MMSE Scaling Enhances Performance in Practical Lattice Codes

Nuwan S. Ferdinand *, Matthew Nokleby †, Brian M. Kurkoski ‡ and Behnaam Aazhang §
nuferdin@ee.oulu.fi *, matthew.nokleby@duke.edu †, kurkoski@jaist.ac.jp‡, aaz@rice.edu§

Abstract—We investigate the value of MMSE scaling for practical lattice codes. For ideal lattices, MMSE scaling has been shown to be a key ingredient in achieving the capacity of the AWGN channel. We demonstrate that MMSE scaling enhances the performance, particularly at low SNR, for practical lattice codes. For example, a dimension \( n = 10000 \) LDLC lattice exhibits approximately 0.6 dB gain when MMSE scaling is used for a rate of 1 bit/dimension. Furthermore, we provide a novel derivation of the MMSE scaling rule, showing that it emerges naturally from principles of belief propagation decoders which account for the transmit power constraint.

I. INTRODUCTION

In the search for structured codes for continuous channels, researchers have shown that several constructions of lattice codes achieve the capacity of the point-to-point AWGN channel. First, it was shown that an appropriate sequence of lattices, intersected with a “thin” shell corresponding to the transmit power constraint, form a capacity-achieving codebook [1]. Then, it was shown that, under a near maximum-likelihood decoding rule, lattices intersected with a hyperspherical shaping region achieve capacity [2]. In each case, the decoding rule is rather complex and eliminates most of the advantages promised by structured codes.

It was then shown that, under a (relatively) low-complexity lattice decoding rule, in which codewords are mapped to the nearest lattice point without regard for shaping, rates below \( 1/2 \log_2(\text{SNR}) \) are achieved; however, this falls well short of capacity when the SNR is small [3]. Finally, it was shown that lattice decoding can achieve the full capacity when the receiver scales the incoming signal prior to decoding. In particular, the receiver scales the signal by the coefficient corresponding to optimal MMSE estimation.

Since the advent of capacity-achieving lattice codes, researchers have searched for lattice codes suitable for practical implementation [4]–[6]. At present, such codes employ simple lattice decoding without regard for the shaping region, which [3] predicts has poor performance at low SNR. Furthermore, it is unclear whether the MMSE scaling proposed in [7] improves performance in practical codes, or if it is only a proof technique.

In this paper, we show that MMSE scaling is beneficial for practical, finite-dimensional lattices. We focus on low-density lattice codes or LDLCs, for which efficient powershaping algorithms, as well as low-complexity lattice decoders, exist. We show experimentally that the use of MMSE scaling improves performance for lattice codes of moderate block length, particularly for low SNR and low rates. Indeed, scaling offers improvements of up to 0.6dB over standard decoders. Finally, we provide an alternate derivation of the MMSE scaling rule from a belief propagation perspective. We show that if we modify the iterative lattice decoder to account for the transmit power constraint, the result is equivalent to applying MMSE scaling at the receiver. That is, MMSE scaling emerges naturally from the design of iterative decoders over power-constrained channels.

II. PRELIMINARIES

A. System Model

We consider a standard additive white Gaussian noise (AWGN) channel in which the transmitter sends a signal \( x \in \mathbb{R}^n \), constrained to have average transmit power \( \|x\|^2 \leq nP_x \); the receiver obtains the transmitted signal, corrupted by noise:

\[
y = x + z
\]

where \( z \) is AWGN with per-element variance \( P_z \).

Each transmit signal \( x \) is a codeword from a finite codebook \( C \subset \mathbb{R}^n \). Define the rate of the codebook as

\[
R = \frac{\log_2 |C|}{n},
\]

measured in bits per channel use. The receiver uses a decoder \( D : \mathbb{R}^n \rightarrow C \) that maps received signals \( y \) to estimates \( \hat{x} \) of the transmitted codeword. We leave the decoder abstract for now; later on we consider several specific examples. Define the average symbol error probability as

\[
P_e = \frac{1}{|C|} \sum_{x \in C} \Pr(\hat{x} \neq x|x),
\]

where each term in (3) is the probability of a decoding error supposing that the codeword \( x \) is sent.

