HDM: Hyper-Dimensional Modulation for Robust Low-Power Communications

Hun-Seok Kim, *Member, IEEE*
EECS Department, University of Michigan,
Ann Arbor, Michigan 48103, U.S.A. Email: hunseok@umich.edu

***Abstract*—This paper introduces hyper-dimensional modulation (HDM), a new class of practical modulation scheme for robust communication among low-power low-complexity devices. Unlike conventional orthogonal modulations, HDM conveys numerous information bits per symbol by combining hyper-dimensional vectors that are not strictly orthogonal to each other. Information bits are spread across many elements in the hyper-dimensional vector, thus HDM is tolerant of element-wise failures in high noise channels. Evaluation results confirm that uncoded HDM with 256-dimension exhibits the bit error rate (BER) comparable to that of low-density parity check (LDPC) and Polar codes, while HDM demodulation complexity is lower than that of LDPC and Polar decoders for the same block length of 256. Moreover, HDM provides graceful tradeoffs between data-rate and signal-to-noise ratio for robust short message communications among power- and complexity-constrained devices.**

I. Introduction

Hyper-dimensional modulation (HDM) is a new class of practical modulation designed for robust communication among complexity-/power-constrained devices. The dimension of modulation determines its information representational space. Conventional modulation schemes utilize a relatively low dimensional space per information symbol. On the contrary, HDM conveys numerous information bits per symbol using a hyper-dimensional space represented by combinations of complex valued components in a hyper-dimensional vector.

HDM is inspired by hyper-dimensional computing [1] where hyper-dimensional vectors are used to represent information and perform cognitive computing. The hyper-dimensional presentation is tolerant of component failure [1]. This robustness comes from redundant representation, in which information symbols are spread across many components in the hyper-dimensional vector. HDM can be considered as a spreading modulation scheme whose spreading gain linearly improves with the dimension, allowing higher error tolerance with increased dimensionality. In low SNR scenarios where each component of the hyper-dimensional vector cannot be reliably demodulated, HDM still achieves successful demodulation of symbols that are spread and superimposed in the hyper-dimensional vector *without explicit error correction encoding*.

Uncoded HDM with a modest block size (or dimension of the vector) achieves the bit error rate (BER) comparable to that of low-density parity check (LDPC) [2], [3], [4] and Polar codes [5], [6], [7] applied to conventional modulation schemes. Demodulation complexity of HDM for a modest block size (e.g., 256) can be kept lower than that of LDPC and/or Polar decoders. HDM can support various coding rates by adjusting modulation parameters without modifying the modulator and demodulator structure. Thus HDM allows low-power low-complexity devices to gracefully adapt to various signal-to-noise ratio (SNR) conditions for dynamic rate $-$ reliability tradeoffs and link adaptations. These unique properties make HDM an attractive solution for low-power low-complexity communication systems that need to exchange relatively short messages reliably at low SNRs.

Consider a hyper-dimensional vector space $\mathbb{C}^D$ where D is the dimension of the hyper-dimensional vector. Similarity between two power-normalized symbols $\mathbf{x}$ and $\mathbf{y}$ can be measured by cross-correlation $\mathbf{x}^*\mathbf{y}$. Here, $\mathbf{x}^*$ stands for transpose conjugate of the vector $\mathbf{x}$. Two identical symbols result in a cross-correlation output that is equal to the symbol power $\mathbf{x}^*\mathbf{x} = ||\mathbf{x}||^2$. The most critical observation that motivates HDM is the fact that two hyper-dimensional vectors whose components are $i.i.d.$ zero-mean random variables have *nearly-orthogonal* cross-correlation; $\mathbf{x}^*\mathbf{y} \approx 0$ for a large D (hyper-dimension). In other words, a hyper-dimensional vector $\mathbf{x}$ in $\mathbb{C}^D$ is nearly-orthogonal to *almost all other* vectors in the hyper-dimensional space $\mathbb{C}^D$ [1]. Randomly selected two vectors in the hyper-dimensional space have very small cross-correlation with high probability. HDM exploits this fact to superimpose multiple nearly-orthogonal vectors to convey numerous information bits very reliably at a low SNR using a single D-dimensional vector. Vector selection in HDM does not have to be carefully done to satisfy near-orthogonality among superimposed vectors when the dimension D is large. Using this property, this paper proposes a frequency-domain HDM scheme to modulate hyper-dimensional vectors with a goal to minimize the demodulation complexity.

