Sliced Lattice Gaussian Sampling: Convergence Enhancement and Decoding Optimization

Zheng Wang, *Member, IEEE,* Cong Ling, *Member, IEEE,* and Ling Liu, *Member, IEEE*

***Abstract*—Sampling from the lattice Gaussian distribution has emerged as a key problem in coding and decoding while Markov chain Monte Carlo (MCMC) methods from statistics offer an effective way to solve it. In this paper, the sliced lattice Gaussian sampling algorithm is proposed to further improve the convergence performance of the Markov chain targeting at lattice Gaussian sampling. We demonstrate that the Markov chain arising from it is uniformly ergodic, namely, it converges exponentially fast to the stationary distribution. Meanwhile, the convergence rate of the underlying Markov chain is also investigated, and we show the proposed sliced sampling algorithm entails a better convergence performance than the independent Metropolis-Hastings-Klein (IMHK) sampling algorithm. On the other hand, the decoding performance based on the proposed sampling algorithm is analyzed, where the optimization with respect to the standard deviation σ is given. After that, a judicious mechanism based on distance judgement and dynamic updating for choosing σ is proposed for a better decoding performance Finally, simulation results based on multiple-input multiple-output (MIMO) detection are presented to confirm the performance gain by convergence enhancement.**

***Index Terms*—Lattice Gaussian sampling, slice sampling, MCMC methods, lattice coding and decoding, MIMO detection.**

I. Introduction

NOwadays, the large-scale multiple-input multiple-output (MIMO) system has become a promising extension of MIMO in 5G, which boosts the network capacity on a much greater scale without extra bandwidth [1]–[4]. However, the dramatically increased system size also places a pressing challenge on the signal detection, which actually belongs to the closest vector problem (CVP) in lattice decoding. On one hand, the advanced detection schemes designed for traditional MIMO systems like lattice-reduction-aided detection show a substantial performance loss with the increment number of antennas [5]–[9]. On the other hand, a number of maximum-likelihood (ML) decoding schemes that aim to reduce the computational complexity of sphere decoding (SD) turn out to be impractical due to the unaffordable complexity in high-dimensional systems [10]–[14]. As for those near-ML decoding schemes like fixed-complexity sphere decoding (FCSD), K-best decoder, etc., they are also inapplicable due to the intensive complexity increment and terrible performance deterioration [15]–[19].

To this end, a number of works have been made by either improving the performance or lowing the complexity [20]–[24]. Among them, sampling detection has become the promising one, which performs lattice decoding by sampling from a discrete Gaussian distribution over lattices (i.e., lattice Gaussian distribution) [25]–[28]. Essentially, sampling detection converts the traditional detection problem into a sampling problem, where the optimal decoding solution with the smallest Euclidean distance entails the largest probability to be sampled. Therefore, if sampling can be efficiently implemented, the decoding problem would be addressed in an effective way. However, the problem of sampling detection chiefly lies on how to successfully sample over the target lattice Gaussian distribution.

In sharp contrast to the continuous Gaussian density, it is by no means trivial even to sample from a low-dimensional discrete Gaussian distribution. Because of this, the pioneer work of sampling detection based on Klein's algorithm only performs the sampling over a Gaussian-like distribution, which means the performance loss due to the distortion by the Gaussian-like distribution is inevitable [27]–[29]. On the other hand, the classic Gibbs algorithm from Markov chain Monte Carlo (MCMC) methods has also been adopted to MIMO detection through sampling from the lattice Gaussian distribution [30]–[34]. However, since the convergence rate of the Markov chain is hard to determine, the Markov mixing turns out to be untractable so that the related decoding analysis is still lacking. Fortunately, a remarkable progress has been made by the independent Metropolis-Hastings-Klein (IMHK) algorithm given in [35], which is not only uniformly ergodic in tackling with lattice Gaussian sampling but also enjoys an accessible convergence rate. In [36], IMHK algorithm was further applied into lattice decoding to solve the CVP, where a better trade-off between performance and complexity in terms of bounded distance decoding (BDD) has been achieved.

Besides lattice decoding, lattice Gaussian sampling has already become a common theme in various research fields. Specifically, in mathematics, Banaszczyk used it to prove the transference theorems for lattices [37]. In coding, it was

This work was partially presented in part at IEEE International Symposium on Information Theory (ISIT), Paris, France, July, 2019. This work was supported in part by the open research fund of Key Laboratory of Dynamic Cognitive System of Electromagnetic Spectrum Space (Nanjing Univ. Aeronaut. Astronaut.), Ministry of Industry and Information Technology, Nanjing, 211106, China (No. KF20181913), the open research fund of National Mobile Communications Research Laboratory, Southeast University (No. 2019D04), the National Natural Science Foundation of China under Grant 61801216, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20180420.

Z. Wang is with College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China; C. Ling is with the Department of Electrical and Electronic Engineering, Imperial College London, London, SW7 2AZ, United Kingdom. L. Liu is with the College of Computer Science and Software Engineering, Shenzhen University, 518060, China (e-mail: z.wang@ieee.org, cling@ieee.org, liulingcs@szu.edu.cn).

applied to achieve the full shaping gain for lattice coding [38], and to achieve the capacity of the Gaussian channel and the secrecy capacity of the Gaussian wiretap channel, respectively [39], [40]. In cryptography, lattice Gaussian distribution has already become a central tool in the construction of many primitives. Specifically, Micciancio and Regev applied it to propose the lattice-based cryptosystems based on the worst-case hardness assumptions [41]. Meanwhile, it also has underpinned the fully-homomorphic encryption for cloud computing [42]. Moreover, lattice Gaussian distribution has been adapted to bidirectional relay network under the compute-and-forward strategy for the physical layer security [43]. Additionally, it is also applied to realize the probabilistic shaping for optical communication systems [44], [45].

In this paper, the state of the art of sampling decoding is advanced from two perspectives. On one hand, in order to improve the convergence performance of MCMC-based sampling algorithm, the proposed sliced lattice Gaussian sampling algorithm is given. Compared to IMHK sampling, auxiliary variables are employed by the proposed sliced sampling to enhance the convergence rate with negligible computational increment. We demonstrate that the Markov chain induced by it is uniformly ergodic, which means the Markov chain converges to the target distribution in an exponential way. Then the convergence analysis is carried out and we show that the convergence rate of the proposed sampling is superior to that of IMHK, thus making it a better choice for lattice Gaussian sampling. Consequently, the performance of sampling decoding can be improved by a more efficient lattice Gaussian sampling algorithm.

On the other hand, in sampling decoding based on MCMC methods there is a latent trade-off with respect to the standard deviation σ of the lattice Gaussian distribution: a large choice of σ naturally leads to a faster convergence rate but the sampling probability of the target point in lattice Gaussian distribution would decrease accordingly, and vice versa. To this end, the selection of σ is fully investigated for a better sampling decoding performance. First of all, a near-optimal choice of $\sigma = d(\Lambda, \mathbf{c})/\sqrt{n}$ is derived and we show it is better than the choice $\sigma = \min_i \|\widehat{\mathbf{b}}_i\|/2\sqrt{\pi}$ provided in [36] when $d(\Lambda, \mathbf{c}) \geq \sqrt{n}\min\|\widehat{\mathbf{b}}_i\|/2\sqrt{\pi}$ (n is the system dimension, $\widehat{\mathbf{b}}_i$'s are the Gram-Schmidt vectors of the lattice basis $\mathbf{B}$, $d(\Lambda, \mathbf{c}) = \min_{\mathbf{x}\in\mathbb{Z}^n} \|\mathbf{B}\mathbf{x} - \mathbf{c}\|$ stands for the Euclidean distance between the query point $\mathbf{c}$ and the lattice Λ with basis $\mathbf{B}$). Based on it, the related decoding complexity as well as decoding radius in terms of BDD is derived, and we show that CVP can be solved with complexity $O(e^{\frac{n}{2}})$ if $d(\Lambda, \mathbf{c}) \leq \sqrt{\frac{n}{2\pi}} \cdot |\det(\mathbf{B})|^{\frac{1}{n}}$. Moreover, a judicious judgement mechanism for choosing σ based on $d(\Lambda, \mathbf{c})$ is proposed, where $d_{\text{initial}}(\Lambda, \mathbf{c})$ based on the output of suboptimal decoding scheme is applied to approximate $d(\Lambda, \mathbf{c})$. By dynamically updating $d_{\text{learning}}(\Lambda, \mathbf{c})$ through the sampled candidates, considerable performance gain can be achieved.

The rest of this paper is organized as follows. Section II introduces the lattice Gaussian distribution and briefly reviews the basics of sampling detection as well as IMHK sampling algorithm. In Section III, based on the traditional slice algorithm in MCMC, the proposed sliced lattice Gaussian sampling algorithm is presented. In Section IV, with respect to the proposed algorithm, the related convergence analysis is carried out, where the demonstration of uniform ergodicity and the convergence rate diagnose are given. The decoding analysis regarding to optimizing the choice of σ in sampling decoding is presented in Section V and simulation results for MIMO detection are illustrated in Section VI. Finally, Section VII concludes the paper.

Notation: Matrices and column vectors are denoted by upper and lowercase boldface letters, and the transpose, inverse, pseudoinverse of a matrix $\mathbf{B}$ by $\mathbf{B}^T, \mathbf{B}^{-1}$, and $\mathbf{B}^\dagger$, respectively. We use $\mathbf{b}_i$ for the ith column of the matrix $\mathbf{B}$, $b_{i,j}$ for the entry in the ith row and jth column of the matrix $\mathbf{B}$. Meanwhile, $\widehat{\mathbf{b}}_i$'s are the Gram-Schmidt vectors of the matrix $\mathbf{B}$. In addition, we use the standard *small omega* notation $\omega(\cdot)$, i.e., $|\omega(g(n))| > k \cdot |g(n)|$ for every fixed positive number $k > 0$. Finally, in this paper, the computational complexity is measured by the number of Markov moves.

II. Preliminaries

In this section, we introduce the background and mathematical tools needed to describe and analyze the proposed sliced lattice Gaussian sampling algorithm.

A. Lattice Gaussian Distribution

Let matrix $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_n] \subset \mathbb{R}^n$ consist of n linearly independent column vectors. The n-dimensional lattice Λ generated by $\mathbf{B}$ is defined by

$$\Lambda = \mathcal{L}(\mathbf{B}) = \{\mathbf{B}\mathbf{x} : \mathbf{x} \in \mathbb{Z}^n\}, \tag{1}$$

where $\mathbf{B}$ is called the lattice basis. We define the Gaussian function centered at $\mathbf{c} \in \mathbb{R}^n$ for standard deviation $\sigma > 0$ as

$$\rho_{\sigma,\mathbf{c}}(\mathbf{z}) = e^{-\frac{\|\mathbf{z}-\mathbf{c}\|^2}{2\sigma^2}}, \tag{2}$$

for all $\mathbf{z} \in \mathbb{R}^n$. When $\mathbf{c}$ or σ are not specified, we assume that they are $\mathbf{0}$ and 1 respectively. Then, the *discrete Gaussian distribution* over Λ is defined as

$$D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x}) = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\mathbf{x})}{\rho_{\sigma,\mathbf{c}}(\Lambda)} = \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}-\mathbf{c}\|^2}}{\sum_{\mathbf{x}\in\mathbb{Z}^n} e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}-\mathbf{c}\|^2}} \tag{3}$$

for all $\mathbf{x} \in \mathbb{Z}^n$, where $\rho_{\sigma,\mathbf{c}}(\Lambda) \triangleq \sum_{\mathbf{B}\mathbf{x}\in\Lambda} \rho_{\sigma,\mathbf{c}}(\mathbf{B}\mathbf{x})$ is just a scaling to obtain a probability distribution. We remark that this definition differs slightly from the one in [41], where σ is scaled by a constant factor $\sqrt{2\pi}$ (i.e., $s = \sqrt{2\pi}\sigma$). In fact, the discrete Gaussian resembles a continuous Gaussian distribution, but is only defined over a lattice. It has been shown that discrete and continuous Gaussian distributions share similar properties, if the *flatness factor* is small [40].

