Low-complexity demapping algorithm for two-dimensional non-uniform constellations in ATSC 3.0

Yu-Sun Liu1 | Chih-Hao Chien1 | Shingchern D. You2

1 Department of Electronic Engineering, National Taipei University of Technology, Taipei, Taiwan
2 Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei, Taiwan

Correspondence
Yu-Sun Liu, Department of Electronic Engineering, National Taipei University of Technology, Taipei, Taiwan.
Email: ysliu@ntut.edu.tw

Funding information
Ministry of Science and Technology, Taiwan, Grant/Award Number: 109-2637-E-027-006

Abstract
The Advanced Television Systems Committee (ATSC) finalized its next-generation digital terrestrial broadcast standard in 2016, known as ATSC 3.0. In order to enhance spectral efficiency, the standard employs two-dimensional non-uniform constellations (2D-NUCs) for signal mapping. As a result, a 2D demapper, with computational complexity much higher than one-dimensional (1D) demappers, has to be designed for the receiver. In this paper, a suboptimal low-complexity 2D-demapping method is proposed for ATSC 3.0. The proposed method takes advantage of the 2D-NUC’s feature that constellation points with same most significant bits tend to locate in the same subregion. Simulation results show that compared with the conventional Max-Log-MAP 2D demapping, the proposed method achieves at least 74% complexity reduction with negligible performance loss. Moreover, compared with another 2D demapper specially designed for ATSC 3.0, the proposed method dispenses with high-complexity exponential and logarithmic operations at a small performance cost of at most 0.14 dB.

1 | INTRODUCTION

The Advanced Television Systems Committee (ATSC) finalized its next-generation digital terrestrial television standard, known as ATSC 3.0, in 2016 [1]. In the standard, non-uniform constellations (NUCs) [2–4], instead of the conventional uniform quadrature-amplitude modulation (QAM) constellations, are used to enhance spectral efficiency. There are two types of NUCs in ATSC 3.0: one-dimensional NUCs (1D-NUCs) and two-dimensional NUCs (2D-NUCs). Both are independently optimized for each code rate based on its target signal-to-noise ratio (SNR) [5]. 1D-NUCs are designed by relaxing the uniform constraint of conventional QAM constellations. In other words, a 1D-NUC with modulation order \( M \) is composed of two 1D non-uniform pulse amplitude modulations (PAMs) with modulation order \( \sqrt{M} \): one for in-phase component and the other for quadrature component. Since the in-phase and quadrature components of constellation points are independent, the demapping process of 1D-NUCs can be accomplished by applying 1D demappers for \( \sqrt{M} \) points [6, 7] to in-phase and quadrature components separately. To further increase spectral efficiency, 2D-NUCs are designed by removing the independence constraint in 1D-NUCs. Since the in-phase and quadrature components of 2D-NUCs are no longer independent, the demapping process cannot be decomposed into two independent 1D demappers. Instead, a 2D demapper for \( M \) points, which takes into account the in-phase and quadrature components simultaneously, is required. Note that one 2D demapper for \( M \) points has significantly higher complexity than two 1D demappers for \( \sqrt{M} \) points [8–10], especially for high-order modulation. Therefore, demapping process for high-order 2D-NUCs becomes one of the bottlenecks in ATSC 3.0 receiver implementation.

In straightforward 2D demapping, Euclidean distances from the received signal to all constellation points are required to compute the demapper output log-likelihood ratio (LLR). To reduce the computational complexity of 2D demappers, most researches focused on decreasing the constellation points (and thus the distances) included in LLR computation. There are three main approaches. The quadrant search reduction (QSR) method was first proposed in [11–13] for the 2D demapper in second-generation terrestrial digital video broadcasting (DVB-T2) [14] and was also employed in [15, 16] for \( M \)-ary phase shift keying (PSK), uniform QAM, and digital video broadcasting via satellite second-generation extensions.
In this paper, we propose a low-complexity 2D-demapping method for ATSC 3.0, in which a QSR method is employed together with CSR to reduce the computational complexity. In the proposed method, the signal plane is divided into subregions. For each subregion, the subset which contains all near-the constellation points are adaptive, the amount of computational overhead is output sensitive. In contrast, Fuentes’ method significantly reduces 2D-demapping complexity with a performance degradation of less than 0.1 dB for ATSC 3.0. The concept of the third approach is to consider only constellation points in the neighbourhood of the received signal. Barrueco et al. [19] proposed an adaptive-subregion demapping algorithm which defines an adaptive-size square region around the received signal based on noise power and then considers only the constellation points inside the region. The adaptive-subregion demapping algorithm, together with CSR, further reduces the number of points in LLR computation, but at the same time imposes the computational overhead for identifying all constellation points inside the subregion. As the size and location of the subregion are adaptive, the amount of computational overhead is output sensitive. In contrast, Fuentes’ method requires fixed amount of computation, which is preferred in hardware implementation.

