Look-up Table Based Deferred-Iteration Aided Low-Complexity Turbo Hybrid ARQ

Hong Chen, Robert G. Maunder, *Member, IEEE* and Lajos Hanzo, *Fellow, IEEE*

***Abstract*—Since the introduction of turbo code aided Hybrid Automatic Repeat reQuest (HARQ) schemes, their complexity reduction has drawn research attention. In our previous work, we proposed an Early Stopping (ES) strategy for a turbo HARQ scheme, which results in a beneficial complexity reduction, while maintaining a high throughput. However, this scheme was designed to strike a compromise across the full range of channel conditions. As a further advance, in this paper, we propose a new Deferred Iterations (DI) strategy, which is specifically designed for taking into account the prevalent channel conditions, as characterized by the Extrinsic Information Transfer (EXIT) chart tunnel opening. More specifically, the DI strategy delays the commencement of turbo decoding until an open EXIT chart tunnel appears. Our simulation results demonstrate that the complexity of the proposed DI aided turbo HARQ schemes is reduced by up to 50%, which is achieved without compromising the Packet Loss Ratio (PLR) or throughput.**

I. Introduction

Turbo codes [1]–[3] are characterized by an iterative exchange of increasingly reliable soft information between the constituent Bahl, Cocke, Jelinek and Raviv (BCJR) [4] decoders, which are concatenated in parallel and separated by an interleaver. Owing to their near-capacity performance, turbo codes can be successfully combined with Hybrid Automatic Repeat reQuest (HARQ) schemes [5], [6], in order to achieve a high throughput. In these turbo HARQ schemes, the transmitter continually transmits turbo-encoded Incremental Redundancy (IR) to the receiver, where BCJR decoding operations may be performed iteratively following the reception of each transmission. Once the message is decoded successfully, the receiver returns an ACKnowledgement (ACK) to the transmitter, in order to cease its transmission of IR.

In general, there is a trade-off between the achievable throughput and the complexity imposed by turbo HARQ schemes. Naturally, the complexity is increased when more than necessary BCJR iterations are performed during the iterative decoding process following the reception of each IR transmission. For example, the early turbo HARQ schemes [6], [7] performed a sufficiently high number of BCJR decoder executions following each and every IR transmission in order to ensure that iterative decoding convergence had been achieved. In this way, they minimized the number of IR transmissions required, hence maximizing the throughput, at the cost of imposing an excessive complexity. On the other hand, the attainable throughput degrades, when sufficient IR contributions have been received for facilitating error-free decoding, but insufficient BCJR iterations have been performed. Our previous solution [8] struck an attractive tradeoff by proposing an Early Stopping (ES) strategy for a turbo HARQ scheme. This ES strategy [8] entirely curtails the iterative decoding process, when the rate of iterative Mutual Information (MI) improvement becomes lower than a pre-determined threshold. Since this approach eradicated unnecessary decoding iterations, the scheme of [8] exhibited a significantly lower complexity than those of [6], [7], without unduly compromising the attainable throughput.

Like the schemes of [6], [7], that of [8] activates iterative decoding following the reception of each transmission. However, a significant further decoding complexity reduction may be expected by complementing the ES strategy of [8] with the novel approach that we propose in this paper, namely by Deferred Iterations (DI). This approach defers the commencement of iterative decoding, until sufficient IR contributions have been received to offer a sufficiently high likelihood of error-free decoding. This may be determined with the aid of the turbo decoder's EXtrinsic Information Transfer (EXIT) chart [9] following the reception of each IR transmission. More specifically, for long packets, having an open EXIT tunnel implies that a sufficient amount of IR has been received to allow the iterative decoding trajectory to reach the $(1, 1)$ point of perfect convergence in the EXIT chart, where a vanishingly low Bit Error Ratio (BER) is achieved. By contrast, having a closed tunnel implies that more IR is required before the decoding iterations should be commenced. The authors of [10] suggested to immediately cease the decoding iterations at Signal Noise Ratios (SNRs) below a certain threshold value given by that particular SNR, where decoding to convergence was 'just' possible for transmissions over an Additive White Gaussian Noise (AWGN) channel. Namely, the EXIT tunnel becomes marginally open at this threshold SNR, which is a unique SNR for transmissions over the AWGN channel. However, they did not consider the corresponding thresholds when transmitting over the quasi-static Rayleigh fading channels employed in [8], and did not exploit the supplemental information provided by IR transmissions in HARQ schemes. In this paper, our approach determines the EXIT chart tunnel's state by using the MI of the IR received so far in order to consult a Look-

Acknowledgements: The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement No. 214625. The financial support of the China-UK Scholarship Council, of the RC-UK under the IU-ATC as well as the China-UK Science Bridge in 4G wireless communication initiative is also gratefully acknowledged.

Hong Chen is a Lecturer with the School of Computer Science and Engineering, University of Electronic Science and Technology of China. She is currently working toward the Ph.D. degree with the School of Electronics and Computer Science, University of Southampton, Southampton, U.K. (e-mail: hc07r@ecs.soton.ac.uk).

Robert G. Maunder and Lajos Hanzo are with the School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K. (e-mail: rm@ecs.soton.ac.uk; lh@ecs.soton.ac.uk).

Up Table (LUT). The novelty of this paper may be detailed as follows:

- We conceive a LUT which requires only a small amount of off-line training and storage. This is achieved by employing a novel semi-analytic design procedure, which avoids time-consuming Monte Carlo simulations. Furthermore, we exploit the gradually evolving nature of the EXIT functions as the channel conditions fluctuate for the sake of minimizing the complexity of the search required for determining the threshold, at which an open EXIT tunnel emerges. Finally, we propose a novel method for exploiting the potential redundancy in the LUT, in order to minimize its size.
- Based on the proposed LUT, our new DI strategy can be applied in its own right to turbo HARQ schemes. However, it may also be additionally combined with the ES strategy of [8] for the sake of achieving a reduced complexity;
- Additionally, we conceive special measures to cater for short packets, for which the Monte-Carlo-decoding trajectory might in fact reach the point of perfect convergence at $(1, 1)$ in the EXIT chart, even when the EXIT tunnel becomes 'just' closed. By contrast, sometimes the $(1, 1)$ point is not reached by the trajectory, even though the EXIT tunnel is open [11]. This inaccuracy is a consequence of failing to generate independent LLRs due to the insufficient packet length.

The proposed DI technique is generically applicable, but in our design example we apply the above-mentioned LUT based DI strategy to turbo code aided HARQ schemes. Importantly, the proposed idea may also be extended to other diverse scenarios, which rely on iterative receivers. For instance in relay networks, a relay node may decide whether it should or should not continue the decoding and transmission of the source's message according to the LUT. Therefore, in Section II-A of this paper, we first highlight the DI philosophy in the context of classic regular Twin-Component Turbo Codes (TCTC), then extend it to Multiple-Component Tubo Code (MCTC) aided HARQ schemes, which has been shown to have an attractive performance even when using the minimal possible constraint length of 2. Hence, Section II also introduces several methods of designing an efficient LUT for MCTC HARQ schemes. Then in Section III, we apply the proposed LUT based DI strategy for further reducing the complexity of MCTC HARQ schemes. Section IV compares the Packet Loss Ratio (PLR), the throughput and the complexity of a suitably parameterized version of our scheme to those of appropriately chosen benchmarkers. Finally, Section V offers our conclusions.

