Improved noisy gradient descent bit-flipping algorithm over Rayleigh fading channel

Reza Biazaran, Hermann Joseph Helgert
Department of Electrical and Computer Engineering, School of Engineering and Applied Science, The George Washington University, Washington, United State of America

Article Info

Article history:

Received May 3, 2021
Revised Dec 18, 2021
Accepted Jan 19, 2022

Keywords:

Bit flipping decoder
Gradient descent bit flip
Low density parity check code
Rayleigh fading channel
Uncorrelated Rayleigh channel

ABSTRACT

Gradient descent bit flipping (GDBF) and its many variants have offered remarkable improvements over legacy, or modified, bit flipping decoding techniques in case of decoding low density parity check (LDPC) codes. GDBF method and its many variants, such as noisy gradient descent bit flipping (NGDBF) have been extensively studied and their performances have been assessed over multiple channels such as binary symmetric channel (BSC), binary erasure channel (BEC) and additive white Gaussian noise (AWGN) channel. However, performance of the said decoders in more realistic channels or channel conditions have not been equally studied. An improved noisy gradient descent bit flipping algorithm is proposed in this paper that optimally decodes LDPC encoded codewords over Rayleigh fading channel and under various fade rates. Comparing to NGDBF method, our proposed decoder provides substantial improvements in both error performance of the code, and in the number of iterations required to achieve the said error performance. It subsequently reduces the end-to-end latency in applications with low or ultra-low latency requirements.

Corresponding Author:

Reza Biazaran
Department of Electrical and Computer Engineering, The George Washington University
1918 F Street, NW, Washington, DC, 20052, United State of America
Email: rbiazaran@gwu.edu

1. INTRODUCTION

Low density parity check (LDPC) codes [1] have become increasingly popular, due to their excellent performance and efficiency, to the extent that they are being considered as the error correcting coding technique of choice for major standard developing bodies such as the third-generation partnership project (3GPP) and the IEEE. Among various decoding methods, decoders based on belief propagation [2], [3], offer superior performance comparing to other hard decision decoding algorithms such as bit flipping (BF). However, their optimal performance is very computationally intensive. On the other hand, decoding techniques such as BF method [1] are easily implemented and offer relatively low computational complexity.

BF algorithms have been extensively studied and coding researchers have offered many variants of the method including weighted bit flipping (WBF) method [4] where soft channel information is incorporated, and a reliability measure is introduced to enhance performance of the BF algorithm. More recently, Wadayama *et al.* [5] devised a class of BF decoding method, i.e., gradient descent bit flipping (GDBF) algorithm that is based on maximum likelihood (ML) decoding problem. The GDBF method follows a gradient descent optimization model for the ML decoding to maximize an objective function [5]. Sundararajan *et al.* [6] introduced a noise perturbation to the objective function in order to avoid being trapped in local extreme, that do not correspond to a solution, during decoding process.

GDBF and noisy gradient descent bit flipping (NGDBF) algorithms are considered probabilistic bit flipping methods, and from that point of view, a few other methods have been proposed and implemented to improve decoding processes. Rasheed *et al.* [7], for instance, introduced a random perturbation to the decoding scheme, and subsequently, a random set of bits (that minimizes the inversion function) are selected to be flipped. The probabilistic gradient descent bit flipping (PGDBF) method proposed in [7], however, is designed for binary symmetric channels (BSC). A few other improvements have also been made, such as the work of Cui and Wang [8], in which a tabu-list is designed to overcome the repeated (the same) bit flipping issue. In other words, the set of flipped bits in the current iteration are added to the tabu list to prevent them from being flipped in the next iteration. Cui and Wang [8] improved their method in [9] by proposing an information storage bit flipping (ISBF) decoder for BSC channels in which the sum of syndromes orthogonal to a bit (also known as energy of the bit) is calculated. Then a bit is flipped with a certain probability if its energy value reaches a certain threshold (Th). By storing the energy value from the previous iteration, the Th can be set as the maximum of both energy values. In this method, both threshold and energy values can be computed in parallel, therefore reducing decoding delays.

The class of NGDBF method initially proposed in [6] as well as other variants of the method so far provide significant improvements over GDBF method for transmissions over additive white Gaussian noise (AWGN) or BSC channel. In fact, Declercq *et al.* [10] have shown that introducing randomness (noise) to an otherwise noiseless decoder can greatly improve error performance of family of BF decoders (including NGDBF and PGDBF algorithms) such that their performance can be compared with soft decision decoders' such as sum product algorithm (SPA) or other variants of belief propagation decoders.

While performance of the NGDBF algorithm and its variants show vast improvements over regular GDBF algorithm there are two major issues associated with them; to achieve performances similar to soft decision algorithms, NGDBF method require large number of decoding iterations [10] which may impose practical limitations on applying the decoders to use cases where low latency is required. It is because end to end latency of the system is directly related to processing delay associated with decoding iterations. Second, NGDBF algorithm, as it is currently designed, performs well over AWGN channel. However, its performance may degrade where it is to function over more realistic channels such as Rayleigh fading channel over which a lot of more modern use cases and applications are envisioned to operate.

In this paper, an improved noisy GDBF method is proposed that efficiently decodes LDPC encoded codewords transmitted over correlated Rayleigh fading channel. We demonstrate in section 3 that our proposed method outperforms regular NGDBF decoder in 1) average number of iterations required to achieve a certain probability of bit error, and 2) error performance in the form of probability of bit error as a function of energy of the bit per noise power spectral density ($^{E_b}/_{N_0}$). In section 3, we explain our proposed method chronologically and follow through each stage of a communication system as bits are encoded, transmitted through the channel, and decoded using our proposed and improved noisy gradient descent bit flipping algorithm. The results of our simulations are presented in section 4.

