IEEE Access
Multidisciplinary Rapid Review Open Access Journal

Received January 14, 2018, accepted February 18, 2018, date of publication March 2, 2018, date of current version March 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2810293

LDPC Decoding Algorithms for Implant to Implant Wireless Body Area Network

ALBASHIR ADEL YOUSSEF[1,2], (Student Member, IEEE), BASSANT ABDELHAMID[2], (Member, IEEE), SALWA HUSSEIN EL-RAMLY[2], (Senior Member, IEEE), HUSSEIN M. ELATTAR[1], (Member, IEEE), AND HAZEM HASSAN ALI[1], (Member, IEEE)

[1]Department of Electronics and Communications Engineering, Arab Academy for Science, Technology and Maritime Transport, Cairo, Egypt
[2]Department of Electronics and Communications Engineering, Ain Shames University, Cairo 11517, Egypt

Corresponding author: Albashir Adel Youssef (albashir.adel@aast.edu)

ABSTRACT Wireless body area network (WBAN) is a promising network aiming at enhancing the communication in medical applications. It is adopted by medical organizations due to its flexibility in remotely monitoring patient health status. WBANs suffer from many limitations due to excessive channel impairments. Low density parity check (LDPC) codes are proposed to mitigate WBAN's impairments concerning the bit error rate, complexity, and dissipated energy. In this paper, a comprehensive performance analysis of various LDPC decoding algorithms is used to improve communication and reduce complexity in implant to implant WBAN channel. Moreover, a novel low complex LDPC decoding algorithm, which has a performance close to soft decision and a decoding time close to hard decision algorithms is proposed to minimize the dissipated energy. The proposed algorithm can be classified as a hybrid decision algorithm. The results demonstrate extensive analysis and comparisons between hard, soft, and hybrid decision algorithms.

INDEX TERMS Low density parity check codes, wireless body area network, implementation efficient reliability ratio weighted bit flipping, forward error correction, turbo codes, hybrid decoding.

I. INTRODUCTION

WBAN is recently adopted by health care providers in remote monitoring of the patient's vital signs using wireless sensors. Despite its importance, the implementation of such networks faces many challenges in size, dissipated energy and quality of communication which impose extra requirements to fulfill those parameters in the presence of aggressive channel impairments [1].

Communication channels of the body are too complicated due to the existence of the organs and the movement of body parts which in turn, impose excessive challenges to the design parameters of these networks especially in the physical (PHY) layer. The main goal of WBANs is to improve the reliability of the sensors inside the body. This can be achieved by utilizing low complex error control algorithms to improve the performance and to save the dissipated energy by reducing the energy required to transmit data. This results in extending the battery life of those sensors.

Error control is classified into two classes [2]: Forward Error Correction (FEC) and Automatic Repeat Request (ARQ). Although, ARQ is more simple than FEC nevertheless, FEC is preferred in WBAN case due to its low number of transmissions to correct errors, which leads to reserving the energy dissipated by sensor nodes. Thus, error control codes with low complex encoding and decoding are introduced as an essential part in wireless sensor networks. In [3] and [4] for WBAN, various classes of codes are investigated, including Reed Solomon (RS) Codes, Convolutional Codes, Turbo Codes and Low Density Parity Check Codes (LDPC). Experimental results in [4] show that LDPC codes are able to fulfill Wireless Sensor Network (WSN) applications performance requirements due to their high coding gain compared to other codes, while the most challenging aspect in LDPC is its decoding algorithms.

LDPC decoding algorithms are divided into three categories: hard decision, soft decision and hybrid decision. LDPC encoding and decoding algorithms mainly depend on using sparse matrices to minimize the decoding time, the required power and the hardware complexity [6]. In [12], an adaptive Belief Propagation (BP) LDPC decoding

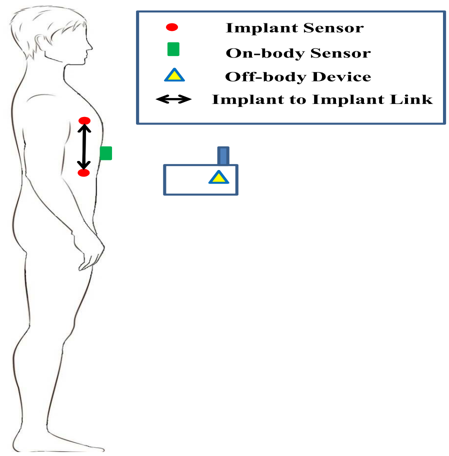


FIGURE 1. WBAN sensors and implant to implant communication link.

algorithm was proposed which is classified as soft decision algorithm to minimize the consumed energy by the decoder in WBAN sensors. However, this decoding algorithm adds overhead hardware complexity in the PHY layer of the sensors. Although LDPC decoding algorithms have never been utilized in reported WBAN channels except for BP algorithm in [12], the hybrid decision LDPC decoding algorithms are expected to reach lower complexity rather than soft decision ones [13]. Furthermore, these algorithms can provide a performance close to those achieved by soft decision ones [13].

In this paper a novel low complex LDPC hybrid decoding algorithm is proposed to improve the communication performance without adding any excessive overhead based on the analysis of existing LDPC decoding algorithms on WBAN.

The proposed low complex hybrid LDPC decoding algorithm is expected to enhance the communication link between implant sensors. Due to the flexibility of the proposed hybrid LDPC decoding algorithms, the hardware complexity is minimized and the Bit Error Rate (BER) performance becomes closer to the soft decision ones. Unlike [12], [14], [15] and [16], the other types of error control algorithms such as convolutional and Reed-Solomon codes are characterized by either excessive complexity or moderate bit error performance in case of the WBAN.

The rest of the paper is organized as follows: Section II discusses the implant to implant WBAN channel model. Section III illustrates briefly LDPC encoding and decoding algorithms. Section IV presents the proposed algorithm. Finally, sections V and VI display the simulation results and conclusions.

II. IMPLANT TO IMPLANT WBAN CHANNEL MODEL

Sensor devices used in WBANs and described in the standard IEEE 802.15.6 [17] are categorized into three types, shown in Fig. 1, as follows:

1) Implant sensor: a device that is planted inside the human body.
2) On-body sensor: a device that is installed on the surface of human skin or at most two centimeters away.
3) Off-body device: a device that is not in contact with the human skin or far away from human body.

Implant to implant communication link is the link between two implant sensors. It is characterized by log-normal shadowing due to the variant electrical properties, movement and continuous change in postures of the body, leading to an excessive fading of the transmitted signal power.

In this paper, the implants are classified into two classes, a deep tissue implant and near surface implant [17]. The channel characteristics of the communication link between these implants are considered as well.

IEEE standard reports that the communication channel between implants can be expressed with a path loss model as follows [17]:

$$PL = PL_o + 10\, n\, log_{10} \frac{d}{d_o} + S \qquad (1)$$

where PL and PL_o are the path loss at distance d and reference distance d_o in dB as d is the separation distance between two implants, n is the path loss exponent and S represents the shadow fading in dB with normal distribution zero mean μ and standard deviation σ_{dB}.

III. LDPC ENCODING AND DECODING ALGORITHMS

LDPC is an outstanding channel coding technique that has impressive performance approaching Shannon limit. It was proposed by Gallager in 1960 [6]. It gained massive attention and became extremely competitive to turbo codes [18]. It has been adopted by various digital communication standards, as an error control coding technique, such as in DVB-S2 [19], DVB-T2 [20], IEEE 802.16 [21] and IEEE 802.11 [22].

LDPC codes are provided by its parity check matrices only with sparse property. Therefore, an efficient encoding procedure using parity check matrix will be applied instead of converting provided parity check matrix to its generator matrix, which provokes destitution of the sparseness property belonging to **H** matrix leading to extra complexity [5]. The encoding procedure maintained in the simulations is motivated by [5]. Likely, the most direct method of building an LDPC encoder is by Gaussian elimination methods which results in identical lower triangular shape as appeared in Fig.2. Split the vector **x** into a systematic part **s**, and a parity part **p**, with the end goal that $\mathbf{x} = [\mathbf{s}, \mathbf{p}]$. Build a precise encoder as tails: i) Fill **s** with (N - M) the coveted data symbols. ii) Determine the **m** parity check symbols utilizing back-substitution.

