b}_{4,i}$, while $\mathcal{S}_1$ comprises the constellation points that imply a value of 1 for the ith bit $\tilde{b}_{4,i}$. The $\max^*$ operator [46] may then be employed to simplify the calculation of (18), according to

$$\begin{aligned} \max{}^*(a,b) &= \log(e^a + e^b) \\ &= \max(a,b) + \log(e^a + e^b) \\ &\approx \max(a,b). \end{aligned} \tag{19}$$

Therefore, the LLR of $b_{4,i}$ can be expressed as

$$\begin{aligned} \text{LLR}(\tilde{b}_{4,i}) = &\max_{X_{k,0} \in \mathcal{S}_0}{}^* \left(-\frac{1}{2\sigma_r^2} \|Y_k - H_k X_{k,0}\| \right) \\ &- \max_{X_{k,1} \in \mathcal{S}_1}{}^* \left(-\frac{1}{2\sigma_r^2} \|Y_k - H_k X_{k,1}\| \right). \end{aligned} \tag{20}$$

Then, the resultant set of $N \log_2 M$ LLRs is distributed among the inputs of the FPTD by its interleaver of Figure 6. As described in [33], the FPTD comprises two rows of N concurrently-operated Processing Elements (PEs), allowing it to process all $N \log_2 M$ LLRs provided in each clock cycle, in contrast to a conventional turbo decoder. Each PE uses registers $\boxed{\triangle}$ to iteratively exchange LLRs with its neighbouring PEs in the same row, as well as with a corresponding PE in the other row, via a second interleaver. The quality of the iteratively exchanged LLRs improves in each clock cycle, until final LLR decisions are obtained for the N bits, with K being the number of message bits, using additions $\oplus$ in the final clock cycle.

IV. Validation

In this section, we validate the correctness of our cumulative FFT and ICI-aware soft QAM demapper by confirming that the resultant LLRs satisfy the consistency condition given in [47]. More specifically, two methods for measuring the Mutual Information (MI) of LLRs are proposed in [47], referred to as the averaging method and the histogram-based method. While the histogram method computes the MI of LLRs by comparing them to the correct bit values, the averaging method does not consider the correct bit values. Instead, it assumes that the LLRs satisfy the consistency condition and that the MI can be correctly computed based on the magnitudes of the LLRs alone. If a vector of LLRs satisfies the consistency condition, then the averaging method will measure the same MI value as the histogram method, which does not assume consistency. Figure 7 illustrates the employment of the averaging and histogram methods of calculating the MI to validate our proposed approach. The results of the comparisons are shown in Figures 8 and 9, where 4QAM and 16QAM are employed, respectively. As shown in Figures 8 and 9, both methods of measuring the MI give similar results across a variety of different E_b/N_0 values, and as successively more symbols D are received. This confirms that the LLRs satisfy the consistency condition and validates the accuracy of the proposed approach. As expected, higher E_b/N_0 values and more received symbols D result in higher LLR reliability, whereas higher-order QAM decreases the LLR reliability, at a given E_b/N_0 value.

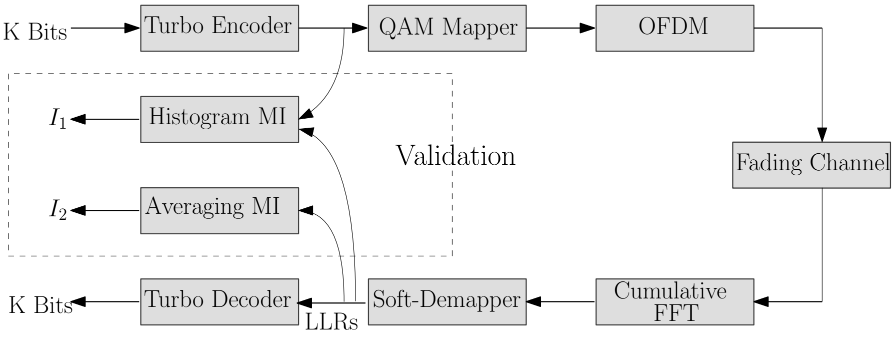

Fig. 7. Validation using the histogram and averaging methods of MI calculation.