II. Look-up Table Design for MCTC HARQ Schemes

In this section, we firstly discuss the DI philosophy and demonstrate that the LUT size is potentially huge for MCTC HARQ schemes. In order to facilitate an efficient design, the following subsections aim for minimizing the LUT size and speed up the training. In Section II-B, we conceive the proposed LUT structure specifically designed for the appropriately sorted MIs and invoke multiple-dimensional linear interpolation for generating its high granularity. Section II-C proposes a fast training procedure for generating the LUT. Section II-D analyzes the memory requirements of the resultant LUT designed for the MCTC HARQ scheme considered.

A. The Deferred Iteration Philosophy

Firstly, let us introduce our LUT concept for the rudimentary example of a classic TCTC using two parallel concatenated accumulators[1], while dispensing with HARQ assistance.

In a non-systematic TCTC encoder, one of the component encoders is employed to convert the information bit sequence $\mathbf{a}_1$ into the parity bit sequence $\mathbf{b}_1$, while the other is used for converting an interleaved version $\mathbf{a}_2$ of $\mathbf{a}_1$ into a second parity bit sequence $\mathbf{b}_2$. The transmitter then conveys these parity bit sequences to the receiver, where a soft demodulator [2] generates the corresponding Logarithmic Likelihood Ratio (LLR) sequences $\tilde{\mathbf{b}}_1$ and $\tilde{\mathbf{b}}_2$. In this paper, we incorporate diacritical tildes into our notation to denote the vectors of LLRs. These vectors of $\tilde{\mathbf{b}}_1$ and $\tilde{\mathbf{b}}_2$ are forwarded to two component decoders, which opt for iteratively exchanging the extrinsic LLR sequences $\tilde{\mathbf{a}}_1^{\mathrm{e}}$ and $\tilde{\mathbf{a}}_2^{\mathrm{e}}$. Following interleaving, these sequences become the *a priori* LLR sequences $\tilde{\mathbf{a}}_2^{\mathrm{a}}$ and $\tilde{\mathbf{a}}_1^{\mathrm{a}}$ [2]. This process may be characterized by a two-dimensional EXIT chart [9], comprising the EXIT functions $I(\tilde{\mathbf{a}}_1^{\mathrm{e}}) = f_1[I(\tilde{\mathbf{a}}_1^{\mathrm{a}}), I(\tilde{\mathbf{b}}_1)]$ and $I(\tilde{\mathbf{a}}_2^{\mathrm{e}}) = f_2[I(\tilde{\mathbf{a}}_2^{\mathrm{a}}), I(\tilde{\mathbf{b}}_2)]$. Here, the MI $I(\tilde{\mathbf{x}})$ quantifies the reliability of the information in the sequence of N LLRs $\tilde{\mathbf{x}} = \{\tilde{x}_j\}_{j=1}^{N}$ pertaining to the particular values of the N bits in the corresponding sequence $\mathbf{x} = \{x_j\}_{j=1}^{N}$, according to [12][2]

$$I(\tilde{\mathbf{x}}) \approx 1 - \frac{1}{N}\sum_{j=1}^{N} H_{\mathrm{b}}\left(\frac{e^{+|\tilde{x}_j|/2}}{e^{+|\tilde{x}_j|/2} + e^{-|\tilde{x}_j|/2}}\right), \quad (1)$$

where H_{b} represents the binary entropy function. In a quasi-static Rayleigh fading channel, consecutive IR transmissions may have MIs that are distributed over the entire legitimate MI range of $[0, 1]$. In general, the MI values may be quantized discretely with a granularity or step-size of $G = 0.01$ without a sacrifice in precision.

However, the receiver may refrain from activating the iterative decoding process, if it deems the MI of the LLR sequences to be insufficient for creating an open tunnel in the EXIT chart [2] [9], indicating that convergence towards the infinitesimally low BER is prevented. We assume that the receiver uses Equation (1) to determine the MI between the estimated hard-decision-based bits and the channel's output LLRs $\tilde{\mathbf{b}}_1$, which is for example $I(\tilde{\mathbf{b}}_1) = 0.15$. In the case where unity-rate accumulators are employed as component codes, this MI corresponds to the EXIT function $I(\tilde{\mathbf{a}}_1^{\mathrm{e}}) = f_1[I(\tilde{\mathbf{a}}_1^{\mathrm{a}}), 0.15]$ of Figure 1.

This EXIT chart analysis reveals that if the channel's output LLRs $\tilde{\mathbf{b}}_2$ have an MI of at least $I(\tilde{\mathbf{b}}_2) = 0.96$, then an open EXIT chart tunnel will be created between $I(\tilde{\mathbf{a}}_1^{\mathrm{e}})$ and the EXIT

[1]We define an accumulator to be a unity-rate recursive convolutional code having feedforward and feedback generator polynomials of $(2, 3)_{\mathrm{o}}$ respectively, which are expressed in an octal representation.

[2]For simplicity of notation, we use $I(\tilde{\mathbf{x}})$ to replace $I(\tilde{\mathbf{x}}, \mathbf{x})$, which more explicitly denotes the MI between the LLRs $\tilde{\mathbf{x}}$ and the corresponding bit sequence $\mathbf{x}$.

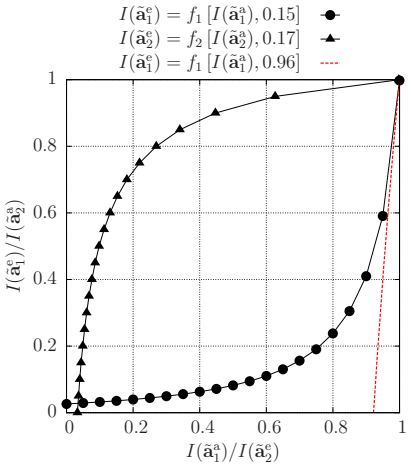


Figure 1. The EXIT chart of a TCTC using two parallel concatenated accumulators.

function $I(\tilde{\mathbf{a}}_2^{\mathrm{e}}) = f_2[I(\tilde{\mathbf{a}}_2^{\mathrm{a}}), 0.96]$. This threshold MI value of $I(\tilde{\mathbf{b}}_2) = 0.96$ could be revealed to the receiver by rounding the value of $I(\tilde{\mathbf{b}}_1) = 0.15$ to two decimal places and using it to index an LUT comprising $(\frac{1}{G} + 1) = 101$ entries, spanning the MI range of $I(\tilde{\mathbf{b}}_1) \in [0, 1]$. However, Equation (1) may actually suggest that the LLRs of $\tilde{\mathbf{b}}_2$ have an MI as low as $I(\tilde{\mathbf{b}}_2) = 0.17$, which is below the MI threshold of 0.96 and corresponds to the EXIT function of $I(\tilde{\mathbf{a}}_2^{\mathrm{e}}) = f_2[I(\tilde{\mathbf{a}}_2^{\mathrm{a}}), 0.17]$ in Figure 1. Using the LUT, the receiver becomes aware that the EXIT chart tunnel is closed and can therefore defer attempting the recovery of the information bit sequence $\mathbf{a}_1$, since its decoding attempt is very unlikely to be successful and yet, it would impose a certain complexity and power drain.