The unpredictability of such encoding scheme ascertained by bringing the matrix **H** into the coveted shape requires $O(n^3)$ operations of preprocessing. The real encoding at that point requires $O(n^2)$ operations since, after the preprocessing, the matrix will never again be sparse. All the more unequivocally, we expect that we require about $n^2 = \frac{r(1-r)}{2}$ XOR operations to fulfill this encoding, where "r" is the rate of the code. Furthermore, encoding is conceivable for those codes. Remarkably, the proposed encoding scheme in [5] still

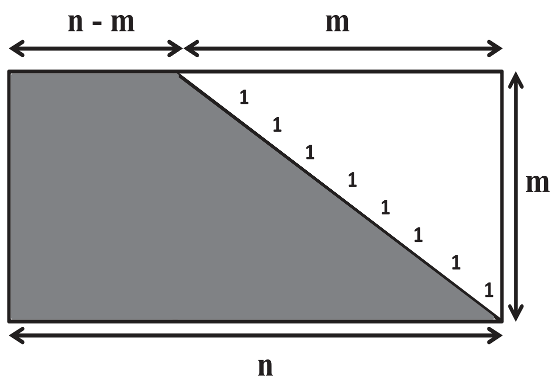


FIGURE 2. An equivalent parity-check matrix in lower triangular form.

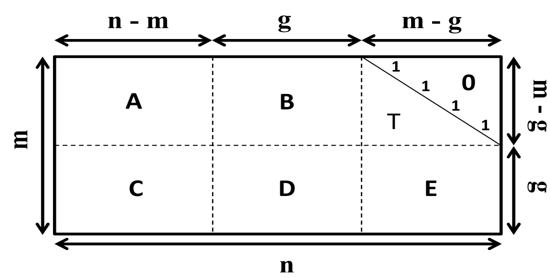


FIGURE 3. The parity-check matrix in approximate lower triangular form.

prompts quadratic encoding, as the constant factor before the n^2 term is regularly insignificant. On the other hand the encoding complexity will remain reasonable even in the case of using expansive block lengths.

The proposed encoder is seen in Fig. 3. Assume that by performing row and column permutations only we can bring the parity-check matrix into the form indicated in Fig. 3. We say that it is in approximate lower triangular form.

Moreover, all these matrices are sparse and are lower triangular with ones along the diagonal. Multiplying this matrix from the left by

$$\begin{pmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{E}\mathbf{T}^{-1} & \mathbf{I} \end{pmatrix} \tag{2}$$

we get

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{T} \\ -\mathbf{E}\mathbf{T}^{-1}\mathbf{A}+\mathbf{T} & -\mathbf{E}\mathbf{T}^{-1}\mathbf{B}+\mathbf{D} & \mathbf{0} \end{pmatrix} \tag{3}$$

let $\mathbf{x} = (\mathbf{s}, \mathbf{p}_1, \mathbf{p}_2)$ where $\mathbf{s}$ denotes the systematic part, $\mathbf{p}_1$ and $\mathbf{p}_2$ combined denote the parity part, $\mathbf{p}_1$ has length (g), and $\mathbf{p}_2$ has length $(m$âĂ"$g)$. The defining equation $\mathbf{H}\mathbf{x}^T = 0$ splits naturally into two equations, namely

$$\mathbf{A}\mathbf{S}^T + \mathbf{B}\mathbf{p}_1^T + \mathbf{T}\mathbf{p}_2^T = 0 \tag{4}$$

and

$$(-\mathbf{E}\mathbf{T}^{-1}\mathbf{A}+\mathbf{C})\mathbf{S}^T + (-\mathbf{E}\mathbf{T}^{-1}\mathbf{B}+\mathbf{D})\mathbf{P}_1^T = 0 \tag{5}$$

define $\boldsymbol{\phi} = -\mathbf{E}\mathbf{T}^{-1}\mathbf{B}+\mathbf{D}$
from (5) we conclude that

$$\mathbf{P}_1^T = -\boldsymbol{\phi}^{-1}(-\mathbf{E}\mathbf{T}^{-1}\mathbf{A}+\mathbf{C})\mathbf{S}^T \tag{6}$$
$$\mathbf{P}_2^T = -\mathbf{T}^{-1}(\mathbf{A}\mathbf{S}^T + \mathbf{B}\mathbf{P}_1^T) \tag{7}$$

Therefore the resultant code word will be in its systematic form as $\mathbf{c} = (\mathbf{s}, \mathbf{p}_1, \mathbf{p}_2)$.

Decoding process starts by using $M \times N$ parity check matrix $\mathbf{H}$ to obtain a $1 \times M$ syndrome vector bits which can be defined as [6]:

$$\mathbf{s} = \mathbf{z}\,\mathbf{H}^T \tag{8}$$

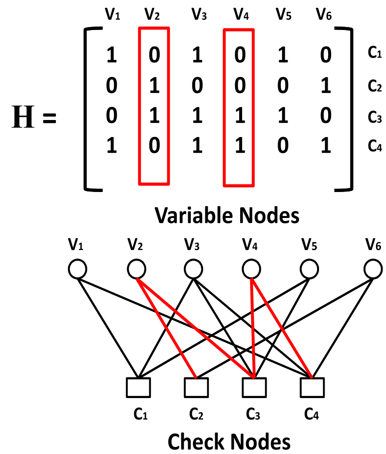


FIGURE 4. Tanner graph for 4 check nodes and 6 variable nodes.

where $\mathbf{z}$ is the hard (binary) values extracted from soft vector $\mathbf{y}$ which is the received vector from the channel. The syndrome is used to check if the received code word is decoded successfully or if it needs further processing to be corrected.

LDPC decoding depends completely on the parity check matrix $\mathbf{H}$ with entries $h(m, n)$. The size of $\mathbf{H}$ influences the BER performance and the complexity of decoding algorithms [24]. As the size of a parity check matrix belonging to certain code is expanded to higher sizes, this will lead to an improvement of the BER performance to reach Shannon limit [24]. Moreover, the complexity of the decoder grows quadratically [24] and the decoding time is escalated.

LDPC codes can be presented graphically using Tanner graph or bipartite graph which is introduced in [25]. Tanner graph has two types of nodes: check and variable nodes which represent message and code word bits, respectively. Fig. 4 shows an example of this graph for a certain code with four check nodes (M = 4) and six variable nodes (N = 6). All decoding algorithms are based on passing the message from check nodes to the variable nodes and vice versa.

Denote the set of bits that are connected to check node m by $\mathcal{N}(m) = \{n : h(m, n) = 1\}$. Similarly, the set of checks in which bit n participates are $\mathcal{M}(n) = \{m : h(m, n) = 1\}$. where $h(m, n) = 1$ is equivalent to ones located in parity check matrix at row m and column n. $\mathcal{N}(m)_{\overline{n}}$ represents the set of variable nodes $\mathcal{N}(m)$ excluding variable node n, and

Algorithm 1 BF Decoding Procedures

Step 1: Compute syndrome by (8). If it reached all zero, halt the decoding.
Step 2: Determine the number of unsatisfied parity-check equ- -ations for each code bit position.
Step 3: Identify bit position which is connected to the largest unsatisfied parity-check equations.
Step 4: Flip identified bit position.
Step 5: Repeat step 2 to step 4 until all parity-check equa- tions
are satisfied or a predetermined number of itera- tions is reached.

$\mathcal{M}(n)_{\overline{m}}$ is the set of check nodes $\mathcal{M}(n)$ excluding check node m.

LDPC decoding algorithms are iterative techniques using a predetermined number of iterations to reach either zero syndrome (no error exists) or an enhanced version of code word sent by the transmitter with minimum number of errors. Decoding algorithms are classified into three classes: Hard decision, Soft decision and Hybrid decision [6]. Hard decision algorithms depend on hard (binary) information of received code word in both detecting and correcting errors, while in soft decision, it relies on soft values (raw values) received from channel to perform either detection or correction of errors. Since hard decision algorithms are characterized by having the least complexity and performance, while soft ones are characterized by impressive performance with upraised complexity, hybrid decision algorithms are introduced to compromise between complexity and BER performance.

A. HARD DECISION DECODING ALGORITHMS

Hard decision algorithms are represented by the original Bit-Flipping (BF) algorithm proposed in [6] and its variants. BF algorithm is characterized by uncomplicated hardware complexity plus subjacent error correcting capability. The BF algorithm is fully expressed in Algorithm 1. The complexity of hard decision iteration is reported in [26] as $O(M\rho + N\gamma)$ where ρ is the number of ones per row in H and γ represents the number of ones per column in H. In the following subsections, some of BF algorithms are illustrated.

1) WEIGHTED BIT FILLIPING (WBF)

It is proposed in [27] and aims at improving BF decoding algorithm to reach better error performance by including some reliability to the information of the received symbols in their decoding decisions. Indeed, extra decoding complexity is mandatory for such performance improvement.

The algorithm initiates the decoding process by identifying the most unreliable variable node associated with each individual check node. It can be expressed as:

$$| y_{n_{min}} | = \{\min | y_n | : n \in \mathcal{N}(m)\} \tag{9}$$

where n_{min} is the index belonging to variable node having the bottom-level soft value.

