Sliced Lattice Gaussian Sampling: Convergence Improvement and Decoding Optimization

Zheng Wang, *Member, IEEE*, Ling Liu, *Member, IEEE*, and Cong Ling, *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 $\sigma > 0$ of the target lattice Gaussian distribution 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 the convergence enhancement and the parameter optimization.**

***Index Terms*— Coding, decoding, MCMC methods, MIMO detection, lattice Gaussian sampling.**

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 uplink MIMO detection in 5G, which aims to recover the transmitted data from the received signal at the receiver. In essence, this actually belongs to the closest vector problem (CVP) in lattice decoding. On one hand, the advanced detection schemes designed for the traditional MIMO systems like lattice-reduction-aided detection show a substantial performance loss with the increment of the antenna number [5]–[8]. 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 [9]–[13]. 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 [14]–[18].

To this end, a numberof works have been made by either improving the performance or reducing the complexity [19]–[23]. 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) [24]–[27]. 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 sampling decoding mainly lies on how to obtain the samples from the target lattice Gaussian distribution. In fact, besides lattice decoding, sampling from lattice Gaussian distribution also has drawn a lot of attentions in various research fields. In coding, lattice Gaussian distribution was employed to obtain the full shaping gain for lattice coding [28]–[30], and to achieve the capacity of the Gaussian channel [31]. It was also used to achieve information-theoretic security in the Gaussian wiretap channel [32], [33] and in the bidirectional relay channel [34], respectively. In cryptography, the lattice Gaussian distribution has become a central tool in the construction of many primitives [35], [36]. Furthermore, lattice Gaussian distribution has been adapted to bidirectional relay network under the compute-and-forward strategy for the physical layer security [34]. Additionally, it is also applied to realize the probabilistic shaping for optical communication systems [37].

In sharp contrast to the continuous Gaussian density, it is by no means trivial even to sample from a low-dimensional

Manuscript received April 14, 2020; revised September 7, 2020; accepted October 11, 2020. Date of publication November 5, 2020; date of current version April 16, 2021. This work was supported in part by the open research fund of National Mobile Communications Research Laboratory, Southeast University (No. 2019D04), 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), by the National Natural Science Foundation of China under Grant 61801216, by the Natural Science Foundation of Jiangsu Province under Grant BK20180420. This article was presented in part at the 2019 IEEE International Symposium on Information Theory (ISIT), Paris, France, July, 2019. The associate editor coordinating the review of this paper and approving it for publication was V. Raghavan. *(Corresponding author: Zheng Wang.)*

Zheng Wang is with the National Mobile Communications Research Laboratory, School of Information Science and Engineering, Southeast University, Nanjing 210096, China, and also with the Key Laboratory of Dynamic Cognitive System of Electromagnetic Spectrum Space, Ministry of Industry and Information Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China (e-mail: z.wang@ieee.org).

Ling Liu is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: liulingcs@szu.edu.cn).

Cong Ling is with the Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, U.K. (e-mail: cling@ieee.org).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCOMM.2020.3036050.

Digital Object Identifier 10.1109/TCOMM.2020.3036050

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 becomes inevitable [26], [27], [38]. 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 [39]–[43]. However, since the convergence rate of the Markov chain is hard to determine, the Markov mixing of Gibbs sampling turns out to be intractable 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 [44], which is not only uniformly ergodic in tackling with lattice Gaussian sampling but also enjoys an accessible convergence rate. In [45], 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.

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.