Let us now consider the application of our LUT technique to an MCTC HARQ scheme. A non-systematic MCTC encoder [13] comprises r number of component encoders, each of which $i \in \{1, 2, \ldots, r\}$ generates a parity bit sequence $\mathbf{b}_i$ by encoding a differently interleaved version $\mathbf{a}_i$ of the information bit sequence $\mathbf{a}_1$ (or use $\mathbf{a}_1$ itself for $i = 1$). In HARQ applications, the number of component encoders r is successively incremented and the corresponding parity bit sequences are successively transmitted until the number of transmissions r reaches a specified maximum IR limit R, or until the receiver acknowledges that the information bit sequence $\mathbf{a}_1$ has been successfully recovered. As each transmission is received at the receiver, the corresponding LLR sequence $\tilde{\mathbf{b}}_r$ is forwarded to a component decoder that is concatenated to those corresponding to the previous $(r-1)$ transmissions. At this point, the receiver may opt for iteratively exchanging the extrinsic LLR sequences $\tilde{\mathbf{a}}_i^{\mathrm{e}}$ among the r component decoders, each of which interleaves and sums the extrinsic LLRs provided by the other decoders, in order to generate the *a priori* LLRs $\tilde{\mathbf{a}}_i^{\mathrm{a}}$. This process may be characterized by an r-dimensional EXIT chart, comprising the r EXIT functions $I(\tilde{\mathbf{a}}_i^{\mathrm{e}}) = f_i[I(\tilde{\mathbf{a}}_i^{\mathrm{a}}), I(\tilde{\mathbf{b}}_i)]$. Alternatively, this process may be characterized by a two-dimensional EXIT chart, comprising the EXIT function $I(\tilde{\mathbf{a}}_r^{\mathrm{e}}) = f_r[I(\tilde{\mathbf{a}}_r^{\mathrm{a}}), I(\tilde{\mathbf{b}}_r)]$ and the composite EXIT function $I(\tilde{\mathbf{a}}_r^{\mathrm{a}}) = f_{\{1,2,\ldots,r-1\}}[I(\tilde{\mathbf{a}}_r^{\mathrm{e}}), I(\tilde{\mathbf{b}}_1), I(\tilde{\mathbf{b}}_2), \ldots, I(\tilde{\mathbf{b}}_{r-1})]$ [13]. This composite EXIT function characterizes the provision of the r^{th} decoder's extrinsic LLRs $\tilde{\mathbf{a}}_r^{\mathrm{e}}$ for the iterative operation of the other $(r-1)$ decoders until convergence is achieved, whereupon the sum of their extrinsic LLRs are provided for the r^{th} decoder for employment as $\tilde{\mathbf{a}}_r^{\mathrm{a}}$, namely as the *a priori* information.

Extending the above example to our MCTC HARQ design example, when the EXIT chart is deemed to be closed for the pair of received MIs contributions $I(\tilde{\mathbf{b}}_1) = 0.15$ and $I(\tilde{\mathbf{b}}_2) = 0.17$, the receiver would opt for deferring the start of the iterative decoding process by returning no acknowledgement to the transmitter and waiting for it to supply more IR. When unity-rate accumulators are employed as the component codes, the MIs $I(\tilde{\mathbf{b}}_1) = 0.15$ and $I(\tilde{\mathbf{b}}_2) = 0.17$ correspond to the composite EXIT function $I(\tilde{\mathbf{a}}_3^{\mathrm{a}}) = f_{\{1,2\}}[I(\tilde{\mathbf{a}}_3^{\mathrm{e}}), 0.15, 0.17]$ of Figure 2.

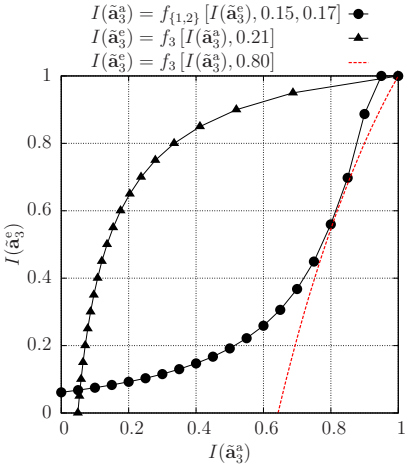


Figure 2. The EXIT chart of a 3-component MCTC.

This EXIT chart analysis reveals that if the channel's output LLRs $\tilde{\mathbf{b}}_3$ supplied by the $r = 3^{\mathrm{rd}}$ transmission have an MI of at least $I(\tilde{\mathbf{b}}_3) = 0.80$, then an open EXIT chart tunnel will be created with the EXIT function $I(\tilde{\mathbf{a}}_3^{\mathrm{e}}) = f_3[I(\tilde{\mathbf{a}}_3^{\mathrm{a}}), 0.80]$. In this case, there is a high probability that the information bit sequence $\mathbf{a}_1$ can indeed be successfully recovered by activating the iterative decoding process. If this is not the case, e.g. we have $I(\tilde{\mathbf{b}}_3) = 0.21$ as seen in Figure 2, then the start of the iterative decoding process can be further deferred, until an open EXIT chart tunnel is deemed to have been created. Here, the threshold value of $I(\tilde{\mathbf{b}}_3) = 0.80$ could be revealed to the receiver by rounding the values of $I(\tilde{\mathbf{b}}_1) = 0.15$ and $I(\tilde{\mathbf{b}}_2) = 0.17$ to two decimal places and using them as the address of an LUT comprising $(101)^2$ entries. This example demonstrates that a naive LUT implementation for an r-component MCTC decoder requires $(101)^{r-1}$ entries and that since r is successively incremented in MCTC HARQ schemes, a total of $\sum_{r=2}^{R}(101)^{r-1}$ entries is required, when a transmission limit of R is imposed. However, in Section II-B and II-C, we will propose a sophisticated LUT design, which significantly reduces the number of LUT entries that must be trained and stored. In this way, a practical DI scheme is devised for MCTC HARQ schemes, facilitating a significantly reduced iterative decoding complexity.

B. Minimizing the Storage Requirements of the MCTC HARQ LUT

The LUT designed for MCTC HARQ schemes may be defined as a set of sub-tables T_i generated for describing the

relationship between any particular set of $(i-1)$ MIs and the minimum supplemental MI $I_{th}(i)$ required for creating an open EXIT chart tunnel, according to:

$$I_{th}(i) = T_i\left[I(\tilde{\mathbf{b}}_1), I(\tilde{\mathbf{b}}_2), \cdots, I(\tilde{\mathbf{b}}_{i-1})\right] \quad . \tag{2}$$

where $2 \leq i \leq (R-1)$. Our DI strategy aided MCTC HARQ schemes does not require the R^{th} sub-table T_R, since the receiver should always exploit its final - namely the R^{th} opportunity of activating the turbo decoder after the reception of the R^{th} IR transmission. The justification of this action is that the correct reception of the information is more valuable than the price of the extra complexity required for performing this last-ditch decoding effort.

The LUT of MCTC HARQ only records the required MIs for a limited set of quantized and sorted $(i-1)$ MI values appearing in an ascending order, i.e. satisfying $I(\tilde{\mathbf{b}}_{\pi(1)}) \leq I(\tilde{\mathbf{b}}_{\pi(2)}) \leq ... \leq I(\tilde{\mathbf{b}}_{\pi(i-1)}) \leq I_{th}(i)$, where π contains the unique integers of $1, ..., (i-1)$ used for appropriately permuting the original IR transmission order. This method avoids storing a large amount of redundant entries, since for example $I_{th}(4) = T_4(0.21, 0.15, 0.17)$ is identical to $I_{th}(4) = T_4(0.15, 0.17, 0.21)$.