II. Hyper-Dimensional Modulation

A. Modulation

The transmitted vector $\mathbf{x}$ is generated by adding V independent vectors $\mathbf{x}_v$, $v = 1, 2, ..., V$ as in (1) where $\mathcal{F}$ stands for discrete Fourier transform. The dimension of $\mathbf{x}_v$ is $D \times 1$ and $\mathbf{P}_v$ is a $D \times D$ permutation matrix randomly selected for each v. It is worth noting that HDM does not strictly require Fourier transform. Other invertible linear transforms can be used but its complexity is desired to be $O(DlogD)$ to lower modulation/demodulation complexity. Each superimposed transmit vector $\mathbf{x}_v$ is obtained by permuting discrete Fourier transform of the sum of $D \times 1$ dimensional K vectors; $\sum_{k=1}^{K} s_{k,v}\mathbf{e}_{p_{k,v}}$

978-1-5386-3180-5/18/$31.00 ©2018 IEEE

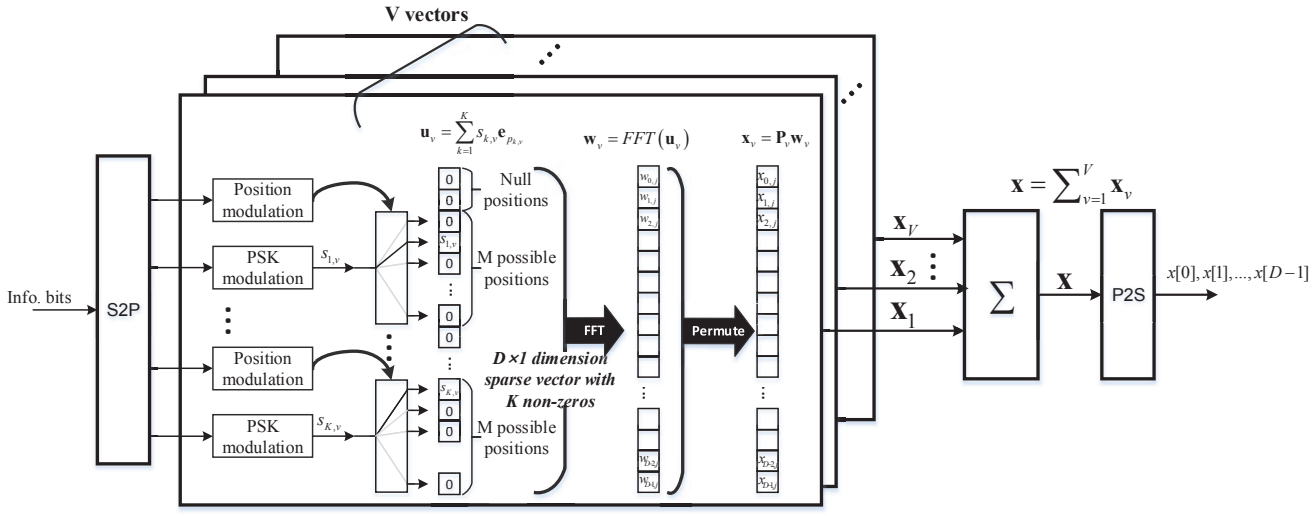

Fig. 1. HDM modulator block diagram. Information sub-symbols are allocated to a set of non-orthogonal vectors. Each sub-symbol is modulated by PSK and its position in the vector. Sub-symbols are merged to a hyper-dimensional vector that is transformed and permuted before it is combined with other vectors. Combined hyper-dimensional vectors are serialized and transmitted sequentially element-wise.

where $s_{k,v}$ is a scalar phase-shift keying (PSK) complex valued symbol from a set $\mathbb{S}$ and $\mathbf{e}_{p_{k,v}} = [e_0, ..., e_{D-1}]^T$ is a $D \times 1$ unit vector with $e_p = 0$ for $\forall p \neq p_{k,v}$ and $e_{p_{k,v}} = 1$. The non-zero position $p_{k,v}$ is selected from a set $\mathbb{P}_k$ whose cardinality is M. Sets $\mathbb{P}_k$, $k = 1, 2, ...K$, do not overlap with each other; $\mathbb{P}_k \cap \mathbb{P}_{k'} = \emptyset$ for $k \neq k'$, and $\mathbb{P}_1 \cup \mathbb{P}_2 ... \cup \mathbb{P}_K \subset \{1, 2, ..., D-1\}$.

$$\mathbf{x} = \sum_{v=1}^{V} \mathbf{x}_v = \sum_{v=1}^{V} \mathbf{P}_v \mathcal{F} \left(\sum_{k=1}^{K} s_{k,v} \mathbf{e}_{p_{k,v}} \right) \tag{1}$$

Note that the DC (i.e., zero frequency) position should be excluded from $\mathbb{P}_k$ otherwise Fourier transform output of $\mathbf{x}_v$ exhibits a DC offset. The cadinality of $\mathbb{S}$ and $\mathbb{P}_k$ is denoted by Q and M, respectively. Each vector $\mathbf{x}_v$ conveys $K\left(log_2 Q + log_2 M\right)$ information bits by the combination of Q-ary PSK and M-ary *symbol position modulation*. Consequently, the modulation rate (or coding rate) C_R of HDM is obtained by (2) as the transmit vector $\mathbf{x}$ contains $VK\left(log_2 Q + log_2 M\right)$ bits with D channel instances (i.e., dimension of the vector).

$$C_R = \frac{VK\left(log_2 Q + log_2 M\right)}{D} \tag{2}$$

The proposed HDM process is summarized in Figure 1. Position based modulation $\mathbf{e}_{p_{k,v}}$ for HDM creates vectors that are sparse before the transform. However, $\mathbf{x}_k$ becomes a *dense* vector after Fourier transform and permutation. This process has some similarity to compressive sensing and its signal recovery techniques [8][9] where random basis functions are used. In HDM, sparse information vectors are transformed and spread to a dense D dimensional vector for robust communication. Combination of FFT and sparse position based modulation significantly lowers the demodulation complexity per information bit as discussed later in Section III.