On the other hand, in the 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 should be fully investigated for a better sampling decoding performance. First of all, a near-optimal choice of the standard deviation $\sigma = d(\Lambda, \mathbf{c})/\sqrt{n}$ is derived and we show it is better than the choice $\sigma = \min_i \|\widehat{\mathbf{b}}_i\|/2\sqrt{\pi}$ provided in [45] 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 $\mathbf{B}$, $d(\Lambda, \mathbf{c}) = \min_{\mathbf{x}\in\mathbb{Z}^n} \|\mathbf{Bx} - \mathbf{c}\|$ 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. By dynamically approaching $d(\Lambda, \mathbf{c})$ 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 decoding and 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 $\mathbf{B}$. Finally, in this paper, the 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

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

where $\mathbf{B}$ is called a generator matrix of a lattice. 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* with respect to $\mathbf{x} \in \mathbb{Z}^n$ is defined as

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

where $\rho_{\sigma,\mathbf{c}}(\Lambda) \triangleq \sum_{\mathbf{Bx}\in\Lambda} \rho_{\sigma,\mathbf{c}}(\mathbf{Bx})$ is just a scaling to obtain a probability distribution. 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 standard deviation $\sigma > 0$ is sufficiently large [33].

B. Decoding by Sampling

Consider the decoding of an $n \times n$ real-valued system. The extension to the complex-valued system is straightforward [26]. Let $\mathbf{x} \in \mathbb{Z}^n$ denote the transmitted signal. The corresponding received signal $\mathbf{c}$ is given by

$$\mathbf{c} = \mathbf{Bx} + \mathbf{w} \tag{4}$$

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 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, i.e.,

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}\in\mathbb{Z}^n} \|\mathbf{c} - \mathbf{B}\mathbf{x}\|^2 = \arg\max_{\mathbf{x}\in\mathbb{Z}^n} D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x}), \tag{6}$$

which means finding the optimal $\widehat{\mathbf{x}}$ with the smallest Euclidean distance $\|\mathbf{c} - \mathbf{B}\mathbf{x}\|$ corresponds to returning the sample $\widehat{\mathbf{x}}$ with the largest sampling probability in $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$. Therefore, the decoding problem in (5) is converted into a sampling problem. If sampling over the lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ can be successfully performed, the solution of CVP will most likely be returned by multiple independent samplings from $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$. It has been demonstrated that lattice Gaussian sampling is equivalent to CVP [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 the decoding by sampling relies on how to successfully sample from the lattice Gaussian distribution.

In [38], 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

$$\begin{aligned} P_{\text{Klein}}(\mathbf{x}) &= \prod_{i=1}^{n} D_{\mathbb{Z},\sigma_i,\widetilde{x}_i}(x_i) = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\mathbf{x})}{\prod_{i=1}^{n}\rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})} \\ &= \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}-\mathbf{c}\|^2}}{\prod_{i=1}^{n}\sum_{x_i\in\mathbb{Z}} e^{-\frac{1}{2\sigma_i^2}\|x_i-\widetilde{x}_i\|^2}}, \end{aligned} \tag{7}$$

where $\widetilde{x}_i = \frac{c_i' - \sum_{j=i+1}^{n} r_{i,j}x_j}{r_{i,i}}$, $\sigma_i = \frac{\sigma}{|r_{i,i}|} = \frac{\sigma}{\|\widehat{\mathbf{b}}_i\|}$, $\mathbf{c}' = \mathbf{Q}^{\dagger}\mathbf{c}$, $r_{i,j}$ denotes the element of the upper triangular matrix $\mathbf{R}$ from the QR decomposition $\mathbf{B} = \mathbf{Q}\mathbf{R}$ and $\widehat{\mathbf{b}}_i$'s are the Gram-Schmidt vectors of $\mathbf{B}$ with $\|\widehat{\mathbf{b}}_i\| = |r_{i,i}|$. 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

$$\sigma \geq \omega(\sqrt{\log\, n}) \cdot \max_{1\leq i\leq n}\|\widehat{\mathbf{b}}_i\|, \tag{8}$$