In this paper, we propose a low-complexity 2D-demapping method for ATSC 3.0, in which a QSR method is employed together with CSR to reduce the computational complexity. In the proposed method, the signal plane is divided into subregions. For each subregion, a subset which contains all nearest constellation points for received signals in the subregion is found. The subsets are then further downsized by the CSR method. Thus, when a signal is received, the LLR is computed by using only the distances to the constellation points in the corresponding subset. Compared with the Log-MAP algorithm, the Max-Log-MAP algorithm [20, 21], and Fuentes’ method, the proposed 2D-demapping method is shown to have a significantly lower complexity. Moreover, the subsets in the proposed method, unlike those in the adaptive-subregion demapping algorithm [19], are determined beforehand. Therefore, the proposed method has fixed complexity and is suitable for hardware implementation. Finally, simulation results show that compared with the Max-Log-MAP algorithm, the complexity reduction is achieved with negligible impact on bit error rate (BER) performance.

The outline of the paper is as follows. The proposed method is described in Section 2. In Section 3, the BER performance of the proposed method is compared with those of the Log-MAP algorithm, the Max-Log-MAP algorithm, and Fuentes’ method for additive white Gaussian noise (AWGN) channel and independent and identically distributed (i.i.d.) Rayleigh fading channel. Complexity comparisons are presented in Section 4. Finally, conclusions are drawn in Section 5.

### 2 PROPOSED DEMAPPING METHOD

Low-density parity-check (LDPC) codes with code rates ranging from 2/15 to 13/15 were adopted for the physical layer specification of ATSC 3.0 [22]. For each code rate, three 2D-NUCs with modulation orders 16, 64, and 256 are designed based on target SNR. We will focus on 2D-NUCs with the highest modulation order, 2D-256NUCs, in the rest of the paper since it has the highest complexity. Figure 1 plots the 2D-256NUCs for code rates 4/15, 8/15, and 12/15. A 2D-256NUC is composed of 256 points, \( h_0, h_1, \ldots, h_{255} \), each representing 8 coded bits, \( b_0, b_1, \ldots, b_7 \). In the Log-MAP algorithm, the optimal maximum-likelihood (ML) LLR is computed for each coded bit \( b_\ell \) as

\[
\Lambda_\ell = \ln \frac{p(b_\ell = 1|y, \sigma^2_r)}{p(b_\ell = 0|y, \sigma^2_r)} = \ln \frac{\sum_{s \in A_{b_\ell=1}} \exp \left( -\frac{|y - h_s|^2}{2\sigma^2_r} \right)}{\sum_{s \in A_{b_\ell=0}} \exp \left( -\frac{|y - h_s|^2}{2\sigma^2_r} \right)}, \tag{1}
\]
where \( y \) and \( \sigma^2_y \) are the received signal and the noise variance after channel equalization, respectively, and \( \mathcal{A}_{b_\ell=0} \) and \( \mathcal{A}_{b_\ell=1} \) are the sets of constellation points with \( b_\ell = 0 \) and \( b_\ell = 1 \), respectively. Fuentes’ demapping method is based on (1) with both \( \mathcal{A}_{b_\ell=\epsilon} \) replaced by their own subsets. It is noted that both the Log-MAP algorithm and Fuentes’ method require a large number of high-complexity exponential and logarithmic operations and thus, are resource intensive. One popular way to solve this problem is to approximate both summations in (1) by their most significant terms to remove the exponential and logarithmic operations. The resulting Max-Log-MAP algorithm computes the Max-log LLR for each coded bit as

\[
\hat{x}_\ell = \frac{1}{2\sigma^2_y} \left( \min_{c \in \mathcal{A}_{I_{y}=1}} |y-c|^2 - \min_{c \in \mathcal{A}_{I_{y}=0}} |y-c|^2 \right), \tag{2}
\]

which is a good approximation of the ML LLR, especially at high SNRs. It is observed that no exponential and logarithmic operations are required in the Max-Log-MAP algorithm. However, to obtain the Max-Log LLRs for all coded bits, the demapper still requires computing Euclidean distances from the received signal to all constellation points. In this paper, we propose a demapping method which significantly reduces the number of distances computed.