2. RESEARCH METHOD

2.1. Encoding and channel model

Assuming binary phase shift keying (BPSK) transmission, a codeword $\boldsymbol{v} = (v_0, v_1, v_2, \dots, v_{n-1})$ is modulated to a sequence $\boldsymbol{x} = (x_0, x_1, x_2, \dots, x_{n-1})$. For our simulation, the generator matrix $g(x) = 1 + x^2 + x^4 + x^6 + x^7 + x^{10}$ is used to generate codeword $\boldsymbol{v}$ resulting a (21,11) projective geometry low density parity check (F) code. The modulation scheme maps each bit x_k to v_k where $x_k = 2v_k - 1$ with $k = 0,1, \dots, n-1$. If $\boldsymbol{x}$ is transmitted through an AWGN channel the received vector $\boldsymbol{y_{awgn}}$ is of the form $\boldsymbol{y_{awgn}} = \left(y_{0,awgn}, \dots, y_{n-1,awgn}\right)$, where $y_{k,awgn} = x_k + n_k$ with n_k being a random variable that is Gaussian and has zero mean and variance $N_0/2$ (noise power spectral density). Assuming a binary operation, $x_k = 2v_k - 1 = +1$ for $v_k = 1$ and $x_k = -1$ for $v_k = 0$. Subsequently, soft decision received bit $y_{k,awgn} = x_k + n_k = 1 + n_k$ and $-1 + n_k$, respectively. On the other hand, if $\boldsymbol{x}$ is transmitted through a Rayleigh (flat) fading channel, the received vector $\boldsymbol{y_{ray}}$ is of the form $\boldsymbol{y_{ray}} = \left(y_{0,ray}, \dots, y_{n-1,ray}\right)$, where $y_{k,ray} = R_k x_k + n_k$, with R_k being a zero-mean complex Gaussian random variable. The absolute value of R_k (the envelope of the channel response) has Rayleigh probability distribution [11]. And finally, $z = (z_0, z_1, z_2, \dots, z_{n-1})$ is the hard decision received vector estimated from y vector. In our simulations, we let $z_i = 1$ if $y_i \geq 0$ and $z_i = 0$ otherwise.

It has been shown that discrete time samples of a Rayleigh fading process must necessarily be correlated and the correlation function is dependent on the Doppler frequency associated with relative motion of the transmitter, receiver, and other parameters such as propagation path and antenna characteristics [12]. If a very long bit interleaver is employed (where bits are spread over an interval much longer than the channel coherence time) then R_k's are independent (and complex) random variables [13]. However, in case of no (or partial) interleaving, R_k's become more correlated random variables which may introduce blocks of bit error in the transmitted frames when the transmitted frame experiences a deep fade.

We used the young model to simulate the channel coefficients R_k's that is based on the inverse discrete Fourier transform (IDFT) technique described in [12]. We generated time samples of Rayleigh fading processes corresponding to different Doppler frequencies (shifts) to simulate Rayleigh fading channel. To ensure that appropriate channel coefficients are generated, we calculated the mean of the generated random variables R_k and observed it to be zero or approximately zero. It was in addition to verifying level crossing rates that vary as a function of Doppler frequency in a Rayleigh fading channel model. To quantify the extent of correlation in a given Rayleigh channel model, we chose a comparative method in which the channel coefficients are compared to the receiver noise that is independent and identically distributed, i.e., the normalized cross correlation of R_k's and n_k's in the received signal $y_{k,ray}$.

2.2. Decoding

BF decoding, in its simplest form, is a syndrome decoding technique where parity check matrix of a code H determines the syndrome of a code s in error detection and correction as described below. Assuming H to have J rows and n columns, the number of 1's in each row and each column of the parity check matrix are the row and column weight of the matrix and are denoted ρ and γ respectively. For an LDPC code, ρ and γ are small comparing to J and n, hence the name low density. The syndromes [1] of the received vector s is then defined:

$$\boldsymbol{s} = (s_1, s_2, \dots, s_J) = z.H^T \tag{1}$$

where z is the hard decision received vector described in sub section 2.1 and the superscript T denotes transpose of the matrix H. When errors are introduced in the received vector, there will be parity failures and some of the syndrome components become 1. BF algorithm calculates all parity check equations (check sums) and changes (flips) the bit(s) that are contained in greater than a set number θ of unsatisfied parity check equations. It has been shown that θ (also referred to as the threshold) is a function of the ρ, γ, error correcting capability of the code and the signal-to-noise ratio (SNR) [14].

BF algorithm first calculates the syndrome bits (parity check equations). If all syndrome bits are zero, then the process stops and the correct codeword is declared. Otherwise, it finds the number of non-zero check sum equations for each bit k, denoted f_k, where $k = 0, 1, \dots, n-1$. The BF algorithm then identifies the set of bits for which f_k is maximum and flips the bits in the set. The process is then repeated until all check sums are zero. To reduce processing delay, this iterative process may be limited, that is the process stops if all check sums are zero or the maximum number of iterations is reached.

To improve hard decision BF decoding, some measure of reliability is introduced to the decoding process. For an AWGN channel, the magnitude of the soft decision received bit $|y_k|$ may be used. The larger $|y_k|$ the more trustworthy the hard decision bit z_k will be. Kolesnik [4] used this concept and developed an improved BF decoding method known as weighted BF (WBF) decoding. In WBF algorithm, a new function E_k is defined for each bit *k*:

$$E_k = \sum_{s_j^{(k)} \in S_k} (2s_j^{(k)} - 1)|y_j|^{(k)}{}_{min} \tag{2}$$

where S_k is the set of check sum equations that contain bit *k* (bit *k* participates in them). And $|y_j|^{(k)}{}_{min}$ is the magnitude of the least reliable soft decision bit. A bit is flipped whose E_k (or energy function) is greater than zero [4]. A few modifications and improvements have been made to WBF algorithm. WBF algorithm determines whether a bit should be flipped based on the contributions from the check sums as shown in (2). Zhang and Fossorier [15] proposed a modification to (2) by incorporating a factor such that the reliability of the individual bits also help determine if a bit should be flipped:

$$E'_k = \sum_{s_j^{(k)} \in S_k} (2s_j^{(k)} - 1)|y_j|^{(k)}{}_{min} - \alpha y_k \tag{3}$$

where α is a real number and $\alpha \geq 0$. The value of parameter α was found to be a function of SNR and the column weight γ of the code, regardless of the type of the LDPC code. As for the flipping threshold of the family of BF decoders, Cho and Sung [16] proposed a method that adaptively adjusted the flipping threshold in order to avoid meaningless no flipping conditions and to improve rate of decoding convergence to an optimal solution. The flipping threshold was shown in [16] to be a function of the number of errors. To avoid no flipping condition, the maximum of all E'_k in (3) E_{max} is set as the new threshold. Initially, any bits whose energy value is larger than the initial threshold is flipped. In case of no flipping condition, the threshold is adjusted to E_{max} and the process is repeated.