B. Signal to Interference Ratio (SIR)

Without loss of generality, it is assumed that the signal power is normalized, satisfying $E\{\mathbf{x}^*\mathbf{x}\} = D$ and $E\left\{\mathbf{x}\mathbf{x}^*\right\} = \mathbf{I}_{\mathbf{D}\times\mathbf{D}}$. With i.i.d. random information symbols $s_{k,v}$ and $\mathbf{e}_{p_{k,v}}$, each transmit vector is assumed to be a complex Gaussian random vector $\mathbf{x} \sim \mathcal{CN}\left(\mathbf{0}, \mathbf{I}_{D\times D}\right)$ and $\mathbf{x}_v \sim \mathcal{CN}\left(\mathbf{0}, \frac{1}{V}\mathbf{I}_{D\times D}\right)$. This assumption holds by the central limit theorem [10] as the dimension D and the number of superimposed vectors V increase. For $v \neq v'$, $\mathbf{x}_v$ and $\mathbf{x}_{v'}$ are independent. As each element of $\mathbf{x}$ has a unit power, the PSK symbol power is set to $|s_{k,v}|^2 = \frac{D}{VK}$, satisfying $\|\mathbf{x}_v\|^2 = D/V$. Cross-correlation of $\mathbf{x}_v$ and the interfering signal $\sum_{j\neq v} \mathbf{x}_j$ can be approximated by a zero mean complex Gaussian random variable $\mathbf{x}_v^* \sum_{j\neq v} \mathbf{x}_j \sim \mathcal{CN}\left(0, \frac{D(V-1)}{V^2}\right)$, while $\mathbf{x}_v^* \mathbf{x}_v = D/V$ holds.

Therefore, the post-correlation signal-to-interference ratio at the transmitter is obtained by (3). Superimposed vector $\mathbf{x}_v$'s are not strictly orthogonal to each other; $\mathbf{x}_v^* \mathbf{x}_{v'} \neq 0$ for $v \neq v'$. However, (3) indicates the post-correlation SIR improves with a larger dimension D. In HDM, increasing V (the number of superimposed vector $\mathbf{x}_v$'s) proportionally degrades SIR. Position based modulation by $\mathbf{e}_{p_{k,v}}$ conveys multiple information bits using a single vector $\mathbf{x}_v$ (especially with a large D and M) so that the post-correlation SIR is restrained with a relatively small V satisfying the target coding rate (2). Position based encoding also allows low complexity decoding at the receiver using the parallelizable maximum correlation search within symbol position sets $\mathbb{P}_k$, $k = 1, ..., K$ that do not overlap.

$$SIR_{vec} = \frac{|\mathbf{x}_v^* \mathbf{x}_v|^2}{E\left\{|\mathbf{x}_v^* \sum_{j\neq v} \mathbf{x}_j|^2\right\}} = \frac{D}{V-1} \quad (3)$$

Note that $\mathbf{e}_{p_{k,v}}$ is orthogonal to other position vectors $\mathbf{e}_{p_{k',v}}$ ($\mathbb{P}_k \cap \mathbb{P}_{k'} = \emptyset$, $k \neq k'$) given the same vector index v. Hence the post-correlation SIR for a symbol $s_{k,v}\mathbf{e}_{p_{k,v}}$ can be expressed by

$$SIR_{sym} = \frac{|s_{k,v}^* \mathbf{e}_{p_{k,v}}^* \mathbf{e}_{p_{k,v}} s_{k,v}|^2}{E\left\{|s_{k,v}^* \mathbf{e}_{p_{k,v}}^* \sum_{j\neq v} \mathbf{x}_j|^2\right\}} = \frac{D}{K(V-1)}. \quad (4)$$

Given the target modulation rate C_R (2), one should strike a balance between SIR_{sym} (4) and the number of bits per symbol (i.e., $log_2(Q) + log_2(M)$) by selecting the optimal parameters K, V, Q, and M to enhance the reliability of HDM while maximizing the data rate. As discussed in Section IV, adapting to a proper combination of various modulation parameters provides graceful tradeoffs in the bit error rate (BER) performance, demodulation complexity, and data rate. It is shown in Section IV that uncoded HDM without explicit error correction encoding achieves the BER comparable to that of LDPC and Polar codes while HDM demodulation complexity is lower for relatively short length messages.