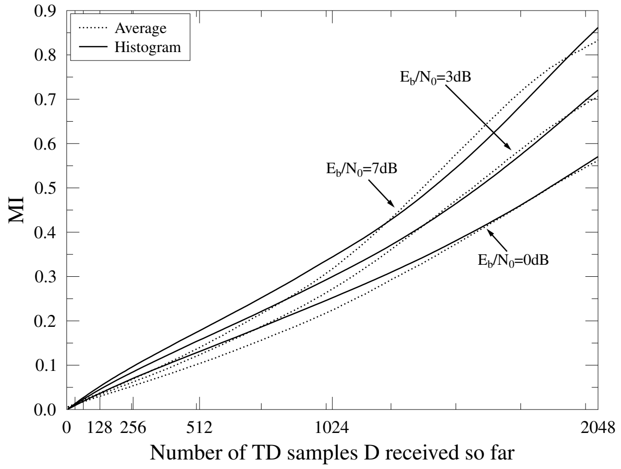

Fig. 8. MI calculated by histogram and averaging methods, when employing a punctured LTE turbo code, Gray-coded 4QAM, OFDM and a quasi-static ETU Rayleigh fading channel, where $K = 1376$ and $N = 2048$.

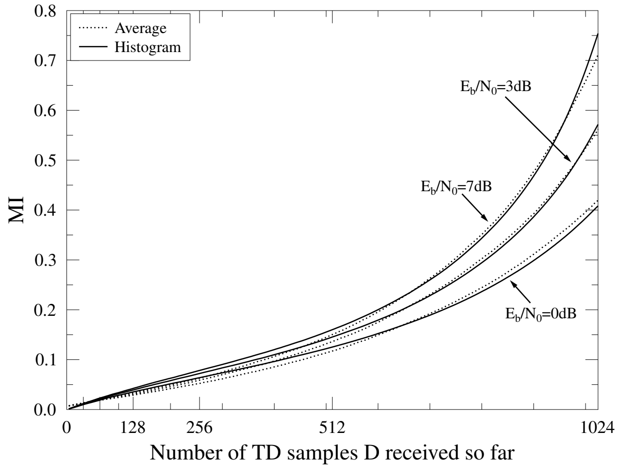

Fig. 9. MI calculated by histogram and averaging methods, when employing a punctured LTE turbo code, Gray-coded 16QAM, OFDM and a quasi-static ETU Rayleigh fading channel, where $K = 1376$ and $N = 1024$.

V. Performance analysis

In this section, we present and benchmark the performance of the proposed concurrent OFDM demodulation and turbo decoding architecture. Figure 10 characterises the performance of the proposed scheme and that of a benchmarker for the case of a punctured LTE turbo code, Gray-coded QAM, OFDM and a quasi-static Extended Typical Urban model (ETU) Rayleigh fading channel [48], where $K = 1376$, $N = 2048$ for 4QAM and $N = 1024$ for 16QAM. Here, the concurrent receive, FFT and turbo decode approach of Figure 1(b) employs the architecture proposed in Figure 6 and $C \in \{128, 256, 512, 1024\}$ clock cycles. This is compared to a benchmarker employing the serial receive, FFT and turbo decode approach of Figure 1(a), when employing a conventional turbo decoder. As the number of clock cycles C is increased, the Bit Error Ratio (BER) performance of the proposed scheme can be seen to converge to that of the benchmarker, proving the concept of the concurrent receive, FFT and turbo decode approach. A similar result may be observed when employing the higher-order QAM, where higher bandwidth efficiency is obtained at the cost of degraded BER performance. However, for the 4QAM modulation scheme, the proposed architecture requires up to $C = 1024$ clock cycles in order to closely match the performance of the benchmarker, while even more clock cycles are required for the 16QAM scheme to approach the benchmarker. A significant reduction in processing energy consumption could be achieved upon employing only $C = 128$ clock cycles, although this is associated with a performance loss of up to 2.5 dB, compared to the benchmarker.

In order to mitigate this performance loss, the architecture of Figure 6 may be further refined. In a first refinement, referred to as the staggered receive, FFT and turbo decode approach, the operation of the FPTD may be staggered relative to that of the cumulative FFT of Figure 6. More specifically, the operation of the FPTD maybe delayed until $S \in [0, N]$ symbols have been received, facilitating a gradated trade-off between latency and processing energy consumption. Figure 11 shows that upon adopting this approach, $C = 128$ clock cycles and a stagger of $S = 3N/8$ symbols is sufficient for closely matching the BER performance of the benchmarker, when employing both 4QAM and 16QAM.