Wadayama *et al.* [5] proposed the original GDBF algorithm in which the maximum likelihood decoding concept presents itself as an objective function for gradient descent optimization. Based on maximum likelihood (ML) decoding principle, the received bit that maximizes the objective function *f(x)*, defined in (4), is declared as the correctly transmitted code bit.

$$f(x) = \sum_{k=0}^{n-1} x_k . y_k - \sum_{s_j^{(k)} \in S_k} (2s_j^{(k)} - 1) \tag{4}$$

The first term in (4) is the sum of the cross correlation of the transmitted code bits and the soft decision received bits which needs to be maximized to correctly decode the transmitted codeword. The second term plays the role of a penalty term. If all check sums are satisfied, the second term has an absolute minimum of $-\rho$ where ρ is the row weight of the parity check matrix. To ensure the objective function achieves its global maximum at the end of the iterative decoding process, partial derivative of objective function E_k, defined in (5) is evaluated bit by bit via a gradient descent optimization process.

$$E_k = x_k . y_k - \sum_{s_j^{(k)} \in S_k} (2s_j^{(k)} - 1) \tag{5}$$

As for mechanics of bit flipping, there are couple of points to consider; the first point is the fact that flipping bits can be deterministic or probabilistic. In the former, a bit that meets certain condition is flipped. In case of BF (and its variants) algorithms, the condition is number of unsatisfied parities checks that are orthogonal to a bit. In the latter, once the condition is met, the bit is flipped with certain probability. One study [17] has shown that the probabilistic BF algorithm is more useful when the number of unsatisfied parity check equations is close to $\left\lceil \frac{J}{2} \right\rceil$ with *J* being the number of rows of the parity check matrix. They showed that for a binary symmetric channel, the probability *p* with which bits are flipped is a design parameter and depends on the cross over probability of the channel. In general, the greater the number of unsatisfied parity check equations the higher the flipping probability *p*. The second point regarding the mechanics of bit flipping, is the number of bits that can simultaneously be flipped. From that perspective, there are three flipping modes: single bit flipping, multi-bit flipping, and a combination of both (or hybrid). In case of GDBF decoding where an optimization process is ensued based on gradient descent (or ascent) method, bit flipping can be considered as steps on a search path toward the solution that is the correctly decoded codeword. Single bit flipping is equivalent to taking small steps on the search path that may converge slowly to the solution. Multi bit flipping, on the other hand, is equivalent to taking larger steps on the search path. The hybrid bit flipping (also referred to as mode switching) usually starts with multi bit flipping and switches to single bit flipping as the search approaches the solution [5]. To determine how many bits are to be flipped in a multi bit flipping process, a threshold θ is defined for inversion function defined in (5), and the bits whose inversion function fall below the threshold θ are flipped. The hybrid and multi bit flipping methods have been shown to have better performances comparing to single bit flipping described here [5], [6], [18].

The objective function defined in (4) is not a linear function and may have many local maxima in addition to the local maximum that corresponds to the transmitted codeword. During the search for the transmitted codeword, it is possible that the search point may be trapped on a local maximum that does not correspond to the transmitted codeword. One method to escape the so-called trap, proposed in [5], is to add a perturbation to a trapped search by switching from single bit flipping to multi bit flipping described above. A disadvantage to this method is that a multi bit flipping process may miss the desired solution due to the fact that the search steps may be too large. Another method to escape the traps, proposed in [6], is by adding a random perturbation to the inversion function (5):

$$E_k = x_k . y_k - (w \sum_{s_j^{(k)} \in S_k} \left(2s_j^{(k)} - 1\right) + q_k \tag{6}$$

where $w \in R^+$ and q_k is a Gaussian random variable with zero mean and variance $\sigma^2 = \alpha^2 \frac{N_0}{2}$ with $0 < \alpha \leq 1$. The study [6] found that w and α are close to 1, and their optimal values are code dependent, however, they are weakly dependent on signal to noise ratio. The decoding method proposed in [6] is referred to as noisy gradient descent bit flipping (NGDBF) algorithm. Single bit flipping and multi bit flipping techniques can both be performed. The study also used adaptive threshold technique when multi bit flipping method was used. As far as error performance of NGDBF method is concerned, the method outperformed the regular GDBF method in almost all scenarios, some of them, however, at the expense of more iterations. A more recent study [19] added more solid theoretical framework to GDBF and NGDBF algorithms that had been otherwise empirically analyzed. In the study, the authors proposed a decoding technique that is based on the conditional probability of message error given the syndrome information, defined in (6). The decoding technique proposed in [19] is also based on maximum likelihood decoding principle and is referred to as probabilistic local maximum likelihood bit flipping (PLMLBF) algorithm.

2.3. Improved NGDBF algorithm

To the best of our knowledge, none of the decoding algorithms described in this paper, considered real life applications where AWGN channel may not appropriately model the channel over which codewords are transmitted. Some of the more modern applications in wireless communications realistically occur where other channel models such as Rayleigh fading channel may be better suited to model the communication channel. In those cases, the error performance of BF decoder and its variants such as NGDBF may be degraded. Additionally, while decoding in non-Gaussian channels such as Rayleigh fading channel, BF method and its variants may suffer large processing delays due to the fact that they may have to decode using greater number of iterations than normal. To put this issue in perspective, one application is here described in the context of intelligent transportation system (ITS), i.e., vehicle to vehicle (V2V) communications.

V2V communication enables vehicles to wirelessly exchange information about their speed, location and heading in order to avoid crashes. The information is broadcast by each vehicle and is received by all other vehicles in a given range. Given crash avoidance is a safety critical application, V2V communication must have high reliability and low latency. The required latency for V2V communication is application dependent and ranges from 10-100 ms [20]. Many V2V applications, described in [20] and [21], may arise while vehicles are moving at relatively fast speed with no direct line of sight. Such mobile channels may experience fading due to Doppler shifts caused by the relative speed of vehicles and may be better described using Rayleigh fading channel model.

To decode LDPC encoded codewords transmitted over Rayleigh fading channel, we propose modifications to inversion function (5):

$$E_k = x_k . y_k - \left(h_k \sum_{s_j^{(k)} \in S_k} \left(2s_j^{(k)} - 1\right) + q_k\right) \tag{7}$$

where h_k's are random variables with Rayleigh probability distribution function and are proportional to the channel coefficients, described in sub section 2.1. And q_k's are Gaussian random variables with zero mean and variance of $\alpha^2 N_0/2$, where $0 < \alpha \leq 1$. The perturbation coefficients h_k and q_k are selected such that they exhibit and track the characteristics of a Rayleigh fading channel as well as added receiver noise, respectively. However, while it is assumed that channel state information (CSI) is known during the decoding process, for practical purposes, h_k's is kept relatively constant across the transmitted frames used in our simulations. Channel estimation in modern cellular communication systems (4G and 5G) is typically performed at OFDM symbol level or subframe level [22], [23]. This assumption also holds true for V2V applications where number of demodulation reference symbols (DMRS) are increased from 3 to 4 to better adapt to fast channel variations due to speed of vehicles [24], [25]. Even with the advances in physical layers and channel estimation techniques for mobile applications, changing perturbation coefficients at transmitted bit level is extremely challenging [26]. Therefore, we found it reasonable to maintain perturbation coefficients relatively constant across our simulated frames. For our simulations, the parity check matrix H was a 21×21 square matrix corresponding with type 1 projective geometry LDPC (PG-LDPC) code (21,11) with row and column weight of 5. Incorporating the negative sign in (7) ensured that the objective function is maximized when all parity check equations are satisfied along with the maximized cross correlation of the transmitted bit and soft decision received bit. The perturbation coefficients add to the complexity of the inversion function as it was originally introduced in (5). Appropriately selecting values of h_k and q_k is extremely important in order to optimize performance of the decoder. In section 3 we will present the results of our complexity and sensitivity analysis on how different values of h_k and q_k impact the performance of the proposed decoder in terms of rate of convergence.