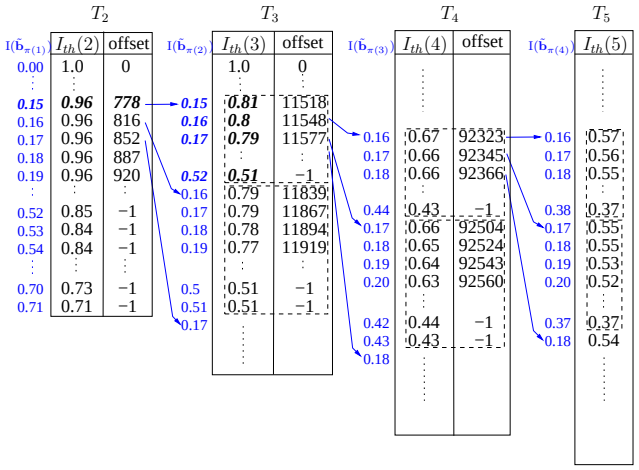


Figure 3. A subset of the MCTC HARQ LUT for $i = 2, 3, 4$ and 5, where the input MI step-size is $G = 0.01$.

Figure 3 displays a subset of the LUTs recorded for BPSK transmission over a quasi-static Rayleigh fading channel, as considered in Section II-C. As seen in Figure 3, the LUT is composed of several sub-tables, each of which corresponds to different IR indices i. Each sub-table T_i - except for the last one having the index $(R-1)$ - contains two columns. The first one stores the threshold MI $I_{th}(i)$, while the other provides an offset which can assist the indexing of the next sub-table T_{i+1}. More explicitly, the first element of each row in the sub-table T_i stores the $I_{th}(i)$ value that is valid for a particular set of MIs $\{I(\tilde{\mathbf{b}}_{\pi(1)}), I(\tilde{\mathbf{b}}_{\pi(2)}), ..., I(\tilde{\mathbf{b}}_{\pi(i-1)})\}$. The second element of each row stores a specific offset of the sub-table T_{i+1}, which indicates the starting index of the rows related to the sub-table T_i within T_{i+1}. Specifically, these rows are displayed in dashed boxes in Figure 3, all of which store the threshold MIs $I_{th}(i+1)$ corresponding to the sets of $\{I(\tilde{\mathbf{b}}_{\pi(1)}), I(\tilde{\mathbf{b}}_{\pi(2)}), ..., I(\tilde{\mathbf{b}}_{\pi(i-1)}), I(\tilde{\mathbf{b}}_{\pi(i)})\}$, where $I(\tilde{\mathbf{b}}_{\pi(i)})$ is varied but the previous $(i-1)$ IR MI values remain the same as in conjunction with the equivalent row in T_2.

Considering $I(\tilde{\mathbf{b}}_{\pi(1)}) = 0.15$ in Figure 3 as an example for providing further explanations, its corresponding threshold information is $I_{th}(2) = T_2(0.15) = 0.96$, implying that the minimum IR MI required for creating an open tunnel is 0.96, when the received MI of the first transmission is 0.15. This threshold is stored at the row index of $I(\tilde{\mathbf{b}}_{\pi(1)}) \cdot \frac{1}{G} = 15$ in sub-table T_2 having $G = 0.01$, as seen in Figure 3. Due to the sorting of $I(\tilde{\mathbf{b}}_{\pi(1)}) \leq I(\tilde{\mathbf{b}}_{\pi(2)}) \leq I_{th}(3)$, the initial value of $I(\tilde{\mathbf{b}}_{\pi(2)})$ is 0.15, thence we fix $I(\tilde{\mathbf{b}}_{\pi(1)}) = 0.15$ and increase $I(\tilde{\mathbf{b}}_{\pi(2)})$ from 0.15 by $G = 0.01$ each step (printed in bold fonts) to explore all possible $I_{th}(3)$ values corresponding to them. The operations continue until the $I_{th}(3)$ value required for perfect convergence becomes less than the current $I(\tilde{\mathbf{b}}_{\pi(2)})$ value, again, owing to the above-mentioned ordering. The resultant $I_{th}(3)$ values corresponding to $I(\tilde{\mathbf{b}}_{\pi(1)}) = 0.15$ and the incremental $I(\tilde{\mathbf{b}}_{\pi(2)})$ values are recorded in a block of continuous rows, as illustrated in the first dashed rectangle of sub-table T_3 in Figure 3. These 38 rows of sub-table T_3 have the offsets ranging from 778 to 815, which correspond to the 38 incremental $I(\tilde{\mathbf{b}}_{\pi(2)})$ values ranging from 0.15 to 0.52, as seen in the index area of sub-table T_3 in Figure 3. Here, the index of this block in sub-table T_3 starts at 778, since the rows 0 to 14 in sub-table T_2 have a total of 778 entries in sub-table T_3.

There are some special cases to be considered in Figure 3, for example, when the fixed $I(\tilde{\mathbf{b}}_{\pi(1)})$ has a larger value, such as $I(\tilde{\mathbf{b}}_{\pi(1)}) = 0.52$. Then, owing to the above-mentioned ordering, the incremental $I(\tilde{\mathbf{b}}_{\pi(2)})$ values start from 0.52. In this situation, the threshold IR MI $I_{th}(3) = 0.15$ corresponding to $\{0.52, 0.52\}$ becomes less than the current $I(\tilde{\mathbf{b}}_{\pi(2)}) = 0.52$. This suggests that no entries will be stored in sub-table T_3 for the fixed $I(\tilde{\mathbf{b}}_{\pi(1)})$ value of 0.52. We tag the corresponding offset as -1 for these cases.

Based on the above structure of the LUT, the search operation may be carried out at a low complexity using the offsets that are stored in the sub-tables. More specifically, the expression of 'offset+$[I(\tilde{\mathbf{b}}_{\pi(i)}) - I(\tilde{\mathbf{b}}_{\pi(i-1)})] \cdot 100$' may be applied recursively, where an initial 'offset' of 0 and $I(\tilde{\mathbf{b}}_0) = 0$ are assumed for $i = 1$, namely searching a certain $I_{th}(2)$ from the sub-table T_2. Once an offset of -1 has been found, this indicates that the EXIT tunnel is definitely open, since the received MIs are always sorted in an ascending order before the search is activated.

Although the LUT only records the sorted MIs, the size of the sub-table T_i increases approximately by an order of magnitude upon increasing i by one. However, increasing the step-size G has the potential of further decreasing the LUT size. In order to guarantee a high accuracy for the determination of the EXIT tunnel's open/closed state, the values of $I_{th}(i)$ in the LUT always maintain a step-size of 0.01, while $I(\tilde{\mathbf{b}}_1), I(\tilde{\mathbf{b}}_2), \cdots, I(\tilde{\mathbf{b}}_{i-1})$ tend to be increased in larger steps. As a result, the sub-table T_4 of Figure 3 changes to only store the threshold MIs for those larger stepped $I(\tilde{\mathbf{b}}_1), I(\tilde{\mathbf{b}}_2), \cdots, I(\tilde{\mathbf{b}}_{i-1})$. For example, for a step-size of $G = 0.1$, it stores the threshold MIs such as $0.80 = T_4(0.1, 0.1, 0.1)$, $0.73 = T_4(0.1, 0.1, 0.2)$, $0.65 = T_4(0.1, 0.2, 0.2)$ and so on. For a set of $(i-1)$ received MIs represented at a higher resolution, we follow the classic multi-dimensional linear interpolation technique for the sake

of estimating the supplemental MI required for achieving perfect decoding convergence and hence for triggering iterative decoding. This operation will require 2^{i-1} memory accesses to the LUT and $\sum_{k=0}^{i-2} 2^k$ linear interpolations for generating $I_{th}(i)$. Nonetheless, the complexity imposed is still far lower than the BCJR decoding complexity, when i is moderate.