The minimum absolute element in the received sequence can be determined as [27] where $| y_n |$ is the absolute value of y_n representing the measure of reliability for a received message. Let z_n be the binary equivalent for soft value y_n. As $| y_n |$ is formidable, the reliability of the hard-decision digit z_n is marked-up. The error-term E_n for each variable node can be determined as:

$$E_n = \sum_{m \in \mathcal{M}(n)} (2s_m - 1) \, | y_{n_{min}} | \tag{10}$$

where s_m is the syndrome bit associated with the m^{th} check node. The value E_n is the weighted check sum which belongs to the code bit position n. WBF algorithm is fully described in Algorithm II.

Algorithm 2 WBF Decoding Procedures

Step 1: Determine syndrome by (8). If all are zero, halt the decoding.
Step 2: Compute E_n based on (10), for $1 \leq n \leq N$.
Step 3: Locate the bit position n where E_n is the bulkiest.
Step 4: Flip the bit z_n which is the hard decision of y_n.
Step 5: Repeat Steps 1 to 4 until all the parity-check equations are satisfied or a pre-set maximum number of iterations is achieved.

2) MODIFIED WEIGHTED BIT FILLIPING (MWBF)

As mentioned, the error-term in (10) is calculated based on information given only by the check node. Zhang *et al.* proposed the MWBF algorithm [28], where the information delivered by the variable node is also examined. The main discrimination between the WBF algorithm and the MWBF algorithm is the second step in the iterative decoding process. The error-term is computed as follows [28]:

$$E_n = \sum_{m \in \mathcal{M}(n)} (2s_m - 1). \, | y_{n_{min}} | - \alpha \cdot | y_{n_{min}} | \tag{11}$$

where α is a predetermined threshold. Comparing (11) and (10), it can be detected that there is an additional term in (11), where the information afforded by the variable node is taken into account. The MWBF algorithm postulates that there are two variable nodes admitting the same error-term computed in (10). Therefore, the two variable nodes have the same probability of being flipped. Moreover, if the magnitudes $| y_n |$ of these two variable nodes are altered, the one owning a lesser magnitude is more unreliable, and hence should be inverted. Therefore, by employing (11) and merging the extra multiplicative term $\alpha \cdot | y_{n_{min}} |$ in the interpretation of the error-term, a more enhanced decision could be made. The performance of the MWBF algorithm leans extremely on the weighting factor α, hence α has to be pre-computed using off-line processing.

3) RELIABILITY RATIO WEIGHTED BIT FILLIPING (RRWBF)

MWBF algorithm is able to outperform the conventional WBF algorithm; however, it intensifies the required operations to get the error term. It has been illustrated in [28] that α term utilized in (11) should be precisely chosen.

Another shortcoming of both the WBF algorithm and the MWBF algorithm is that both of them treat the violation of a check node as unreliable variable node. However, all the variable nodes connected with this check node are liable to the violation of this particular check node. In other words, all variable nodes might be liable to change if the check node that they are engaged in is violated. For two distinct variable nodes sharing the same violated parity check, the probability that the check node is violated depends on variable node with an upraised soft-magnitude is lower than that associated with the slightest soft magnitude. Hence, in [29], a new quantity called Reliability Ratio (RR) is introduced to solve this problem, and can be defined as:

$$R_{mn} = \beta \frac{| y_n |}{| y_{n_{max}} |} \tag{12}$$

where β is a normalization factor imported for ensuring that $\sum_{n \in \mathcal{N}(m)} R_{mn} = 1$. To obtain the preeminent soft magnitude of all engaged variable nodes in the m^{th} check node:

$$| y_{n_{max}} | = \{max \mid y_n \mid : n \in \mathcal{N}(m)\} \tag{13}$$

where n_{max} is the index belonging to variable node having the superlative soft value. Therefore, rather than calculating the error-term E_n as in (11) using $y_{n_{min}}$, authors in [29] proposed the employment of the following formula instead of the one which belongs to MWBF:

$$E_n = \sum_{m \in \mathcal{M}(n)} (2s_m - 1)/R_{mn} \tag{14}$$

RRWBF steps are the same as the conventional WBF algorithm except that E_n in step 2 is calculated using (14).

4) IMPLEMENTATION EFFICIENT RELIABILITY RATIO WEIGHTED BIT FILLIPING (IERRWBF)

It is exhibited in [29] so that the reliability ratio based bit flipping (RRWBF) algorithm outperforms existing bit flipping based algorithms. While in [30] the authors proposed a new term to reduce the decoding time in undersized number of iterations. The reliability ratio is replaced by T_m:

$$T_m = \sum_{n \in \mathcal{N}(m)} | y_n | \tag{15}$$

and the error-term is determined by:

$$E_n = \frac{1}{| y_n |} \sum_{m \in \mathcal{M}(n)} (2s_m - 1) T_m \tag{16}$$

The rest of the algorithm will follow the same procedure as the conventional WBF algorithm, by wavering the calculations for E_n by (16).

5) MODIFIED IMPLEMENTATION EFFICIENT RELIABILITY RATIO WEIGHTED BIT FILLIPING (MIERRWBF)

One of the undeniable shortcomings of the previous iterative decoders is that the algorithm consumes an extensive percentage of decoding time at the variable node step and the check node step [31].

In [30], as the assigned number of iterations for the decoding algorithm is exaggerated, more processing time is required per iteration without any further contribution in BER in addition to altitudinous delay due to the oscillation phenomenon.

Hard-decision based bit-flipping algorithms sometimes fail to decode the received code word especially at low Signal to Noise Ratio (SNR), as the decoding in such a case causes the syndrome vector to be a non-zero vector, which leads to failure in decoding of the received code word. In addition, with monumental iteration number appointed for the algorithm (1000 iterations and more), there will be no expressive promotion in the error performance. It has been monitored that in aforesaid low SNR the syndrome vector will advance to be a non-zero vector and the decoding will still flip ceaselessly wasting computational power without any advancement in BER performance.

In [31], it is proposed to add a conditional step to detect such cases as described in the previous paragraph and limit the iteration loop by choosing whether to continue decoding or terminate the iteration loop and halt the algorithm to have a final decoded code word. Thus, a technique of the extra condition is initiated by determining syndrome vector at the end of each iteration and store its equivalent hard code word from the decoder in 3-entry register. The register is picked to be of a minimum size which is equal to three parts. At least two iterations are required to get the same decoded code word and the same syndrome vector, starting from initial vector, if the same bit in the decoded code word is being flipped two times to return to the initial state. Therefore, the three entries correspond to the initially received binary code word in the register, the code word accomplished after first iteration and the code word after second iteration. Thus, size three is the lowest size of register that could be used for accumulating received binary vectors for correlation between the first and the third (last one) where each new entry is stocked at the top of the register and the remaining entries are transited down to displace the last entry. As a result, when the point of oscillation is attained which means, that no more advancement to performance will exist, decoding will be terminated.

This supplementary condition results in an incomparable trimming of the complexity beyond any influence on performance compared to IERRWBF algorithm. The oscillation phenomenon discussed in the previous paragraph is presented in Fig. 5 for further clarification.

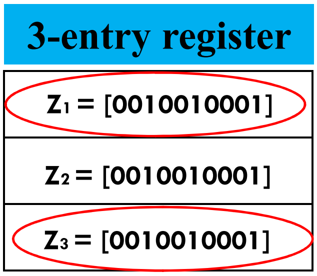


FIGURE 5. Oscillation case that occurs in IERRWBF algorithm where z is the hard decoded bits from IERRWBF decoder.

B. SOFT DECISION DECODING ALGORITHMS

Soft decision algorithms are derived from original Belief Propagation (BP) algorithm proposed in [6]. It is characterized by decoding iteration complexity of $O(2M\rho + 4N\gamma)$ [26]. In this subsection, BP algorithm and its variants are illustrated and discussed. For LDPC codes with Tanner graphs, the BP decoding includes two steps in each iteration: Firstly, processing on check nodes (Horizontal Step) and secondly, processing on variable nodes (Vertical Step). During each iteration, all check nodes receive messages from their neighbor variable nodes, process them, and send back rejuvenated messages to the neighbor variable nodes; then the same procedure takes place for all variable nodes. The decoding of LDPC codes is decentralized, each check node (or variable node) can be regarded as a processor, and the processing in all the check nodes (or variable nodes) are carried out simultaneously.

The BP algorithm can be called the probabilistic BP algorithm or the sum product algorithm, depending on how to represent messages. For the probabilistic BP algorithm, all messages are represented in its probabilities. This is the general form of the BP algorithm that also works for non-binary LDPC codes with symbols over $GF(q)$, $q > 2$. Alternatively, for the binary LDPC codes, messages are represented by LLR values, and consequently the BP algorithm is called the sum product algorithm.