Our proposed decoding algorithm can perform single bit flipping, multi bit flipping and hybrid (mode switching) bit flipping. The second modification we made was to the manner by which mode switching occurs. Our proposed mode switching method is as following; Start the decoding process by initially flipping m bits. As the number of iterations becomes half of the maximum number of iterations I_{max}, lower the number of bits to be flipped to $\frac{m}{2}$. And finally, once all individual inversion functions $E_k's$ become positive, switch to single bit flipping (m=1). This method ensures the large steps, associated with multi bit flipping, do not miss the desired local maximum, as a known characteristic of multi bit flipping [5], [6], and at the same time ensures the rate of convergence is faster than it is of single bit flipping method. The proposed decoding process:

Improved NGDBF algorithm

Step 1: Compute all syndrome components in $s = (s_1, s_2, \dots, s_J) = z.H^T(1)$. If all s_i's are zero output z as correctly received codeword, otherwise move on to next step (step 2).

Step 2: Determine the inversion function $E_k = x_k.y_k - (h_k \sum_{s_j^{(k)} \in S_l} (2s_j^{(k)} - 1) + q_k)$. Initiate decoding by flipping m bits whose corresponding inversion function is the least.

Step 3: Put the flipped bits in step 2 aside so they do not get flipped in the next immediate iteration.

Step 4: If the current number of iteration I is equal to $\frac{I_{max}}{2}$ then lower the number of bits to be flipped to $\frac{m}{2}$. If the individual $E_k's$ are no longer negative then switch to single bit flipping mode, i.e., flip the bit whose inverstion function is the least. Otherwise keep multi bit flipping mode.

Step 4: Repeat the process until all syndromes are satisfied or the number of iterations has reached I_{max}.

3. SIMULATION RESULTS AND DISCUSSION

3.1. Channel model

To simulate a wireless communication use case, where codewords are transmitted over Rayleigh fading channel, a V2V communication scenario was envisioned where relative speed of approaching vehicles varied from 1 mi/h (0.5 m/s) to 120 mi/h (53.6 m/s) corresponding to Doppler frequencies f_m of 10 Hz to 1,000 Hz, respectively. Table 1 shows the mean of all Rayleigh distributed random variables used in our simulations as channel fading coefficients. In all scenarios, mean of the generated Rayleigh distributed coefficients are approximately zero or very close to zero as it should be [18]. Figure 1 also shows the envelop of the signal level (transmitted over Rayleigh channel) corresponding to Doppler shifts of 10 Hz, 100 Hz and 1,000 Hz. As Doppler frequency (shift) increases the level crossing rate also increases which demonstrates slow to fast fading characteristics of the Rayleigh channel [27]. For clarity, only one realization of the signal envelopes (2^{11} samples) corresponding to Doppler frequencies of 10 Hz, 100 Hz, and 1,000 Hz are shown in Figure 1.

Table 1. Mean of generated Rayleigh distributed channel coefficients

Doppler Shift (Hz)	10	100	1,000
Mean of fading coefficients	0.003	0.010	0.034

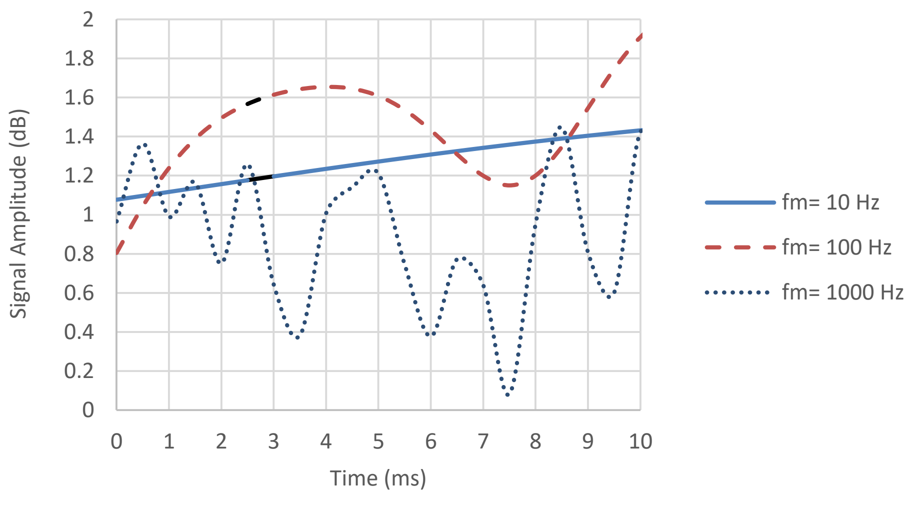

Figure 1. Signal amplitudes in three scenarios corresponding to Doppler frequencies of 10 Hz, 100 Hz, and 1,000 Hz

Observing the fading rates displayed in Figure 1, we noticed that when the fading rate is the slowest (corresponding to Doppler frequency f_m of 10 Hz) the fade values change at such a slow rate that the signal level is relatively constant across small enough transmit time intervals. Assuming channel state information is known, channel coefficients become highly correlated, and the channel is practically Gaussian [28]. The other extreme is when the fading rate is the fastest (corresponding to Doppler frequency f_m of 1,000 Hz), and the fading is sufficiently fast for the adjacent symbols to be almost uncorrelated. As the Doppler frequency increases the amount of correlation (of the received symbols) transitions from highly correlated to almost fully uncorrelated. To quantify this phenomenon, normalized Doppler frequency f_mT is usually used where T is the symbol rate [28]. Obviously, where the transition (from fully correlated signals to uncorrelated ones) occurs depends on the system design, the symbol rate, and the depth of interleaving of the code bits. For our analysis, we performed the decoding at Doppler frequencies that are decade apart, i.e., 10 Hz, 100 Hz and 1,000 Hz, respectively.