III. HDM Demodulation

An additive white Gaussian noise (AWGN) channel model is used for HDM performance analysis and evaluation. The received vector is represented by (5), where $\mathbf{n}$ is a complex Gaussian noise vector with a covariance of $E\{\mathbf{n}\mathbf{n}^*\} = N_0\mathbf{I}_{D\times D}$. The SNR is $1/N_0$ as the signal $\mathbf{x}$ has the unit covariance $E\{\mathbf{x}\mathbf{x}^*\} = \mathbf{I}_{\mathbf{D}\times\mathbf{D}}$.

$$\mathbf{y} = \mathbf{x} + \mathbf{n} \quad (5)$$

A. Demodulation Algorithm and Architecture

Figure 2 depicts the proposed HDM demodulation process using successive interference cancellation. Estimation of each superimposed transmission vector $\hat{\mathbf{x}}_v$ can be performed in parallel to enhance overall demodulation throughput. For the path v, the residual received vector $\hat{\mathbf{y}}_v^{(i)}$ for iteration i is obtained by subtracting estimated interference vectors from the original received signal $\mathbf{y}$ as in (6), where $\hat{s}_{k,w}^{(i)}$ and $\hat{\mathbf{e}}_{p_{k,w}^{(i)}}$ are the estimated PSK symbol and position vector for the i_{th} iteration. For the initial iteration, $\hat{s}_{k,w}^{(0)} = 0$ and $\hat{\mathbf{e}}_{p_{k,w}^{(0)}} = \mathbf{0}$.

$$\hat{\mathbf{y}}_v^{(i)} = \mathbf{y} - \sum_{w\neq v} \mathbf{P}_w \mathcal{F}\left(\sum_{k=1}^{K} \hat{s}_{k,w}^{(i)} \hat{\mathbf{e}}_{p_{k,w}^{(i)}}\right) \quad (6)$$

To obtain $\hat{\mathbf{x}}_v^{(i)}$, the residual vector $\hat{\mathbf{y}}_v^{(i)}$ is permuted (multiplied) by $\mathbf{P}_v^{-1}$, and then inverse Fourier transform is performed on the permuted vector as in (7).

$$\hat{\mathbf{x}}_v^{(i)} = \mathcal{F}^{-1}\left(\mathbf{P}_v^{-1}\hat{\mathbf{y}}_v^{(i)}\right) = \sum_{k=1}^{K} s_{k,v}\mathbf{e}_{p_{k,v}} + \sum_{w\neq v} \mathbf{d}_{v,w}^{(i)} \quad (7)$$

The residual distortion vector $\sum_{w\neq v} \mathbf{d}_{v,w}^{(i)}$ satisfies (8) where $\mathbf{r}_{k,w}$ represents the residual error vector $\mathbf{r}_{k,w} = \left\{s_{k,w}\mathbf{e}_{p_{k,w}} - \hat{s}_{k,w}^{(i)}\hat{\mathbf{e}}_{p_{k,w}}^{(i)}\right\}$.

$$\mathbf{d}_{v,w}^{(i)} = \mathcal{F}^{-1}\left(\mathbf{P}_v^{-1}\left(\mathbf{P}_w\mathcal{F}\left(\sum_{k=1}^{K}\mathbf{r}_{k,w}\right)\right) + \mathbf{n}\right) \quad (8)$$

For the next iteration $i+1$, the demodulated symbol $\hat{s}_{k,v}^{(i+1)}$ and $\hat{\mathbf{e}}_{p_{k,v}^{(i+1)}}$ is obtained from $\hat{\mathbf{x}}_v^{(i)}$ by (9). Notice that (9) can be performed in parallel for each symbol and vector index k and v. Specifically, PSK symbol $s_{k,v}^{(i)}$ and position $p_{k,v}^{(i)}$ demodulation is performed by only inspecting vector elements whose indices belong to the set $\mathbb{P}_k$ as $\mathbb{P}_k \cap \mathbb{P}_{k'} = \emptyset$ for $k \neq k'$. The element with the maximum cross-correlation as in (9) provides the estimated (demodulated) PSK symbol and position vector for the next iteration $i+1$.

$$\hat{s}_{k,v}^{(i+1)}\hat{\mathbf{e}}_{p_{k,v}}^{(i+1)} = \underset{\forall s\in\mathbb{C},\ p\in\mathbb{P}_k}{\operatorname{argmax}} \ \Re\left\{s^*\mathbf{e}_p^*\hat{\mathbf{x}}_v^{(i)}\right\} \quad (9)$$

Iteration stops when $\hat{s}_{k,v}^{(i+1)} = \hat{s}_{k,v}^{(i)}$ and $\hat{\mathbf{e}}_{p_{k,v}}^{(i+1)} = \hat{\mathbf{e}}_{p_{k,v}}^{(i)}$ for all k and v. To guarantee a constant demodulation throughput, it is also possible to terminate the iteration after a predetermined number of iterations. As discussion in Section IV, the average number of iterations at a reasonably low BER is < 3 including the initial iteration with $i = 0$.