B. Lattice Codes

We focus on codebooks constructed from lattices. An \( n \)-dimensional lattice \( \Lambda \) is a discrete additive subgroup of \( \mathbb{R}^n \). Any lattice \( \Lambda \) may be expressed in terms of a (non-unique) generator matrix \( G \in \mathbb{R}^{n \times n} \) by the following definition:

\[
\Lambda = \{ \lambda = Gb : b \in \mathbb{Z}^n \}.
\]

We introduce a few basic definitions used throughout the paper; for further details about lattice codes we refer the reader to [8]. The lattice quantizer \( Q_\Lambda \) maps any point in \( \mathbb{R}^n \) to the nearest lattice point in \( \Lambda \) in the Euclidean sense, with ties
broken arbitrarily:

\[ Q_{\Lambda}(x) = \arg \min_{x \in \Lambda} |\lambda - x|^2. \]  

(5)

The fundamental Voronoi region of \( \Lambda \) is the region of \( \mathbb{R}^n \) closer to the origin than to any other lattice point:

\[ V = \{ x : Q_{\Lambda}(x) = 0 \} \]  

(6)

The modulo operation with respect to \( \Lambda \) returns the quantization error:

\[ [x] \mod \Lambda = x - Q_{\Lambda}(x). \]  

(7)

The modulo operation partitions \( \mathbb{R}^n \) into Voronoi regions associated with each \( \lambda \in \Lambda \), or the subsets of \( \mathbb{R}^n \) closest to each \( \lambda \).

Next, \( V = \text{Vol}(V) \) denotes the volume of the fundamental Voronoi region. The second moment of a lattice, defined as

\[ \sigma^2_{\Lambda} = \frac{1}{\text{Vol}(V)} - \frac{1}{n} \int_V |x|^2 \, dx, \]  

(8)

specifies the average power of a random variable uniformly distributed across \( V \). The normalized second moment (NSM) of \( \Lambda \) is given by

\[ G(\Lambda) = \frac{\sigma^2_{\Lambda}}{\overline{V}^2}. \]  

(9)

For any lattice, \( \frac{1}{2\pi e} \leq G(\Lambda) \leq \frac{1}{12}, \) where the minimum value corresponds to a hyperspherical Voronoi region, and the maximum corresponds to a hypercube. Poltyrev [9] showed that there exists a sequence of lattices such that \( G(\Lambda) \rightarrow \frac{1}{2\pi e} \) as \( n \rightarrow \infty \).

Suppose codewords are lattice points in \( \Lambda \) and the receiver uses \( Q_{\Lambda}(y) \) as the decoder. Then, an error occurs precisely when the noise escapes the fundamental Voronoi region, and the symbol error probability is

\[ P_e = \Pr(x \notin V) \]  

(10)

Then, we define the volume to noise ratio (VNR) as

\[ \mu(\Lambda, P_e) = \frac{V^2}{\overline{P}^2}, \]  

(11)

where \( \overline{P} \) is the noise variance such that the probability of error is \( P_e \). Poltyrev [9] showed also that there exists a sequence of lattices such that \( \mu(\Lambda, P_e) \rightarrow 2\pi e \), which is the maximum theoretical value.

### C. Low-Density Lattice Codes

A low-density lattice \( \Lambda \subset \mathbb{R}^n \) is defined by a generator matrix \( G \in \mathbb{R}^{n \times n} \) such that the inverse \( G^{-1} = H \) exists and is sparse [4]. In order to construct a finite-rate codebook, we choose a finite subset of the low-density lattice \( \Lambda \), defined by a subset of \( \mathbb{Z}^n \) to which we apply the generator matrix \( G \). Define the set

\[ \mathcal{L} = \{ b \in \mathbb{Z}^n : 0 \leq b_i \leq L_i - 1 \}, \]  

(12)

where each \( L_i \) is a positive integer. Then we define the codebook

\[ C = \{ Gb : b \in \mathcal{L} \} \subset \Lambda. \]  

(13)

Observing that \( |\mathcal{L}| = \prod_{i=1}^{n} L_i \), the rate of this code is

\[ R = \frac{\sum_{i=1}^{n} \log_2(L_i)}{n}. \]  

(14)

In general, the codewords in \( C \) do not obey the power constraint, so we must employ shaping methods in order to create a suitable codebook. In particular, we employ shaping methods developed in [6] for LDLCs. They require LDLCs with a lower-triangular parity-check matrix \( H \), which enables fast computation and manipulation of lattice points. As in [7], the shaping algorithms are based on nested lattices. In hypercube shaping, the shaping lattice \( \Lambda_1 \) is a scaled integer lattice; in nested lattice shaping \( \Lambda_1 \) is an integer multiple of the coding lattice \( \Lambda \). In each case, the codewords are \( x = [c] \mod \Lambda_1 \) for lattice points \( c = \Lambda \). See [6] for details.