3.2. Characteristics and performance of the proposed decoder

3.2.1. Complexity analysis of the perturbation coefficients h_k and q_k

Our investigation has shown that rate of convergence (to the actual solution, i.e., correctly decoded codeword) is a function of the perturbation coefficients h_k and q_k. In other words, decoding performance is sensitive to the choice of both h_k and q_k. To demonstrate this sensitivity, we first kept $h_k = 1$ and varied q_k. Figure 2 shows the rate of convergence of the proposed decoder as a function of q_k. The case with $q_k = 0$ (blue trace in Figure 2) is the non-perturbed or legacy case and is considered the baseline. It can be seen that the decoder has the fastest rate of convergence when q_k exhibits the same characteristics of the added noise. This finding is in good agreement with the finding in [6] as well. On the other hand, if $q_k's$ are not properly chosen the performance of the decoder is even worse than the legacy (non-perturbed) decoder. The optimum q_k did not show any sensitivity to Rayleigh channel coefficients (or Doppler frequency) nor to the signal to noise ratio. Figure 2 shows the rate of convergence for the worst case signal to noise ratio in our study that is SNR=0 dB.

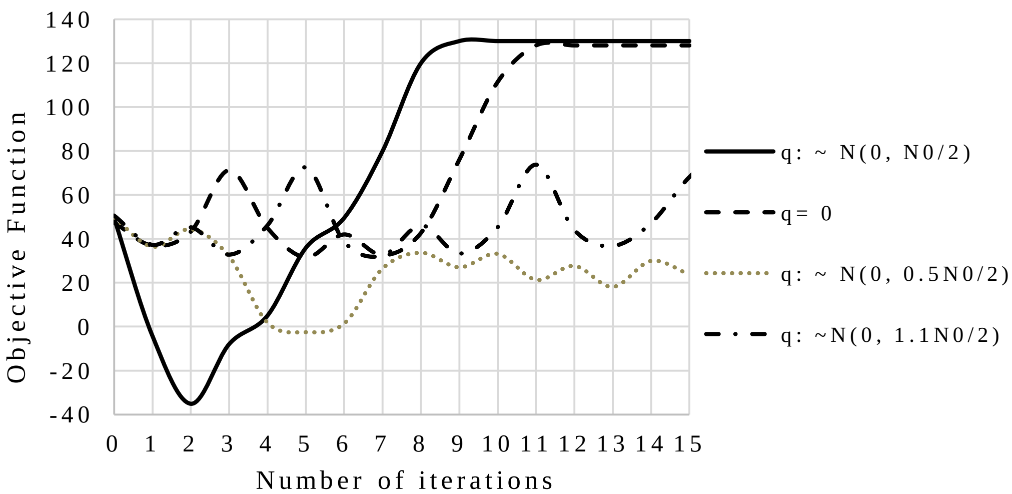

Figure 2. Rate of convergence as a function of perturbation coefficient q_k, $SNR = 0\ dB$

The next step was to determine the value for the second perturbation coefficient h_k that optimizes the decoder performance. To do so, we kept the coefficient q_k the random variable drawn from Gaussian distribution that produced the fastest rate of convergence shown in Figure 2 and then varied the h_k in multiple scenarios with different Doppler frequencies. Figures 3 and 4 show the rate of convergence as a function of perturbation coefficient h_k where Doppler frequency was 10 Hz, and 1,000 Hz, respectively with no measurable difference between Doppler frequencies of 10 Hz and 100 Hz. It can be seen that the decoder has the fastest rate of convergence when h_k is approximately equal to the channel coefficient R_k in both scenarios. The perturbation coefficients h_k did not demonstrate any measurable sensitivity to SNR. The proposed decoder's performance is shown for the worst case signal to noise ratio in our study that is SNR=0 dB.

Our introduced perturbation coefficients q_k and h_k, when properly selected, improve the performance of the decoder as compared to the non-perturbed or legacy decoder and as shown in Figures 2 through 4. However, the improvement comes at a cost and that is both the receiver noise variance and channel coefficients have to be known by estimators that are external to the decoder, hence added complexity to the decoder and receiver design. With the added complexity in mind, our proposed decoder is better suited for applications such as V2V, described in section 2, where capability to estimate noise variance and channel coefficients is assumed.

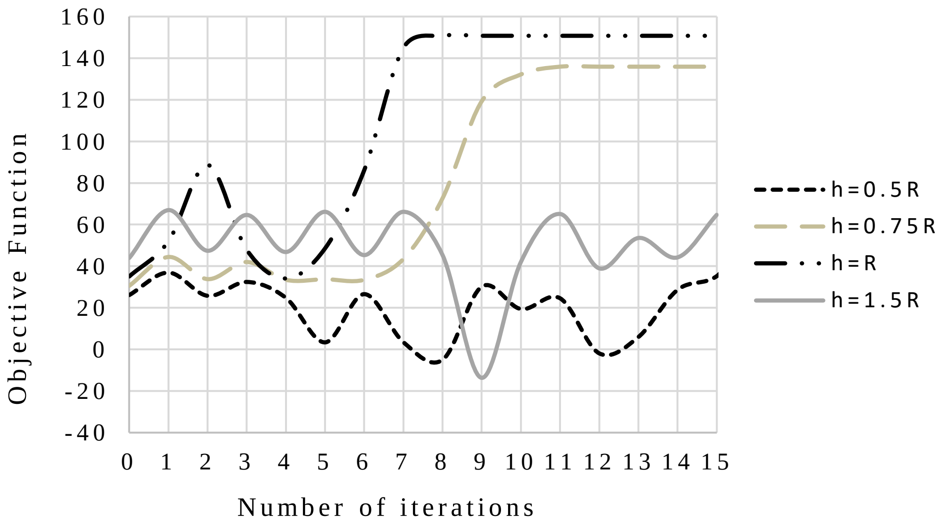

Figure 3. Rate of convergence as a function of perturbation coefficient h_k, $SNR = 0\ dB$, $f_m = 10\ Hz$

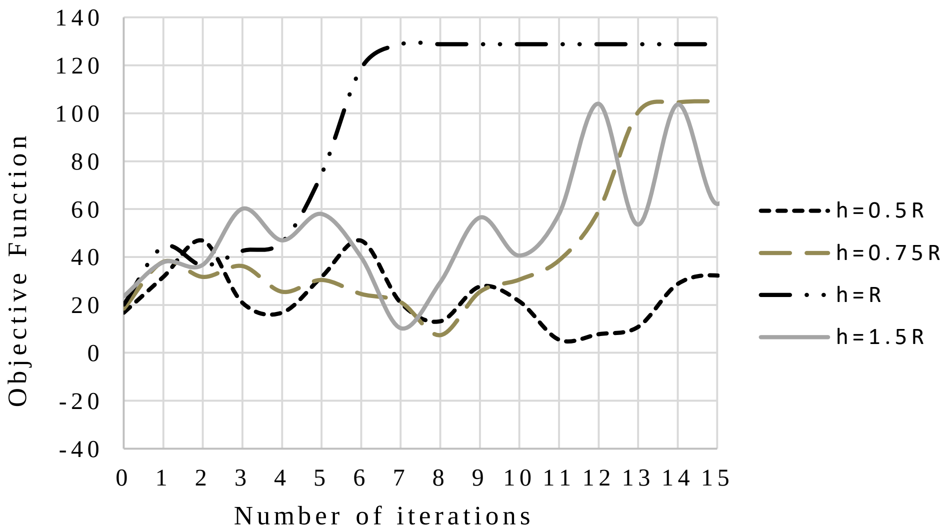

Figure 4. Rate of convergence as a function of perturbation coefficient h_k, $SNR = 0\ dB$, $f_m = 1{,}000\ Hz$