In a second refinement, referred to as the scaled concurrent receive FFT and turbo decode approach, the BER performance loss can also be mitigated by reducing the weighting of the

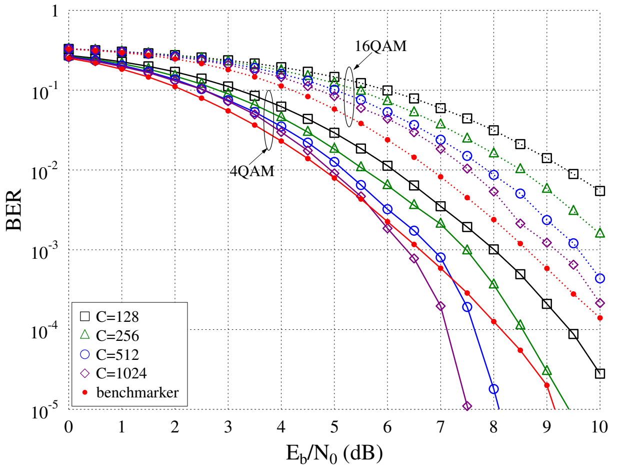

Fig. 10. BER of the proposed concurrent architecture, when using $C \in \{128,$ 256, 512, 1024, 2048$\}$ and employing a punctured LTE turbo code, Gray-coded QAM, OFDM and a quasi-static ETU Rayleigh fading channel, where $K = 1376$, $N = 2048$ for 4QAM and $N = 1024$ for 16QAM.

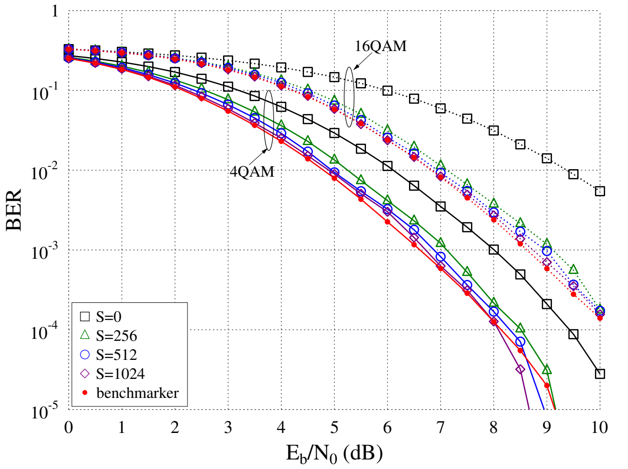

Fig. 11. BER of the staggered architecture, when using $C = 128$ and $S \in \{0, 256, 512, 1024\}$, employing a punctured LTE turbo code, Gray-coded QAM, OFDM and a quasi-static ETU Rayleigh fading channel, where $K = 1376$, $N = 2048$ for 4QAM and $N = 1024$ for 16QAM.

ICI-dominated LLRs provided by the soft demappers during the early clock cycles. This can be achieved by applying a gradually increasing scaling factor imposed on the LLRs in successive clock cycles according to an exponential function $y = \exp x$. Figure 12 shows that decreasing the weighting of LLRs provided in early cycles improves the overall BER performance by around 0.2 dB.

By designing the proposed architecture of Figure 6 to implement the staggered and/or scaled receive, FFT and turbo decode approach using $C = 128$ and a clock frequency of at least 176 MHz, sub-microsecond physical layer latencies will be facilitated for applications such as low-latency capital market trading or robotic cars. In applications such at LTE, where a transmission latency of 70 μs is imposed, the operation of the cumulative FFT and the FPTD can be spread over this duration, in order to achieve a significantly improved hardware resource efficiency or processing energy consumption.

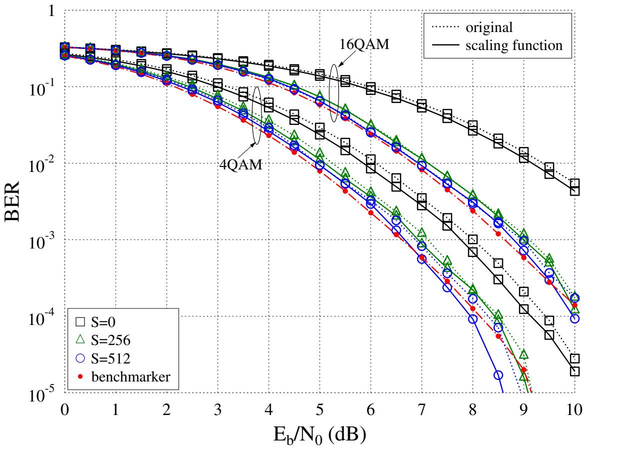

Fig. 12. BER of the proposed architecture with an exponential scaling factor function, when using C = 128 and $S \in \{0,\ 256,\ 512\}$ and employing a punctured LTE turbo code, Gray-coded QAM, OFDM and a quasi-static ETU Rayleigh fading channel, where $K = 1376$, $N = 2048$ for 4QAM and $N = 1024$ for 16QAM.