Low-density lattice codes can be decoded with a linear-complexity, iterative algorithm similar to those for low-density parity check codes [4], [5], [10]. Here we briefly sketch the algorithm from [5], which is based on a Gaussian-mixture approximation.

The message-passing algorithm has two types of nodes. Let \( V_k \) be the \( k \)th variable node, which is ensures that the estimated codeword matches the \( k \)th element of the received signal, and let \( C_l \) be the \( l \)th check node, which ensures that the \( l \)th element of \( Hx \) is an integer. Variable and check nodes iteratively exchange “messages”, which are density functions of the codeword from each node’s vantage point. These messages are computed as follows.

- Let \( y_k = x_k + z_k \) be the \( k \)th element of \( y \). With knowledge only of \( y_k \), the density of \( x_k \) is \( \mathcal{N}(y_k, P_z) \). The variable node \( V_k \) sends this density to every connected check node, or every node \( C_l \) such that \( H_{k,m} \neq 0 \).
- Each check node \( C_l \) enforces the integer constraint. Given the incoming densities from each connected variable node and the integer constraint, the density of \( x_k \) is the convolution of the incoming densities, scaled down by \( H_{k,l} \), and periodically extended by the integers. For tractability, the scaled convolution is approximated by a Gaussian density, and only a few integer replicas are computed, resulting in a Gaussian mixture. For each connected \( V_k \), the approximate density is computed and sent.
- Returning to the variable node \( V_k \), the density of \( x_k \) given \( y_k \) and the incoming densities is the product of the incoming densities and the Gaussian \( \mathcal{N}(y_k, P_z) \). The variable node approximates the product by a Gaussian distribution, and sends the result to each connected check node.
- After sufficiently many iterations, the variable nodes \( V_k \) take the product of the incoming densities, and find the peak values \( x_k^* \) of the result. Finally, the decoder
estimates the information vector $\mathbf{b}$ by choosing

$$b^* = [\mathbf{Hx}^*],$$

where $[\cdot]$ denotes element-wise rounding to the nearest integer.

Fig. 1 illustrates the Gaussian mixture LDLC algorithm where it shows the operation of $m^{th}$ check node and $k^{th}$ variable node. We refer the reader to [4], [5], [10] for details.

We emphasize that this decoder does not consider the shaping region in choosing the optimum codeword, which increases the probability of error. As we show in the next section, the use of scaling can greatly improve error performance, particularly at low SNR.

### III. SCALING FOR LDLCs

#### A. Scaling Algorithm

The difficulty with lattice decoding is that it is easy for the receiver to erroneously decode points near the boundary of the shaping region to lattice points that are not in the codebook. As the dimension increases, a rather large fraction of the codewords are near the boundary, and the error performance degrades. Scaling obviates this problem by “inflating” the equivalent lattice so that codewords are further from the boundary. However this advantage comes at the cost of self-noise, as we describe in the next subsection.

Scaling itself is a simple process. The received signal $y$, as given in (1), is multiplied by $\alpha$ at the receiver:

$$y' = \alpha y = \alpha (x + z)$$

where the effective noise $z' = (\alpha - 1)x + \alpha z$. Then it uses the LDLC decoder with input probability distribution $\mathcal{N}(\alpha y, (\alpha - 1)^2 P_x + \alpha^2 P_z)$.

#### B. Analytical Evaluation

The achievable rate for optimal nested lattice codes with large block length is known and equal to AWGN capacity [7]. In this section we analyze the achievable rates of practical lattice codes, both with and without MMSE scaling. Let generator matrices $G$ and $\mathcal{L}G$ correspond to the fine lattice $\Lambda_1$ and coarse lattice $\Lambda_2$. Let $\mathcal{V}_1$ and $\mathcal{V}_2$ be the Voronoi regions of $\Lambda_1$ and $\Lambda_2$ respectively. We first obtain the second moment of effective noise $z'$ as given in (16)

$$\sigma_{z'}^2 = \frac{1}{n} E[(1 - \alpha)x + \alpha z]^2$$

$$= (1 - \alpha)^2 P_x + \alpha^2 P_z$$

(17)

Simple optimization shows that the average power of the effective noise is minimized when $\alpha$ equals the MMSE coefficient

$$\alpha = \alpha_m = \frac{P_x}{P_x + P_z}$$

(18)

For this $\alpha$, the corresponding effective noise power is $\sigma_{z'}^2 = P_x P_z/(P_x + P_z)$. If the receiver uses an unconstrained lattice decoder on $y'$, the error probability is

$$P_e = Pr[z' \notin \mathcal{V}_1]$$

(19)