B. Decoding by Sampling

Consider the decoding of an $n \times n$ real-valued system. The extension to the complex-valued system is straightforward [27]. Let $\mathbf{x} \in \mathbb{Z}^n$ denote the transmitted signal. The corresponding received signal $\mathbf{c}$ is given by

$$\mathbf{c} = \mathbf{B}\mathbf{x} + \mathbf{w} \tag{4}$$

Algorithm 1 IMHK Sampling Algorithm

Require: $\mathbf{B}, \sigma, \mathbf{c}, \mathbf{x}_0, t_{\text{mix}}(\epsilon)$;
Ensure: $\mathbf{x} \sim D_{\Lambda,\sigma,\mathbf{c}}$;
let $\mathbf{X}_0 = \mathbf{x}_0$
for t =1,2, …, **do**
  let $\mathbf{x}$ denote the state of $\mathbf{X}_{t-1}$
  sample $\mathbf{y}$ from the proposal distribution $q(\mathbf{x}, \mathbf{y})$ in (8)
  calculate the acceptance ratio $\alpha(\mathbf{x}, \mathbf{y})$ in (9)
  generate a sample u from the uniform density $U[0, 1]$
  if $u \leq \alpha(\mathbf{x}, \mathbf{y})$ **then**
    let $\mathbf{X}_t = \mathbf{y}$
  else
    $\mathbf{X}_t = \mathbf{x}$
  end if
  if $t \geq t_{\text{mix}}(\epsilon)$ **then**
    output $\mathbf{x}$
  end if
end for

where $\mathbf{w}$ is the noise vector with zero mean and variance σ_w^2, $\mathbf{B}$ is an $n \times n$ full column-rank matrix of channel coefficients. Typically, the conventional maximum likelihood (ML) reads

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x} \in \mathbb{Z}^n} \|\mathbf{c} - \mathbf{B}\mathbf{x}\|^2 \tag{5}$$

where $\|\cdot\|$ denotes the Euclidean norm. Clearly, ML decoding corresponds to the closest vector problem (CVP) in lattices. If the received signal $\mathbf{c}$ is the origin, then ML decoding reduces to shortest vector problem (SVP).

Intuitively, the CVP given in (5) can be solved by the lattice Gaussian sampling. Since the distribution is centered at the query point $\mathbf{c}$, the closest lattice point $\mathbf{B}\mathbf{x}$ to $\mathbf{c}$ is assigned the largest sampling probability. Therefore, by multiple samplings, the solution of CVP is the most likely to be returned. It has been demonstrated that lattice Gaussian sampling is equivalent to CVP via a polynomial-time dimension-preserving reduction [46]. Meanwhile, the standard deviation σ of the discrete Gaussian distribution can be optimized to improve the sampling probability of the target point. By adjusting the sample size, the sampling decoder enjoys a flexible trade-off between performance and complexity. However, the premise behind decoding by sampling relies on how to successfully sample from lattice Gaussian distribution.

In [29], Klein's algorithm that samples from a Gaussian-like distribution was proposed for lattice decoding. Specifically, by sequentially sampling from the 1-dimensional conditional Gaussian distribution $D_{\mathbb{Z},\sigma_i,\widetilde{x}_i}$ in a backward order from x_n to x_1, the Gaussian-like distribution arising from Klein's algorithm is given by

$$P_{\text{Klein}}(\mathbf{x}) = \prod_{i=1}^{n} D_{\mathbb{Z},\sigma_i,\widetilde{x}_i}(x_i) = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\mathbf{x})}{\prod_{i=1}^{n} \rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})}. \tag{6}$$

In [47], $P_{\text{Klein}}(\mathbf{x})$ has been demonstrated to be close to $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ within a negligible statistical distance if

$$\sigma \geq \omega(\sqrt{\log\, n}) \cdot \max_{1 \leq i \leq n} \|\widehat{\mathbf{b}}_i\|. \tag{7}$$

Unfortunately, such a requirement of σ is sufficiently large, rendering Klein's algorithm inapplicable to most cases of lattice Gaussian sampling.

C. IMHK Sampling for Lattice Gaussian Distribution

In order to sample from a target lattice Gaussian distribution, Markov chain Monte Carlo (MCMC) methods were introduced [35], [36]. In principle, it randomly generates the next Markov state conditioned on the previous one; after the burn-in time, the Markov chain will step into a stationary distribution, where samples from the target distribution can be obtained thereafter [48]. From MCMC perspective, $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ can be viewed as a complex target distribution lacking direct sampling methods, and the independent Metropolis-Hastings-Klein (IMHK) sampling that fully exploits the potential of MCMC was therefore proposed in [35].

In particular, given the current Markov state $\mathbf{X}_t = \mathbf{x}$, $P_{\text{Klein}}(\mathbf{x})$ from Klein's algorithm is used to serve as the proposal distribution $q(\mathbf{x}, \mathbf{y})$ in IMHK:

$$q(\mathbf{x}, \mathbf{y}) = P_{\text{Klein}}(\mathbf{y}) = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\mathbf{y})}{\prod_{i=1}^{n} \rho_{\sigma_i,\widetilde{y}_i}(\mathbb{Z})}, \tag{8}$$

where the generation of the state candidate $\mathbf{y}$ is actually independent of $\mathbf{x}$. Then, regarding to the state candidate $\mathbf{y}$, the acceptance ratio α is calculated by

$$\alpha(\mathbf{x}, \mathbf{y}) = \min\left\{1, \frac{\pi(\mathbf{y}) q(\mathbf{y}, \mathbf{x})}{\pi(\mathbf{x}) q(\mathbf{x}, \mathbf{y})}\right\} = \min\left\{1, \frac{\prod_{i=1}^{n} \rho_{\sigma_i \widetilde{y}_i}(\mathbb{Z})}{\prod_{i=1}^{n} \rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})}\right\}, \tag{9}$$

where $\pi = D_{\Lambda,\sigma,\mathbf{c}}$. In the sequel, the decision for whether accept $\mathbf{X}_{t+1} = \mathbf{y}$ or not is made based on $\alpha(\mathbf{x}, \mathbf{y})$, thus completing a Markov move. Overall, the transition probability of the IMHK sampling over two Markov states is

$$P_{\text{IMHK}}(\mathbf{x}, \mathbf{y}) = q(\mathbf{x}, \mathbf{y}) \cdot \alpha(\mathbf{x}, \mathbf{y}) = \min\left\{P_{\text{Klein}}(\mathbf{y}), \frac{\pi(\mathbf{y}) P_{\text{Klein}}(\mathbf{x})}{\pi(\mathbf{x})}\right\} \tag{10}$$

Theorem 1 ([35])**.** *Given the invariant lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}$, the Markov chain established by the IMHK algorithm is uniformly ergodic:*

$$\|P^t(\mathbf{x}, \cdot) - D_{\Lambda,\sigma,\mathbf{c}}(\cdot)\|_{TV} \leq (1 - \delta)^t \tag{11}$$

with

$$\delta \triangleq \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{\Lambda})}{\prod_{i=1}^{n} \rho_{\sigma_i}(\mathbb{Z})} \tag{12}$$

for all $\mathbf{x} \in \mathbb{Z}^n$.

Clearly, the exponential decay coefficient δ is the key to determine the convergence rate. More specifically, the convergence rate of a Markov chain is dominated by its spectral gap $\tau = 1 - |\lambda_{\text{max}}|$, where $|\lambda_{\text{max}}| \neq 1$ denotes the second largest eigenvalue of the transition matrix (the largest eigenvalue is always 1) [48].

III. Slice Sampling for Lattice Gaussian Distribution

In this section, we present the conventional slice sampling in MCMC and give the proposed sliced lattice Gaussian sampling

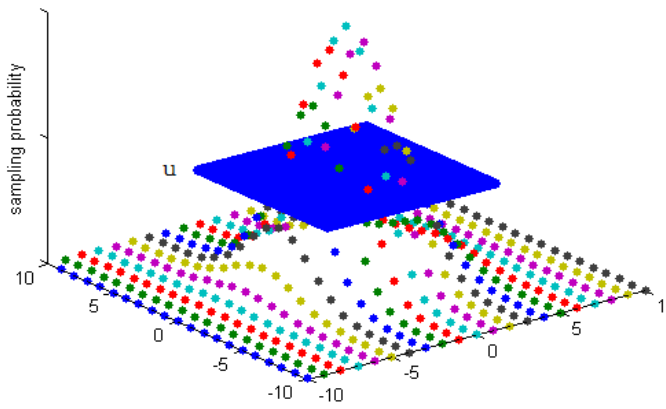

Fig. 1. Illustration of a two-dimensional lattice Gaussian distribution and a slice (blue plane) with $u \geq 0$ over it.

algorithm. Note that the Markov chain that we are concerned with here has a countably infinite state space, i.e., the lattice Λ with $\mathbf{x} \in \mathbb{Z}^n$.

A. Slice Sampling

The classical slice sampling was generalized by Neal in [49]. In principle, it relies on the fact that uniformly sampling from the region under the curve of a density function is actually equal to drawing samples directly from that distribution. Take a multi-dimensional target distribution $\pi(\mathbf{x})$ as an example, auxiliary variable $u \geq 0$ is introduced to sample from target distribution $\pi(\mathbf{x})$ by sampling from the uniform distribution over the set $S = \{(\mathbf{x}, u) : 0 \leq u \leq \pi(\mathbf{x})\}$ and marginalizing out u coordinate. To achieve this, slice sampling alternatively updates $\mathbf{x}$ and u from uniform distributions $p(\mathbf{x} \mid u) \sim Uni(S)$ and $p(u \mid \mathbf{x}) \sim Uni(0, \pi(\mathbf{x}))$ respectively, thus forming a valid Markov chain with joint distribution $\Pi(\mathbf{x}, u)$. Consequently, samples of $\mathbf{x}$ can be easily drawn from the marginal distribution $\pi(\mathbf{x})/Z$, where $Z > 0$ is a constant scalar. Overall, the operation of slice sampling can be summarized as follows:

1) *Sample u_t from the conditional distribution*

$$p(u_t \mid \mathbf{x}_{t-1}) \sim Uni(0, \pi(\mathbf{x}_{t-1})). \tag{13}$$

2) *Sample $\mathbf{x}_t$ from the conditional distribution*

$$p(\mathbf{x}_t \mid u_t) \sim Uni(S_u), \tag{14}$$

where $S_u = \{\mathbf{x} : \pi(\mathbf{x}) \geq u\}$.

Clearly, the samples of $\mathbf{x}$ are obtained by simply ignoring the values of u while only uniform sampling is required over the set S_u. However, in many cases of interest, determining the set S_u may be tricky especially for multi-modal distributions. Compared to the conventional Metropolis-Hastings (MH) sampling, a salient feature of slice sampling is that the sampled candidate $\mathbf{x}$ from (14) will be accepted as $\mathbf{X}_t = \mathbf{x}_t$ without uncertainty. In this way, the underlying Markov chain effectively avoids the risk of getting stuck, thus making the traverse of the state space of the Markov chain more efficiently. Hence, if the identity of S_u can be carried out, then slice sampling becomes preferable due to the considerable convergence gain. In fact, as lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ is simply unimodal, finding the slice and sampling from it could be straightforward, which motivates us to incorporate slice sampling into lattice Gaussian distribution for a better sampling performance.