C. Minimizing the Training Complexity of the LUT

An offline training process was developed for generating the LUT of Figure 3 by finding the specific $I_{th}(i)$ values for all possible values of $I(\tilde{\mathbf{b}}_{\pi(1)}), I(\tilde{\mathbf{b}}_{\pi(2)}) \dots I(\tilde{\mathbf{b}}_{\pi(i-1)})$. This process scans through the entire input MI range in steps of 0.01 - regardless of the specific value of G - in order to find the minimum MI for creating a marginally open EXIT tunnel, when it is combined with a certain set of MIs $I(\tilde{\mathbf{b}}_{\pi(1)}), I(\tilde{\mathbf{b}}_{\pi(2)}) \dots I(\tilde{\mathbf{b}}_{\pi(i-1)})$. For the sake of efficient execution, we abandon the traditional LLR-histogram based experimental EXIT chart creation and instead we use a model based on the spline functions $I(\tilde{\mathbf{a}}_i^{\mathrm{e}}) = f_i'[I(\tilde{\mathbf{a}}_i^{\mathrm{a}}), I(\tilde{\mathbf{b}}_i)]$ by fitting the corresponding EXIT functions f_i. Each spline function f_i' consists of a group of linear polynomials having the following form

$$I(\tilde{\mathbf{a}}_i^{\mathrm{e}}) = \begin{cases} m_{i1} \cdot I(\tilde{\mathbf{a}}_i^{\mathrm{a}}) + n_{i1} & (0.0 \le I(\tilde{\mathbf{a}}_i^{\mathrm{a}}) < 0.1) \\ m_{i2} \cdot I(\tilde{\mathbf{a}}_i^{\mathrm{a}}) + n_{i2} & (0.1 \le I(\tilde{\mathbf{a}}_i^{\mathrm{a}}) < 0.2) \\ \vdots & \\ m_{i10} \cdot I(\tilde{\mathbf{a}}_i^{\mathrm{a}}) + n_{i10} & (0.9 \le I(\tilde{\mathbf{a}}_i^{\mathrm{a}}) \le 1.0) \\ , & \end{cases} \tag{3}$$

where $m_{i1} \cdots m_{i10}$ and $n_{i1} \cdots n_{i10}$ were determined in advance for any $I(\tilde{\mathbf{b}}_i) \in [0, G, 2G, \cdots, 1]$.

The training is accelerated by iteratively invoking r spline functions f_i' corresponding to the MCTC decoder's r constituent components, since each call of this function only involves simple calculations. More specifically, all $I(\tilde{\mathbf{a}}_i^{\mathrm{e}}), i = 1, 2, \cdots, r$ are firstly initialized to 0. Then, we progress from f_1' to f_r', each time with a corresponding $I(\tilde{\mathbf{a}}_i^{\mathrm{a}})$ input calculated according to the following equation:

$$I(\tilde{\mathbf{a}}_i^{\mathrm{a}}) = J\left(\sqrt{\sum_{j=1, j\neq i}^{r} \left[J^{-1}(I(\tilde{\mathbf{a}}_j^{\mathrm{e}}))\right]^2}\right), \tag{4}$$

where the function $J(\cdot)$ and $J^{-1}(\cdot)$ can be found in the Appendix of [14]. This procedure continues until the output extrinsic information $I(\tilde{\mathbf{a}}_i^{\mathrm{e}})$ approaches 1 or until the affordable number of iterations is exhausted.

The training process can be accelerated by exploiting the monotonically decreasing nature of the function T_i, which ensures that each consecutive row in the LUT of Figure 3 requires a slightly lower MI value for achieving convergence than the previous one. For example, where we have $I(\tilde{\mathbf{b}}_{\pi(1)}), I(\tilde{\mathbf{b}}_{\pi(2)}), I(\tilde{\mathbf{b}}_{\pi(3)}) = \{0.15, 0.15, 0.17\}$, the additional MI required is $I_{th}(4) = 0.66$ in Figure 3, which is slightly lower than the previous value of $I_{th}(4) = 0.67$, corresponding to $I(\tilde{\mathbf{b}}_{\pi(1)}), I(\tilde{\mathbf{b}}_{\pi(2)}), I(\tilde{\mathbf{b}}_{\pi(3)}) = \{0.15, 0.15, 0.16\}$. This implies that the investigation of each new threshold test may commence from the value of the previous one, in decremental steps of 0.01. Statistically speaking, only two or three steps are required for determining the additional MI that yields an open EXIT tunnel.

D. Storage Requirements of the LUT

Based on the above LUT structure and training process, Table I shows the number of entries for T_2 to T_7 sub-tables of the LUT for different step-sizes, namely $G \in \{0.01, 0.05, 0.1\}$.

Table I
THE NUMBER OF ENTRIES FOR 6 SUB-TABLES OF THE LUT FOR THE STEP-SIZES OF 0.01, 0.05 AND 0.1.

	T_2	T_3	T_4	T_5	T_6	T_7
0.01	71	1261	13337	95388	508767	2159981
0.05	16	80	264	602	1108	1752
0.1	9	32	70	110	137	164

Observe from Table I that the total number of entries is significantly reduced for the step-size of $G = 0.1$ in comparison to 0.01. Our simulation results not included here owing to space-limitation demonstrated that the PLR, throughput and complexity curves of $G = 0.01, 0.05$ and 0.1 recorded for the LUT based DI aided MCTC HARQ are almost identical. Therefore, $G = 0.1$ is the best choice for the LUT, since it requires the minimum storage. When designing T_2, T_3, T_4 and T_5 with the step-size of 0.1 for our later simulations supporting $R = 6$ IR transmissions, only 212 MI thresholds have to be stored in these four sub-tables of the LUT. The complexity of the associated multiple-dimensional interpolation includes at most 16 memory accesses and 15 simple linear interpolations. Furthermore, if the LUT has to support more than $R = 6$ IR transmissions, e.g. $R = 8$, we may combine T_2, T_3 having $G = 0.01$ with T_4 to T_7 also having $G = 0.1$ in order to strike a trade-off between the memory requirements and the multi-dimensional interpolation cost.