The final step of the HDM demodulation is the cyclic redundancy check (CRC) based error correction. HDM demodulation errors at a relative low BER typically involve only a small number of PSK position symbol ($\hat{s}_{k,v}\hat{\mathbf{e}}_{p_{k,v}}$) errors; mostly one or two out of KV symbols within a received vector $\mathbf{y}$. To correct these errors, one can construct a *second-trial list* that consists of PSK position symbols whose correlation values are closest to the solution of (9). Entries in the second-trial list are sorted by the ratio between their correlation values over the max correlation value from (9). When CRC fails, the demodulation process attempts to correct one error by testing a modest number (e.g., tens) of entries in the second-trial list replacing the least confident solution of (9) based on the correlation value ratio. Error correction is confirmed by CRC validation. If all single error correction trails fail, double error correction is tried, and so forth. A reasonable number of entries in the second-trial list is $<$100. At a low BER, this single/double/triple error correction scheme does not incur significant complexity overhead when the maximum number of trials is limited to $<$500 because it is only invoked when CRC error occurs and most errors are resolved during single error correction trials ($\ll$100 cycles). The SNR gain of the proposed CRC based error correction is 1.75dB at a BER of 10^{-5} for D =256 or 512. Triple error correction is the practical limit as quadruple error correction rarely succeeds.

B. Signal-to-Interference-Noise Ratio of HDM

With the central limit theorem for a large D and V, each element of the distortion vector $\sum_{w\neq v} \mathbf{d}_{v,w}^{(i)}$ (8) can be modeled as a zero-mean complex Gaussian vector with a

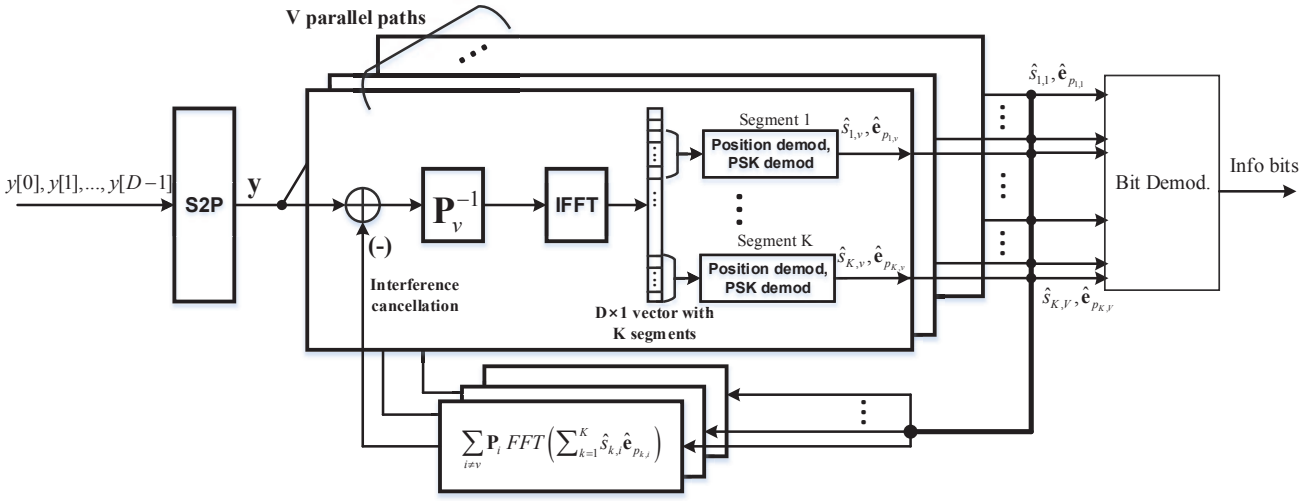

Fig. 2. HDM demodulator block diagram. Each superimposed hyper-dimensional vector can be demodulated in parallel. Estimated interference from other vectors are subtracted first, and then inverse permutation and transform are performed on the received vector. After inverse transform, subsymbols can be demodulated in parallel from non-overlapping segments of the vector. Demodulated subsymbols are fed back for interference cancellation in the next iteration.

covariance of $\left(N_0 + \frac{1}{D}\sum_{w\neq v}\sum_{k=1}^{K}\mathbf{r}_{k,w}^{*}\mathbf{r}_{k,w}\right)\mathbf{I}_{D\times D}$. At the initial iteration ($i=0$), this covariance is $\left(N_0+\frac{V-1}{V}\right)\mathbf{I}_{D\times D}$, implying the initial signal-to-interference-noise ratio (SINR) of $\frac{D}{VK\left(N_0+\frac{V-1}{V}\right)}$ to demodulate each symbol $s_{k,v}\mathbf{e}_{p_{k,v}}$. When this initial SINR is sufficiently high, the residual error vector $\mathbf{r}_{k,w}$ diminishes as iteration continues via successive interference cancellation. The SINR eventually converges to $\frac{D}{VKN_0}$ with successive interference cancellation [11], providing the HDM SNR gain factor of γ_{SNR} (10) compared to the original channel SNR of $\frac{1}{N_0}$.

$$\gamma_{SNR} = \frac{D}{VK} \tag{10}$$

C. HDM Demodulation Complexity

To estimate the complexity of HDM demodulation, we count the number of operations involved in the algorithm proposed in Section III.A. The 'operation' includes multiplication, addition, comparison, and memory read/write. Each iteration of the demodulation process consists of three steps: 1) interference computation and subtraction, 2) inverse permutation and FFT, and 3) PSK and position demodulation.