Algorithm 1 IMHK Sampling Algorithm

Input: $\mathbf{B}, \sigma, \mathbf{c}, \mathbf{x}_0, t_{\text{mix}}(\epsilon)$;
Output: $\mathbf{x} \sim D_{\Lambda,\sigma,\mathbf{c}}$;
let $\mathbf{X}_0 = \mathbf{x}_0$
for $t = 1,2, \ldots,$ **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 (10)
  calculate the acceptance ratio $\alpha(\mathbf{x}, \mathbf{y})$ in (11)
  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 $\omega(\log n)$ is a superlogarithmic function.[1] 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 [44], [45]. In principle, they randomly generate the next Markov state conditioned on the previous one; after the mixing time for convergence, the distribution of samples from the Markov chain is statistically close to the stationary distribution, where the samples from the target distribution can be obtained thereafter. As an important parameter to measure the time required by a Markov chain to get close to its stationary distribution, the *mixing time* t_{mix} is defined as follows [48]

$$t_{\text{mix}}(\epsilon) = \min\{t : \max\|P^t(\mathbf{x}, \cdot) - \pi(\cdot)\|_{TV} \leq \epsilon\}, \tag{9}$$

where the superscript t denotes the number of Markov moves, $P^t(\mathbf{x}, \cdot)$ represents a row of the transition matrix $\mathbf{P}$ for t Markov moves, π is the target invariant distribution and $\|\cdot\|_{TV}$ represents the total variation distance.

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 [44]. 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:

$$q(\mathbf{x}, \mathbf{y}) = P_{\text{Klein}}(\mathbf{y}) = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\mathbf{y})}{\prod_{i=1}^{n}\rho_{\sigma_i,\widetilde{y}_i}(\mathbb{Z})}, \tag{10}$$

where the generation of the state candidate $\mathbf{y}$ is actually independent of $\mathbf{x}$. Then, regarding the state candidate $\mathbf{y}$,

[1] Here we use the standard *small omega* notation $\omega(\cdot)$, i.e., $|\omega(g(n))| > k \cdot |g(n)|$ for every fixed positive number $k > 0$.

the acceptance ratio α is calculated by

$$\begin{aligned}\alpha(\mathbf{x},\mathbf{y}) &= \min\left\{1, \frac{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})}\right\} \\ &= \min\left\{1, \frac{\prod_{i=1}^{n}\rho_{\sigma_i,\widetilde{y}_i}(\mathbb{Z})}{\prod_{i=1}^{n}\rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})}\right\}, \end{aligned} \tag{11}$$

where the stationary distribution $\pi = D_{\Lambda,\sigma,\mathbf{c}}$. 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

$$\begin{aligned}P_{\text{IMHK}}(\mathbf{x},\mathbf{y}) &= q(\mathbf{x},\mathbf{y})\cdot\alpha(\mathbf{x},\mathbf{y}) \\ &= \min\left\{P_{\text{Klein}}(\mathbf{y}), \frac{\pi(\mathbf{y})P_{\text{Klein}}(\mathbf{x})}{\pi(\mathbf{x})}\right\}, \end{aligned} \tag{12}$$

for cases $\mathbf{x} \neq \mathbf{y}$.

Proposition 1 [44]: Given the invariant lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}$, the Markov chain established by the IMHK algorithm is uniformly ergodic:

$$\|P^t(\mathbf{x},\cdot) - D_{\Lambda,\sigma,\mathbf{c}}(\cdot)\|_{TV} \leq (1-\delta)^t \quad \textit{for all } \mathbf{x} \in \mathbb{Z}^n \tag{13}$$

with

$$\delta \triangleq \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{\Lambda})}{\prod_{i=1}^{n}\rho_{\sigma_i}(\mathbb{Z})} > 0. \tag{14}$$

Clearly, the exponential decay coefficient $0 < \delta < 1$ is the key to determine the convergence rate.