There are many variants of BP algorithm that have been reported in literatures [7], [8], and [9]. Noticeably, the proposed informed dynamic scheduling (IDS) strategies regulated the message passing between variable and check nodes in the Tanner graph. So, the BER performance of the BP algorithm was improved. Also [9] stated that while preserving the same message generation functions and trimming the total number of messages propagated in the Tanner graph, IDS could enhance the BER performance at the expense of an incremental complexity to select the messages to be propagated. In addition, a modified BP algorithms are implemented in [10] and [11] by mixing genetic algorithm (GA) with original BP producing improved BER performance with extraneous complexity. So, all variants of BP algorithm, characterized by formidable BER performance and complexity which are not applicable for WBANs, will consume the available power and reduce the sensors battery life.

Some reduced complexity decoding algorithms can be derived from the sum product algorithm that suit WBANs, as it has the nethermost complexity compared to other soft decision algorithms [26]. These decoding algorithms will be tackled in the next paragraphs.

Algorithm 3 MIN-SUM Decoding Procedures

Step 1: Initialization:

$$F_n = \frac{4}{N_Q} y_n$$

N_o is noise power spectral density

Step 2: Horizontal Step:

$$L_{mn} \approx \prod_{n' \in \mathcal{N}(m)\bar{n}} sgn(z_{mn}). \min_{n' \in \mathcal{N}(m)\bar{n}} | z_{mn'} |$$

Step 3: Vertical Step: $z_{mn} = F_n + \sum_{m' \in \mathcal{M}(n)\bar{m}} L_{m'n}$

Step 4: For hard decision: $z_n = F_n + \sum_{m' \in \mathcal{M}(n)} L_{m'n}$

One of the most common soft decision algorithms is Min-Sum Algorithm. It is extracted from BP algorithm as stated in [26]. The process of decoding maintained in this algorithm is shown in Algorithm 3 [26]. As F_n is the initialization value of bit n while n' is one of the variable nodes connected to check node m except variable node n. m', is one of the check nodes connected to variable node n except check node m. z_n is the hard value of y_n after applying min-sum algorithm for each iteration, which is utilized in hard decision step.

The hard decision step is used for syndrome check to terminate iterations when they are converged to all zero syndrome or they reach the maximum predetermined number of iterations. Decoding is halted and last code word is assumed to be a correct message.

C. HYBRID DECISION DECODING ALGORITHMS

Hybrid decision decoding algorithms are a mixture of the soft and the hard decision algorithms for accomplishing and achieving both formidable performance of soft decision algorithms and subsided complexity of hard decision algorithms. In the following subsections, hybrid decoding algorithms are delineated.

1) BOOTSTRAPPED MODIFIED IMPLEMENTATION EFFICIENT RELIABILITY RATIO WEIGHTED BIT FILLIPING (BMIERRWBF)

To improve the performance of MIERRWBF, a new decoding algorithm is proposed in [32] called BMIERRWBF. It is based on inserting bootstrap step [32] implemented in [33].

The main idea of BMIERRWBF is in the bootstrap step representing the use of adaptive hybrid decision algorithm to limit the number of iterations belonging to soft decision algorithm which has an excessive complexity plus reaching BER performance close to soft decision algorithms. Multitudinous attempts use hybrid decision algorithms in [13]. Authors in [13] used multifarious complete iterations of soft decision algorithms to improve the BER of hard decision

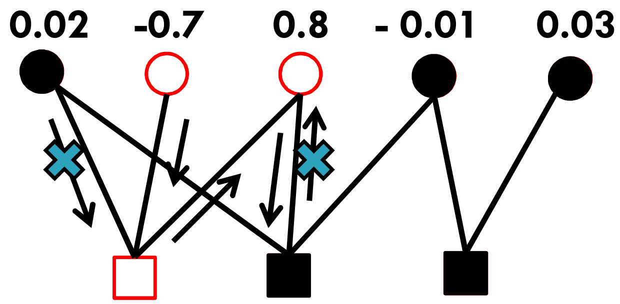


FIGURE 6. Bootstrap process on the received code word.

represented by MIERRWBF, which leads to having excessive complexity to achieve surpassing performance.

So, the main idea is to achieve closer BER performance of hybrid decision algorithms as implemented in [13] and soft decision algorithms by using bootstrap step as adaptive hybrid decision technique plus having comparable complexity compared to them.

It is worthy noted that the received bit y_n is unreliable if $| y_n | < \beta$. A check node is referred to as reliable with respect to an unreliable variable node if all the other variable nodes connected to that check node are reliable.

The decoding algorithm is initiated by identifying and erasing all the unreliable variable nodes, then assigning improved values and liabilities to the erased bits by passing messages from the reliable variable nodes through the reliable check nodes. The new value y'_n that substitutes y_n for an erased variable n is computed by (17) [33]. Also, for further clarification, an example of the process of bootstrap step is exposed in Fig. 6. As shown in Fig. 6, the received soft values from the channel are exposed to preassigned threshold β for differentiating between reliable messages and unreliable ones.

The next step is to determine reliable check nodes and unreliable ones using Tanner graph, as the check node connected to more than one unreliable message node counts as unreliable check node. After that, a blockage process is initiated for messages received from unreliable check nodes to their reliable variable nodes while, allowing the ones connected to reliable ones to transfer through connections of Tanner graph. After many iterations, unreliable variable nodes will be transformed to be reliable ones due to reliable message passing through Tanner graph connections.

$$y'_n = y_n + \sum_{m \in \mathcal{M}(n)} \prod_{n' \in \mathcal{N}(m)\bar{n}} sgn(y_{n'}). \min_{n' \in \mathcal{N}(m)\bar{n}} | y_{n'} | \quad (17)$$

$$L_{mn} \approx \prod_{n' \in \mathcal{N}(m)\bar{n}} sgn(Z_{mn}). \min_{n' \in \mathcal{N}(m)\bar{n}} | Z_{mn'} | \quad (18)$$

$$Z_{mn} = F_n + \sum_{m \in \mathcal{M}(n)\bar{m}} L_{m'n} \quad (19)$$

The combination of (18) and (19) produce the bootstrap step (17) which is inserted in WBF [33] and BMIERRWBF [32] to improve the reliability of received unreliable bits.

Algorithm 4 BMIERRWBF Decoding Procedures

Step 1: Extract unreliable variable nodes by applying threshold β on received code word $\boldsymbol{y}$.

Step 2: Apply bootstrap step on unreliable variable no- -des using (17).

Step 3: Compute syndrome by (8). If all are zeros, halt the decoding.

Step 4: Calculate E_n using (16), given that $1 \leq n \leq N$.

Step 5: Locate the bit position n where E_n is the highest.

Step 6: Flip the bit z_n which is the hard decision of y_n.

Step 7: Return to step 3, if the syndrome not all zero and save it in the three cell shift register.

Step 8: Check if the first and last cells of shift register are the same. Halting the decoding and last decoded vector will be the output of the decoder.

Complexity of the algorithm proposed in [13] for each complete iterations is $O(N_s(2M\rho + 4N\gamma) + N_h(M\rho + N\gamma))$ where N_s is the number of iterations belonging to soft decision algorithm and N_h is the number of iterations belonging to hard decision algorithm [13]. The complete iterations complexity of BMIERRWBF is $O((2M'\rho + 4N'\gamma) + N_h(M\rho + N\gamma))$ where $M' < M$ and $N' < N$ are the number of unreliable check nodes and unreliable variable nodes belonging to each unreliable check node, respectively. Thus, bootstrap step enhanced the usage of soft decision algorithm by contributing to most unreliable bits affected by channel impairments. This leads to improving performance and lowering complexity of hybrid algorithm proposed by [13]. Algorithm 4 manifests the unexpurgated decoding procedure of BMIERRWBF algorithm for further clarification.

IV. THE PROPOSED ALGORITHM

BMIERRWBF algorithm has an undeniable drawback which is the offline calculation of threshold value β used to Different between unreliable and reliable variable nodes of the received vector y. The offline calculation of the threshold value β promotes excessive overhead and it does not assure accuracy, for threshold β is determined according to the channel state at the initial step of decoding. Moreover, channel condition of WBANs suffers from excessive variations due to incommensurable tissue permeability and movement of human body parts. Therefore, the need for an alternative way to differentiate between reliable and unreliable variable nodes is essential in WBANs channels with its circumscribed resources of time and complexity. Therefore, proposing an alternative method is essential for the sake of limiting the time taken for predetermining bootstrap threshold, which induces a lower efficiency for the WBAN.

The new algorithm personified by Modified BMIERRWBF (MBMIERRWBF) is commenced by upgrading the bootstrap step. To fulfill, this requirement, a predetermined offline prior MIERRWBF algorithm is initiated instead of threshold β. The new algorithm is executed by checking the

TABLE 1. Hardware complexity analysis.