2D-NUCs in ATSC 3.0 are quadrant symmetric, and the quadrant of a constellation point is determined by the two most significant bits, \( b_0 \) and \( b_1 \). Thus, the bit-LLRs of a received signal \( y \), which contains one subset of constellation points. Let \( R_{00}, R_{01}, R_{10}, \) and \( R_{11} \) denote the region which contains \( \mathcal{B}_{00}, \mathcal{B}_{01}, \mathcal{B}_{10}, \) and \( \mathcal{B}_{11} \), respectively. The sets \( \mathcal{B}_{00}, \mathcal{B}_{01}, \mathcal{B}_{10}, \mathcal{B}_{11} \), the lines \( L_1(I, Q), L_2(I, Q), \) and regions \( R_{00}, R_{01}, R_{10}, R_{11} \) are plotted in Figure 2 for code rates 4/15, 8/15, and 12/15. In the following, we will use received signals located inside \( R_{01} \) to illustrate the proposed method.

Denote the constellation points that attain the minimums in (2) for a received signal \( y \) by

\[
\hat{\mathcal{S}}_{I_{y}=\epsilon}(y) = \arg \min_{c \in \mathcal{A}_{I_{y}=\epsilon}} |y-c|^2. \tag{4}
\]

In other words, \( \hat{\mathcal{S}}_{I_{y}=\epsilon}(y) \) is the nearest constellation point with bit \( b_\ell \) equal to \( \epsilon \). Let \( \mathcal{C}_{I_{y}=\epsilon}(R_{01}) \) be the set of constellation points each of which is the nearest point \( \hat{\mathcal{S}}_{I_{y}=\epsilon}(y) \) for some received signal \( y \) in \( R_{01} \). We first consider the Max-log LLR for bits \( b_4 \), \( b_5 \), \( b_6 \), and \( b_7 \). Note that subset \( \mathcal{B}_{01} \) contains 16 constellation points corresponding to all possible combinations of bits \( b_4 \), \( b_5 \), \( b_6 \), and \( b_7 \). Thus, for most of the received signals \( y \) in \( R_{01} \), the nearest constellation point \( \hat{\mathcal{S}}_{I_{y}=\epsilon}(y) \) can be found in subset \( \mathcal{B}_{01} \) for \( \ell = 4, 5, 6, 7 \). The exception occurs only if the received signal \( y \) lies near the boundary of \( R_{01} \). In such cases, by the quadrant symmetry of the constellation, it can be easily proved that the nearest constellation point \( \hat{\mathcal{S}}_{I_{y}=\epsilon}(y) \) is still a constellation point in the first quadrant, that is, \( \hat{\mathcal{S}}_{I_{y}=\epsilon}(y) \in \mathcal{S} \). However, it may be an element of set \( \mathcal{S} \setminus \mathcal{B}_{01} \) and lie near the boundary of \( R_{01} \). Figure 3a shows \( \mathcal{C}_{b_4=0}(R_{01}) \) and \( \mathcal{C}_{b_4=1}(R_{01}) \) for code rate 12/15, which confirms the analysis above. Note that \( R_{01} \) is the upper left region in the first quadrant and \( \mathcal{B}_{01} \) is the set of 16 points inside.