3.2.2. Error performance of the proposed decoder

Figure 5 shows the error performance of our proposed decoder comparing to NGDBF decoder while codewords are transmitted over Rayleigh fading channel with Doppler frequency of 10 Hz. Our proposed decoder outperforms the legacy NGDBF algorithm across the full range of measured signal to noise ratios. Another performance metric, that becomes especially important in use cases with low latency requirements, is the decoder's rate of convergence since it directly relates to the overall processing delay and ultimately end to end latency. Figure 6 depicts the average number of iterations over hundred frames to achieve the error performance of our proposed decoder shown in Figure 5. Again, our proposed decoder outperforms the legacy NGDBF decoder across the full range of measured signal to noise ratios. On average, our proposed decoder shows 19.98% improvement in average number of iterations required to obtain the said error performance.

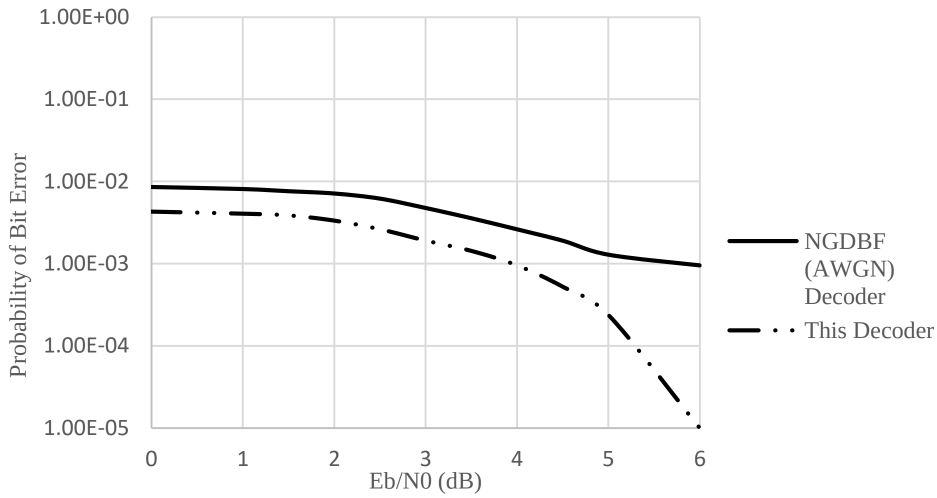

Figure 5. BER curves of the improved and legacy NGDBF decoders over 100 frames, max iterations: 15, $f_m = 10\ Hz$

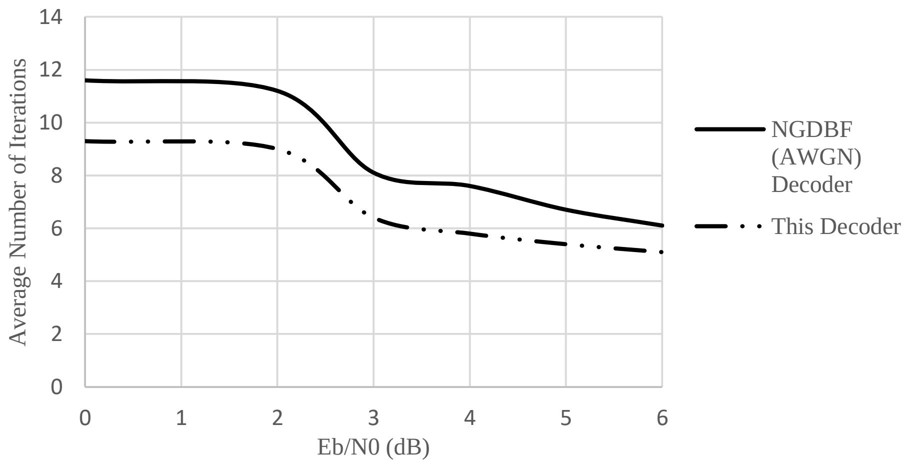

Figure 6. Average rate of convergence for the improved and legacy NGDBF decoders over 100 frames, Max iteration: 15, $f_m = 10\ Hz$

There was no measurable difference in the performance of the decoders over Rayleigh fading channel with $f_m = 100\ Hz$. Our decoder's performance, however, was improved in case of $f_m = 1{,}000\ Hz$ as it can be seen in Figure 7. The improvement should be expected since LDPC codes are random error correcting codes, and they perform best if introduced errors into the codewords are random in nature and statistically independent. As the Doppler frequency increases the fading rate becomes sufficiently fast so the adjacent symbols are uncorrelated, and the errors appear random. Our finding agrees with the studies in [11] and [28] as well. Figure 8 depicts the average number of iterations over hundred frames to obtain the error performance of our proposed decoder shown in Figure 7. Again, our proposed decoder outperforms the legacy NGDBF decoder across the full range of measured signal to noise ratios. On average, our proposed decoder shows 16.87% improvement in average number of iterations required to obtain the said error performance.

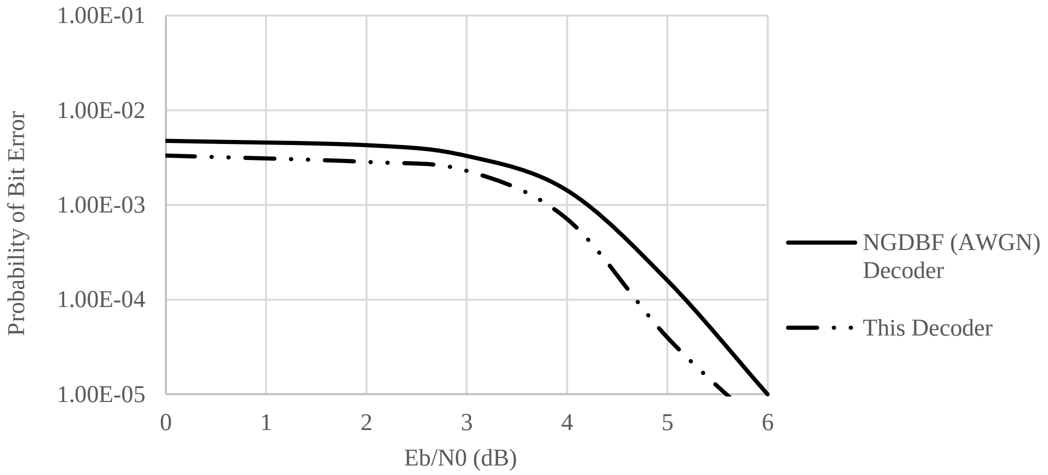

Figure 7. BER curves of the improved and legacy NGDBF decoders over 100 frames, max iterations: 15, $f_m = 1{,}000\ Hz$