	Operation	Min-Sum	WBF	IERRWBF	MIERRWBF	BMIERRWBF	Proposed Algorithm
Steps at Check-Node	**abs**	1	-	1	1	1	1
	adder	$w_r - 1$	$w_r - 1$	$w_r - 1$	$w_r - 1$	$(w_r - 1) + w'_r$	$w_r - 1 + w'_r$
	multiplier	$w_r - 1$	-	-	-	M'	M'
	min	1	-	-	-	1	1
	flip-flops	-	-	-	$3M$	$3M$	$3M$
Steps at Variable-Node	**abs**	1	-	-	-	1	1
	adder	$w_c - 1$	$w_c - 1$	$w_c - 1$	$w_c - 1$	$w_c - 1$	$w_c - 1$
	min	1	1	-	-	-	-
	multiplier	$w_c - 1$	-	-	-	-	-
	comparator	-	-	-	-	$w_c - 1$	-

unreliable check nodes through calculating syndrome vector using (8) which differentiates between unreliable and reliable check nodes. Thus, every non-zero syndrome bit s_m is equivalent to unreliable check node; otherwise, it is considered to be a reliable check node.

Variable nodes connected to each unreliable check nodes represented by $\mathcal{N}(m)$ will be extracted. Decoding algorithm will track down the variable nodes with the lowest soft values connected to each unreliable check node m, as it will be the most probably unreliable variable nodes causing unreliability of the check node which they belong to. This operation will be executed using (9). After obtaining the variable nodes with the slightest soft values connected to each unreliable check nodes, a bootstrap step will be initiated to replace unreliable soft values by more reliable ones. Finally, all unreliable variable nodes have been extracted with formidable accuracy without the need for sagacious channel state information or pre-set threshold as performed in the BMIERRWBF algorithm. Complexity of the proposed algorithm is less subordinate than the complexity of BMIERRWBF algorithm by the canceled comparator which uses predetermined threshold to discriminate unreliable bits. Complexity of the proposed algorithm will be $O((2M'\rho + 4N'\gamma) + N_h(M\rho + N\gamma))$ where $M' < M$ and $N' < N$. The values of N' and M' are predetermined using (8) plus minimum function for each check node to get the undermost soft value of variable nodes kinship to it. For further clarification Algorithm 5 will epitomize the new algorithm in detail.

V. HARDWARE COMPLEXITY FOR LDPC DECODERS

Table 1 elucidates the hardware complexity of the maintained decoding algorithms including the proposed algorithm for check node and variable node, respectively. It is observed that the hardware complexity of BMIERRWBF and proposed algorithm have a trivial increment in complexity. Owing to adding extra w'_r adders which pertain to bootstrap and Modified bootstrap steps on unreliable variable nodes. Also it allegories the complexity reduction of the proposed algorithm done by trimming the comparator employed in BMIERRWBF algorithm which ends in a reduction in its complexity.

Algorithm 5 MBMIERRWBF Decoding Procedures (Proposed Algorithm)

Step 1: Extract unreliable check nodes by applying (8).
Step 2: Obtain indexes of non-zero syndrome elements s_m which are equivalent to unreliable check nodes.
Step 3: Extract $\mathcal{N}(m)$ which represents variable nodes lin--ked to each unreliable check node.
Step 4: Calculate the lowest variable node belonging to every unreliable check node using (9) which is the most probably unreliable bit causing unrelia--bility of check node it belongs to.
Step 5: Update all extracted unreliable variable nodes using step 4 with new more reliable soft values using (17).
Step 6: Proceed by MIERRWBF algorithm procedures.

According to Table 1, the complete complexity of exhibited algorithms proves that the proposed algorithm has a comparable complexity with good BER performance.

VI. SIMULATION RESULTS

The selected scenario or channel model for WBAN is the first class (deep tissue implant to implant) representing capsule endoscope application with $PL_o = 35.04$ dB, $n = 6.26$ and $\sigma_{dB} = 8.18$ dB. The machine capabilities used for simulations are 2.27 GHz Intel Core i3 with 2 GB random access memory (RAM).

Simulations are performed on a number of LDPC codes using BPSK modulation: for one regular PEG LDPC code [34] and one regular Gallager code [6]. The first code has $N = 504$ and $M = 252$ while the second one has $N = 204$ and $M = 102$. The received vectors are decoded simultaneously using different decoders. All algorithms are applied under WBAN channel with maximum number of iterations equal to 50 iterations for a fair comparison between algorithms except Min-Sum algorithm having only 10 iterations due to the prominent BER performance of soft decision algorithms. From the point of view of biological safety, the maximum transmitting power level is 20 mW or 13 dBm.

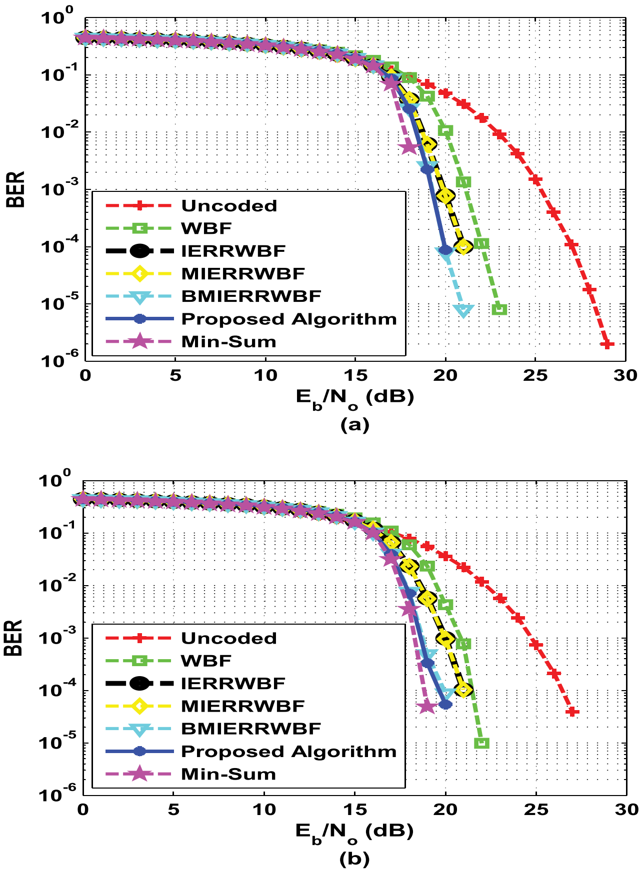


FIGURE 7. BER performance comparison: (a) LDPC regular PEG(504,252) code in WBAN channel, (b) LDPC regular (204,102) code in WBAN channel.

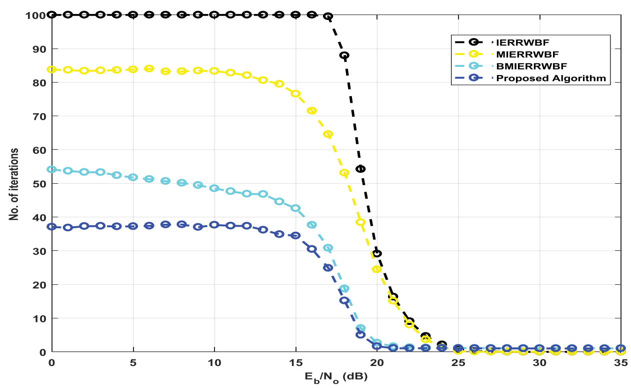


FIGURE 8. Number of iterations: A comparison between IERRWBF, MIERRWBF, BMIERRWBF and Proposed Algorithm LDPC regular PEG(504,252) code in WBAN channel.

This will be acceptable since it will never induce a Specific Absorption Rate (SAR) in the human tissue $> 2\ W/kg$ in any $10\ g$ [1].

For WBAN channel, the BER of uncoded and various decoding algorithms are plotted in Figs. 7a and 7b. It is shown that BER of the proposed algorithm is superior to over all hybrid decision algorithms and approaches soft decision algorithm represented by Min-Sum algorithm using the mentioned previous WBAN channel. As indicated in [1] the

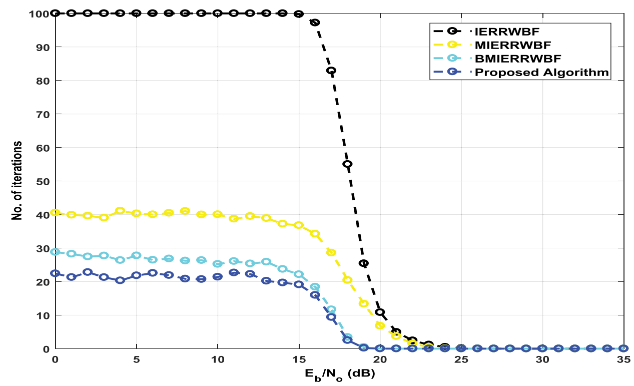


FIGURE 9. Number of iterations: A comparison between IERRWBF, MIERRWBF, BMIERRWBF and Proposed Algorithm LDPC regular (204,102) code in WBAN channel.

minimum required BER for Implant to implant communications is 10^{-3} at $E_b/N_o = 26$ dB. The proposed algorithm reaches lower BER at the same E_b/N_o for further proof of superiority of the proposed algorithm. As shown in Figs. 7a and 7b, the size of the parity check matrix has an effective influence on BER performance where the size of N and M increments results in improvement of the BER for all algorithms under study.