Next, consider the Max-log LLR for bits \( b_2 \) and \( b_3 \). Note that for every constellation point in \( \mathcal{B}_{01} \), \( b_2 = 0 \) and \( b_3 = 1 \). Therefore, for most of the received signals \( y \) in \( R_{01} \), the nearest
FIGURE 3 Subsets for received signals in $R_{01}$, Code rate 12/15: (a) bit $b_6$; (b) bit $b_2$; (c) bit $b_0$; (d) all bits

From the constructive method above, it is easy to see that $A_{b_6=\cdot}(R_{01})$ is a subset $A_{b_6=\cdot}$ for all $c$ and $\cdot$. Moreover, the Max-log LLR value remains unchanged if the constellation point sets $A_{b_6=\cdot}(R_{01})$ in (2) are replaced by their subsets $C_{b_6=\cdot}(R_{01})$. Therefore, calculating the Max-log LLR by using

$$\frac{1}{2\sigma^2_h} \left( \min_{s \in C_{b_6=0}(R_{01})} |y-s|^2 - \min_{s \in C_{b_6=1}(R_{01})} |y-s|^2 \right)$$

(5)

can decrease the demapping complexity with no performance degradation (compared to the Max-Log-MAP algorithm). As for how many distances are required to calculate the Max-log LLRs for all eight bits, only Euclidean distances to all constellation points in

$$C(R_{01}) = \bigcup_{\ell=0,..,7} \bigcup_{c=0,1} C_{b_\ell=\cdot}(R_{01})$$

(6)

are required. Figure 3d shows $C(R_{01})$ for code rate 12/15. From the figure, it is obvious that the number of distance calculations in the proposed method is much smaller than 256.
The number of constellation points in $C_{b_i=0}(R_{01})$ can be further reduced by CSR method. If there exist clusters of constellation points in $C_{b_i=0}(R_{01})$, it suffices for the demapper to calculate a single distance from the received signal to a representing point in each cluster, and use it to approximate the distances to the other points in the same cluster. By retaining only the representing point for each cluster and discarding the rest, the CSR method decreases the number of points in $C_{b_i=0}(R_{01})$ and thus, the demapping complexity. However, the improvement comes at a price of BER performance degradation induced by distance approximation. Barrueco et al. showed that to achieve a balance between demapping complexity and BER performance, the distance between any two points in a cluster is better limited to less than 0.1 [19]. In this paper, we adopt 0.05 as the distance limit. It is noted that by this limit, no cluster forms in 2D-256NUCs for code rates 12/15 and 13/15. With a slight abuse of notation, $C_{b_i=0}(R_{00})$ and $C(R_{00})$ are used to denote the subsets after CSR in the rest of the paper if not specified otherwise.

Until now, we have used the received signal located in $R_{01}$ as an example to illustrate the proposed method. By the same argument, we can obtain the constellation sets $C_{b_i=0}(\cdot)$ and union of sets $C(\cdot)$ for the other three regions $R_{00}$, $R_{10}$, and $R_{11}$. Figure 4 shows $C_{b_i=0}(R_{00})$, $C_{b_i=1}(R_{00})$, and $C(R_{00})$ for code rate 8/15. Note that $R_{00}$ is the lower left region in the first quadrant and $B_{00}$ is the set of 16 points inside. By an argument similar to the one for $R_{01}$, it follows that before CSR, all 16 constellation points in $B_{00}$ are either an element of $C_{b_i=0}(R_{00})$ or $C_{b_i=1}(R_{00})$. From the figure, it is observed that after applying CSR, only 7 constellation points (out of 16) remain in $C(R_{00})$. In fact, the number of points in $C(R_{00})$ is reduced from 40 to 30 by CSR. Table 1 lists the boundary lines, $L_1(I,Q)$ and $L_2(I,Q)$, and the numbers of constellation points in $C(R_{00})$, $C(R_{01})$, $C(R_{10})$, and $C(R_{11})$ for all code rates.

### 3 | PERFORMANCE EVALUATION

A bit-interleaved coded modulation (BICM) module compliant with ATSC 3.0 is used in the simulation, which provides results comparable with ATSC 3.0 systems for AWGN channel and...
TABLE 1 Boundary lines and cardinalities of sets \( \mathcal{C}(\cdot) \)