1) Interference computation and subtraction: In this step, interference is subtracted from the received vector $\mathbf{y}$ as in (6). In fact, this step does not require explicit Fourier transform because $\hat{s}_{k,v}^{(i)}\hat{\mathbf{e}}_{p_{k,v}^{(i)}}$ has only one non-zero element, and $\mathcal{F}\left(\hat{s}_{k,v}^{(i)}\hat{\mathbf{e}}_{p_{k,v}^{(i)}}\right)$ can be obtained by traversing the unit circle in the complex plane from the initial point $\hat{s}_{k,v}^{(i)}$ with a constant phase rotation rate given by $p_{k,v}^{(i)}$. The permutation by $\mathbf{P}_w$ requires $2D$ memory read/write operations. Notice that for each vector index v in (6), a different interference vector needs to be computed. This can be simplified by computing the summation $\hat{\mathbf{y}}^{(i)} = \mathbf{y} - \sum_{v=1}^{V}\mathbf{P}_v\mathcal{F}\left(\sum_{k=1}^{K}\hat{s}_{k,v}^{(i)}\hat{\mathbf{e}}_{p_{k,v}^{(i)}}\right)$ first, and then compute $\hat{\mathbf{y}}_v^{(i)} = \hat{\mathbf{y}}^{(i)} + \mathbf{P}_v\mathcal{F}\left(\sum_{k=1}^{K}\hat{s}_{k,v}^{(i)}\hat{\mathbf{e}}_{p_{k,v}^{(i)}}\right)$ for each index v. Therefore, the total number of operations to process all V vectors in this step is $(3KV-2K-2)D$ per iteration for the iteration index $i>0$. This step is omitted for the initial iteration with $i=0$.

2) Inverse permutation and Fourier transform: This step computes $\hat{\mathbf{x}}_v^{(i)}$ (7). Inverse permutation and Fourier transform requires $2D$ memory read/write and $4Dlog_2D-6D+8$ (real) arithmetic operations per vector.

3) PSK and position demodulation: Given index v, demodulating symbols $\hat{s}_{k,v}^{(i)}\hat{\mathbf{e}}_{p_{k,v}^{(i)}}$ for different k's can be performed in parallel using K non-overlapping segments of the vector (7). For each segment, (9) can be simplified to finding the maximum power symbol first and then identifying the closest valid PSK symbol. Therefore, the total number of operations to compute (9) for all K segments and all V vectors is $\leq 5D+6$ per iteration for $Q\in\{1,2,4,8\}$.

Summarizing all three steps, the complexity of HDM demodulation for I_{iter} iterations (including the initial iteration with $i=0$) can be approximated by (11) in the number of operations per information bit.

$$O_{bit} = \frac{I_{iter}(3KV-2K+4log_2D-3)-3KV+2K+2}{C_R} \tag{11}$$

The complexity overhead of CRC based single/double/triple error correction can be ignored because the average number of error correction trials per information bit is $<10^{-3}$ for a relatively low BER ($<10^{-4}$).

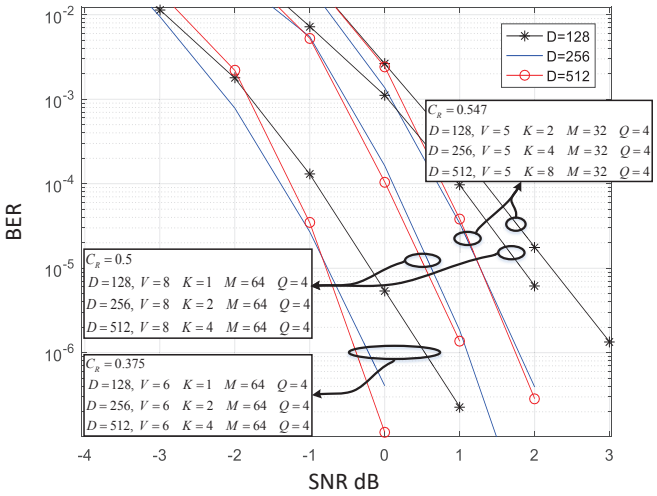

Fig. 3. BER of HDM in AWGN for various C_R settings.

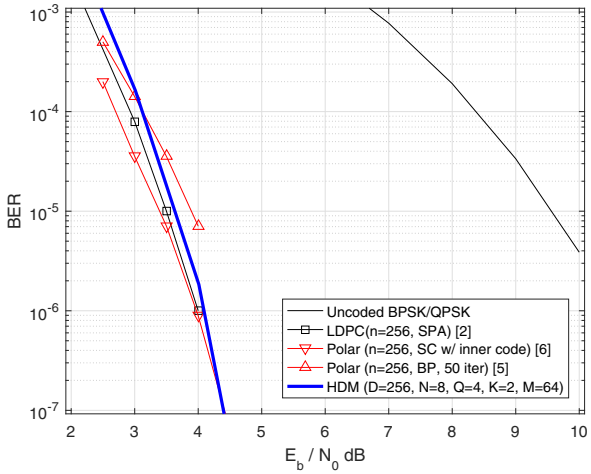

Fig. 4. Bit error rate of HDM vs. LDPC vs. Polar for a short block length of 256 and $C_R = 0.5$. E_b is the energy per information bit. E_b/N_0 = SNR/C_R.