III. THE LUT BASED DI AIDED MCTC HARQ SCHEME

Based on our efficient LUT, Figure 4 illustrates the flow-chart of the receiver when the DI strategy is applied to our MCTC HARQ design example. In detail, if the EXIT tunnel is deemed to be closed and the retransmission retry limit has not been exhausted, the receiver waits for the next IR transmission. By contrast, if the tunnel is sufficiently close to becoming open, the r-component turbo decoder is activated and iterative decoding proceeds as in the original ES aided MCTC HARQ scheme[3], commencing from the box labeled as 'choose the least recently operated BCJR decoder' which is copied from the flow-chart of the decoding process seen in Figure 3 of [8]. Since no BCJR decoding has been performed at this point, the first BCJR decoder is activated, followed by gradually proceeding one-by-one to the last one. Therefore, 'choosing the least recently operated BCJR decoder' implies that r BCJR

[3]In our LUT based DI aided HARQ scheme, the convergence of the MCTC decoder is defined as reaching the state, when the MI increment of *every* component BCJR decoder becomes lower than a particular threshold. This is slightly different from the approach of [8], which declares that convergence was reached once *any one* component decoder satisfies this stopping condition. This modification continues the decoding process for longer and is justified, because the proposed scheme does not commence decoding until there is a good chance that it will become successful.

decoders will be iteratively activated commencing from the first to the last. The iterative decoding process continues, until the Cyclic Redundancy Check (CRC) or the ES criterion is satisfied. The ES strategy employed was detailed in [8], which implies that the iterative decoding stops if any of the BCJR decoder's MI increment becomes less than a pre-defined minimum threshold. In the DI aided MCTC HARQ, since iterative decoding is only activated after the EXIT tunnel is deemed to be open, the iterative decoding process of this MCTC decoder is typically capable of achieving error-free decoding of the current packet. In the rare cases when the EXIT tunnel is only marginally open, the MCTC decoder may converge to a legitimate but incorrect decoding decision, which is spotted by the CRC assumed to be perfectly reliable, hence triggering an IR transmission. The packet will be deemed lost or dropped, when the number of transmissions reaches a retry limit.

The DI process is detailed in the upper dashed rectangle of Figure 4. As seen in Figure 4, the MI $I(\tilde{\mathbf{b}}_r)$ is firstly calculated using Equation 1. Next, the MI-sorting operation is performed in order to obtain the sorted MIs $I(\tilde{\mathbf{b}}_{\pi(1)}), I(\tilde{\mathbf{b}}_{\pi(2)}), \cdots, I(\tilde{\mathbf{b}}_{\pi(r-1)}), I(\tilde{\mathbf{b}}_{\pi(r)})$, where the largest MI $I(\tilde{\mathbf{b}}_{\pi(r)})$ is used for estimating the EXIT tunnel's open/closed state. This is carried out by checking, whether it is above the threshold $I_{th}(r) = T_r[I(\tilde{\mathbf{b}}_{\pi(1)}), I(\tilde{\mathbf{b}}_{\pi(2)}), ..., I(\tilde{\mathbf{b}}_{\pi(r-1)})]$ from the LUT. When the packet length N is sufficiently high, a straightforward comparison between $I(\tilde{\mathbf{b}}_{\pi(r)})$ and $I_{th}(r)$ may be used for confidently estimating the EXIT tunnel's open/closed state. Specifically, if $I(\tilde{\mathbf{b}}_{\pi(r)}) \leq I_{th}(r)$, the EXIT tunnel is deemed to be closed, otherwise it is deemed to be open.

However, satisfying the condition of $I(\tilde{\mathbf{b}}_{\pi(r)}) \leq I_{th}(r)$ cannot always provide a sufficiently reliable judgement of whether the trajectory can or cannot navigate through the EXIT tunnel to the $(1, 1)$ point. This is, because for short packets the trajectory may sometimes navigate through the tunnel that is marginally closed and vice versa [11]. Since our primary objective is to approach the maximum possible throughput, rather than waiting for the EXIT tunnel to open, it is desirable to allow iterative decoding to commence, even if the tunnel is marginally closed, especially when the packet length is short. This is achieved by modifying the threshold test according to $I(\tilde{\mathbf{b}}_{\pi(r)}) \leq I_{th}(r) - I_{diff}$, where I_{diff} is chosen to be the appropriate MI 'safety margin' for the specific packet length N employed. More particularly, if I_{diff} is chosen to be too high, then iterative decoding might commence at too low MI values, when there is no chance for the trajectory to navigate through the tunnel, hence unnecessarily increasing the complexity. By contrast, if I_{diff} is chosen to be too low, then iterative decoding will be deferred, even when there is a chance for the trajectory to navigate through the 'just' closed tunnel. This may potentially reduce the throughput. For this reason, we conceived the simulations detailed in Section IV to determine appropriate values for I_{diff} for a range of packet lengths.

IV. Performance Results

In this section, we evaluate the PLR, throughput and complexity of our previously proposed MCTC HARQ scheme

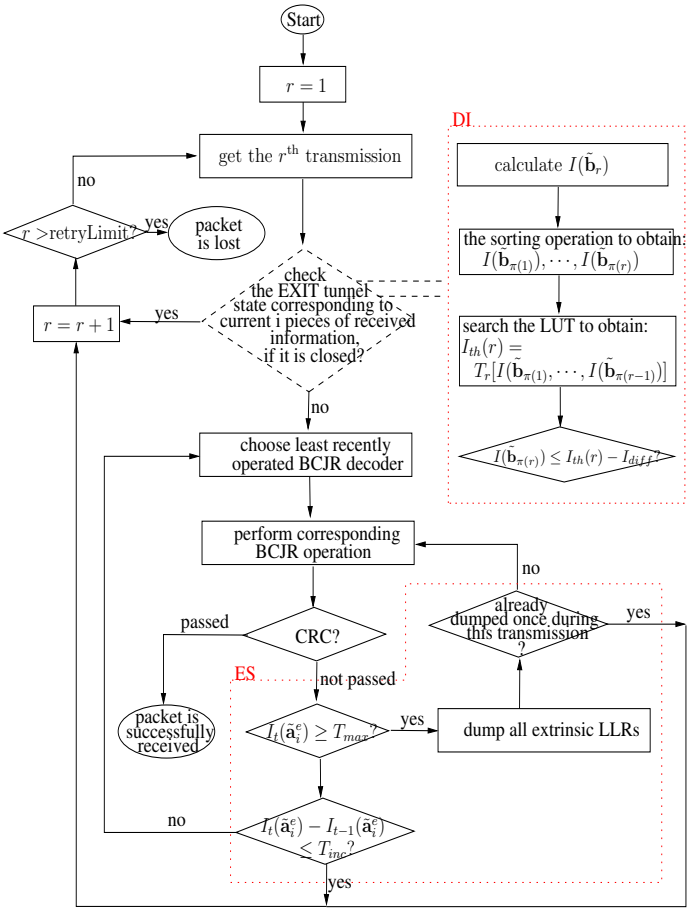
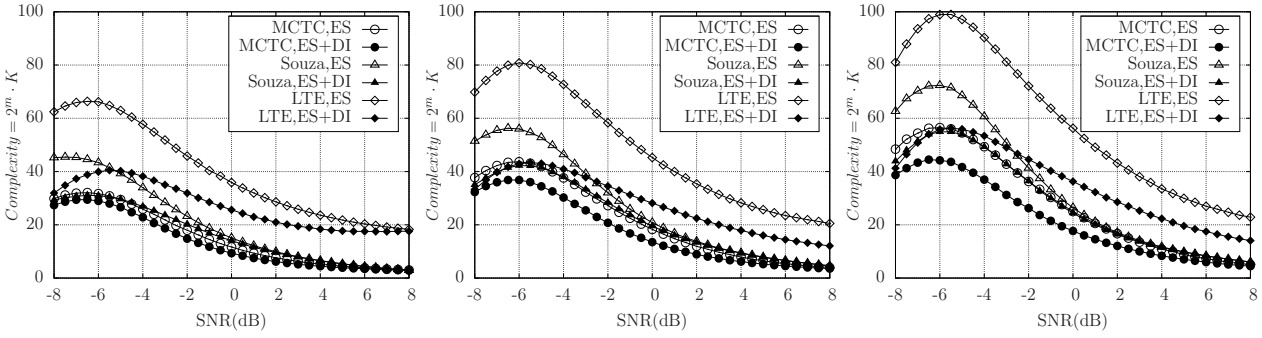


Figure 4. The flow chart of the DI and ES aided MCTC HARQ scheme based on the look-up table.