According the definition of VNR, the volume of the fine lattice should be $V_1 = [\mu(\Lambda_1, P_e)\sigma_{z'}^2]^{\frac{1}{2}}$ for a target probability of $P_e$. Now we find the achievable rate

$$R = \frac{1}{n} \log_2 \left( \frac{V_2}{V_1} \right)$$

(20)

According the the definition of normalized second moment

$$R_s = \frac{1}{2} \log_2 \left[ \frac{P_x/G(\Lambda_2)}{\mu(\Lambda_1, P_e)\sigma_{z'}^2} \right]$$

$$= \frac{1}{2} \log_2 \left( 1 + \frac{P_x}{P_z} \right) - \frac{1}{2} \log_2 \left[ G(\Lambda_2)\mu(\Lambda_1, P_e) \right]$$

(21)

When block length $n \rightarrow \infty$, there exists a good nested lattice pair for quantization and AWGN coding such that $G(\Lambda_2) \rightarrow 1/2 \pi e$ and $\mu(\Lambda_1, P_e) \rightarrow 2 \pi e$. In that scenario $\frac{1}{2} \log_2 \left[ G(\Lambda_2)\mu(\Lambda_1, P_e) \right] \rightarrow 0$; hence, lattice codes achieve the AWGN capacity [7].

However, in practice $G(\Lambda_2) > 1/2 \pi e$ due suboptimal shaping, and $\mu(\Lambda_1, P_e) > 2 \pi e$ due to suboptimal coding. Let $\beta_s$ and $\beta_c$ be the shaping and coding loss for any practical lattice scheme in general. Then we obtain second moment of $\Lambda_2$ and VNR for LDLC

$$G(\Lambda_2) = \beta_s 2 \pi e$$

(22)

$$\mu(\Lambda_1, P_e) = 2 \pi e \beta_c$$

(23)

Then, we find the achievable rate for practical lattices, such as LDLCs, in terms of their shaping and coding loss. With MMSE scaling,

$$R_s = \frac{1}{2} \log_2 \left[ \frac{P_x/G(\Lambda_2)}{\mu(\Lambda_1, P_e)\sigma_{z'}^2} \right]$$

$$= \frac{1}{2} \log_2 \left( 1 + \frac{P_x}{P_z} \right) - \frac{1}{2} \log_2 (\beta_s \beta_c)$$

(24)

If we use $\alpha = 0$, that is, if we do not employ scaling, the rate
The hypercube scenario with

\[ \beta \]

scenario where

\[ \beta \]

scenario where

\[ \beta \]

Figure 2. Three cases are plotted; 1) The perfect nested lattice

\[ R \]

power, the effect of scaling is reduced.

\[ R \]

losses, increases the power requirement to achieve the fix rate

\[ R \]

increase of coding and shaping losses, the power gain with the

achievable rate expression. One can observe that with the

shaping and coding losses due to the addition of 1 in the

impact for low rates even for the practical lattice codes with

we have plotted

\[ P \]

In order to observe how scaling is effective for different rates,

Fig. 2. Required power comparison to achieve a fix rate \( R \) with and without

scaling.

is

\[ R_{ns} = \frac{1}{2} \log_2 \left( \frac{P_x}{P_z} \right) - \frac{1}{2} \log_2 (\beta_x \beta_c) \]  \hspace{1cm} (25)

**(Example):** For LDLC for block length equal to 10000 and rate

\[ = 2.935\text{bits/dimension} \] the shaping and coding combine loss is

roughly 2.8dB for hypercube shaping and 2.4dB for nested lattice shaping [6].

We obtain the transmit power requirement to achieve fixed

rate of \( R \) for scaling and non-scaling cases assuming noise variance \( P_z = 1 \) without of loss of generality.

\[ P_x^s = 2^{2R} \beta_x \beta_c - 1 \]  \hspace{1cm} (26)

\[ P_x^{ns} = 2^{2R} \beta_x \beta_c \]  \hspace{1cm} (27)

In order to observe how scaling is effective for different rates,

we have plotted \( P_x^{ns}/P_x^s \) in dB scale for different rates in

Figure 2. Three cases are plotted; 1) The perfect nested lattice scenario where \( \beta_x = \beta_c = 1 \), 2) The hypercube shaping scenario where \( \beta_x = 2\pi e/12 \) (1.53dB) and no coding loss and 3) The hypercube scenario with \( \beta_c = 0.5\text{dB} \)  \footnote{For LDLC with block length 100000, the coding loss is 0.5dB}.