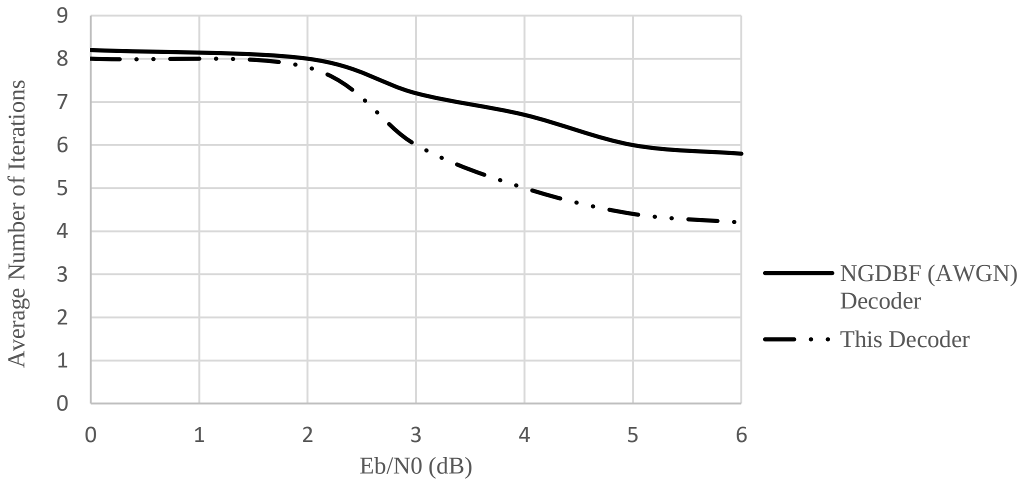

Figure 8. Average rate of convergence for the improved and legacy NGDBF decoders over 100 frames, max iteration: 15, $f_m = 1{,}000\ Hz$

The results of our simulation, can also be analyzed from an information theory perspective. Codewords that are transmitted over a Rayleigh fading channel may experience block error (in case of slow fading) when transmitted frames experience deep fades, or they can undergo a fast fading when introduced errors are less correlated and error patterns are more random in nature. In case of slow fading, corresponding with Doppler frequencies of 10 Hz and 100 Hz in our simulations, the outage probability can be approximated when knowledge of channel state information is available at the receiver. The outage probability in this case has been shown to be inversely proportional to signal to noise ratio at a given transmission rate [29]–[31]. It is equivalent to the fact that the effective channel capacity increases as signal to noise ration increases. Our results shown in Figures 5 and 7 also show the same trend and agree well with established results in [29] and [30]. In case of fast fading scenario, corresponding with Doppler frequency of 1,000 Hz in our simulation, channel capacity has been shown to be directly proportional to signal to noise ratio [29] as well and the error probability decreases as the signal to noise ratio increases. Additionally, [28] has shown that the error performance improves as channel transitions from slow fading to fast fading. Our proposed decoder's performance improves by a factor of 10 (probability of bit error of $1x10^{-5}$ at SNR of 5.5 dB in Figure 7) when fast fading is experienced comparing to probability of bit error of $1x10^{-4}$ at the same SNR as shown in Figure 5 when slow fading scenario is experienced. Again, our results are in very good agreement with results in [28].

4. CONCLUSION

In this paper, we proposed an improved noisy gradient descent bit flipping decoder that performs well over Rayleigh fading channel that is a more realistic channel model for a lot of modern applications with low latency requirement. We demonstrated that our proposed decoder outperformed both non-perturbed and legacy NGDBF decoders that are designed for AWGN channels. Measurable improvements are observed in terms of error performance and processing delay of the decoder. Additionally, our proposed decoder outperformed legacy NGDBF decoder across a wide range of mobile channel conditions quantified in terms of Doppler frequency to simulate slow and fast fading channels. The fact that our decoder's performance improved as fade rate increased agrees very well with established results reported in other studies. Our empirical results also prove the hypothesis that the decoder performance is optimum (or near optimum) when the perturbation coefficients exhibit and track the statistical characteristics of the channel coefficients.