[8] relying on the proposed LUT based DI strategy. This was achieved by simulating the transmission of a statistically relevant number of packets over a Binary Phase-Shift Keying (BPSK)-modulated quasi-static Rayleigh fading channel. We also apply the LUT based DI strategy to Souza's systematic TCTC HARQ [6] and to the LTE system's systematic TCTC HARQ [15]. These three HARQ schemes which rely on the ES strategy proposed in [8] are used as our benchmarkers.

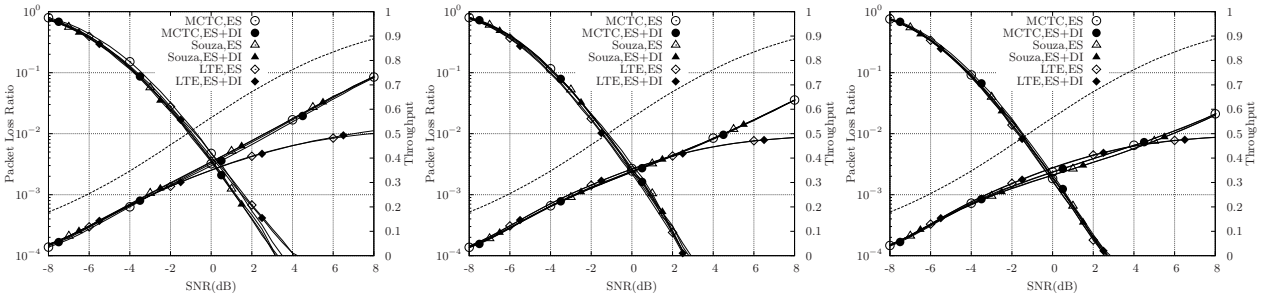
Souza's systematic TCTC HARQ transmits the systematic bit sequences $\mathbf{a}$ and the two parity bit sequences $\mathbf{b}_1$ and $\mathbf{b}_2$. The receiver activates iterative decoding between two parallel concatenated BCJR decoders after the third IR transmission. From the fourth IR transmission onwards, the repeated frame replica's LLRs are added to those gleaned from the previous transmissions. In contrast to the MCTC HARQ scheme, which may employ a unity-rate accumulator for obtaining the desirable PLR and throughput performances, Souza's HARQ scheme relies on the Recursive Systematic Convolutional (RSC) codes using octally represented memory-3 generator polynomials of $(17, 15)_o$ for achieving similar results, as seen in [8].

The LTE HARQ scheme adopts RSC codes having different memory-3 polynomials of $(15, 13)_o$. The LTE standard specifies a particular interleaver, and a so-called 'rate matching' operation for selecting specific transmitted bits rather than transmitting all bits [15]. More explicitly, the standard defines its own interleaver between two parallel concatenated turbo



(a) 48bits (b) 480bits (c) 4800bits

Figure 5. Complexity versus the quasi-static Rayleigh fading channel SNR for message packets of length a) 48 bits, b) 480 bits and c) 4800 bits.



(a) 48bits (b) 480bits (c) 4800bits

Figure 6. PLR and throughput versus the quasi-static Rayleigh fading channel SNR for message packets of length a) 48 bits, b) 480 bits and c) 4800 bits. The dashed line represents the DCMC capacity.

encoders/decoders for a range of specific packet lengths. Furthermore, the systematic bit sequence $\mathbf{a}$ and the two parity bit sequences $\mathbf{b}_1$, $\mathbf{b}_2$ are interleaved again, according to the standard's so-called sub-block interleavers. The interleaved systematic bits are entered into a circular buffer. The interleaved parity bits fill in the rear part of the circular buffer, where the odd positions are from $\mathbf{b}_1$ and the even positions are from $\mathbf{b}_2$. Next, N transmitted bits are continuously selected from a specific starting point of this circular buffer. This starting point advances along the circular buffer, based on a standard-specific equation, which is a function of the transmission frame index. As a result, turbo decoding can be activated right after the first frame's transmission, since it contains some of the systematic bits as well as some of the two parity bit sequences. The repeated LLRs are also Chase combined with the corresponding previously received replicas at the receiver.

For each of these HARQ schemes, the transmission retry limit was set to $R = 6$ in order to prevent any particular message packet from unduly reserving the network resources, when communicating in hostile environments requiring a high number of retransmissions. For the sake of fair comparison, we increase the maximum number of transmissions defined as $R = 4$ in the LTE HARQ scheme to $R = 6$, following the standardized rule of locating the starting point of the circular buffer for each IR transmission. Our results were collected by transmitting source message packets comprising 48, 480 and 4800 bits over quasi-static Rayleigh fading channels, since these packet lengths can be supported by the LTE HARQ scheme. Additionally, the appropriate I_{diff} values were selected for these packet lengths in order to limit the maximum normalized throughput loss imposed by the DI strategy to be as low as 0.003. Table II shows the preferred I_{diff} values for the three HARQ schemes considered.

Table II
THE PREFERRED I_{diff} VALUES FOR 48, 480 AND 4800 BITS PACKET LENGTHS, WHEN THE ALLOWED MAXIMUM THROUGHPUT LOSS IS 0.003.

	48 bits	480 bits	4800 bits
MCTC HARQ	0.08	0.01	0.0
Souza's HARQ	0.0	0.0	0.0
LTE HARQ	0.11	0.0	0.0

Observe from Table II that for Souza's HARQ scheme, the preferred I_{diff} values are all zeros, regardless of how short the packet length is. This is because the determination of the EXIT tunnel's open/closed state only starts after the third transmission, and because there is seldom a 'just' open or 'just' closed EXIT tunnel.

Figure 5 shows the complexity versus SNR performance for the three HARQ schemes both with and without the DI strategy. We employ the same complexity metric as in [8], which was formulated as $Complexity = 2^m \cdot K$, where m is the number of memory elements employed in the convolutional encoders' generator polynomials and K is the total number of BCJR decoder executions performed during iterative decoding. As shown in Figure 5, the 'MCTC,ES+DI' HARQ scheme offers complexity reductions of approximately 10%, 20% and 20% for the packet lengths of 48, 480 and 4800 bits respectively, when compared to the 'MCTC,ES' scheme. However, when the LUT based DI is applied to Souza's systematic TCTC HARQ, the complexity reductions become about 35%, 32% and 30% for the 48, 480 and 4800-bit packet lengths, since Souza's scheme only relied on the ES

strategy. Furthermore, the 'LTE,ES+DI' arrangement obtained the highest complexity reductions of up to 50% for all three packet lengths, since the LTE HARQ scheme activates the turbo decoding right away from the first transmission. The LUT-based DI aided MCTC HARQ scheme shows the lowest complexity among all HARQ schemes.