VI. Conclusions

In this paper, we proposed a concurrent OFDM demodulation and turbo decoding architecture employing a novel cumulative FFT technique for significantly reducing the associated processing latency. Rather than completing the reception, FFT and turbo decoding operations one after another, these three processes are performed concurrently in our proposed approach. In this way, the overall receiving, demodulation and decoding latency is only a third of that of the conventional serial architecture, which makes it promising application for URLLC. In order to allow the trade off between latency and complexity to be adjusted and to improve the associated BER performance, we also proposed staggered and scaled refinements to the proposed approach.

Our future work will investigate the performance of the proposed architecture in the case of mobile terminals such as autonomous vehicles, where time-varying channels and Doppler shift must be considered. The extension of the proposed architecture to other transmission schemes, such as LDPC-coded modulation and diverse multicarrier communications schemes will also be explored. It may be expected that these advanced techniques will enable further enhancements to the proposed technique.

Appendix A
Derivation of (13)

$$
\begin{aligned}
Y_{z,c} &= H_z \frac{1}{N} \sum_{k=0}^{N-1} \sum_{n=0}^{N-1} X_n W_N^{nk} d_k W_N^{-kz} + N_z \\
&= H_z \frac{1}{N} \sum_{k=0}^{N-1} \sum_{n=0}^{N-1} X_n W_N^{(n-z)k} d_k + N_z \\
&= H_z \frac{1}{N} \sum_{n=0}^{N-1} X_n \sum_{k=0}^{N-1} \left[W_N^{(n-z)k} d_k \right] + N_z \\
&= H_z \frac{1}{N} X_z \sum_{k=0}^{N-1} \left[W_N^{0} d_k \right] \\
&+ H_z \frac{1}{N} \sum_{n=0, n\neq z}^{N-1} X_n \sum_{k=0}^{N-1} \left[W_N^{(n-z)k} d_n \right] + N_z \\
&= \underbrace{H_z \frac{D}{N} X_z}_{\text{what we expect}} + \underbrace{H_z \frac{1}{N} \sum_{n=0, n\neq z}^{N-1} X_n \sum_{k=0}^{D-1} \left[W_N^{(n-z)k} \right]}_{\text{ICI}} + \underbrace{N_z}_{\text{noise}} \\
&= \frac{D}{N} \left(H_z X_z + \frac{H_z}{D} \sum_{n=0, n\neq z}^{N-1} \sum_{k=0}^{D-1} X_n W_N^{(n-z)k} + \frac{N}{D} N_z \right) \\
&= \frac{D}{N} \left(H_z X_z + \underbrace{\frac{H_z}{D} \sum_{n=0, n\neq z}^{N-1} \sum_{k=0}^{N-1} X_n W_N^{(n-z)k}}_{=0} \right. \\
&\left. - \frac{H_z}{D} \sum_{n=0, n\neq z}^{N-1} \sum_{k=D}^{N-1} X_n W_N^{(n-z)k} + \frac{N}{D} N_z \right) \\
&= \frac{D}{N} \left(H_z X_z - \frac{H_z}{D} \sum_{n=0, n\neq z}^{N-1} \sum_{k=D}^{N-1} X_n W_N^{(n-z)k} + \frac{N}{D} N_z \right)
\end{aligned}
$$

References

[1] C. Hoymann, D. Astely, M. Stattin, G. Wikstrom, J.-F. Cheng, A. Hoglund, M. Frenne, R. Blasco, J. Huschke, and F. Gunnarsson, "LTE release 14 outlook," *IEEE Communications Magazine*, vol. 54, no. 6, pp. 44–49, 2016.

[2] P. Guan, D. Wu, T. Tian, J. Zhou, X. Zhang, L. Gu, A. Benjebbour, M. Iwabuchi, and Y. Kishiyama, "5G field trials: OFDM-based waveforms and mixed numerologies," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1234–1243, 2017.

[3] K. David and H. Berndt, "6G vision and requirements: Is there any need for beyond 5G?," *IEEE Vehicular Technology Magazine*, vol. 13, no. 3, pp. 72–80, 2018.