B. Sliced Lattice Gaussian Sampling Algorithm

Inspired by the works of slice sampling [50]–[52], we now present the proposed sliced sampling algorithm for lattice Gaussian distribution. First of all, a Markov chain $\{\mathbf{X}_t, U_t\}_{t=0}^{\infty}$ with joint distribution $\Pi(\mathbf{x}, u)$ should be set up. Typically, given the factorization of the target distribution

$$\pi(\mathbf{x}) = D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x}) = P_{\text{Klein}}(\mathbf{x}) \cdot l(\mathbf{x}) \tag{15}$$

with

$$l(\mathbf{x}) \triangleq \frac{\prod_{i=1}^{n} \rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})}{\rho_{\sigma,\mathbf{c}}(\Lambda)}, \tag{16}$$

we can establish the joint distribution as

$$\Pi(\mathbf{x}, u) = P_{\text{Klein}}(\mathbf{x}) \cdot I_{u<l(\mathbf{x})}(\mathbf{x}), \tag{17}$$

where $\rho_{\sigma,\mathbf{c}}(\Lambda)$ is a scalar and $I_A(\mathbf{x})$ is the indicator function of the set A. Compared to the original slice sampling, the factorization of the target distribution in (15) is adopted to the proposed slice sampling, which results in a slightly different expression shown in (17). In fact, it was pointed out by Besag & Green in [53] that the usage of decomposition is rather effective in multidimensi onal problems (especially when $P_{\text{Klein}}(\mathbf{x})$ has a simpler structure than $\pi(\mathbf{x})$).

More specifically, the conditional uniform distribution of u lies on the interval $(0, l(\mathbf{x}))$ by incorporating u and $l(\mathbf{x})$ together. By doing this, u and $\mathbf{x}$ are iteratively updated by respectively sampling from uniform distribution on $(0, l(\mathbf{x}))$ given $\mathbf{x}$ and from $P_{\text{Klein}}(\mathbf{x})$ restricted to the set $A_u = \{\mathbf{x} : l(\mathbf{x}) > u\}$, i.e.,

1) *Sample u_t from the conditional distribution*

$$p(u_t \mid \mathbf{x}_{t-1}) \sim Uni(0, l(\mathbf{x}_{t-1})). \tag{18}$$

2) *Sample $\mathbf{x}_t$ from the conditional distribution*

$$p(\mathbf{x}_t \mid u_t) \sim P_{\text{Klein}}^{A_{u_t}}(\mathbf{x}), \tag{19}$$

where $\mathbf{x} \in A_{u_t} = \{\mathbf{x} : l(\mathbf{x}) > u_t\}$.

Following [53], it is straightforward to verify that the iterative samplings from (18) and (19) lead to the joint distribution $\Pi(\mathbf{x}, u)$ in (17).

Intuitively, with respect to (19), sampling from $P_{\text{Klein}}(\mathbf{x})$ can be efficiently implemented by Klein's algorithm with complexity $O(n^2)$, whereas the restriction of $\mathbf{x} \in A_{u_t}$ can be simply addressed by resorting to rejection sampling. If $\mathbf{x} \notin A_{u_t}$, then repeat the sampling until a qualified candidate is found for $\mathbf{x}_t$. In fact, since the candidates $\mathbf{x}$ with large sampling probabilities are most likely to be obtained by Klein's sampling, the rejection ratio of the rejection sampling normally turns out to be low. Meanwhile, to avoid the risk of being rejected, one also can sample $\mathbf{x}$ from $P_{\text{Klein}}(\mathbf{x})$ under the

restriction $\mathbf{x} \in A_{u_t}$ directly[1]. Interestingly, the numerator in (16) has already been calculated by Klein's algorithm during the sampling while the denominator in (16) only serves as a scalar that can be set free for non-negative values. Therefore, the computational cost by incurring rejection sampling tends to be quite low.

To summarize, the proposed sliced lattice Gaussian sampling algorithm is presented in Algorithm 2. More precisely, the complexity of slice sampling in each single Markov move is easily accepted with $O(n^2)$, which is the same order with the IMHK and Gibbs sampling algorithms. For this reason, in MCMC the complexity of each Markov move is often insignificant, whereas the number of Markov moves is more critical. Additionally, we emphasize that the framework of slice sampling actually contributes several degrees of freedom: the choice of the conditional distribution of the auxiliary variable $p(u_t \mid \mathbf{x}_{t-1})$, the decomposition way of $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$, and the update schedule scheme between $\mathbf{x}$ and u. Here, for convenience, the systematic update scheme that updates $\mathbf{x}$ and u sequentially is considered through the context. In a word, reasonable performance gain is available by further exploiting these freedom degrees.

Algorithm 2 Sliced Lattice Gaussian Sampling Algorithm

Input: $\mathbf{B}, \sigma, \mathbf{c}, \mathbf{x}_0, t_{\text{mix}}(\epsilon)$;
Output: $\mathbf{x} \sim D_{\Lambda,\sigma,\mathbf{c}}$;

```
for t =1,2, ..., do
    calculate l(x_{t-1}) according to (16)
    uniformly draw u_t from the interval (0, l(x_{t-1}))
    for k =1,2, ..., do
        sample x_t from P_klein(x) shown in (6)
        calculate l(x_t) according to (16)
        if l(x_t) > u_t then
            break
        end if
    end for
    if t ≥ t_mix(ε) then
        output x_t
    end if
end for
```

IV. Convergence Analysis

In this section, convergence analysis with respect to the proposed sliced lattice Gaussian sampling algorithm is given. The uniform ergodicity is firstly demonstrated, followed by the convergence diagnosis to show the superiority of the proposed sampling algorithm over IMHK.

A. Uniform Ergodicity

Consider the marginal distribution $\pi(\mathbf{x}) = D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ with respect to the joint distribution $\Pi(\mathbf{x}, u)$, clearly, the marginal chain $\{\mathbf{X}_1, \mathbf{X}_2, ...\}$ regarding to $\mathbf{x}$ is not only a valid Markov chain with transition probability $P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) > 0$, but also turns out to be reversible (also known as detailed balance) due to

$$\begin{aligned}
\pi(\mathbf{x}_t)P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) &= \pi(\mathbf{x}_t)\int \Pi(u_{t+1}|\mathbf{x}_t)\Pi(\mathbf{x}_{t+1}|u_{t+1})du_{t+1} \\
&= \int \Pi(\mathbf{x}_t|u_{t+1})\Pi(u_{t+1}|\mathbf{x}_t)\Pi(\mathbf{x}_{t+1}|u_{t+1})p(u_{t+1})du_{t+1} \\
&= \pi(\mathbf{x}_{t+1}) \int \Pi(u_{t+1}|\mathbf{x}_t)\Pi(\mathbf{x}_t|u_{t+1})du_{t+1} \\
&= \pi(\mathbf{x}_{t+1}) \int \Pi(u_t|\mathbf{x}_{t+1})\Pi(\mathbf{x}_t|u_t)du_t \\
&= \pi(\mathbf{x}_{t+1})P_{\text{Slice}}(\mathbf{x}_{t+1}, \mathbf{x}_t),
\end{aligned} \tag{20}$$

where $\pi(\mathbf{x}) = \int \Pi(\mathbf{x}, u)du = \int \Pi(\mathbf{x}|u)p(u)du$. Subsequently, based on the sub-Markov chain $\{\mathbf{X}_1, \mathbf{X}_2, ...\}$, its transition probability can be derived as

$$\begin{aligned}
P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) &= \int_0^{l(\mathbf{x}_t)} p(\mathbf{x}_{t+1}|u_{t+1})p(u_{t+1}|\mathbf{x}_t)du_{t+1} \\
&= \int_0^{l(\mathbf{x}_t)} P_{\text{Klein}}^{A_{u_{t+1}}}(\mathbf{x}_{t+1})p(u_{t+1}|\mathbf{x}_t)du_{t+1} \\
&= \frac{1}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)} P_{\text{Klein}}^{A_{u_{t+1}}}(\mathbf{x}_{t+1})du_{t+1} \\
&\overset{(a)}{=} \frac{1}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)} \frac{P_{\text{Klein}}(\mathbf{x}_{t+1})I_{u_{t+1}<l(\mathbf{x}_{t+1})}(\mathbf{x}_{t+1})}{P_{\text{Klein}}(A_{u_{t+1}})}du_{t+1} \\
&= \frac{P_{\text{Klein}}(\mathbf{x}_{t+1})}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)\wedge l(\mathbf{x}_{t+1})} \frac{1}{P_{\text{Klein}}(A_{u_{t+1}})}du_{t+1}
\end{aligned} \tag{21}$$

where (a) recalls *Bayes' theorem* and "$\wedge$" yields the smaller choice between two terms. Moreover, it follows that

$$\frac{1}{P_{\text{Klein}}(A_{u_{t+1}})} = \frac{1}{\sum_{\mathbf{x}\in\{\mathbf{x}:l(\mathbf{x}_t)>u_{t+1}\}} P_{\text{Klein}}(\mathbf{x})} \geq 1, \tag{22}$$

where the equality happens if and only if u_{t+1} is selected to be 0. Therefore, the following relationship holds

$$\int_0^{l(\mathbf{x}_t)\wedge l(\mathbf{x}_{t+1})} \frac{1}{P_{\text{Klein}}(A_{u_{t+1}})}du_{t+1} = \beta \int_0^{l(\mathbf{x}_t)\wedge l(\mathbf{x}_{t+1})} du_{t+1} \tag{23}$$

with the constant $\beta \geq 1$. Here, we point out that the case of $u_{t+1} = 0$ would rarely happen as it is randomly generated from the interval $(0, l(\mathbf{x}_t))$, which means the constant β is normally larger than 1 in practice.

According to (23), we can rewrite $P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1})$ as

$$\begin{aligned}
P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) &= \beta \cdot \frac{P_{\text{Klein}}(\mathbf{x}_{t+1})}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)\wedge l(\mathbf{x}_{t+1})} du_{t+1} \\
&= \beta \cdot P_{\text{Klein}}(\mathbf{x}_{t+1}) \left[1 \wedge \frac{l(\mathbf{x}_{t+1})}{l(\mathbf{x}_t)}\right] \\
&= \beta \cdot \left[P_{\text{Klein}}(\mathbf{x}_{t+1}) \wedge \frac{\pi(\mathbf{x}_{t+1})P_{\text{Klein}}(\mathbf{x}_t)}{\pi(\mathbf{x}_t)}\right] \\
&= \beta \cdot P_{\text{Klein}}(\mathbf{x}_{t+1}) \cdot \alpha(\mathbf{x}_t, \mathbf{x}_{t+1}) \\
&= \beta \cdot P_{\text{IMHK}}(\mathbf{x}_t, \mathbf{x}_{t+1}) \\
&\overset{(b)}{\geq} \beta \cdot \delta \cdot \pi(\mathbf{x}_{t+1}),
\end{aligned} \tag{24}$$

where the inequality (b) follows the fact that [41]

$$\frac{P_{\text{Klein}}(\mathbf{x})}{\pi(\mathbf{x})} = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{\Lambda})}{\prod_{i=1}^n \rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})} \geq \delta \tag{25}$$

[1]For more details of implementing Klein's sampling at each layer with finite searching space, reader are referred to [27].

for all Markov state $\mathbf{x} \in \mathbb{Z}^n$.