IV. HDM Evaluation

A. Bit Error Rate Performance

To evaluate HDM performance, simulations were conducted in the AWGN channel as explained in Section III. With the unit transmit power $E\{\mathbf{x}\mathbf{x}^*\} = \mathbf{I}_{D\times D}$, SNR is defined by $1/N_0$. Figure 3 shows the BER vs. SNR for various HDM parameters satisfying $C_R = 0.375, 0.5$ and 0.547. The impact of the dimension D is shown in Figure 3. When modulation parameters such as M, K and Q are chosen properly, using a larger D for the same C_R generally improves the BER performance but the gain diminishes when $D \geq 256$. This is predicted by the relationship between the SNR gain factor (10) and the cording rate (2). That is, when $\frac{D}{VK}$ in C_R (2) is kept the same to maintain the equal coding rate, using a larger D does not significantly improve BER as the SNR gain (10) also stays unchanged. Using $D = 128$, however, shows significant degradation.

With a proper selection of HDM parameters that minimize SINR for $C_R = 0.5$, HDM with a modest D (=256) exhibits the BER performance matches to that of LDPC and Polar codes with $1/2$ rate and the 256-bit block length (equivalent to D in HDM). Figure 4 shows the BER comparison between HDM and LDPC / Polar from [2] [6] [5] using the block length of 256. E_b stands for the energy per information bit in Figure 4. Hence E_b/N_0 is obtained by SNR/C_R. It validates that, for short message communications, the performance of HDM, LDPC and Polar are similar for the BER range of $10^{-5}-10^{-7}$ (or block error rate of $10^{-3} - 10^{-5}$).

B. HDM Receiver Complexity

The number of iterations for HDM demodulation is shown in Figure 5. At $\geq$ 1dB SNR, the average number of iterations including the initial iteration ($i = 0$) is less than 3. Using a higher dimension D would require more iterations to converge but it also provides improved BER as shown in Fig. 3.

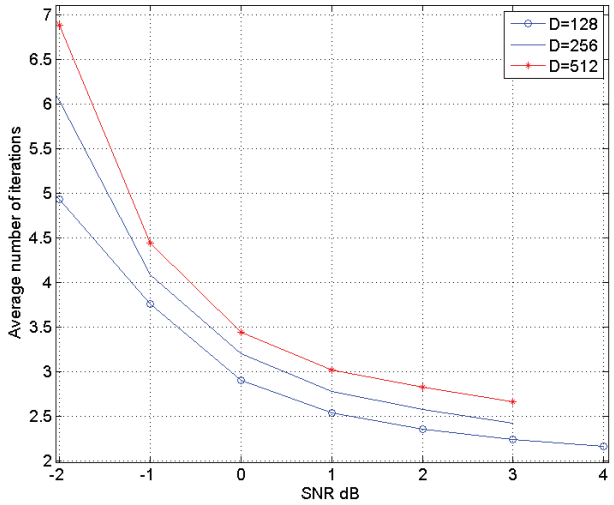

Fig. 5. Average number of iterations I_{iter} (including the initial iteration $i = 0$) vs. SNR for the setting $C_R = 0.5$; $D = 128, 256$, and 512.

The average number of operations for HDM with $D = 256$ is summarized in Table I. Complexity comparison among HDM, LDPC, and Polar code demodulation/decoding is also provided in the same table.

TABLE I
AVERAGE NUMBER OF ITERATIONS AND NUMBER OF OPERATIONS PER INFORMATION BIT FOR $C_R = 0.5$ AND $E_b/N_0 = 4dB$.

	HDM	LDPC [4]	LDPC [2]	Polar [7]
D or blk. length	256	256	256	256
Avg. I_{iter}	2.5	5	2.7	6
Operations per bit	280	518*o	127*+o	336*

* Not including demodulation (soft-decision) complexity.
\+ $tanh$ function is counted as a single operation.
o Message passing complexity ignored.

For a relatively small block length of 256, the HDM operation count is comparable to or lower than that of LDPC or Polar decoding. It must be noted that the number of operations is just a first-order indicator of the decoding complexity. It is well known that LDPC requires a large amount of data movement via irregular interconnects between 'check' and 'variable' nodes which typically becomes the bottleneck for high-performance and low-power hardware implementations [3]. The interconnect data movement complexity overhead is not captured by the number of operations in Table I, thus underestimating the LDPC complexity for practical implementation. It is well known that LDPC decoders have higher complexity than Polar decoders with the same block length. Similar to the proposed HDM demodulator architecture, data movement patterns for Polar decoders [5] [7] typically have 'FFT-like' efficient structures. Thus the operation count in Table I allows more direct complexity comparison between HDM and Polar code.

Although Polar codes have advantages over LDPC for lower complexity implementation, its serialized successive cancelling (SC) architecture prevents realizing parallelized hardware implementation for Polar codes. Modified algorithms such as belief propagation (BP) allow efficient parallel implementation but their efficiency comes at the cost of degraded BER or increased number of iterations. Although Polar decoder implementation complexity of [7] in Table I is comparable to that of HDM, its BP based algorithm with a block length of 256 has a noticeably worse BER compared to a SC based decoder [6] and/or HDM with $D = 256$.