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 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 $S = \{(\mathbf{x},u) : 0 \leq u \leq \pi(\mathbf{x})\}$. To achieve this, slice sampling alternatively updates $\mathbf{x}$ and u from uniform distributions $p(\mathbf{x} \mid u) \sim \text{Uni}(S)$ and $p(u \mid \mathbf{x}) \sim \text{Uni}(0,\pi(\mathbf{x}))$ 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 the introduced variable u is marginalized out and $Z > 0$ is a constant scalar. Overall, the operation of slice sampling can be summarized as follows:

1) *Sample u_t from the conditional distribution*

$$p(u_t \mid \mathbf{x}_{t-1}) \sim \text{Uni}(0, \pi(\mathbf{x}_{t-1})). \tag{15}$$

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

$$p(\mathbf{x}_t \mid u_t) \sim \text{Uni}(S_u), \tag{16}$$

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. 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 it 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 following factorization

$$\pi(\mathbf{x}) = D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x}) = P_{\text{Klein}}(\mathbf{x})\cdot l(\mathbf{x}) \tag{17}$$

with

$$l(\mathbf{x}) \triangleq \frac{\prod_{i=1}^{n}\rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})}{\rho_{\sigma,\mathbf{c}}(\Lambda)}, \tag{18}$$

we can establish the joint distribution as

$$\Pi(\mathbf{x},u) = P_{\text{Klein}}(\mathbf{x})\cdot I_{u<l(\mathbf{x})}(\mathbf{x}), \tag{19}$$

where $\rho_{\sigma,\mathbf{c}}(\Lambda)$ is a constant 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 (17) is adopted to the proposed slice sampling, which results in the joint distribution in (19). In fact, it was pointed out by Besag and Green in [53] that the usage of decomposition is rather effective in multidimensional 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 $P_{\text{Klein}}^{A_{u_t}}(\mathbf{x})$ which restricts the set of $\mathbf{x}$ in $P_{\text{Klein}}(\mathbf{x})$ into the set $A_u = \{\mathbf{x} : l(\mathbf{x}) > u\}$:

1) *Sample u_t from the conditional distribution*

$$p(u_t \mid \mathbf{x}_{t-1}) \sim \text{Uni}(0, l(\mathbf{x}_{t-1})). \tag{20}$$

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

$$p(\mathbf{x}_t \mid u_t) \sim P_{\text{Klein}}^{A_{u_t}}(\mathbf{x}), \tag{21}$$

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 (20) and (21) lead to the joint distribution $\Pi(\mathbf{x},u)$ in (19). Intuitively, with respect to (21), 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

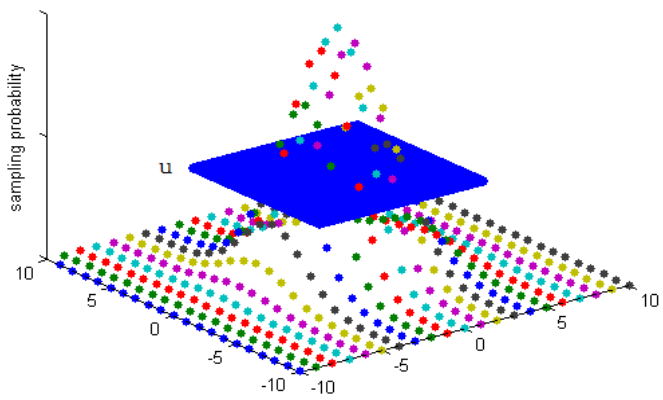

Fig. 1. Illustration of a two-dimensional lattice Gaussian distribution and a slice (blue plane) with $u > 0$ over it.