It is observed in Figure 2 that the scaling plays a significant

impact for low rates even for the practical lattice codes with

shaping and coding losses due to the addition of 1 in the

achievable rate expression. One can observe that with the

increase of coding and shaping losses, the power gain with the

use of scaling decreases. This is due to the fact that increase of

losses, increases the power requirement to achieve the fix rate

\( R \) irrespective of scaling. Hence, with the increase of transmit

power, the effect of scaling is reduced.

C. Lattice Scaling Using Belief Propagation

Belief propagation principles can be applied to obtain the

MMSE scaling coefficient given by Erez and Zamir. While

the MMSE scaling coefficient was obtained using information

theoretic approaches [7], the significance here is that belief

propagation principles alone can be used to arrive at the same

result [11]. Recall that when the MMSE scaling coefficient

\[ \alpha_m = P_x/(P_x + P_z) \] is used, the effective noise is \( P_x/P_z \) [7, eqn. (29)].

The belief propagation perspective is as follows. The belief

propagation decoder should be modified to take into account

the transmitted codebook. In the ideal case, the transmitted

signal is a mean zero, variance \( P_z \) Gaussian, due to the shaping

region. Under belief-propagation principles, this contribution

of the shaping region should be multiplied at each variable

node \( k \), since it is information about the transmitted signal \( x_k \). Because this message is iteration-independent, multiply the

shaping region message and the channel message \( \mathcal{N}(y_k/P_z) \)

to obtain a new variable node input message, as in Fig. 3-(a).

Since both are Gaussian, the product will also be a Gaussian

(exactly):

\[ \mathcal{N}(y_k', P_z') = \mathcal{N}(0, P_x) \times \mathcal{N}(y_k/P_z), \]  \hspace{1cm} (28)

with mean \( y_k' \) and variance \( P_z' \). The mean and variance of the

new Gaussian can be found as:

\[ y_k' = \frac{P_x}{P_x + P_n} y_k, \]  \hspace{1cm} (29)

and

\[ P_z' = \frac{P_x P_z}{P_x + P_z}. \]  \hspace{1cm} (30)

Observe that the new mean \( y_k' \) is scaling by the MMSE coefficient,

that is, we found the MMSE coefficient by using belief-

propagation arguments alone. In addition, the corresponding

new variance is the same as the effective noise found using

information theoretic arguments. This is illustrated in Fig. 3-

(b).

IV. Numerical Section

Fig. 4 shows the symbol error rate versus average SNR for

block length \( n = 100 \). We have fixed the rate to 0.65

bits/dimension by having zeros for last 35 integer locations and

\( L_i = 2, \forall i = \{1, \ldots, 65\} \). One can notice a better symbol error rate (SER) performance when scaling is used. We observe

\( \sim 1\text{dB} \) gain by scaling before decoding. This is encouraging

since this gain is obtained without adding any complexity to

existing LDLC decoding algorithm. Moreover, nested lattice
shaping (with scaling) curve shows it has close to $\sim 2\text{dB}$ gain over hypercube shaping without scaling.

Fig. 4 illustrates the SER versus average SNR for different $\alpha$. We have plotted for $\alpha = \alpha_m + [0 + 0.05 - 0.02 - 0.05 - 0.1 - 0.15]$. Since we have not used dither, effective noise is correlated with signal [7]. Due to this correlation, the optimal $\alpha$ is not equal to $\alpha_m$ for small dimension as shown in [7], however, for large dimensions $\alpha \to \alpha_m$. It is observed from the figure that $\alpha_m$ outperforms other cases and as expected the SER increases as $\alpha$ deviates from $\alpha_m$. Hence, we conclude that the optimal $\alpha = \alpha_m$ for dimension as small as $n = 100$.

The symbol error rate versus average SNR for $n = 10000$ is plotted in Fig. 6. The degree of rows of $H$ and constellation sizes are selected as in [6] which correspond to $R = 1$ bits/dimension and used nested lattice shaping. We observe $\sim 0.6 \text{ dB}$ performance gain with the use of scaling. Further, it is only $\sim 0.7\text{dB}$ away from the bound of uniform input distribution without scaling. We have to note that due the use of coarse constellation for the unprotected integers, there is a power loss of $\sim 0.8\text{dB}$ and this loss can be further reduce by selecting better constellation sizes. We conclude that with the use of scaling, the LDLC even perform well at low rates.

V. Conclusion

We have studied the value of MMSE scaling for low density lattice codes, showing significant performance improvement at low SNR and low rates. Further, we have shown that, belief propagation principles alone can be applied to obtain the MMSE scaling coefficient by considering the shaping region. MMSE scaling has potential to substantially improve the performance of practical lattice schemes.

REFERENCES