Let us now define the throughput as the ratio of the number of successfully delivered source message packets to the total number of transmitted packets. The left and right axes of Figure 6, respectively, illustrate the PLR and throughput performances, which are similar, regardless of which turbo HARQ scheme is used and whether the DI is employed, for all the three packet lengths considered. There is one exception, where the throughput of the LTE HARQ scheme becomes significantly lower than that of the other two HARQ schemes, namely at high SNRs[4]. This is because many packets may be successfully received after the first transmission attempt in the other two HARQ schemes, while in the LTE HARQ scheme, a minimum of two transmissions are needed for recovering the source packet. Furthermore, the dashed curve seen in Figure 6 reveals the gap between the Discrete-input Continuous-output Memoryless Channel's (DCMC) capacity and the throughput that these three HARQ schemes can achieve.

V. Conclusions

In this paper, a generically applicable low-complexity DI aided turbo HARQ design was proposed and characterized. Complexity is a critical issue for any communication scheme employing turbo codes, especially for applications like HARQ, which may have to activate iterative decoding multiple times. As demonstrated in [8], the total complexity of the HARQ schemes dispensing with ES strategies may be particularly high. By contrast, the ES strategy aided turbo HARQ scheme of [8] was shown to significantly decrease this complexity. For the sake of decreasing the complexity further, this paper proposed a more sophisticated DI strategy, which exploits the EXIT tunnel's open/closed state for determining, when iterative decoding should commence. At the cost of storing a modest LUT, the proposed scheme has been shown to decrease the complexity by 10% to 50% in the context of three recent turbo HARQ benchmarker schemes. This was achieved without imposing a significant degradation upon the throughput or PLR performance.

Although our LUT concept is specific for a particular code and channel, the LUT combined with MCTCs exhibited better performance than the LUT relying on TCTCs. Hence, the LUT combined with MCTCs may become the preferred design option for other scenarios, for example for relay networks. Furthermore, the LUT only has to be trained once for different scenarios, if the same code is used and similar channel conditions are encountered. Hence, when the LUT combined with MCTCs is applied in other scenarios, no additional training may be required. Therefore, the LUT based approach may be deemed general. Our future work will consider the application of the proposed DI strategy to other turbo coded schemes, including their distributed version used in relaying.

[4]In our simulations, the LTE HARQ has been implemented without the aid of other schemes specified in the LTE standard, for example the adaptive modulations.

References

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Proceedings of the IEEE International Conference on Communications (ICC)*, vol. 2, May 1993, pp. 1064 –1070.

[2] L. Hanzo, T. H. Liew, B. L. Yeap, R. Y. S. Tee, and S. X. Ng, *Turbo Coding, Turbo Equalisation and Space-Time Coding for Transmission over Fading Channels*. John Wiley & Sons, Ltd, 2011.

[3] L. Hanzo, J. P. Woodard, and P. Robertson, "Turbo decoding and detection for wireless applications," *Proceedings of the IEEE*, vol. 95, no. 6, pp. 1178–1200, June 2007.

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

[5] K. R. Narayanan and G. L. Stuber, "A novel ARQ technique using the turbo coding principle," *IEEE Communications Letters*, vol. 1, no. 2, pp. 49–51, March 1997.

[6] R. D. Souza, M. E. Pellenz, and T. Rodrigues, "Hybrid ARQ scheme based on recursive convolutional codes and turbo decoding," *IEEE Transactions on Communications*, vol. 57, no. 2, pp. 315–318, February 2009.

[7] H. Chen, R. G. Maunder, and L. Hanzo, "Multi-level turbo decoding assisted soft combining aided hybrid ARQ," in *Proceedings of the IEEE Vehicular Technology Conference Spring (VTC 2010-Spring)*, May 2010.

[8] ——, "Low-complexity multiple-component turbo decoding aided hybrid ARQ," *IEEE Transactions on Vehicular Technology*, vol. 60, pp. 1571–1577, May 2011.

[9] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Transactions on Communications*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.

[10] F.-M. Li and A.-Y. Wu, "On the new stopping criteria of iterative turbo decoding by using decoding threshold," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5506 –5516, nov. 2007.

[11] J. Lee and R. Blahut, "Generalized EXIT chart and BER analysis of finite-length turbo codes," in *Proceedings of the IEEE Global Telecommunications Conference, (GLOBECOM)*, vol. 4, 1-5 2003, pp. 2067 – 2072 vol.4.

[12] J. Hagenauer, "The EXIT chart - introduction to extrinsic information transfer in iterative processing," in *Proceedings of the 12th European Signal Processing Conference (EUSIPCO)*, 2004, pp. 1541–1548.

[13] H. Chen, R. Maunder, and L. Hanzo, "An EXIT-chart aided design procedure for near-capacity N-component parallel concatenated codes," in *Proceedings of the IEEE Global Communications Conference, (GLOBECOM)*, December 2010.

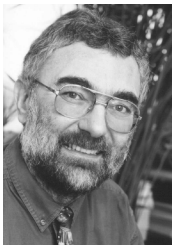
[14] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Transactions on Communications*, vol. 52, no. 4, pp. 670 – 678, April 2004.

[15] "3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding (Release 10)," 3GPP TS 36.212, December, 2010, Downloadable at http://www.3gpp.org/FTP/Specs/html-info/26346.htm.

Hong Chen received the M.E. degree in computer science from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2000. She is currently working toward the Ph.D. degree with the Communications Research Group, School of Electronics and Computer Science, University of Southampton, Southampton, U.K.

Since 2000, she has been a Lecturer with UESTC. Her current research interests include cross-layer optimization of wireless networks and turbo-coding aided hybrid ARQ.

Robert G. Maunder (MIEEE'03) has studied with the School of Electronics and Computer Science, University of Southampton, UK, since October 2000. He was awarded a first class honours BEng in Electronic Engineering in July 2003, as well as a Ph.D. in Wireless Communications and a lectureship in December 2007, all from University of Southampton, UK. His research interests include video coding, joint source/channel coding and iterative decoding. He has published a number of IEEE papers in these areas.



Lajos Hanzo (FREng, FIEEE'04, FIET, DSc) received his first-class degree in electronics in 1976 and his doctorate in 1983, both from Technical University of Budapest, Hungary. He was also awarded the Doctor of Sciences (Dsc) degree by University of Southampton, UK, in 2004, and the honorary doctorate "Doctor Honaris Causa" by Technical University of Budapest, Hungary, in 2009.

During his 35-year career in telecommunications he has held various research and academic posts in Hungary, Germany and the UK. Since 1986 he has been with the School of Electronics and Computer Science, University of Southampton, UK, where he holds the chair in telecommunications. He has co-authored 20 John Wiley/IEEE Press books on mobile radio communications totalling in excess of 10 000 pages, published about 990 research entries at IEEE Xplore, acted as TPC Chair of IEEE conferences, presented keynote lectures and been awarded a number of distinctions. Currently he is directing an academic research team, working on a range of research projects in the field of wireless multimedia communications sponsored by industry, the Engineering and Physical Sciences Research Council (EPSRC) UK, the European IST Programme and the Mobile Virtual Centre of Excellence (VCE), UK. He is an enthusiastic supporter of industrial and academic liaison and he offers a range of industrial courses.

Lajos is a Fellow of the Royal Academy of Engineering (FREng), UK, a Fellow of both the IEEE and the IET, and is also a Governor of the IEEE VTS. Since 2008 he has been the Editor-in-Chief of the IEEE Press and since 2009 a Chaired Professor also at Tsinghua University, Beijing. For further information on research in progress and associated publications please refer to http://www-mobile.ecs.soton.ac.uk