Actually, the result of $P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) \geq \beta \cdot \delta \cdot \pi(\mathbf{x}_{t+1})$ for all the Markov states is accordance with the definition of *small set* in literatures of MCMC [48]. Furthermore, given (24), for a reversible Markov chain, it is straightforward to demonstrate the *uniform ergodicity* of the underlying Markov chain through *coupling technique*. Here, for the consideration of simplicity, the related proof is omitted while more details about the proof can be found in [36].

Theorem 2. *Given the invariant lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}$, the sub-chain $\{\mathbf{X}_1, \mathbf{X}_2, ...\}$ established by the proposed sliced lattice Gaussian sampling algorithm is uniformly ergodic as:*

$$\|P^t(\mathbf{x}, \cdot) - D_{\Lambda,\sigma,\mathbf{c}}(\cdot)\|_{TV} \leq (1 - \delta')^t \tag{26}$$

with $\delta' = \beta \cdot \delta$ for all $\mathbf{x} \in \mathbb{Z}^n$.

Intuitively, compared to the convergence rate given in Theorem 1, the convergence performance of the proposed sliced sampling is better than that of IMHK sampling due to a larger size δ'.

B. Convergence Improvement

Similar to IMHK sampling, the proposed sliced sampling for lattice Gaussian distribution is uniformly ergodic as well, where the convergence advantage can be found from

$$P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) \geq P_{\text{IMHK}}(\mathbf{x}_t, \mathbf{x}_{t+1}). \tag{27}$$

For a better understanding, we now recall the concept of *Peskun ordering* to verify the convergence improvement of the proposed sliced sampling. Specifically, with respect to sampling from $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$, it always follows that

$$P_{\text{Slice}}(\mathbf{X}_t{=}\mathbf{x}, \mathbf{X}_{t+1}{=}\mathbf{y}) \geq P_{\text{IMHK}}(\mathbf{X}_t = \mathbf{x}, \mathbf{X}_{t+1} = \mathbf{y}) \tag{28}$$

for $\mathbf{x} \neq \mathbf{y}$, which means each off-diagonal element in transition matrix $\mathbf{P}_{\text{Slice}}$ is no smaller than that of $\mathbf{P}_{\text{IMHK}}$. From literatures of MCMC, such a case is known as *Peskun ordering* written by

$$P_{\text{Slice}}(\mathbf{X}_t, \mathbf{X}_{t+1}) \succeq P_{\text{IMHK}}(\mathbf{X}_t, \mathbf{X}_{t+1}). \tag{29}$$

We then invoke the following Theorem to show the convergence performance from Peskun ordering. Here, $L^2(\pi)$ denote the set of all function $f(\cdot)$ that are square integrable with respect to π and $v(f, \mathbf{P})$ is defined as sampler's asymptotic efficiency by

$$v(f, \mathbf{P}) = \lim_{n \to \infty} \frac{1}{n} \text{var} \left\{ \sum_{t=1}^{n} f(\mathbf{X}_t) \right\}, \tag{30}$$

where $\mathbf{X}_0, \ldots, \mathbf{X}_t$ establish the corresponding Markov chain.

Theorem 3 ([54])**.** *Suppose $\mathbf{P}_1$ and $\mathbf{P}_2$ are reversible transition matrices with the same invariant distribution and $\mathbf{P}_2 \geq \mathbf{P}_1$. Then, for all any function $f \in L_0^2(\pi) = \{f \in L^2(\pi) : E\{f\} = 0\}$, we have*

$$v(f, \mathbf{P}_1) \geq v(f, \mathbf{P}_2). \tag{31}$$

Clearly, from Theorem 3, the proposed sliced sampling has a smaller asymptotic variance of sample path averages than IMHK for every function that obeys the central limit theorem (CLT). Theoretically, the insight behind Peskun ordering is that a Markov chain has smaller probability of remaining in the same position explores the state space more efficiently. Hence, convergence performance is improved by shifting probabilities off the diagonal of the transition matrix, which corresponds to decreasing the rejection probability of the proposed moves. Moreover, in [50], Mira shows that if two transition matrices are Peskun ordered as $\mathbf{P}_2 \geq \mathbf{P}_1$, then their corresponding second largest eigenvalues satisfy

$$|\lambda_{\max,1}| \geq |\lambda_{\max,2}|, \tag{32}$$

where convergence rate in uniform ergodicity is exactly characterized by the second largest eigenvalue $|\lambda_{\max}|$. Therefore, we can easily arrive at the following result to show the convergence gain of the proposed sliced sampling.

Corollary 1. *The proposed sliced sampling algorithm is more efficient than the IMHK sampling algorithm to converge to the target lattice Gaussian distribution due to a better convergence rate by*

$$|\lambda_{\max}|_{\text{Slice}} \leq |\lambda_{\max}|_{\text{IMHK}} \tag{33}$$

for all $\mathbf{x} \in \mathbb{Z}^n$.

As an important parameter to measure the time required by a Markov chain to get close to its stationary distribution, the *mixing time* is defined as [48]

$$t_{\text{mix}}(\epsilon) = \min\{t : \max\|P^t(\mathbf{x}, \cdot) - \pi(\cdot)\|_{TV} \leq \epsilon\}. \tag{34}$$

Obviously, given the value of $\delta' < 1$, the mixing time of the Markov chain can be calculated by (34) and (26), that is,

$$t_{\text{mix}}(\epsilon) = \frac{\ln \epsilon}{\ln(1 - \delta')} \leq (-\ln \epsilon) \cdot \left(\frac{1}{\delta'} \right), \quad \epsilon < 1 \tag{35}$$

where we use the bound $\ln c < c - 1$ for $0 < c < 1$. Therefore, the mixing time is proportional to $1/\delta'$, and becomes $O(1)$ as $\delta' \to 1$. From this point of view, the superiority of the proposed sliced sampling over IMHK is determined by $\beta \geq 1$. On the other hand, it is straightforward to see that $P_{\text{Klein}}(A_{u_{t+1}})$ decreases with the improvement of σ. This is actually in line with the fact that a larger σ corresponds to a faster convergence rate. Clearly, if σ is sufficiently large, then sampling from $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ can be realized immediately.

V. Decoding Analysis

In this section, we apply the proposed sliced sampling to solve the CVP and analyze its complexity with respect to the choice of the standard deviation σ. As mentioned before, the decoding complexity of MCMC is evaluated by the number of Markov moves. In MCMC, samples from the stationary distribution tend to be correlated with each other. Thus one leaves a gap, which is the mixing time t_{mix}, to pick up the desired independent samples (alternatively, one can run multiple Markov chains in parallel to guarantee i.i.d. samples). Therefore, following the configuration in [36], the complexity of solving CVP by MCMC is defined as follows.

Definition 1. *Let $d(\Lambda,\mathbf{c})=\min_{\mathbf{x}\in\mathbb{Z}^n}\|\mathbf{Bx}-\mathbf{c}\|$ denote the Euclidean distance between the query point $\mathbf{c}$ and the lattice Λ with basis $\mathbf{B}$, and let $\widehat{\mathbf{x}}$ be the lattice point achieving $d(\Lambda,\mathbf{c})$. The complexity (i.e., the number of Markov moves t) of solving CVP by MCMC is*

$$C_{\text{CVP}}\triangleq\frac{t_{mix}}{D_{\Lambda,\sigma,\mathbf{c}}(\widehat{\mathbf{x}})}. \tag{36}$$

According to (35) and (36), the decoding complexity of the proposed sliced sampling for CVP can be upper bounded by

$$\begin{aligned}C_{\text{slice}}&<\log\left(\frac{1}{\epsilon}\right)\cdot\frac{1}{\delta'}\cdot\frac{\rho_{\sigma,\mathbf{c}}(\mathbf{\Lambda})}{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\widehat{\mathbf{x}})}\\&\leq\log\left(\frac{1}{\epsilon}\right)\cdot\frac{1}{\beta}\cdot\frac{\prod_{i=1}^n\rho_{\sigma_i}(\mathbb{Z})}{\rho_{\sigma,\mathbf{c}}(\mathbf{\Lambda})}\cdot\frac{\rho_{\sigma,\mathbf{c}}(\mathbf{\Lambda})}{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\widehat{\mathbf{x}})}\\&=\log\left(\frac{1}{\epsilon}\right)\cdot\frac{1}{\beta}\cdot\frac{\prod_{i=1}^n\rho_{\sigma_i}(\mathbb{Z})}{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\widehat{\mathbf{x}})}\\&=\log\left(\frac{1}{\epsilon}\right)\cdot\frac{1}{\beta}\cdot C(\sigma),\end{aligned} \tag{37}$$

where

$$C(\sigma)\triangleq\frac{\prod_{i=1}^n\rho_{\sigma_i}(\mathbb{Z})}{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\widehat{\mathbf{x}})}=\left(\prod_{i=1}^n\sum_{x_i\in\mathbb{Z}}e^{-\frac{\|\tilde{\mathbf{b}}_i\|^2}{2\sigma^2}\cdot x_i^2}\right)\cdot e^{\frac{d^2(\Lambda,\mathbf{c})}{2\sigma^2}}. \tag{38}$$

Clearly, given $\mathbf{B}$ and $d(\Lambda,\mathbf{c})$, the decoding complexity is determined by the choice of σ. Note that since slice sampling achieves a better mixing time than IMHK sampling, its decoding complexity is also superior to that of IMHK sampling for a better decoding performance [36]. Based on (38), further analysis is given to optimize the choice of the standard deviation σ in what follows, thus leading to a better decoding performance.

A. Optimization of σ

In [36], the choice of $\sigma_A=\min_i\|\widehat{\mathbf{b}}_i\|/2\sqrt{\pi}$ is proposed as a suboptimal choice for solving CVP by IMHK sampling decoding. By simply substituting it into (38) for the proposed sliced sampling, we have

$$C(\sigma_A)=e^{\frac{2\pi}{\min^2\|\tilde{\mathbf{b}}_i\|}\cdot d^2(\Lambda,\mathbf{c})}\cdot\prod_{i=1}^n\vartheta_3\left(\frac{2\|\widehat{\mathbf{b}}_i\|^2}{\min^2\|\widehat{\mathbf{b}}_i\|}\right) \tag{39}$$

with *Jacobi theta* function $\vartheta_3(\tau)=\sum_{n=-\infty}^{+\infty}e^{-\pi\tau n^2}$. Since $\vartheta_3(\tau)$ is monotonically decreasing with $\tau>0$, it is shown that

$$C(\sigma_A)\leq e^{\frac{2\pi}{\min_i\|\tilde{\mathbf{b}}_i\|^2}\cdot d^2(\Lambda,\mathbf{c})}\cdot\vartheta_3^n(2)=1.0039^n\cdot e^{\frac{2\pi}{\min_i\|\tilde{\mathbf{b}}_i\|^2}\cdot d^2(\Lambda,\mathbf{c})}. \tag{40}$$

Nevertheless, $C(\sigma_A)$ is sensitive with $d(\Lambda,\mathbf{c})$ due to the exponentially increasing component $e^{d^2(\Lambda,\mathbf{c})}$.

In order to obtain a better σ for solving CVP, we start with considering the optimal choice of σ with respect to

$$C'(\sigma)\triangleq\left(\prod_{i=1}^n\int_{-\infty}^{\infty}e^{-\frac{\|\tilde{\mathbf{b}}_i\|^2}{2\sigma^2}\cdot x_i^2}dx_i\right)\cdot e^{\frac{d^2(\Lambda,\mathbf{c})}{2\sigma^2}}, \tag{41}$$

which is a continuous version of (38). According to the fact

TABLE I
VALUES OF ϑ_3.