Moreover, the number of iterations comparison for IERRWBF, MIERRWBF, BMIERRWBF and proposed algorithm are plotted in Figs. 8 and 9 under WBAN channel by assigning predetermined maximum number of iterations equal to 100 iterations. The chosen limit of iterations, in the number of iterations comparison of the presented algorithms, is for further exploration of the performance of algorithms in case of soaring number of iterations. It is shown that the proposed algorithm has the lowest required number of iterations over the algorithms under comparison, as a further proof of superiority of the proposed algorithm over other algorithms in BER performance and required number of iterations.

Another metric which must be calculated is the decoding time for all decoding algorithms. This is demonstrated in Figs. 10 and 11, under WBAN channel using predetermined maximum number of iterations equal to 50 iterations for, fair comparison between presented algorithms. It is observed from Figs. 10 and 11 that the proposed algorithm maintains the third lowest decoding time compared to MIERRWBF algorithm in the range of low values of E_b/N_o from 0 to 15 dB. While in the range of 15 to 35 dB, decoding time of the proposed algorithm is approximately equal to the decoding time of BMIERRWBF and min-sum, though MIERRWBF is still the lowest. Moreover, min-sum algorithm has more subordinate decoding time than that of proposed algorithm, while its hardware complexity is remarkably complicated which in real applications will induce a massive hardware complexity more than the proposed algorithm.

Statistical calculations are done to explore the effect of the proposed algorithms on the required number of iterations in

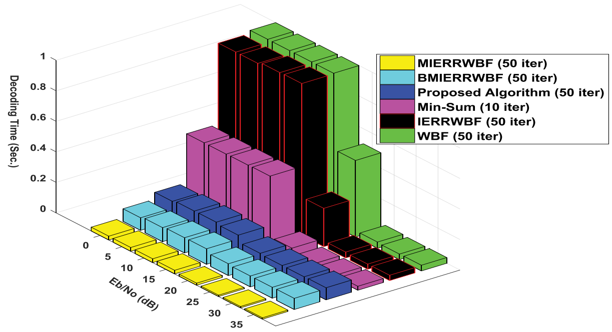


FIGURE 10. Decoding time: A comparison between WBF, IERRWBF, MIERRWBF, BMIERRWBF, Proposed Algorithm and Min-Sum algorithms LDPC regular PEG(504,252) code in WBAN channel.

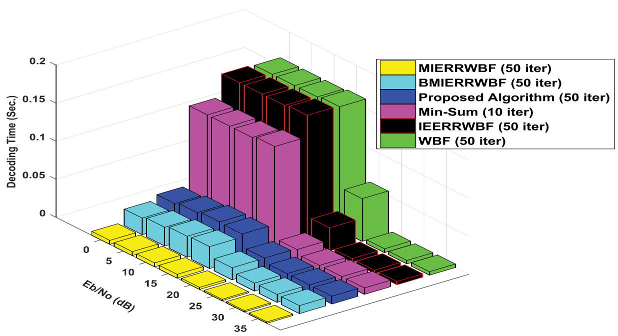


FIGURE 11. Decoding time: A comparison between WBF, IERRWBF, MIERRWBF, BMIERRWBF, Proposed Algorithm and Min-Sum algorithms LDPC regular (204,102) code in WBAN channel.

WBAN channel using PEG 504×252 code. Table 2 illustrates the mean (μ) and standard deviation (σ) of the number of iterations for WBAN channel. The proposed algorithm has maintained the least value of all the algorithms in comparison: IERRWBF, MIERRWBF and BMIERRWBF especially in the range of E_b/N_o from 0 to 10 dB. As shown in Table 2, the IERRWBF algorithm achieves its maximum limit of iterations without successfully decoding the received vectors because of excessive power loss that occurs in case of WBAN. Moreover, comparing algorithms as MIEERRWBF, BMIERRWBF and proposed algorithm, it is observed that the μ of the proposed algorithm is the lowest of all values of E_b/N_o except at $E_b/N_o = 15$ due to the shadowing in WBAN channel. While σ of the proposed algorithm is surpassing all values of E_b/N_o in the range from 0 to 20 dB compared to MIERRWBF and BMIERRWBF. This variation in σ is due to the shadowing that occurs in the WBAN channel.

For further exploration of proposed algorithm, another metric is investigated. This is the decoding algorithm parameters being compared to all decoding algorithms under study as depicted in Table 3. The proposed algorithm achieves comparable parameters compared to other LDPC decoding algorithms. For further investigation of the algorithms presented, a throughput of all algorithms is extracted to prove

TABLE 2. Statistics represented by mean (μ) and standard deviation (σ) of the number of decoding iterations for (504,252) PEG-LDPC code in WBAN case.

Decoding Algorithm	Statistical Parameter	E_b/N_o (dB)				
		0	5	10	15	20
IERRWBF	μ	100	100	100	58.8	3.9
	σ	0	0	0	29.9	1.9
MIERRWBF	μ	83.7	83.7	79.9	39.7	2,9
	σ	8.5	8.5	8.4	10.6	1.9
BMIERRWBF	μ	54.9	52.4	47.7	11.7	0.8
	σ	11.6	10.9	10.7	10.1	1.4
Proposed Algorithm	μ	39.7	38.7	37.7	24.7	0.6
	σ	12.8	14.4	12.8	14.8	0.5

TABLE 3. Comparison of decoding parameters for different LDPC decoding algorithms.

Decoding Algorithm	BER	Decoding Time	Complexity	Number of Iterations
WBF	Low	High	Low	Very High
IERRWBF	Medium	Medium	Medium	High
MIERRWBF	Medium	Very Low	Low	Very Low
BMIERRWBF	High	Low	Medium	Medium
Proposed Algorithm	High	Low	Medium	Low
Min-Sum	Very High	Very High	Very High	Low

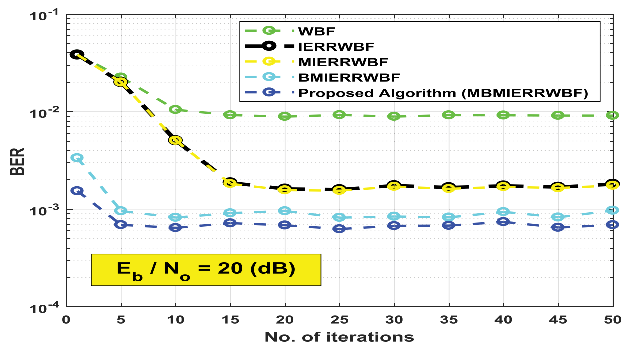


FIGURE 12. Convergence comparison between WBF, IERRWBF, MIERRWBF, BMIERRWBF, Proposed Algorithm and Min-Sum algorithms LDPC regular (204,102) code in WBAN channel.

the predomination of the proposed algorithm over other algorithms. According to Figs. 14 and 15, the proposed algorithm maintained the uppermost throughput compared to other algorithms especially in large LDPC **H** matrices, as shown in Fig. 15. The proposed algorithm occupies the 3rd place in throughput at low $\mathbf{E}_b/\mathbf{N}_o$s in case of small LDPC **H** matrix as portrayed in Fig. 14.

A crucial parameter for LDPC decoding algorithms that should be scrutinized is the convergence of algorithms.

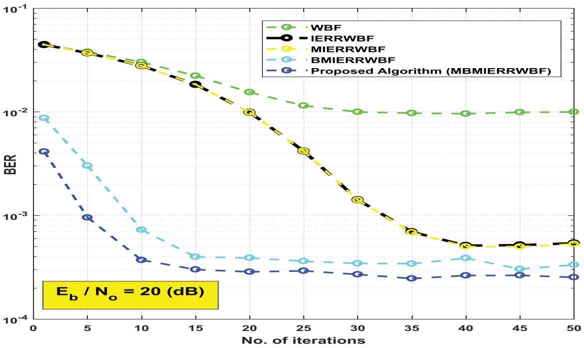


FIGURE 13. Convergence comparison between WBF, IERRWBF, MIERRWBF, BMIERRWBF, Proposed Algorithm and Min-Sum algorithms LDPC regular PEG(504,252) code in WBAN channel.