The proposed HDM decoding algorithm is fully parallelizable without performance degradation. The Fourier transform based architecture allows very efficient software and hardware implementations. Unlike LDPC, HDM does not require irregular data movements. Permutation operations in HDM are also easily implementable in both software and hardware.

C. Rate−SNR Tradeoffs

Fig. 6 summarizes the HDM tradeoff space emperically obtained by BER vs. SNR simulations for various sets of modulation parameters. In Fig. 6, the dimension is kept to $D = 256$ or 512. HDM can provide graceful tradeoffs between SNR and coding rate by changing its modulation parameters such as D, V, K, M, and Q.

D. Applications

HDM can be augmented by various conventional modulation methods such as OFDM [12] [13] and MIMO [14]. That is, each element of an HDM vector can be mapped to a subcarrier of OFDM or a spatial stream symbol of MIMO spatial multiplexing. The BER degradation from using a smaller D (e.g., 512 vs. 256) is negligible as shown in Fig. 3, and in analysis combines (2) and (10). HDM is particularly useful for short message communications among low power devices in latency-critical applications where adaptive and robust communications are required.

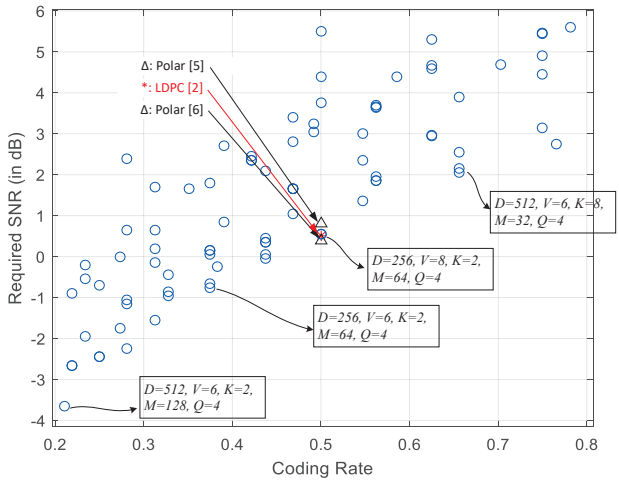

Fig. 6. HDM tradeoff space: SNR for 10^{-5} BER vs. coding rate (C_R). Each point represent a distinct parameter setting. $D = 256$ or 512.

V. Conclusion

This paper introduces HDM for robust communication among low-power and low-complexity devices. HDM with a short message length exhibits BER performances comparable to state-of-the-art LDPC and Polar codes. HDM complexity is also shown to be similar to or less than that of Polar and LDPC. HDM provides graceful tradeoffs between data rate and SNR for robust short message communications.

References

[1] P. Kanerva, "Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors," *Cognitive Computation*, vol. 1, no. 2, pp. 139–159, 2009.

[2] B. Moision, "Decoding Complexity and Performance of Short-Block LDPC Codes Over GF(q)," in *IPN Progress Report 42-194*, 2013, p. 1.

[3] Z. Zhang, V. Anantharam, M. J. Wainwright, and B. Nikolic, "An Efficient 10GBASE-T Ethernet LDPC Decoder Design With Low Error Floors," *IEEE Journal of Solid-State Circuits*, April 2010.

[4] M. P. C. Fossorier, M. Mihaljevic, and H. Imai, "Reduced complexity iterative decoding of low-density parity check codes based on belief propagation," *IEEE Transactions on Communications*, May 1999.

[5] A. Pamuk, "An FPGA implementation architecture for decoding of polar codes," in *2011 8th Int'n Symp. on Wireless Comm. Systems*, Nov 2011.

[6] M. Seidl and J. B. Huber, "Improving successive cancellation decoding of polar codes by usage of inner block codes," in *2010 6th International Symposium on Turbo Codes Iterative Information Processing*, Sept 2010.

[7] S. Sun and Z. Zhang, "Architecture and optimization of high-throughput belief propagation decoding of polar codes," in *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2016, pp. 165–168.

[8] R. G. Baraniuk, "Compressive sensing [lecture notes]," *IEEE signal processing magazine*, vol. 24, no. 4, pp. 118–121, 2007.

[9] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. on Info. Theory*, 2009.

[10] A. Leon-Garcia, *Probability and Random Processes For EE's (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2007.

[11] P. Patel and J. Holtzman, "Analysis of a simple successive interference cancellation scheme in a DS/CDMA system," *IEEE journal on selected areas in communications*, vol. 12, no. 5, pp. 796–807, 1994.

[12] J. Proakis, *Digital Communications*. McGraw-Hill, 2001.

[13] R. v. Nee and R. Prasad, *OFDM for wireless multimedia communications*. Artech House, Inc., 2000.

[14] A. J. Paulraj, D. A. Gore, R. U. Nabar, and H. Bolcskei, "An overview of MIMO communications-a key to gigabit wireless," *Proceedings of the IEEE*, vol. 92, no. 2, pp. 198–218, 2004.