$\vartheta_3(1)$	$\sum_{n=-\infty}^{+\infty}e^{-1\pi n^2}$	$\frac{\sqrt[4]{\pi}}{\Gamma(\frac{3}{4})}$	1.087
$\vartheta_3(2)$	$\sum_{n=-\infty}^{+\infty}e^{-2\pi n^2}$	$\frac{\sqrt[4]{6\pi+4\sqrt{2}\pi}}{2\Gamma(\frac{3}{4})}$	1.0039
$\vartheta_3(3)$	$\sum_{n=-\infty}^{+\infty}e^{-3\pi n^2}$	$\frac{\sqrt[4]{27\pi+18\sqrt{3}\pi}}{3\Gamma(\frac{3}{4})}$	1.00037
$\vartheta_3(4)$	$\sum_{n=-\infty}^{+\infty}e^{-4\pi n^2}$	$\frac{\sqrt[4]{8\pi}+2\sqrt[4]{\pi}}{4\Gamma(\frac{3}{4})}$	1.0002
$\vartheta_3(5)$	$\sum_{n=-\infty}^{+\infty}e^{-5\pi n^2}$	$\frac{\sqrt[4]{225\pi+100\sqrt{5}\pi}}{5\Gamma(\frac{3}{4})}$	1.0001

$\int_{-\infty}^{\infty}e^{-\frac{x^2}{2\sigma^2}}dx=\sqrt{2\pi}\sigma$, it follows that

$$C'(\sigma)=\frac{(2\pi)^{\frac{n}{2}}}{|\det(\mathbf{B})|}\cdot\sigma^n\cdot e^{\frac{d^2(\Lambda,\mathbf{c})}{2\sigma^2}} \tag{42}$$

and the derivative of function $C'(\sigma)$ with respect to σ can be easily obtained. Furthermore, by letting

$$\frac{\partial C'(\sigma)}{\partial\sigma}=0, \tag{43}$$

then we have

$$\sigma=\frac{d(\Lambda,\mathbf{c})}{\sqrt{n}}. \tag{44}$$

Here, for notational simplicity, we apply $\sigma_B=d(\Lambda,\mathbf{c})/\sqrt{n}$ as the choice for $C(\sigma)$ shown in (38), and it follows that

$$\begin{aligned}C(\sigma_B)&=e^{\frac{n}{2}}\cdot\left(\prod_{i=1}^n\sum_{x_i\in\mathbb{Z}}e^{-\frac{n\|\tilde{\mathbf{b}}_i\|^2}{2d^2(\Lambda,\mathbf{c})}\cdot x_i^2}\right)\\&=e^{\frac{n}{2}}\cdot\prod_{i=1}^n\vartheta_3\left(\frac{n\|\widehat{\mathbf{b}}_i\|^2}{2\pi d^2(\Lambda,\mathbf{c})}\right).\end{aligned} \tag{45}$$

Clearly, the choice $\sigma_B=d(\Lambda,\mathbf{c})/\sqrt{n}$ is still suboptimal because it was found through the continuous case. Nevertheless, significant potential still can be obtained. More precisely, according to $\int_{-\infty}^{\infty}e^{-\frac{x^2}{2\sigma^2}}dx=\sqrt{2\pi}\sigma$, $C(\sigma_B)$ is upper bounded by

$$\begin{aligned}C(\sigma_B)&\leq e^{\frac{n}{2}}\cdot\left(\prod_{i=1}^n\int_{-\infty}^{\infty}e^{-\frac{n\|\tilde{\mathbf{b}}_i\|^2}{2d^2(\Lambda,\mathbf{c})}\cdot x_i^2}dx_i\right)\\&=e^{\frac{n}{2}}\cdot\prod_{i=1}^n\left(\sqrt{2\pi}\cdot\frac{d(\Lambda,\mathbf{c})}{\sqrt{n}\|\widehat{\mathbf{b}}_i\|}\right)\\&=\left(\frac{2\pi e}{n}\right)^{\frac{n}{2}}\cdot\frac{d^n(\Lambda,\mathbf{c})}{|\det(\mathbf{B})|}.\end{aligned} \tag{46}$$

Intuitively, when $n>2\pi e$ (i.e., $n\geq 18$), $C(\sigma_B)$ is mainly dominated by the relationship between $d(\Lambda,\mathbf{c})$ and $|\det(\mathbf{B})|$. More precisely, according to (46), the complexity of $O(e^{\frac{n}{2}})$ for solving CVP can be achieved by $C(\sigma_B)$ if

$$d(\Lambda,\mathbf{c})\leq\sqrt{\frac{n}{2\pi}}\cdot|\det(\mathbf{B})|^{\frac{1}{n}}. \tag{47}$$

Furthermore, by substituting (46) into (37), the complexity of solving CVP via the proposed sliced sampling decoding

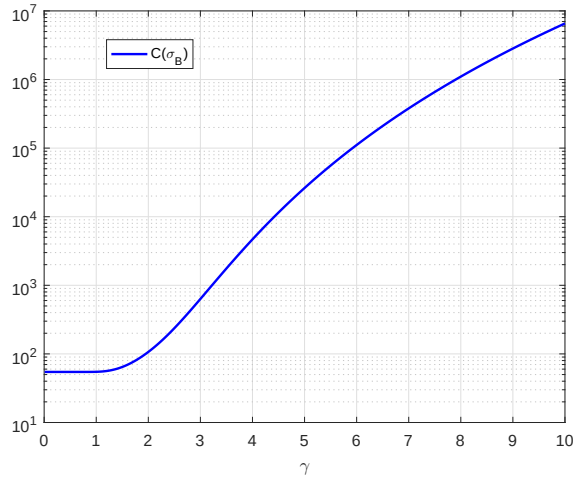

Fig. 2. $C(\sigma_{\mathrm{B}})$ versus $d(\Lambda,\mathbf{c}) = \gamma \cdot \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$ for the lattice basis $\mathbf{B} \in \mathbb{R}^{8\times 8}$ with Gaussian coefficients.

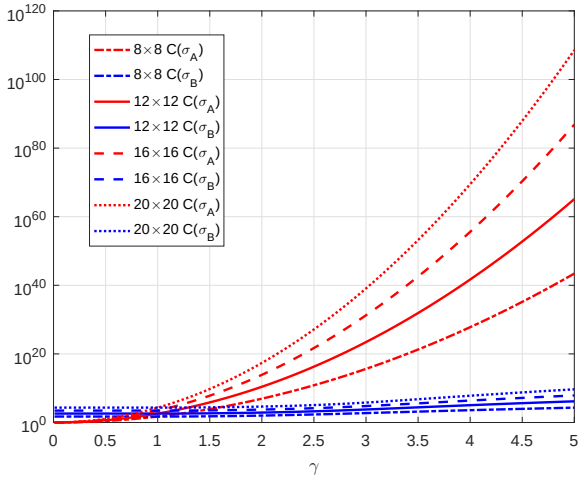

Fig. 3. $C(\sigma)$ versus $d(\Lambda,\mathbf{c}) = \gamma \cdot \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$ for 8×8, 12×12, 16×16 and 20×20 lattice basis $\mathbf{B}$ with Gaussian coefficients.

can be derived as

$$C_{\mathrm{CVP}} \leq \log\left(\frac{1}{\epsilon}\right)\cdot\left(\frac{2\pi e}{n}\right)^{\frac{n}{2}}\cdot\frac{d^n}{|\det(\mathbf{B})|} \tag{48}$$

and its decoding radius follows that

$$d_{\sigma_{\mathrm{B}}}(\Lambda,\mathbf{c}) = \sqrt{\frac{n}{2\pi e}}\cdot\left(\frac{C_{\mathrm{CVP}}}{\log(\frac{1}{\epsilon})}\right)^{\frac{1}{n}}\cdot|\det(\mathbf{B})|^{\frac{1}{n}}, \tag{49}$$

which is comparable to the one based on σ_A as

$$d_{\sigma_{\mathrm{A}}}(\Lambda,\mathbf{c}) = \sqrt{\frac{1}{2\pi}\cdot\ln\frac{C_{\mathrm{CVP}}}{\log\left(\frac{1}{\epsilon}\right)}}\cdot\min_{1\leq i\leq n}\|\widehat{\mathbf{b}}_i\|. \tag{50}$$

B. Relationship between σ_A and σ_B

In order to investigate the relationship between σ_A and σ_B, we arrive at the following Proposition, whose proof is omitted due to simplicity. From it, the following analysis can be carried out thereafter.

Proposition 1. *Given $C(\sigma_A)$ and $C(\sigma_B)$ in (39) and (45) respectively, it follows that*

1) *If $d(\Lambda,\mathbf{c}) = \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, the choices σ_A and σ_B are equivalent due to the same value of $C(\sigma)$.*
2) *If $d(\Lambda,\mathbf{c}) < \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, then the choice σ_A is better than σ_B due to a smaller $C(\sigma)$.*
3) *If $d(\Lambda,\mathbf{c}) > \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, then the choice σ_B is better than σ_A due to a smaller $C(\sigma)$.*

Insight into $C(\sigma_A)$ and $C(\sigma_B)$, both of them increase with the improvement of $d(\Lambda,\mathbf{c})$. In particular, due to the constant term $e^{n/2}$, $C(\sigma_B)$ is less efficient than $C(\sigma_A)$ when $d(\Lambda,\mathbf{c})$ is small. However, given an increasing $d(\Lambda,\mathbf{c})$, the product term $\prod_{i=1}^{n}\vartheta_3\left(\frac{n\|\widehat{\mathbf{b}}_i\|^2}{2\pi d^2(\Lambda,\mathbf{c})}\right)$ in $C(\sigma_B)$ is not as sensitive as the term $e^{\frac{2\pi}{\min_i\|\widehat{\mathbf{b}}_i\|^2}\cdot d^2(\Lambda,\mathbf{c})}$ in $C(\sigma_A)$, thus leading to a significant superiority in decoding complexity for $d(\Lambda,\mathbf{c}) > \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$. To summarize, in order to achieve a better decoding performance, σ should obey the following choice

$$\sigma = \begin{cases}\sigma_A = \min_i\|\widehat{\mathbf{b}}_i\|/2\sqrt{\pi} & \text{if } d(\Lambda,\mathbf{c}) \leq \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}};\\ \sigma_B = d(\Lambda,\mathbf{c})/\sqrt{n} & \text{if } d(\Lambda,\mathbf{c}) > \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}.\end{cases} \tag{51}$$

For a better understanding, the average value of $C(\sigma_B)$ versus $d(\Lambda,\mathbf{c}) = \gamma\cdot\frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, $\gamma > 0$ for an 8×8 lattice basis $\mathbf{B}$ with Gaussian coefficients is illustrated by Monte Carlo methods in Fig. 2. Intuitively, when $d(\Lambda,\mathbf{c}) > \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$ (i.e., $\gamma > 1$), the average value of $C(\sigma_B)$ grows rapidly. Moreover, the comparisons between $C(\sigma_A)$ and $C(\sigma_B)$ with respect to various lattice basis $\mathbf{B}$ with Gaussian coefficients are further presented in Fig. 3. As expected, $C(\sigma_A)$ and $C(\sigma_B)$ are same when $d(\Lambda,\mathbf{c}) = \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, i.e., $\gamma = 1$. Meanwhile, it is clear that $C(\sigma_A) < C(\sigma_B)$ when $d(\Lambda,\mathbf{c}) < \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$ and $C(\sigma_A) > C(\sigma_B)$ for $d(\Lambda,\mathbf{c}) > \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$. Most importantly, compared to $C(\sigma_A)$, the increment of $C(\sigma_B)$ is much milder, thus making $C(\sigma_B)$ a promising choice for $d(\Lambda,\mathbf{c}) > \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$.