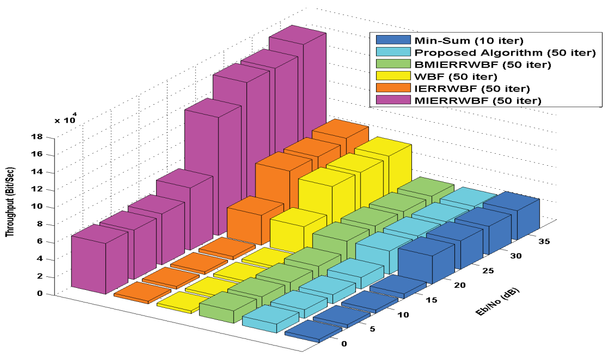


FIGURE 14. Throughput comparison between WBF, IERRWBF, MIERRWBF, BMIERRWBF, Proposed Algorithm and Min-Sum algorithms LDPC regular (204,102) code in WBAN channel.

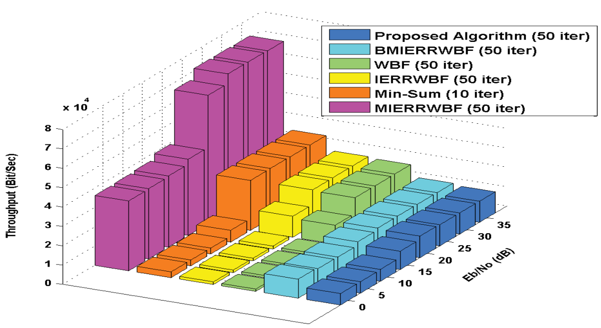


FIGURE 15. Throughput comparison between WBF, IERRWBF, MIERRWBF, BMIERRWBF, Proposed Algorithm and Min-Sum algorithms LDPC regular PEG(504,252) code in WBAN channel.

Fast convergence algorithms are characterized by prostrate consuming power to decode received information successfully, which is vital for WBANs. As delineated in Figs. 12 and 13, the proposed algorithm attained the fastest convergence compared to other algorithms, which validates enhancement in complexity of proposed algorithm over conferred algorithms. The proposed algorithm is outstanding in case of WBAN. The proposed algorithm's number of operations are the least compared to other algorithms expressed at modest

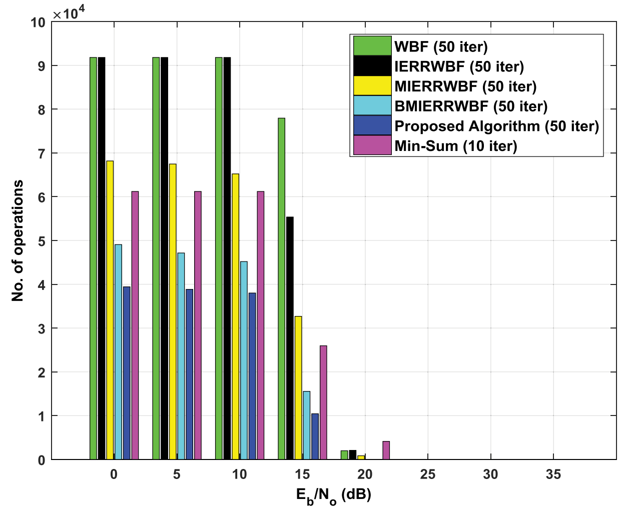


FIGURE 16. Number of operations comparison between WBF, IERRWBF, MIERRWBF, BMIERRWBF, Proposed Algorithm and Min-Sum algorithms LDPC regular (204,102) code in WBAN channel.

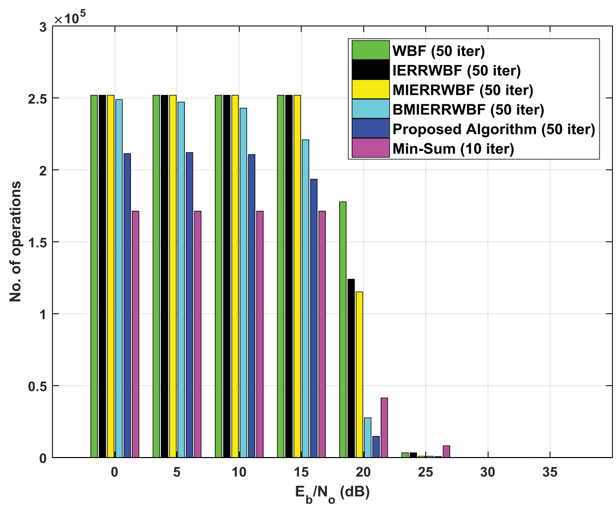


FIGURE 17. Number of operations comparison between WBF, IERRWBF, MIERRWBF, BMIERRWBF, Proposed Algorithm and Min-Sum algorithms LDPC regular PEG(504,252) code in WBAN channel.

LDPC **H** matrices as demonstrated in Fig. 16. In Fig. 17 the proposed algorithm occupy the 2nd place in number of operations consumed exactly at low $\mathbf{E}_b/\mathbf{N}_o$s while reaching minimum number of operations at more advanced $\mathbf{E}_b/\mathbf{N}_o$s, due to escalation of size belonging to **H** which results in increment of hard decision part operations especially at low $\mathbf{E}_b/\mathbf{N}_o$s.

Finally, according to the obtained results, the proposed algorithm interpreted by MBMIERRWBF has maintained the finest performance compared to other decoding algorithms used in comparison with respect to BER, required number of iterations, decoding time, resultant throughput, decoding convergence agility and number of operations performed by decoders.

VII. CONCLUSION

The proposed novel low complex LDPC hybrid decoding algorithm has shown to be prominent in the BER, decoding iterations, hardware complexity, number of operations, decoders convergence, decoders throughput, statistical properties and decoding time. Owing to using the least number of required iterations, the proposed algorithm achieved the least level complexity. Besides it occupied the minimum number of operations especially in modest LDPC matrices and bottommost E_b/N_os. Moreover, the proposed algorithm accomplished a lower decoding time close to MIERRWBF and BMIERRWBF, realized a radical number of decoding operations and reached the highest resultant throughput over all existing algorithms. Furthermore, the proposed algorithm accomplished the fastest convergence compared to other algorithms and achieved a nonpareil statistical analysis while maintaining comparable decoding parameters.