| code rate | \( L_1(I, Q) \) | \( L_2(I, Q) \) | \( |\mathcal{C}(R_{00})| \) | \( |\mathcal{C}(R_{01})| \) | \( |\mathcal{C}(R_{10})| \) | \( |\mathcal{C}(R_{11})| \) |
|------------|-----------------|-----------------|-------------------|-------------------|-------------------|-------------------|
| 13/15      | \( I = 0.1 + Q = 0.55 \) | \( 10I - Q = 4.8 \) | 59                | 45                | 46                | 34                |
| 12/15      | \( I = 0.5 \)     | \( Q = 0.5 \)    | 59                | 46                | 45                | 35                |
| 11/15      | \( I = Q \)       | \( I^2 + Q^2 = 0.875^2 \) | 38                | 37                | 56                | 52                |
| 10/15      | \( I = Q \)       | \( I^2 + Q^2 = 0.875^2 \) | 40                | 36                | 53                | 45                |
| 9/15       | \( I = Q \)       | \( I^2 + Q^2 = 0.875^2 \) | 30                | 37                | 40                | 49                |
| 8/15       | \( I = Q \)       | \( I^2 + Q^2 = 0.875^2 \) | 23                | 29                | 34                | 43                |
| 7/15       | \( I = Q \)       | \( I^2 + Q^2 = 0.875^2 \) | 19                | 24                | 31                | 38                |
| 6/15       | \( I = Q \)       | \( I^2 + Q^2 = 0.875^2 \) | 14                | 17                | 23                | 27                |
| 5/15       | \( I = Q \)       | \( I^2 + Q^2 = 0.875^2 \) | 29                | 29                | 12                | 15                |
| 4/15       | \( I = Q \)       | \( I^2 + Q^2 = 0.875^2 \) | 10                | 10                | 7                 | 7                 |
| 3/15       | \( I = Q \)       | \( I^2 + Q^2 = 0.875^2 \) | 8                 | 8                 | 5                 | 7                 |

i.i.d. Rayleigh fading channel. A block length of 64,800 bits is adopted for the LDPC code, and the 2D-256NUCs for different code rates are adopted for modulation. In the receiver, perfect channel estimation is assumed, and the sum–product algorithm [23] with a maximum of 50 decoding iterations is employed by the LDPC decoder. Finally, the SNR values in the subsequent results are given by \( E_s/N_0 \) where \( E_s \) is the average energy of a 2D-NUC signal, and \( N_0/2 \) is the power spectral density of AWGN. Figures 5 and 6 plot the BER performance for code rate 4/15 in AWGN channel and i.i.d. Rayleigh fading channel, respectively; Figures 7 and 8 for code rate 7/15; Figures 9 and 10 for code rate 12/15. The performances of the Log-MAP algorithm (which computes the optimal ML LLR) and the Max-Log-MAP algorithm (which computes the suboptimal Max-log LLR) are also shown in the figures as benchmarks. We first compare the performances of the Log-MAP and the Max-Log-MAP algorithms. It is observed that the Max-Log-MAP algorithm has worse performance compared with the Log-MAP algorithm. Moreover, the higher the code rate is, the smaller the performance loss becomes. Since a high code rate is meant to be used at a high target SNR, this observation is in agreement with the fact that the Max-log LLR is a good approximation of the ML LLR for high SNR.

We then compare the performance of the proposed method with the Max-Log-MAP algorithm. The performance losses (in dB measured at BER \( 10^{-6} \)) of the proposed method are listed in Table 2 for all code rates, where \( \Delta_{A,ML} \) and \( \Delta_{A,Max} \) are used to denote the performance losses relative to the Log-MAP and the Max-Log-MAP algorithms, respectively, for AWGN channel; and \( \Delta_{R,ML} \) and \( \Delta_{R,Max} \) for i.i.d. Rayleigh fading channel. Note that the LLR value calculated by the Max-Log-MAP algorithm is the same as that calculated by the proposed method.
TABLE 2  Performance loss (in dB) and demapping complexity