Another point should be emphasized is the application of lattice reduction technique. It is well known that after Lenstra-Lenstra-Lovász (LLL) reduction, vectors in the matrix $\mathbf{B}$ (i.e., lattice basis) become relatively short and orthogonal to each other. Meanwhile, LLL reduction is able to significantly improve $\min_i\|\widehat{\mathbf{b}}_i\|$ while reduce $\max_i\|\widehat{\mathbf{b}}_i\|$ [55]. To this end, LLL reduction is encouraged to serve as a preprocessing stage with polynomial computational complexity $O(n^3\log n)$ since it could significantly improve the decoding performance of the choice σ_A [56]. Note that as shown in (45) and (49), LLL reduction does not alter the volume of lattice, which corresponds to a constant $|\det(\mathbf{B})|$. In this condition, since $C(\sigma_A)$ is better than $C(\sigma_B)$ when $d(\Lambda,\mathbf{c}) < \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, the usage of LLL reduction also expands the active range of the choice σ_A.

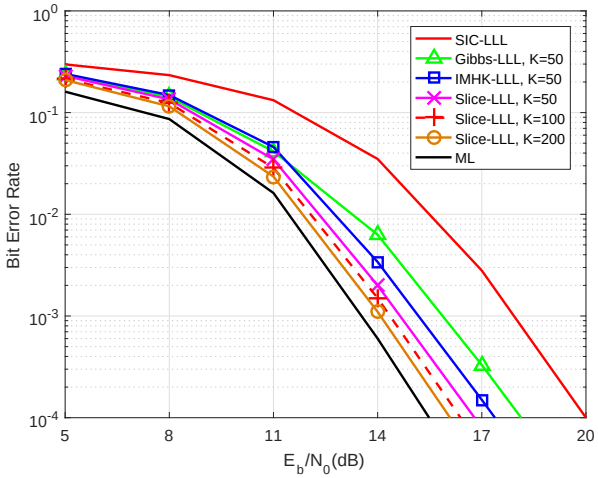

Fig. 4. Bit error rate versus average SNR per bit for the uncoded 8×8 MIMO system using 16-QAM.

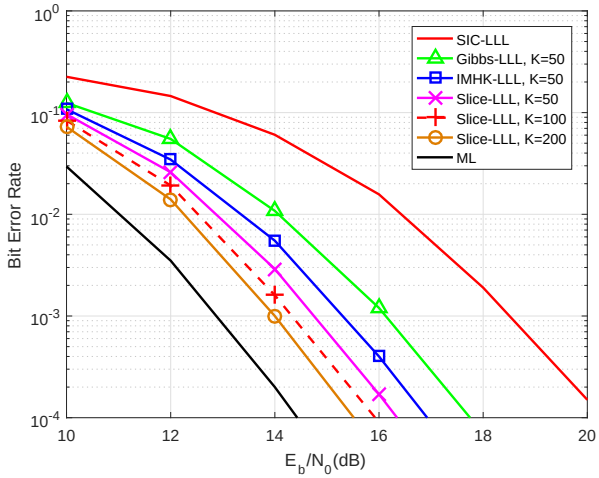

Fig. 5. Bit error rate versus average SNR per bit for the uncoded 12×12 MIMO system using 16-QAM.

C. Dynamic Update of σ

However, given the choice of σ in (51), two natural questions raise, which should be considered carefully:

- A judgement based on $d(\Lambda, \mathbf{c})$ is required to determine the choice of σ.
- The premise $d(\Lambda, \mathbf{c})$ corresponds to solving the CVP, which is difficult to obtain at the very beginning.

Therefore, we now try to answer these two questions in what follows.

Specifically, as $d(\Lambda, \mathbf{c})$ is hard to get, the initial distance $d_{\text{initial}}(\Lambda, \mathbf{c}) = \|\mathbf{B}\mathbf{x}_{\text{sic-lll}} - \mathbf{c}\|$ is applied as an approximation, where $\mathbf{x}_{\text{sic-lll}}$ is the decoding result of SIC-LLL decoding[2]. Note that other decoding results $\widehat{\mathbf{x}}$ of sub-optimal decoding schemes can also be applied, and the more accurate $d_{\text{initial}}(\Lambda, \mathbf{c})$ to $d(\Lambda, \mathbf{c})$, the better decoding performance. Meanwhile, the result of $\mathbf{x}_{\text{sic-lll}}$ can also be applied as the initial starting state of the Markov chain, which is helpful to the Markov mixing [57].

Then, given the fact $d(\Lambda, \mathbf{c}) \leq d_{\text{initial}}(\Lambda, \mathbf{c})$, the related judgement can be carried out to determine the choice of σ. Clearly, if $d_{\text{initial}}(\Lambda, \mathbf{c}) \leq \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, then σ_A is selected as a judicious choice. Otherwise, $\sigma = d_{\text{initial}}(\Lambda, \mathbf{c})/\sqrt{n}$ is applied at the beginning, and σ is updated dynamically by learning from the collected samples as

$$\sigma_{\text{dynamic}} = \frac{d_{\text{update}}(\Lambda, \mathbf{c})}{\sqrt{n}} \triangleq \frac{\min_{\mathbf{x}\in S}\|\mathbf{B}\mathbf{x} - \mathbf{c}\|}{\sqrt{n}}, \tag{52}$$

where set S contains all the samples of $\mathbf{x}$ collected by the sampling. Undoubtedly, along with the sampling, $d_{\text{update}}(\Lambda, \mathbf{c})$ shrinks gradually, thus leading to a more accurate estimation of σ to σ_B.

Proposition 2. *Given $d(\Lambda, \mathbf{c}) > \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, the choice $\sigma = \frac{d_{\text{initial}}(\Lambda,\mathbf{c})}{\sqrt{n}}$ is better than σ_A due to a smaller $C(\sigma)$ if*

$$d_{\text{initial}}(\Lambda, \mathbf{c}) \leq |\det(\mathbf{B})|^{\frac{1}{n}} \cdot \sqrt{\frac{n}{2\pi e}} \cdot e^{\frac{2\pi\alpha^2}{n}}. \tag{53}$$

Proof: First of all, by substituting $\sigma_{\text{initial}} = \frac{d_{\text{initial}}(\Lambda,\mathbf{c})}{\sqrt{n}}$ into (46), we have

$$C(\sigma_{\text{initial}}) \leq \left(\frac{2\pi e}{n}\right)^{\frac{n}{2}} \cdot \frac{d^n_{\text{initial}}(\Lambda, \mathbf{c})}{|\det(\mathbf{B})|}. \tag{54}$$

Then, given (39), in order to make sure $C(\sigma_{\text{initial}}) < C(\sigma_A)$, it follows that

$$\left(\frac{2\pi e}{n}\right)^{\frac{n}{2}} \cdot \frac{d^n_{\text{initial}}(\Lambda,\mathbf{c})}{|\det(\mathbf{B})|} < e^{\frac{2\pi}{\min^2\|\widehat{\mathbf{b}}_i\|}\cdot d^2(\Lambda,\mathbf{c})} \cdot \prod_{i=1}^{n} \vartheta_3\left(\frac{2\|\widehat{\mathbf{b}}_i\|^2}{\min^2\|\widehat{\mathbf{b}}_i\|}\right) \tag{55}$$

so as to

$$\begin{aligned} d_{\text{initial}}(\Lambda, \mathbf{c}) &< |\det(\mathbf{B})|^{\frac{1}{n}} \cdot \sqrt{\frac{n}{2\pi e}} \cdot e^{\frac{2\pi\cdot d^2(\Lambda,\mathbf{c})}{n\cdot\min^2\|\widehat{\mathbf{b}}_i\|}} \cdot \prod_{i=1}^{n} \vartheta_3\left(\frac{2\|\widehat{\mathbf{b}}_i\|^2}{\min^2\|\widehat{\mathbf{b}}_i\|}\right) \\ &\leq |\det(\mathbf{B})|^{\frac{1}{n}} \cdot \sqrt{\frac{n}{2\pi e}} \cdot e^{\frac{2\pi\cdot d^2(\Lambda,\mathbf{c})}{n\cdot\min^2\|\widehat{\mathbf{b}}_i\|}} \cdot \vartheta_3^n(2) \\ &\approx |\det(\mathbf{B})|^{\frac{1}{n}} \cdot \sqrt{\frac{n}{2\pi e}} \cdot e^{\frac{2\pi\cdot d^2(\Lambda,\mathbf{c})}{n\cdot\min^2\|\widehat{\mathbf{b}}_i\|}}, \end{aligned} \tag{56}$$

completing the proof. ■

From Proposition 2, the choice of $\sigma = \frac{d_{\text{initial}}(\Lambda,\mathbf{c})}{\sqrt{n}}$ is superior to σ_A when d_{initial} is close to $d(\Lambda, \mathbf{c}) > \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$ within a certain level. Nevertheless, as the initial distance $d_{\text{initial}}(\Lambda, \mathbf{c})$ may be quite far away from $d(\Lambda, \mathbf{c})$ due to a poor suboptimal detection, the estimation of σ in (52) has the risk to be excessively large. To prevent such a problem, it is necessary to set an upper bound for σ as

$$\sigma_{\text{dynamic}} = \min\left\{\frac{d_{\text{update}}(\Lambda, \mathbf{c})}{\sqrt{n}}, \gamma \cdot \frac{\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}\right\}, \tag{57}$$

where $d(\Lambda, \mathbf{c}) = \gamma \cdot \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$ and the coefficient γ is suggested to choose from the range $[1.2, 1.4]$. We emphasize

[2]The successive interference cancelation (SIC) decoding is also known as Babai's nearest plane algorithm in lattice decoding.

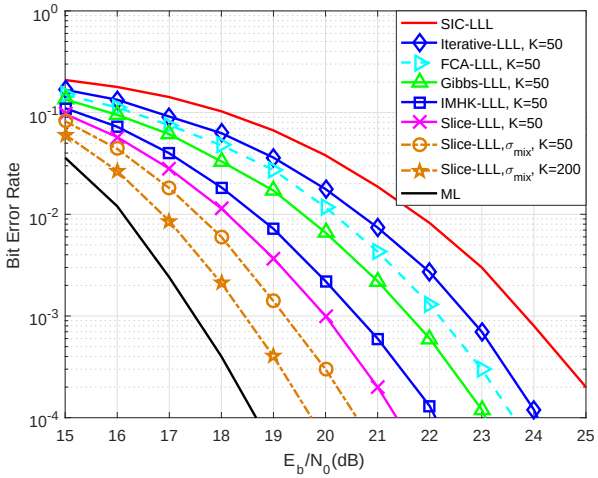

Fig. 6. Bit error rate versus average SNR per bit for the uncoded 16×16 MIMO system using 64-QAM.

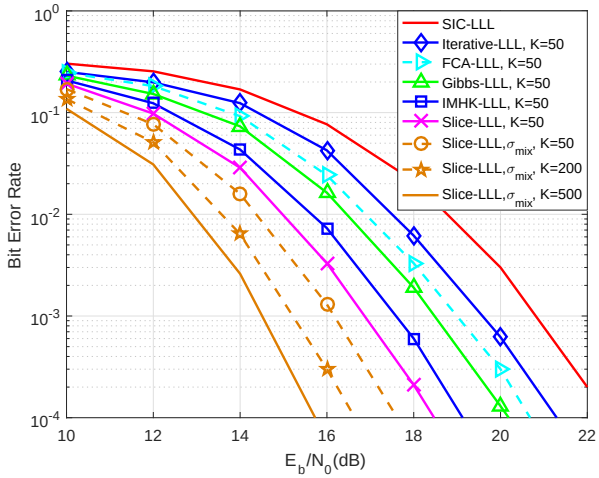

Fig. 7. Bit error rate versus average SNR per bit for the uncoded 24×24 MIMO system using 16-QAM.

the significance of the above upper bound, which is important especially for decoding cases with limited state space, i.e., $\mathbf{x} \in \mathcal{X}^n \subseteq \mathbb{Z}^n$.