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 $t = 1,2, \ldots,$ **do**
  calculate $l(\mathbf{x}_{t-1})$ according to (18)
  uniformly draw u_t from the interval $(0, l(\mathbf{x}_{t-1}))$
  for $k = 1,2, \ldots,$ **do**
    sample $\mathbf{x}_t$ from $P_{\text{Klein}}(\mathbf{x})$ shown in (7)
    calculate $l(\mathbf{x}_t)$ according to (18)
    if $l(\mathbf{x}_t) > u_t$ **then**
      break
    end if
  end for
  if $t \geq t_{\text{mix}}(\epsilon)$ **then**
    output $\mathbf{x}_t$
  end if
end for

qualified candidate is found for $\mathbf{x}_t$. Here one also can sample $\mathbf{x}$ from $P_{\text{Klein}}(\mathbf{x})$ approximately under the restriction $\mathbf{x} \in A_{u_t}$ directly.[2] Interestingly, the numerator in (18) has already been calculated by Klein's algorithm during the sampling while the denominator in (18) only serves as a scalar that can be set free for non-negative values. In addition, considerable sampling potential can be further exploited to improve the sampling efficiency, which is one of our work in future. To summarize, the proposed sliced lattice Gaussian sampling algorithm is presented in Algorithm 2. More precisely, the complexity of each Markov move in slice sampling is $O(n^2)$, which is the same order with the IMHK and Gibbs sampling algorithms. For this reason, in MCMC the computational cost of each Markov move is often insignificant in complexity comparison, whereas the number of Markov moves is more critical.

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 advantages of the proposed sampling over IMHK sampling.

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 $\{\mathbf{X}_1, \mathbf{X}_2, \ldots\}$ 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

$$\begin{aligned}
&\pi(\mathbf{x}_t) P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) \\
&= \pi(\mathbf{x}_t) \int \Pi(u_{t+1}|\mathbf{x}_t)\Pi(\mathbf{x}_{t+1}|u_{t+1}) du_{t+1} \\
&= \int \Pi(\mathbf{x}_t|u_{t+1})\Pi(u_{t+1}|\mathbf{x}_t)\Pi(\mathbf{x}_{t+1}|u_{t+1}) p(u_{t+1}) du_{t+1} \\
&= \pi(\mathbf{x}_{t+1}) \int \Pi(u_{t+1}|\mathbf{x}_t)\Pi(\mathbf{x}_t|u_{t+1}) du_{t+1} \\
&= \pi(\mathbf{x}_{t+1}) \int \Pi(u_t|\mathbf{x}_{t+1})\Pi(\mathbf{x}_t|u_t) du_t \\
&= \pi(\mathbf{x}_{t+1}) P_{\text{Slice}}(\mathbf{x}_{t+1}, \mathbf{x}_t),
\end{aligned} \tag{22}$$

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 $\{\mathbf{X}_1, \mathbf{X}_2, \ldots\}$, its transition probability can be derived as

$$\begin{aligned}
&P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) \\
&= \int_0^{l(\mathbf{x}_t)} p(\mathbf{x}_{t+1}|u_{t+1}) p(u_{t+1}|\mathbf{x}_t) du_{t+1} \\
&= \int_0^{l(\mathbf{x}_t)} P_{\text{Klein}}^{A_{u_{t+1}}}(\mathbf{x}_{t+1}) p(u_{t+1}|\mathbf{x}_t) du_{t+1} \\
&= \frac{1}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)} P_{\text{Klein}}^{A_{u_{t+1}}}(\mathbf{x}_{t+1}) du_{t+1} \\
&= \frac{1}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)} \frac{P_{\text{Klein}}(\mathbf{x}_{t+1}) I_{u_{t+1} < l(\mathbf{x}_{t+1})}(\mathbf{x}_{t+1})}{P_{\text{Klein}}(A_{u_{t+1}})} du_{t+1}
\end{aligned} \tag{23}$$

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

for cases $\mathbf{x}_t \neq \mathbf{x}_{t+1}$, where $P_{\text{Klein}}(A_{u_{t+1}})$ denotes the probability summation of $P_{\text{Klein}}(\mathbf{x})$ over set $A_{u_{t+1}} = \{\mathbf{x} : l(\mathbf{x}) > u_{t+1}\}$, (23) recalls *Bayes' theorem* and "$\wedge$" yields the smaller choice between two terms.