| code rate | 2/15 | 3/15 | 4/15 | 5/15 | 6/15 | 7/15 |
|-----------|------|------|------|------|------|------|
| method    | Ours | [10] | Ours | [10] | Ours | [10] | Ours | [10] | Ours | [10] | Ours | [10] |
| Δₐ,ML     | 0.2  | N/A  | 0.15 | N/A  | 0.18 | N/A  | 0.2  | N/A  | 0.19 | N/A  | 0.17 | N/A  |
| Δₐ,ML     | 0.17 | 0.1  | 0.18 | 0.1  | 0.24 | 0.1  | 0.24 | 0.1  | 0.21 | 0.1  | 0.2  | 0.1  |
| Δₐ,Max    | < 0.005 | N/A | < 0.005 | N/A | < 0.005 | N/A | 0.02 | N/A | 0.01 | N/A | 0.02 | N/A |
| Δₐ,Max    | < 0.005 | N/A | < 0.005 | N/A | < 0.005 | N/A | 0.04 | N/A | 0.01 | N/A | 0.03 | N/A |
| N₉         | 8    | 16   | 10   | 16   | 29   | 39   | 27   | 53   | 38   | 61   | 43   | 68   |
| N₄         | 8 × 8 | 8 × 16 | 8 × 10 | 8 × 16 | 8 × 29 | 8 × 39 | 8 × 27 | 8 × 53 | 8 × 38 | 8 × 61 | 8 × 43 | 8 × 68 |
| N₁         | 16   | 2 × 16 | 18   | 2 × 16 | 37   | 2 × 39 | 35   | 2 × 39 | 46   | 2 × 61 | 51   | 2 × 68 |
| N₉         | 0    | 24   | 0    | 24   | 0    | 47   | 0    | 61   | 0    | 69   | 0    | 76   |

[10] stands for Fuentes’ method.
N/A means that there is no data available for comparison.

FIGURE 7  BER for AWGN channel, code rate 7/15

FIGURE 8  BER for i.i.d. Rayleigh fading channel, code rate 7/15

if the CSR method is not employed. This is demonstrated in Figures 9 and 10 for 2D-256NUC of code rate 12/15, in which all constellation points are well separated, and thus, the CSR method does not apply. Hence, the proposed method achieves the same performance as the Max-Log-MAP algorithm. The same also holds true for code rate 13/15 as shown in Table 2. In contrast, for the 2D-256NUC with code rate less than or equal to 11/15, the CSR method becomes applicable to some...
clusters of constellation points, which leads to performance loss (as shown in Figures 5–8 and Table 2). Upon analysis, it is found that the performance loss is affected by two factors: one is the number of constellation points appearing in clusters; the other is noise variance $\sigma^2_h$ at target SNR. On the one hand, as code rate decreases, the number of constellation points appearing in clusters increases, (which can be observed from Figure 1), and thus, more constellation points are substituted by the representing points; consequently, the performance deteriorates. On the other hand, as code rate decreases, the target SNR decreases, and thus, noise variance increases, which compensates for the performance loss caused by CSR method as shown by (5). This observation is supported by Figures 5–8 and Table 2, whereas code rate decreases, the performance loss first tends to increase from code rates 11/15 to 5/15 and then decrease when code rate is less than 5/15. It is noted that although the proposed method suffers performance loss compared to the Max-Log-MAP algorithm for code rates less than or equal to 11/15, the loss is negligible (less than 0.02 dB for AWGN channel and 0.04 dB for i.i.d. Rayleigh fading channel).

Next, we compare the performance of the proposed method with Fuentes’ method [10]. The performance losses (in dB measured at BER $10^{-4}$) of Fuentes’ method are listed in Table 2 for all code rates. From Table 2, it is observed that the proposed method achieves a similar performance as Fuentes’ method for high code rates (from 10/15 to 13/15). In contrast, for medium and low code rates, the proposed method is outperformed by Fuentes’ method. However, the performance loss (less than 0.14 dB for i.i.d. Rayleigh fading channel) is chiefly inherited from the Max-Log-MAP algorithm.

Finally, we compare the performance of the proposed method with both the Log-MAP algorithm (which computes the optimal ML LLR). It is observed that the performance loss is more significant for i.i.d. Rayleigh fading channel compared with AWGN channel. However, even for i.i.d. Rayleigh fading channel, the proposed method suffers a performance loss less than 0.12 dB for high code rates (from 10/15 to 13/15) and less than 0.24 dB for other code rates.