REFERENCES

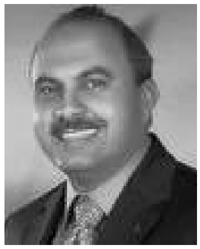
[1] J. Wang and Q. Wang, *Body Area Communication Performance in Body Area Communication: Channel Modeling, Communication Systems, and EMC*, 1st ed. Piscataway, NJ, USA: IEEE Press, 2013.
[2] S. Lin and D. J. Costello, *Error Control Coding*, vol. 2. Englewood Cliffs, NJ, USA: Prentice-Hall, 2004.
[3] E. Shih, S. Cho, F. S. Lee, B. H. Calhoun, and A. Chandrakasan, "Design considerations for energy-efficient radios in wireless microsensor networks," *J. VLSI Signal Process. Syst. Signal, Image Video Technol.*, vol. 37, no. 1, pp. 77–94, 2004.
[4] S. L. Howard, C. Schlegel, and K. Iniewski, "Error control coding in low-power wireless sensor networks: When is ECC energy-efficient?" *EURASIP J. Wireless Commun. Netw.*, vol. 2006, Dec. 2006, Art. no. 074812.
[5] T. J. Richardson and R. L. Urbanke, "Efficient encoding of low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 638–656, Feb. 2001.
[6] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.
[7] A. I. V. Casado, M. Griot, and R. D. Wesel, "Improving LDPC decoders via informed dynamic scheduling," in *Proc. Inf. Theory Workshop (ITW)*, Sep. 2007, pp. 208–213.
[8] A. I. V. Casado, M. Griot, and R. D. Wesel, "Informed dynamic scheduling for belief-propagation decoding of LDPC codes," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2007, pp. 932–937.
[9] A. I. V. Casado, M. Griot, and R. D. Wesel, "LDPC decoders with informed dynamic scheduling," *IEEE Trans. Commun.*, vol. 58, no. 12, pp. 3470–3479, Dec. 2010.
[10] Z.-R. Deng and X.-C. Liu, "Improved BP-based decoding algorithms integrated with GA for LDPC codes," in *Fuzzy Information and Engineering*, vol. 2. Berlin, Germany: Springer-Verlag, 2009, pp. 717–725.
[11] G.-J. Han and X.-C. Liu, "Dynamic schedules based on variable nodes residual for LDPC codes," in *Proc. 5th Int. Conf. Wireless Commun. Netw. Mobile Comput. (WiCom)*, Sep. 2009, pp. 1–3.
[12] N. Javaid, O. Rehman, N. Alrajeh, Z. A. Khan, B. Manzoor, and S. Ahmed, "AID: An energy efficient decoding scheme for LDPC codes in wireless body area sensor networks," *Procedia Comput. Sci.*, vol. 21, pp. 449–454, 2013.
[13] H. R. Zeidan and M. M. Elsabrouty, "Modified iterative two-stage hybrid decoding algorithm for low-density parity-check (LDPC) codes," in *Proc. IEEE 69th Veh. Technol. Conf. (VTC)*, Apr. 2009, pp. 1–5.
[14] B. Sansoda, W. Khattiya, and S. Choomchuay, "Performance comparison of convolutional codes for UWB-WBAN applications," in *Proc. Humanitarian Technol. Conf. (R10-HTC)*, Aug. 2013, pp. 336–339.
[15] R. McSweeney, C. Spagnol, and E. Popovici, "Comparative study of software vs. hardware implementations of shortened Reed–Solomon code for wireless body area networks," in *Proc. 27th Int. Conf. Microelectron. (MIEL)*, May 2010, pp. 223–226.
[16] C.-H. Yang, T.-Y. Huang, M.-R. Li, and Y.-L. Ueng, "A 5.4 μW soft-decision BCH decoder for wireless body area networks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 9, pp. 2721–2729, Sep. 2014.
[17] K. Y. Yazdandoost, *Channel Model for Body Area Network (BAN)*, Standard IEEE802.15-07-0943-00-0ban, IEEE P802. 15 Working Group for Wireless Personal Area Networks (WPANs), 2007
[18] N. Okubo, N. Miki, Y. Kishiyama, K. Higuchi, and M. Sawahashi, "Performance comparison between turbo code and rate-compatible LDPC code for evolved UTRA downlink OFDM radio access," *IEICE Trans. Commun.*, vol. 92, no. 5, pp. 1504–1515, 2009.
[19] ETSI and EBU, "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broadband satellite applications," ETSI, Sophia Antipolis, France, Tech. Rep. v1.1.1, 2005.
[20] ETSI and ETS, "Digital video broadcasting (DVB): Frame structure, channel coding and modulation for digital terrestrial television (DVB-T)," ETSI, Sophia Antipolis, France, Tech. Rep. 5, 1997.
[21] C. Eklund, R. B. Marks, K. L. Stanwood, and S. Wang, "IEEE standard 802.16: A technical overview of the WirelessMAN air interface for broadband wireless access," *IEEE Commun. Mag.*, vol. 40, no. 6, pp. 98–107, Jun. 2002.
[22] J. Weinmiller, M. Schläger, A. Festag, and A. Wolisz, "Performance study of access control in wireless LANs—IEEE 802.11 DFWMAC and ETSI RES 10 Hiperlan," *Mobile Netw. Appl.*, vol. 2, no. 1, pp. 55–67, 1997.
[23] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, "Progressive edge-growth tanner graphs," in *Proc. Global Telecommun. Conf. (GLOBECOM)*, Nov. 2001, pp. 995–1001.
[24] C. Chavet and P. Coussy, *Advanced Hardware Design for Error Correcting Codes*. Berlin, Germany: Springer-Verlag, 2015.
[25] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inf. Theory*, vol. IT-27, no. 5, pp. 533–547, Sep. 1981.
[26] G. D. Forney, Jr., "On iterative decoding and the two-way algorithm," in *Proc. Int. Symp. Turbo Codes Related Topics*, 1997, p. 1–14.
[27] Y. Kou, S. Lin, and M. P. C. Fossorier, "Low-density parity-check codes based on finite geometries: A rediscovery and new results," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2711–2736, Nov. 2001.
[28] J. Zhang and M. P. C. Fossorier, "A modified weighted bit-flipping decoding of low-density Parity-check codes," *IEEE Commun. Lett.*, vol. 8, no. 3, pp. 165–167, Mar. 2004.
[29] F. Guo and L. Hanzo, "Reliability ratio based weighted bit-flipping decoding for low-density parity-check codes," *Electron. Lett.*, vol. 40, no. 21, pp. 1356–1358, Oct. 2004.
[30] C. H. Lee and W. Wolf, "Implementation-efficient reliability ratio based weighted bit-flipping decoding for LDPC codes," *Electron. Lett.*, vol. 41, no. 13, pp. 755–757, Jun. 2005.
[31] H. R. Zeidan and M. M. Elsabrouty, "Low complexity iterative decoding algorithm for low-density parity-check (LDPC) codes," in *Proc. 1st IFIP IEEE Wireless Days WD*, Nov. 2008, pp. 1–5.
[32] A. A. Mohamed, M. M. Elsabrouty, and S. H. El-Ramly, "Bootstrapped iterative decoding algorithms for low density parity check (LDPC) codes," in *Proc. 5th Int. Conf. Syst. Netw. Commun. (ICSNC)*, Nice, France, Aug. 2010, pp. 335–339.
[33] A. Nouh and A. H. Banihashemi, "Bootstrap decoding of low-density parity-check codes," *IEEE Commun. Lett.*, vol. 6, no. 9, pp. 391–393, Sep. 2002.
[34] David JC. MacKay, *Encyclopedia of Sparse Graph Codes*. 2005.

ALBASHIR ADEL YOUSSEF received the B.S. and M.S. degree in electronics and communications engineering from the Arab Academy for Science, Technology and Maritime Transport, Cairo, Egypt, in 2008 and 2011, respectively. He is currently pursuing the Ph.D. degree with the Electronics and Communications Department, Faculty of Engineering, Ain Shames University, Cairo.

BASSANT ABDELHAMID received the B.Sc. and M.Sc. degrees in communications and electronic engineering from Ain Shams University, Egypt, in 2006 and 2010, respectively, and the Ph.D. degree in communications and electronic engineering from E-JUST University in 2013. She was with Ain Shams University as a Demonstrator, a Teaching Assistant, and an Assistant Professor, in 2006, 2010, and 2013, respectively. She was a Special Research Student with Kyushu University, Japan. She is specialized in communications engineering, digital signal processing, mobile communications, and statistical signal processing.

SALWA HUSSEIN EL-RAMLY received the B.Sc. and the M.Sc. degrees from the Faculty of Engineering, Ain Shams University, Egypt, in 1967 and 1972, respectively, and the Ph.D. degree from Nancy University, France, in 1976. She is currently an Emeritus Professor with the Electronics and Communications Engineering Department, Faculty of Engineering, Ain Shams University, where she was the Head of the Department from 2004 to 2006. She published over 100 scientific papers and supervised over 100 M.Sc. and Ph.D. thesis. Her research field of interest is wireless communication systems and signal processing, language engineering, coding, encryption, and radars. She received the Ain Shams Award of Appreciation in Engineering Sciences in 2010, the Award of Excellence from the Society of Communications Engineers in 2009, and the Award of Excellence from the Egyptian Society of Language Engineering. She is the IEEE Signal Processing Chapter Chair in Egypt. She shared in the establishment of the Egyptian Society of Language Engineering in 1996, where she is currently President.

HUSSEIN M. ELATTAR received the B.Sc. degree from the Arab Academy for Science, Technology and Maritime Transport in 1998, the M.Sc. degree from Ain Shams University, Egypt, in 2004, and the Ph.D. degree from Ain Shams University, Cairo, Egypt, in 2011, all in electrical engineering (electronics and communications). He is currently an Associate Professor with the Arab Academy for Science, Technology and Maritime Transport, AASTMT, Department of Electronics and Communications Engineering, Cairo, Egypt. His current research interests include radio resource management, cognitive networks, optimization techniques, IoT, and 5G applications.



HAZEM HASSAN ALI received the bachelor's degree in electrical engineering from Kuwait University, Kuwait, in 1987, the M.Sc. degree in electrical engineering (electrophysics: physical electronics) and the Ph.D. degree in systems and circuits from The George Washington University, Washington, DC, USA, in 1990 and 1993, respectively. He joined the faculty of the Department of Electronics and Communication Engineering, Arab Academy for Science, Technology and Maritime Transport (AASTMT), in 1993. In 2002, he became the Associate Dean of the College of Engineering, Cairo Campus-for Student Affairs. In 2006, he became the Chairman of Electronics and Communication Engineering Department-Cairo Campus, then in 2011, he became the Dean of Education and Research, Cairo Campus. In 2012, he became the Director, AASTMT Smart Village Campus. In 2014, he became the Director of AASTMT Misr Algadida Campus. He has been a Professor with the Electronics and Communication Engineering Department, AASTMT, Cairo Campus, since 2002. His research interests are integrated circuits and systems and microelectromechanical Systems.