To summarize, the proposed choice of standard deviation σ_{mix} is given as follows, where the judgement is carried out based on $d_{\text{initial}}(\Lambda, \mathbf{c})$.

Remark 1.
- *If* $d_{initial}(\Lambda, \mathbf{c}) \leq \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, *let* $\sigma_{mix} = \sigma_A$ *for the sampling decoding.*
- *If* $d_{initial}(\Lambda, \mathbf{c}) > \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, *update* $\sigma_{mix} = \sigma_{dynamic}$ *dynamically for the sampling decoding.*

Note that updating σ by learning dynamically is compatible with the mechanism of MCMC, which is known as adaptive MCMC (see [58], [59] for more details). Meanwhile, this is also in line with the concept of simulated annealing (SA) by gradually cooling down the temperature of the Markov chain, which is widely applied in various research fields [60].

VI. Simulations

In this section, the performances of MCMC-based sampling schemes are exemplified in the context of MIMO detection, whose system model can be expressed as

$$\mathbf{c} = \mathbf{H}\mathbf{x} + \mathbf{w}. \tag{58}$$

Here, the ith entry of the transmitted signal $\mathbf{x}$, denoted as x_i, is a modulation symbol taken independently from an M-QAM constellation $\mathcal{X}$ with Gray mapping. The channel matrix $\mathbf{H}$ contains uncorrelated complex Gaussian fading gains with unit variance and remains constant over each frame duration and $\mathbf{w}$ is the Gaussian noise with zero mean and variance σ_w^2. Intuitively, this decoding problem of $\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}\in\mathcal{X}^n} \|\mathbf{H}\mathbf{x} - \mathbf{c}\|^2$ can be solved by sampling over the discrete Gaussian distribution

$$P_{\mathcal{L}(\mathbf{H}),\sigma,\mathbf{c}}(\mathbf{x}) = \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{H}\mathbf{x}-\mathbf{c}\|^2}}{\sum_{\mathbf{x}\in\mathcal{X}^n} e^{-\frac{1}{2\sigma^2}\|\mathbf{H}\mathbf{x}-\mathbf{c}\|^2}}, \tag{59}$$

and the closest lattice point $\mathbf{H}\mathbf{x}$ will be returned with the highest probability.

In Fig. 4, the bit error rates (BERs) of MCMC sampling detectors are evaluated against the number of Markov moves (i.e., iterations) in a 8×8 uncoded MIMO system with 16-QAM. Here, LLL reduction-aided SIC decoding serves as a performance baseline for a better comparison. Meanwhile, LLL reduction is also applied to other decoding schemes as a fair comparison, where the trade-off coefficient $1/4 < \eta < 1$ in Lovász condition is set as 0.99 for a relatively orthogonal lattice basis. Clearly, there is a substantial performance gap between lattice reduction-aided decoding scheme and sampling decoding schemes. In particular, with the standard deviation $\sigma_A = \min_i \|\widehat{\mathbf{b}}_i\|/(2\sqrt{\pi})$, the proposed sliced lattice Gaussian sampling algorithm achieves a better decoding performance than IMHK under the same number of Markov moves (i.e., $K = 50$). On the other hand, with the increase of Markov moves, the decoding performance improves gradually due to a larger decoding radius. As expected, near-optimal decoding performance can be obtained when $K = 200$. In addition, Gibbs sampling from MCMC is also added for the comparison, whose setting of standard deviation σ_{distance} comes from [61]. However, as LLL can not be perfectly adopted to Gibbs sampling, considerable performance loss turns out to be inevitable compared to sliced or IMHK samplings.

In Fig. 5, the BERs of MCMC sampling decoding schemes are evaluated against the number of Markov moves in a 12×12 uncoded MIMO system with 16-QAM. Clearly, the proposed sliced lattice Gaussian sampling algorithm is still superior to IMHK sampling in terms of decoding performance, thus implying a better convergence rate. Note that with the increase of the system dimension, the performance gap between ML and sampling decoding schemes is enlarged, and more complexity cost will be consumed. Therefore, to achieve the near-optimal decoding performance, a larger number of Markov moves is required. In other words, the proposed sliced lattice Gaussian sampling is flexible, where its decoding trade-off between performance and complexity can be adjusted through

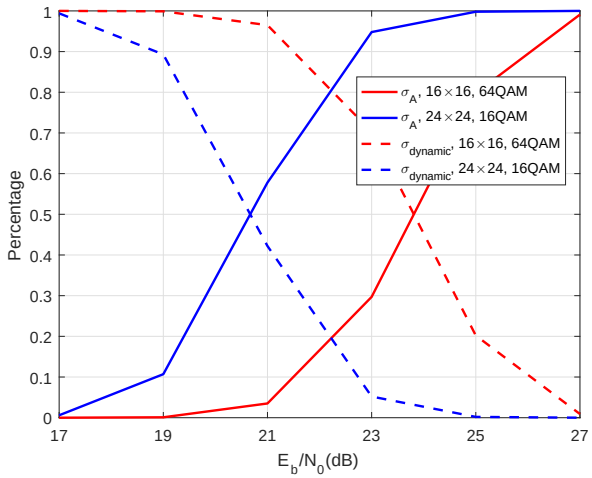

Fig. 8. Choice percentage of σ_{mix} versus average SNR per bit for the uncoded 16×16 MIMO system with 64-QAM and 24×24 MIMO system with 16-QAM respectively.

the number of Markov moves.

In order to show the performance comparison with different choices of the standard deviations, Fig. 6 is given to illustrate the BER performance of σ_A and σ_{dynamic} in a 16×16 uncoded MIMO system with 64-QAM. As shown in Proposition 1, σ_A is only advantageous in a limited range with $d(\Lambda, \mathbf{c}) < \frac{\sqrt{n} \min \|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$, which means considerable decoding potential can be further exploited. For this reason, σ_{dynamic} shown in (57) is given, and it takes the advantages of the initial starting state (i.e., $\mathbf{x}_0 = \mathbf{x}_{\text{sic-lll}}$) of the underlying Markov chain, thereby leading to the mixed version σ_{mix} based on the judgement of $d_{\text{initial}}(\Lambda, \mathbf{c})$. More specifically, the coefficient γ we choose in $d_{\text{initial}}(\Lambda, \mathbf{c})$ is 1.3. In Fig. 6, as expected, compared to the sliced sampling with σ_A, considerable decoding performance can be obtained by the sliced sampling with σ_{mix} under the same number of Markov moves $K = 50$. In particular, the gain of the choice σ_{mix} with $K = 50$ is approximately 1 dB for a BER of 10^{-4}, which can be further improved with the increase of the Markov moves. Additionally, the decoding performance of the fixed candidates algorithm (FCA) in [62] and iterative list decoding in [63] with 50 samples are also presented as a comparison.

In Fig. 7, the performance comparison with different choices of the standard deviations is presented to show the BER performance in a 24×24 uncoded MIMO system with 16-QAM. This corresponds to a lattice decoding scenario with restricted state space in dimension $n = 48$. It is clear that under the help of LLL reduction, all the decoding schemes are able to achieve the full receive diversity. However, with the increase of system dimension, more number of Markov moves is needed for approaching the ML performance. Nevertheless, the decoding gain of the choice σ_{mix} over σ_A still can be observed, which is further improved with the increase of Markov moves. Note that as a mixed strategy for choosing σ, σ_{mix} selects σ_A or σ_{dynamic} according to the judgement based on $d_{\text{initial}}(\Lambda, \mathbf{c})$, where the related details will be explicitly described in the following.

To further study the choice of σ_{mix} between σ_A and σ_{dynamic}, Fig. 8 is given to show the choice percentage of σ_{mix} in two different decoding cases, namely, 16×16 uncoded MIMO system with 64-QAM and 24×24 uncoded MIMO system with 16-QAM. Both the numbers of Markov moves of the sliced sampling algorithm in these two cases are set as $K = 50$ with coefficient $\gamma = 1.3$. Clearly, in the low SNR region, the choice percentages of σ_A for σ_{mix} are rather limited. This is because $d_{\text{initial}}(\Lambda, \mathbf{c})$ normally turns out to be relatively large due to the effect of noises. With the increase of SNR, the effects of noises are constrained gradually, thus resulting in a smaller size of $d_{\text{initial}}(\Lambda, \mathbf{c})$. In this case, the choice of σ_A becomes a better choice than σ_{dynamic} and its percentage of being selected goes up subsequently. On the contrary, the choice percentages of σ_{dynamic} decrease with the increase of SNR. When the SNR per bit is larger than 27dB, the percentage of choosing σ_{dynamic} is approaching to 0. Nevertheless, it still plays a dominant role especially in the low SNR regions. For this reason, the decoding performance gain of σ_{dynamic} over σ_A is reliable in most cases of MIMO detection. Another thing should be pointed out is that the modulation scheme also has an impact upon the choice of σ since the high order modulation suffers from the noises more severely in average.

VII. Conclusion

In this paper, the sliced lattice Gaussian sampling algorithm was proposed to sample from the lattice Gaussian distribution. By introducing an auxiliary random variable, the underlying Markov chain of the proposed sliced sampling not only achieves uniformly ergodicity, but also shows a better mixing performance than that of the independent Metropolis-Hastings-Klein (IMHK) sampling algorithm. On the other hand, with respect to lattice decoding by the sliced lattice Gaussian sampling algorithm, comprehensive analysis is carried out while a better choice of the standard deviation σ is derived for certain cases. To further exploit the decoding potential, a judicious judgement based on the Euclidean distance is proposed for a better choice of σ. By doing this, the proposed sliced lattice Gaussian sampling algorithms suits well for the various decoding requirements, where the decoding trade-off between the performance and complexity is flexibly adjusted through the number of Markov moves.

References

[1] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, pp. 3590–3600, November 2010.

[2] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186–195, February 2014.

[3] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 40–60, Jan 2013.

[4] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, "Energy and spectral efficiency of very large multiuser MIMO systems," *IEEE Transactions on Communications*, vol. 61, no. 4, pp. 1436–1449, April 2013.

[5] L. Babai, "On Lovász' lattice reduction and the nearest lattice point problem," *Combinatorica*, vol. 6, no. 1, pp. 1–13, 1986.

[6] H. Yao and G. Wornell, "Lattice-reduction-aided detectors for MIMO communication systems," in *Proc. IEEE Globecom*, Taipei, China, Nov. 2002, pp. 424–428.

[7] M. Taherzadeh, A. Mobasher, and A. Khandani, "LLL reduction achieves the receive diversity in MIMO decoding," *IEEE Trans. Inform. Theory*, vol. 53, pp. 4801–4805, Dec. 2007.

[8] J. Jalden and P. Elia, "DMT optimality of LR-aided linear decoders for a general class of channels, lattice designs, and system models," *IEEE Trans. Inform. Theory*, vol. 56, no. 10, pp. 4765–4780, Oct. 2010.

[9] C. Ling, "On the proximity factors of lattice reduction-aided decoding," *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2795–2808, Jun. 2011.