### 4 DEMAPPING COMPLEXITY

In this section, we compare the computational complexity of the proposed method with those of the Log-MAP algorithm, the Max-Log-MAP algorithm, and Fuentes’ method. The complexities are evaluated in terms of number of distance calculations $N_D$, number of additions $N_A$, number of multiplications $N_M$, and number of exponential and logarithmic operations $N_E$. Note that as the complexity of one comparison operation is similar to that of one addition, each comparison is counted as an addition and is included in $N_A$. From (1), it is obvious that in order to compute the ML LLR for all eight bits, the Log-MAP algorithm requires 256 distance calculations, $8 \times 256$ additions, $2 \times 256$ multiplications (including 256 multiplications for the squares of distances and 8 divisions), and 264 exponential and logarithmic operations. Fuentes’ method is similar to the Log-MAP algorithm except replacing $A_{b, l} = c$ in (1) by its subset. Thus, the sizes of the subsets, which depend on the signal constellation, determine the computational complexity. The numbers of operations, $N_D$, $N_A$, $N_M$, and $N_E$, of Fuentes’ method are listed in Table 2. It is observed that Fuentes’ method indeed significantly decreases the complexity of the Log-MAP algorithm. However, both the Log-MAP algorithm and Fuentes’ method employ a large number of exponential and logarithmic operations that are more expensive than distance calculations, additions, and multiplications. In contrast, both the Max-Log-MAP algorithm and the proposed method require no exponential and logarithmic operations. From (2), it follows that the Max-Log-MAP algorithm requires 256 distance calculation, $8 \times 256$ additions, and 264 multiplications (including 256 multiplications for the squares of distances and 8 divisions). The proposed method replaces $A_{b, l} = c$ in (2) by $C_{b, l} = c$. 

![Figure 9 BER for AWGN channel, code rate 12/15](image)

![Figure 10 BER for i.i.d. Rayleigh fading channel, code rate 12/15](image)
and thus reduces the computational complexity. The numbers of operations $N_{D_2}, N_{A_3}$, and $N_{M}$ of the proposed method are listed in Table 2. Note that the number of distance calculations equals $|C(R_{00})|$, $|C(R_{11})|$, or $|C(R_{11})|$, depending on the location of the received signal. The value of $N_{D_2}$ in the table is the maximum among the four. The same rule is applicable to $N_{A_3}$ and $N_{M}$. Comparing the proposed method with the Max-Log-MAP algorithm, it is obtained that the proposed method drastically reduces the number of operations required at a nominal cost of computing $l_{1}(I, Q)$ and $l_{2}(I, Q)$. Specifically, for both $N_{D_2}$ and $N_{M}$, it exhibits a reduction ranging from 77% at the highest code rate to 96% at the lowest code rate; for $N_{A_3}$, it exhibits a reduction ranging from 74% at the highest code rate to 93% at the lowest code rate. Finally, we compare the proposed method with Fuentes’ method. The proposed method not only is free of exponential and logarithmic operations, but also shows a reduction of both $N_{D_2}$ and $N_{A_3}$ ranging from 25% to 50% and a reduction of $N_{M}$ ranging from 43% to 67%.

5 | CONCLUSION

We propose a low-complexity demapping algorithm for the 2D-NUCs in ATSC 3.0. The proposed method partitions the first quadrant into four subregions and for each subregion, finds the set of constellation points which are the nearest points to some received signals in the subregion. Therefore, the demapper can find the nearest constellation point of a received signal by searching through the corresponding set instead of the whole constellation set. We further reduce the size of the sets by using CSR strategy. Simulation results show that compared with the Max-Log-MAP algorithm, the proposed method reduces the computational complexity by at least 74% with almost no performance loss (less than 0.02 dB for AWGN channel and 0.04 dB for i.i.d. Rayleigh fading channel). Moreover, compared with Fuentes’ method, the proposed method not only is free of exponential and logarithmic operations but also significantly reduces the numbers of distance calculations, additions, and multiplications at the cost of a small performance loss (less than 0.14 dB for i.i.d. Rayleigh fading channel). Moreover, the performance loss occurs mainly in low and medium code rates. For high code rates, the proposed method achieves great computational saving without compromising performance when compared with Fuentes’ method.

ACKNOWLEDGEMENT

This work was supported by the Ministry of Science and Technology, Taiwan, under Project 109-2637-E-027-006.