[4] H. Shariatmadari, S. Iraji, R. Jantti, P. Popovski, Z. Li, and M. A. Uusitalo, "Fifth-generation control channel design: Achieving ultrareliable low-latency communications," *IEEE Vehicular Technology Magazine*, vol. 13, no. 2, pp. 84–93, 2018.

[5] A. Aissioui, A. Ksentini, A. M. Gueroui, and T. Taleb, "On enabling 5G automotive systems using follow me edge-cloud concept," *IEEE Transactions on Vehicular Technology*, vol. 67, pp. 5302–5316, June 2018.

[6] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2307–2359, 2010.

[7] M. Condoluci, M. Dohler, G. Araniti, A. Molinaro, and K. Zheng, "Toward 5G densenets: architectural advances for effective machine-type communications over femtocells," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 134–141, 2015.

[8] M. Luvisotto, Z. Pang, and D. Dzung, "Ultra high performance wireless control for critical applications: challenges and directions," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1448–1459, 2017.

[9] P.-H. Chiu, P.-H. Tseng, and K.-T. Feng, "Interactive mobile augmented reality system for image and hand motion tracking," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 10, pp. 9995–10009, 2018.

[10] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned aerial vehicle with underlaid device-to-device communications: performance and tradeoffs," *IEEE Transactions on Wireless Communications*, vol. 15, no. 6, pp. 3949–3963, 2016.

[11] M. Bennis, M. Debbah, and H. V. Poor, "Ultra-reliable and low-latency wireless communication: Tail, risk and scale," *arXiv preprint arXiv:1801.01270*, 2018.

[12] M. S. Elbamby, C. Perfecto, M. Bennis, and K. Doppler, "Toward low-latency and ultra-reliable virtual reality," *IEEE Network*, vol. 32, no. 2, pp. 78–84, 2018.

[13] G. P. Fettweis, "The tactile internet: applications and challenges," *IEEE Vehicular Technology Magazine*, vol. 9, no. 1, pp. 64–70, 2014.

[14] B. Soret, P. Mogensen, K. I. Pedersen, and M. C. Aguayo-Torres, "Fundamental tradeoffs among reliability, latency and throughput in cellular networks," in *Globecom Workshops (GC Wkshps), 2014*, pp. 1391–1396, IEEE, 2014.

[15] D. Schneider, "The microsecond market," *IEEE spectrum*, vol. 6, no. 49, pp. 66–81, 2012.

[16] M. Simsek, A. Aijaz, M. Dohler, J. Sachs, and G. Fettweis, "5G-enabled tactile internet," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 3, pp. 460–473, 2016.

[17] G. P. Fettweis, "A 5G wireless communications vision," *Microwave Journal*, vol. 55, no. 12, pp. 24–36, 2012.

[18] TSGRANGRA, Network, "Evolved universal terrestrial radio access (E-UTRA); multiplexing and channel coding," *3rd Generation Partnership Project (3GPP)*, vol. TS 36, 2009.

[19] L. Hanzo, M. Münster, B. Choi, and T. Keller, *OFDM and MC-CDMA for broadband multi-user communications, WLANs and broadcasting*. John Wiley & Sons, 2005.

[20] Access, Evolved Universal Terrestrial Radio, "Multiplexing and channel coding," *3rd Generation Partnership Project Std. 3GPP*, vol. TS 36, p. V8, 2008.

[21] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Transactions on communications*, vol. 44, no. 10, pp. 1261–1271, 1996.

[22] C. Berrou and A. Glavieux, "Turbo codes," *Encyclopedia of Telecommunications*, 2003.

[23] L. Xu, J. Yang, D. Huang, and A. Cantoni, "Exploiting cyclic prefix for Turbo-OFDM receiver design," *IEEE Access*, vol. 5, pp. 15762–15775, 2017.

[24] I. Shubhi and Y. Sanada, "Joint turbo decoding for overloaded MIMO-OFDM systems," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 433–442, 2017.

[25] J. C. S. Arenas, T. Dudda, and L. Falconetti, "Ultra-low latency in next generation LTE radio access," in *SCC 2017; 11th International ITG Conference on Systems, Communications and Coding; Proceedings of*, pp. 1–6, VDE, 2017.

[26] T. Fehrenbach, R. Datta, B. Göktepe, T. Wirth, and C. Helge, "URLLC services in 5G-low latency enhancements for LTE," *Accepted for publication at IEEE Vehicular Technology Conference (VTC)*, Fall 2018.