REFERENCES

[1] R. Gallager, "Low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962, doi: 10.1109/TIT.1962.1057683.
[2] N. Wiberg, H.-A. Loeliger, and R. Kotter, "Codes and iterative decoding on general graphs," 2001, doi: 10.1109/ISIT.1995.550455.
[3] R. J. McEliece, D. J. C. MacKay, and Jung-Fu Cheng, "Turbo decoding as an instance of pearl's 'belief propagation' algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 2, pp. 140–152, 1998, doi: 10.1109/49.661103.
[4] V. D. Kolesnik, "Probabilistic decoding of majority codes," *Problemy Peredachi Informatsii*, vol. 7, no. 3, pp. 3–12, 1971.
[5] T. Wadayama, K. Nakamura, M. Yagita, Y. Funahashi, S. Usami, and I. Takumi, "Gradient descent bit flipping algorithms for decoding LDPC codes," *IEEE Transactions on Communications*, vol. 58, no. 6, pp. 1610–1614, Jun. 2010, doi: 10.1109/TCOMM.2010.06.090046.
[6] G. Sundararajan, C. Winstead, and E. Boutillon, "Noisy gradient descent bit-flip decoding for LDPC codes," *IEEE Transactions on Communications*, vol. 62, no. 10, pp. 3385–3400, Oct. 2014, doi: 10.1109/TCOMM.2014.2356458.
[7] O. Al Rasheed, P. Ivanis, and B. Vasic, "Fault-tolerant probabilistic gradient-descent bit flipping decoder," *IEEE Communications Letters*, vol. 18, no. 9, pp. 1487–1490, Sep. 2014, doi: 10.1109/LCOMM.2014.2344031.
[8] H. Cui, J. Lin, and Z. Wang, "An improved gradient descent bit-flipping decoder for LDPC codes," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 8, pp. 3188–3200, Aug. 2019, doi: 10.1109/TCSI.2019.2909653.
[9] H. Cui, J. Lin, and Z. Wang, "Information storage bit-flipping decoder for LDPC codes," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 11, pp. 2464–2468, Nov. 2020, doi: 10.1109/TVLSI.2020.3009270.
[10] D. Declercq, C. Winstead, B. Vasic, F. Ghaffari, P. Ivanis, and E. Boutillon, "Noise-aided gradient descent bit-flipping decoders approaching maximum likelihood decoding," in *2016 9th International Symposium on Turbo Codes and Iterative Information Processing (ISTC)*, Sep. 2016, pp. 300–304, doi: 10.1109/ISTC.2016.7593125.
[11] E. Biglieri, J. Proakis, and S. Shamai, "Fading channels: information-theoretic and communications aspects," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2619–2692, 1998, doi: 10.1109/18.720551.
[12] D. J. Young and N. C. Beaulieu, "On the generation of correlated Rayleigh random variates by inverse discrete fourier transform," in *Proceedings of ICUPC - 5th International Conference on Universal Personal Communications*, 1996, vol. 1, pp. 231–235, doi: 10.1109/ICUPC.1996.557875.
[13] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 44, no. 3, pp. 927–946, May 1998, doi: 10.1109/18.669123.
[14] Yu Kou, Shu Lin, and M. P. C. Fossorier, "Low density parity check codes based on finite geometries: a rediscovery," ISlT 2000, Sorrento, Italy, 2000, doi: 10.1109/ISIT.2000.866498.
[15] J. Zhang and M. P. C. Fossorier, "A modified weighted bit-flipping decoding of low-density parity-check codes," *IEEE Communications Letters*, vol. 8, no. 3, pp. 165–167, Mar. 2004, doi: 10.1109/LCOMM.2004.825737.
[16] J. Cho and W. Sung, "Adaptive threshold technique for bit-flipping decoding of low-density parity-check codes," *IEEE Communications Letters*, vol. 14, no. 9, pp. 857–859, Sep. 2010, doi: 10.1109/LCOMM.2010.072310.100599.
[17] N. Miladinovic and M. P. C. Fossorier, "Improved bit-flipping decoding of low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1594–1606, Apr. 2005, doi: 10.1109/TIT.2005.844095.
[18] M.-R. Li, L.-M. Jhuang, and Y.-L. Ueng, "A modified gradient descent bit flipping decoding scheme for LDPC codes," in *2017 IEEE International Workshop on Signal Processing Systems (SiPS)*, Oct. 2017, pp. 1–6, doi: 10.1109/SiPS.2017.8109969.
[19] R. R. Mathews and C. Winstead, "Decoding LDPC codes with probabilistic local maximum likelihood bit flipping," in *2020 International Symposium on Information Theory and Its Applications (ISITA)*, 2020, pp. 205–209.
[20] K. Lee, J. Kim, Y. Park, H. Wang, and D. Hong, "Latency of cellular-based V2X: perspectives on TTI-proportional latency and TTI-independent latency," *IEEE Access*, vol. 5, pp. 15800–15809, 2017, doi: 10.1109/ACCESS.2017.2731777.
[21] A. Kanavos, D. Fragkos, and A. Kaloxylos, "V2X communication over cellular networks: capabilities and challenges," *Telecom*, vol. 2, no. 1, pp. 1–26, Jan. 2021, doi: 10.3390/telecom2010001.
[22] Qingchuan Zhang, Xudong Zhu, Tao Yang, and Jin Liu, "An enhanced DFT-based channel estimator for LTE-A uplink," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 9, pp. 4690–4696, Nov. 2013, doi: 10.1109/TVT.2013.2264293.
[23] A. Kiayani, L. Anttila, Y. Zou, and M. Valkama, "Channel estimation and equalization in multiuser uplink OFDMA and SC-FDMA systems under transmitter RF impairments," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 82–99, Jan. 2016, doi: 10.1109/TVT.2015.2397277.
[24] R. Sattiraju, A. Weinand, and H. D. Schotten, "Channel estimation in C-V2X using deep learning," in *2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, Dec. 2019, pp. 1–5, doi: 10.1109/ANTS47819.2019.9117972.
[25] R. Munagala and R. Budhiraja, "Channel estimator designs for emerging 5G new radio cellular systems," in *2020 National Conference on Communications (NCC)*, Feb. 2020, pp. 1–6, doi: 10.1109/NCC48643.2020.9056078.
[26] S. Zeadally, M. A. Javed, and E. Ben Hamida, "Vehicular communications for ITS: standardization and challenges," *IEEE Communications Standards Magazine*, vol. 4, no. 1, pp. 11–17, Mar. 2020, doi: 10.1109/MCOMSTD.001.1900044.
[27] A. Abdi, K. Wills, H. A. Barger, M.-S. Alouini, and M. Kaveh, "Comparison of the level crossing rate and average fade duration

of Rayleigh, rice and nakagami fading models with mobile channel data," in *Vehicular Technology Conference Fall 2000. IEEE VTS Fall VTC2000. 52nd Vehicular Technology Conference (Cat. No.00CH37152)*, 2000, vol. 4, pp. 1850–1857, doi: 10.1109/VETECF.2000.886139.
[28] R. van Nobelen and D. P. Taylor, "Analysis of the pairwise error probability of non-interleaved codes on the Rayleigh-fading channel," in *1994 IEEE GLOBECOM. Communications: The Global Bridge*, 1994, pp. 181–185, doi: 10.1109/CTMC.1994.512601.
[29] A. Goldsmith, "Capacity of broadcast fading channels with variable rate and power," in *Proceedings of GLOBECOM'96. 1996 IEEE Global Telecommunications Conference*, 1996, pp. 92–96, doi: 10.1109/GLOCOM.1996.586775.
[30] P. Varzakas, "Channel capacity per user in a power and rate adaptive hybrid DS/FFH-CDMA cellular system over Rayleigh fading channels," *International Journal of Communication Systems*, vol. 25, no. 7, pp. 943–952, Jul. 2012, doi: 10.1002/dac.1298.
[31] M. Amjad, L. Musavian, and M. H. Rehmani, "Effective capacity in wireless networks: a comprehensive survey," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 4, pp. 3007–3038, 2019, doi: 10.1109/COMST.2019.2929001.

BIOGRAPHIES OF AUTHORS

Reza Biazaran is a PhD candidate in the department of electrical and computer engineering the George Washington University. His research interests are wireless communications systems and networks, error correction coding, and electromagnetic compatibility. Email: rbiazaran@gwu.edu.

Hermann Joseph Helgert holds a PhD in electrical engineering from the State University of New York. He is a professor in the department of electrical and computer engineering, the George Washington University. He conducts research in the areas of communication networks, with specialization in networks employing wireless communication channels. Areas of concentration include transmission systems for satellite and cellular wireless channels and security protocols. Current topics include the development of adaptive coding and modulation schemes, multicarrier systems, position location algorithms, and multiple access algorithms for wireless local area networks. Email: hhelgert@gwu.edu.