ORCID

Yu-Sun Liu https://orcid.org/0000-0003-3182-2202

REFERENCES

1. Advanced Television Systems Committee. ATSC standard: Physical layer protocol. Doc. A/322:2017 (2017)
2. Muhammad, N.S.: Coding and modulation for spectral efficient transmission. PhD thesis, Inst. Nachrichtenübertragung, University of Stuttgart, Germany; 2010
3. Zoellner, J., Loghin, N.: Optimization of high-order non-uniform QAM constellations. In: Proceedings of the 2013 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, pp. 1–6, London (2013)
4. Loghin, N.S., et al. Non-uniform constellations for ATSC 3.0. IEEE Trans. Broadcast. 62, 197–203 (2016)
5. Michael, L., Gómez-Barquero, D.: Bit-interleaved coded modulation (BICM) for ATSC 3.0. IEEE Trans. Broadcast. 62, 181–188 (2016)
6. Akay, E., Ayanoglu, E.: Low complexity decoding of bit-interleaved coded modulation for M-ary QAM. In: Proceedings of the 2004 IEEE International Conference on Communications, vol. 2, pp. 901–905, Paris, France (2004)
7. Chang, C.W., Chen, P.N., Han, Y.S.: A systematic bit-wise decomposition of M-ary symbol metric. IEEE Trans. Wireless Commun. 5, 2742–2751 (2006)
8. Kim, K., et al.: One-dimensional soft-demapping algorithms for rotated QAM and software implementation on DSP. IEEE Trans. Signal Process. 61, 3918–3930 (2013)
9. Zhang, M., Kim, S.: Efficient soft demodulation scheme for digital video broadcasting via satellite - second generation system. IET Commun. 8, 124–132 (2014)
10. Fuentes, M., Vargas, D., Gómez-Barquero, D.: Low-complexity demapping algorithm for two-dimensional non-uniform constellations. IEEE Trans. Broadcast. 62, 375–383 (2016)
11. Li, M., et al.: Design of rotated QAM mapper/demapper for the DVB-T2 standard. In: Proceedings of the 2009 IEEE Workshop on Signal Processing Systems, pp. 18–23, Tampere, Finland (2009)
12. Pérez-Calderón, D., et al.: Rotated constellation demapper for DVB-T2. Electron. Lett. 47, 31–32 (2011)
13. Pérez-Calderón, D., et al.: Simplified rotated constellation demapper for second generation terrestrial digital video broadcasting. IEEE Trans. Broadcast. 59, 160–167 (2013)
14. European Telecommunications Standard Institute. Digital video broadcasting (DVB); frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2). ETSI EN 302 755 V1.1.1, 2009.
15. Zhang, M., Kim, S.: Universal soft demodulation schemes for M-ary phase shift keying and quadrature amplitude modulation. IET Commun. 10, 316–326 (2016)
16. Zhang, M., Kim, S.: Efficient soft demodulation schemes for the DVB-S2x system. Int. J. Satell. Commun. Netw. 34, 833–858 (2016)
17. Stott, J.: BICM limits for condensed QAM, further results, document DVB TM-MIMO0023, DVB, Geneva, Switzerland (2013)
18. Kwon, S., et al.: Simplified non-uniform constellation demapping scheme for the next broadcasting system. In: Proceedings of the 2015 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, pp. 1–3, Ghent, Belgium (2015)
19. Barrueco, J., et al.: Low complexity adaptive demapper for 2-D non-uniform constellations. IEEE Trans. Broadcast. 65, 10–19 (2019)
20. Wang, Q., et al.: A universal low-complexity symbol-to-bit soft demapper. IEEE Trans. Veh. Technol. 63, 119–130 (2014)
21. Liu, X., Kosakowski, M.: Max-log-MAP soft demapper with logarithmic complexity for M-PAM signals. IEEE Signal Process. Lett. 22, 50–53 (2015)
22. Kim, K.J., et al.: Low-density parity-check codes for ATSC 3.0. IEEE Trans. Broadcast. 62, 189–196 (2016)
23. Papathanalabos, S., et al.: Modified sum-product algorithms for decoding low-density parity-check codes. IET Commun. 1, 294–300 (2007)

How to cite this article: Liu Y-S, Chien C-H, You SD. Low-complexity demapping algorithm for two-dimensional non-uniform constellations in ATSC 3.0. IET Commun. 2021;15:815–823. https://doi.org/10.1049/cmu2.12123