[27] L. Xiang, M. F. Brejza, R. G. Maunder, B. M. Al-Hashimi, and L. Hanzo, "Arbitrarily parallel turbo decoding for ultra-reliable low latency communication in 3GPP LTE," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 4, pp. 826–838, 2019.

[28] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex fourier series," *Mathematics of computation*, vol. 19, no. 90, pp. 297–301, 1965.

[29] W. T. Cochran, J. W. Cooley, D. L. Favin, H. D. Helms, R. A. Kaenel, W. W. Lang, G. Maling, D. E. Nelson, C. M. Rader, and P. D. Welch, "What is the fast fourier transform?," *Proceedings of the IEEE*, vol. 55, no. 10, pp. 1664–1674, 1967.

[30] D. Soldani, Y. J. Guo, B. Barani, P. Mogensen, I. Chih-Lin, and S. K. Das, "5G for ultra-reliable low-latency communications," *IEEE Network*, vol. 32, no. 2, pp. 6–7, 2018.

[31] B. Tahir, S. Schwarz, and M. Rupp, "BER comparison between convolutional, turbo, LDPC, and polar codes," in *Telecommunications (ICT), 2017 24th International Conference on*, pp. 1–7, IEEE, 2017.

[32] T. Richardson and S. Kudekar, "Design of low-density parity check codes for 5G new radio," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 28–34, 2018.

[33] R. G. Maunder, "A fully-parallel turbo decoding algorithm," *IEEE Transactions on Communications*, vol. 63, no. 8, pp. 2762–2775, 2015.

[34] S. Weinstein and P. Ebert, "Data transmission by frequency-division multiplexing using the discrete fourier transform," *IEEE transactions on Communication Technology*, vol. 19, no. 5, pp. 628–634, 1971.

[35] S. Darlington, "On digital single-sideband modulators," *IEEE Transactions on Circuit Theory*, vol. 17, no. 3, pp. 409–414, 1970.

[36] B. Hirosaki, "An orthogonally multiplexed QAM system using the discrete fourier transform," *IEEE Transactions on Communications*, vol. 29, no. 7, pp. 982–989, 1981.

[37] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions on Information Theory*, vol. 20, no. 2, pp. 284–287, 1974.

[38] S. Yoon and Y. Bar-Ness, "A parallel MAP algorithm for low latency turbo decoding," *IEEE Communications Letters*, vol. 6, no. 7, pp. 288–290, 2002.

[39] C. Studer, C. Benkeser, S. Belfanti, and Q. Huang, "Design and implementation of a parallel turbo-decoder ASIC for 3GPP-LTE," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 8–17, 2011.

[40] H. Luo, Y. Zhang, W. Li, L.-K. Huang, J. Cosmas, D. Li, C. Maple, and X. Zhang, "Low latency parallel turbo decoding implementation for future terrestrial broadcasting systems," *IEEE Transactions on Broadcasting*, vol. 64, no. 1, pp. 96–104, 2018.

[41] A. Li, L. Xiang, T. Chen, R. G. Maunder, B. M. Al-Hashimi, and L. Hanzo, "VLSI implementation of fully parallel LTE turbo decoders," *IEEE Access*, vol. 4, pp. 323–346, 2016.

[42] E. Boutillon, W. J. Gross, and P. G. Gulak, "VLSI architectures for the MAP algorithm," *IEEE Transactions on Communications*, vol. 51, no. 2, pp. 175–185, 2003.

[43] J. Sun and O. Y. Takeshita, "Interleavers for turbo codes using permutation polynomials over integer rings," *IEEE Transactions on Information Theory*, vol. 51, no. 1, pp. 101–119, 2005.

[44] O. Y. Takeshita, "On maximum contention-free interleavers and permutation polynomials over integer rings," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1249–1253, 2006.

[45] L. Piazzo and P. Mandarini, "Analysis of phase noise effects in ofdm modems," *IEEE Transactions on Communications*, vol. 50, no. 10, pp. 1696–1705, 2002.

[46] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Communications, 1995. ICC'95 Seattle,'Gateway to Globalization', 1995 IEEE International Conference on*, vol. 2, pp. 1009–1013, IEEE, 1995.

[47] J. Hagenauer, "The EXIT chart-introduction to extrinsic information transfer in iterative processing," in *Signal Processing Conference, 2004 12th European*, pp. 1541–1548, IEEE, 2004.

[48] Access, Evolved Universal Terrestrial Radio, "Base station (BS) radio transmission and reception," *3GPP TS 36.104 V14*, vol. 3, 2009.