[10] R. Kannan, "Improved algorithms for integer programming and related lattice problems," in *Proc ACM Symp. Theory of Computing*, Boston, Apr. 1983, pp. 193–206.

[11] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inform. Theory*, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.

[12] B. Hassibi and H. Vikalo, "On the sphere-decoding algorithm I. Expected complexity," *IEEE Trans. Signal Process.*, vol. 53, pp. 2806–2818, Aug. 2005.

[13] M. O. Damen, H. E. Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2389–2401, Oct. 2003.

[14] J. Jalden and B. Ottersen, "On the complexity of sphere decoding in digital communications," *IEEE Trans. Signal Process.*, vol. 53, pp. 1474–1484, Aug. 2005.

[15] A.D.Murugan, H. E. Gamal, M. Damen, and G.Caire, "A unified framework for tree search decoding: rediscovering the sequential decoder," *IEEE Trans. Inform. Theory*, vol. 52, pp. 933– 953, 2006.

[16] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative decoding for MIMO channels via modified sphere decoding," *IEEE Trans. Wireless Commun.*, vol. 3, no. 6, pp. 2299 – 2311, Nov. 2004.

[17] J. Jalden, L. Barbero, B. Ottersten, and J. Thompson, "The error probability of the fixed-complexity sphere decoder," *IEEE Trans. Signal Process.*, vol. 57, pp. 2711–2720, Jul. 2009.

[18] S. Chen, T. Zhang, and Y. Xin, "Relaxed K-best MIMO signal detector design and VLSI implementation," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 15, no. 3, pp. 328–337, March 2007.

[19] L. Luzzi, D. Stehlé, and C. Ling, "Decoding by embedding: correct decoding radius and DMT optimality," *IEEE Trans. Inform. Theory*, vol. 59, no. 5, pp. 2960–2973, 2013.

[20] N. Srinidhi, T. Datta, A. Chockalingam, and B. S. Rajan, "Layered Tabu search algorithm for large-MIMO detection and a lower bound on ML performance," *IEEE Transactions on Communications*, vol. 59, no. 11, pp. 2955–2963, Nov. 2011.

[21] L. Dai, X. Gao, X. Su, S. Han, C. I, and Z. Wang, "Low-complexity soft-output signal detection based on Gauss-Seidel method for uplink multiuser large-scale MIMO systems," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 10, pp. 4839–4845, Oct. 2015.

[22] A. Lu, X. Gao, Y. R. Zheng, and C. Xiao, "Low complexity polynomial expansion detector with deterministic equivalents of the moments of channel Gram matrix for massive MIMO uplink," *IEEE Transactions on Communications*, vol. 64, no. 2, pp. 586–600, Feb. 2016.

[23] S. Wu, L. Kuang, Z. Ni, J. Lu, D. Huang, and Q. Guo, "Low-complexity iterative detection for large-scale multiuser MIMO-OFDM systems using approximate message passing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 902–915, Oct. 2014.

[24] P. Som, T. Datta, N. Srinidhi, A. Chockalingam, and B. S. Rajan, "Low-complexity detection in large-dimension MIMO-ISI channels using graphical models," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 8, pp. 1497–1511, Dec 2011.

[25] B. Hassibi, M. Hansen, A. Dimakis, H. Alshamary, and W. Xu, "Optimized Markov Chain Monte Carlo for signal detection in MIMO systems: An analysis of the stationary distribution and mixing time," *IEEE Transactions on Signal Processing*, vol. 62, no. 17, pp. 4436–4450, Sep. 2014.

[26] T. Datta, N. Kumar, A. Chockalingam, and B. Rajan, "A novel Monte Carlo sampling based receiver for large-scale uplink multiuser MIMO systems," *IEEE Transactions on Vehicular Technology,*, vol. 62, no. 7, pp. 3019–3038, Sep. 2013.

[27] S. Liu, C. Ling, and D. Stehlé, "Decoding by sampling: A randomized lattice algorithm for bounded distance decoding," *IEEE Trans. Inform. Theory*, vol. 57, pp. 5933–5945, Sep. 2011.

[28] Z. Wang, S. Liu, and C. Ling, "Decoding by sampling - Part II: Derandomization and soft-output decoding," *IEEE Trans. Commun.*, vol. 61, no. 11, pp. 4630–4639, Nov. 2013.

[29] P. Klein, "Finding the closest lattice vector when it is unusually close," in *ACM-SIAM Symp. Discr. Algorithms*, 2000, pp. 937–941.

[30] Z. Wang, C. Ling, and G. Hanrot, "Markov chain Monte Carlo algorithms for lattice Gaussian sampling," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Honolulu, USA, Jun. 2014, pp. 1489–1493.

[31] Z. Wang and C. Ling, "On the geometric ergodicity of gibbs algorithm for lattice gaussian sampling," in *2017 IEEE Information Theory Workshop (ITW)*, Nov 2017, pp. 269–273.

[32] P. Aggarwal and X. Wang, "Multilevel sequential Monte Carlo algorithms for MIMO demodulation," *IEEE Transactions on Wireless Communications*, vol. 6, no. 2, pp. 750–758, Feb. 2007.

[33] H. Zhu, B. Farhang-Boroujeny, and R.-R. Chen, "On performance of sphere decoding and Markov chain Monte Carlo detection methods," *IEEE Signal Processing Letters*, vol. 12, no. 10, pp. 669–672, 2005.

[34] J. Choi, "An MCMC-MIMO detector as a stochastic linear system solver using successive overrelexation," *IEEE Transactions on Wireless Communications*, vol. 15, no. 2, pp. 1445–1455, Feb. 2016.

[35] Z. Wang and C. Ling, "On the geometric ergodicity of Metropolis-Hastings algorithms for lattice Gaussian sampling," *IEEE Transactions on Information Theory*, vol. 64, no. 2, pp. 738–751, Feb. 2018.

[36] Z. Wang and C. Ling, "Lattice Gaussian sampling by Markov chain Monte Carlo: Bounded distance decoding and trapdoor sampling," *IEEE Transactions on Information Theory*, vol. 65, no. 6, pp. 3630–3645, June 2019.

[37] W. Banaszczyk, "New bounds in some transference theorems in the geometry of numbers," *Math. Ann.*, vol. 296, pp. 625–635, 1993.

[38] G. Forney and L.-F. Wei, "Multidimensional constellations–Part II: Voronoi constellations," *IEEE J. Sel. Areas Commun.*, vol. 7, no. 6, pp. 941–958, Aug. 1989.

[39] C. Ling and J.-C. Belfiore, "Achieiving the AWGN channel capacity with lattice Gaussian coding," *IEEE Trans. Inform. Theory*, vol. 60, no. 10, pp. 5918–5929, Oct. 2014.

[40] C. Ling, L. Luzzi, J.-C. Belfiore, and D. Stehlé, "Semantically secure lattice codes for the Gaussian wiretap channel," *IEEE Trans. Inform. Theory*, vol. 60, no. 10, pp. 6399–6416, Oct. 2014.

[41] D. Micciancio and O. Regev, "Worst-case to average-case reductions based on Gaussian measures," in *Proc. Ann. Symp. Found. Computer Science*, Rome, Italy, Oct. 2004, pp. 372–381.

[42] C. Gentry, "Fully homomorphic encryption using ideal lattices," *STOC*, pp. 169–178, 2009.

[43] S. Vatedka, N. Kashyap, and A. Thangaraj, "Secure compute-and-forward in a bidirectional relay," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2531–2556, May 2015.

[44] T. Fehenberger, D. Lavery, R. Maher, A. Alvarado, P. Bayvel, and N. Hanik, "Sensitivity gains by mismatched probabilistic shaping for optical communication systems," *IEEE Photonics Technology Letters*, vol. 28, no. 7, pp. 786–789, Apr. 2016.

[45] T. Fehenberger, A. Alvarado, G. Böcherer, and N. Hanik, "On probabilistic shaping of quadrature amplitude modulation for the nonlinear fiber channel," *Journal of Lightwave Technology*, vol. 34, no. 21, pp. 5063–5073, July 2016.

[46] N. Stephens-Davidowitz, "Discrete Gaussian sampling reduces to CVP and SVP," submitted for publication. [Online]. Available: http://arxiv.org/abs/1506.07490.

[47] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proc. 40th Ann. ACM Symp. Theory of Comput.*, Victoria, Canada, 2008, pp. 197–206.

[48] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Time*, American Mathematical Society, 2008.

[49] R. Neal, "Slice sampling," *Ann. Statist.*, vol. 31, pp. 705–767, 2003.

[50] A. Mira and L. Tierney, "Efficiency and convergence properties of slice samplers," *Scandinavian Journal of Statistics*, vol. 29, pp. 1–12, 2002.

[51] C. Dubois, A. Korattikara, M. Welling, and P. Smyth, "Approximate slice sampling for Bayesian posterior inference," in *Proc. 17th International Conference on Artificial Intelligence and Statistics (AISTATS)*, Reykjavik, Iceland, 2014, pp. 185–193.

[52] Y. Zhang, X. Wang, C. Chen, R. Henao, K. Fan, and L. Carin, "Towards unifying Hamiltonian Monte Carlo and slice sampling," in *Proc. 30th Conference on Neural Information Processing Systems (NIPS)*, Barcelona, Spain, 2016, pp. 1741–1749.

[53] J. Besag and P. J. Green, "Spatial statistics and Bayesian computation," in *J. Roy. Statist. Soc. Ser.*, vol. 55, 1993, pp. 25–37.

[54] P. H. Peskun, "Optimal Monte Carlo sampling using Markov chains," *Biometrika*, vol. 60, pp. 607–612, 1973.

[55] A. K. Lenstra, H. W. Lenstra, and L. Lovasz, "Factoring polynomials with rational coefficients," *Math. Annalen*, vol. 261, pp. 515–534, 1982.

[56] C. Ling and N. Howgrave-Graham, "Effective LLL reduction for lattice decoding," in *Proc IEEE Int. Symp. Inform. Theory*, Nice, France, Jun. 2007, pp. 196–200.
[57] J. S. Rosenthal, "Minorization conditions and convergence rates for Markov chain Monte Carlo," *J. Amer. Statist. Assoc.*, vol. 90, pp. 558–566, 1995.
[58] K. Latuszyński, G. O. Roberts, and J. S. Rosenthal, "Adaptive Gibbs samplers and related MCMC methods," *The Annals of Applied Probability*, vol. 23, no. 1, pp. 66–98, 2013.
[59] C. Andrieu and J. Thoms, "On the ergodicity properties of some adaptive MCMC algorithms," *Ann. Appl. Probab.*, vol. 16, pp. 1462–1505, 2006.
[60] C. J. Geyer and E. A. Thompson, "Annealing Markov chain Monte Carlo with applications to ancestral inference," *J. Amer. Statist. Assoc.*, vol. 90, pp. 909–920, 1995.
[61] Z. Wang, Y. Huang, and S. Lyu, "Lattice-Reduction-Aided Gibbs algorithm for lattice Gaussian sampling: Convergence enhancement and decoding optimization," *IEEE Transactions on Signal Processing*, vol. 67, no. 16, pp. 4342–4356, Aug 2019.
[62] W. Zhang and X. Ma, "Low-complexity soft-output decoding with lattice-reduction-aided detectors," *IEEE Trans. Commun.*, vol. 58, no. 9, pp. 2621–2629, Sep. 2010.
[63] T. Shimokawa and T. Fujino, "Iterative lattice reduction aided MMSE list detection in MIMO system," in *Proc. IEEE International Conference on Advanced Technologies for Communications*, Oct. 2008, pp. 50–54.