Here, the inequality (24) follows the fact that [35]

$$\frac{P_{\text{Klein}}(\mathbf{x})}{\pi(\mathbf{x})} = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{\Lambda})}{\prod_{i=1}^{n} \rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})} \geq \delta > 0 \tag{25}$$

[2]For more details of implementing Klein's sampling at each layer with finite searching space, the reader is referred to [26].

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

Theorem 1: Given the invariant lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}$, the sub-chain $\{\mathbf{X}_1, \mathbf{X}_2, \ldots\}$ established by the proposed sliced lattice Gaussian sampling algorithm is uniformly ergodic as:

$$\|P^t(\mathbf{x}, \cdot) - D_{\Lambda,\sigma,\mathbf{c}}(\cdot)\|_{TV} \leq (1-\delta)^t \quad \textit{for all } \mathbf{x} \in \mathbb{Z}^n. \tag{26}$$

Actually, the result of $P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) \geq \delta \cdot \pi(\mathbf{x}_{t+1})$ for all the Markov states is accordance with the definition of *small set* in the literature on 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 [44], [45]. Intuitively, because of $P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) \geq P_{\text{IMHK}}(\mathbf{x}_t, \mathbf{x}_{t+1})$ for $\mathbf{x}_t \neq \mathbf{x}_{t+1}$, the sliced sampler performs better than the IMHK sampling in terms of the convergence performance.

B. Convergence Improvement

Similar to IMHK sampling, the proposed sliced sampling for lattice Gaussian distribution is uniformly ergodic as well. 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

$$P_{\text{Slice}}(\mathbf{x}, \mathbf{y}) \geq P_{\text{IMHK}}(\mathbf{x}, \mathbf{y}) \tag{27}$$

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 the literature on MCMC, this is known as *Peskun ordering* defined in [54]

$$P_{\text{Slice}} \succeq P_{\text{IMHK}}. \tag{28}$$

We then invoke the following Proposition 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

$$v(f, \mathbf{P}) = \lim_{n\to\infty} \frac{1}{n} \text{var}\left\{\sum_{t=1}^{n} f(\mathbf{X}_t)\right\}, \tag{29}$$

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

Proposition 2 [55]: Suppose $\mathbf{P}_1$ and $\mathbf{P}_2$ are reversible transition matrices with the same invariant distribution and $\mathbf{P}_2 \succeq \mathbf{P}_1$. Then, for any function $f \in L_0^2(\pi)$ with zero mean $E\{f\} = 0$, we have

$$v(f, \mathbf{P}_1) \geq v(f, \mathbf{P}_2). \tag{30}$$

Clearly, from Proposition 2, 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 $\mathbf{P}_2 \succeq \mathbf{P}_1$, then their corresponding second largest eigenvalues satisfy

$$|\lambda_{\max,1}| \geq |\lambda_{\max,2}|, \tag{31}$$

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

$$|\lambda_{\max}|_{\text{Slice}} \leq |\lambda_{\max}|_{\text{IMHK}} \tag{32}$$

to confirm the convergence gain of the proposed sliced sampling.

Obviously, given the value of $\delta < 1$, the mixing time of the Markov chain can be calculated by (9) and (26), that is,

$$t_{\text{mix}}(\epsilon) = \frac{\ln \epsilon}{\ln(1-\delta)} \leq (-\ln \epsilon) \cdot \left(\frac{1}{\delta}\right), \quad \epsilon < 1 \tag{33}$$

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 $\delta \to 1$. 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 directly.

V. Decoding Analysis

In this section, we apply the proposed sliced sampling algorithm 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. Therefore one can leave a sampling 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 [45], the statistical complexity of solving CVP by MCMC is defined as follows.

Definition 1 [45]: Let $d(\Lambda, \mathbf{c}) = \min_{\mathbf{x}\in\mathbb{Z}^n} \|\mathbf{B}\mathbf{x} - \mathbf{c}\|$ 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 statistical complexity (i.e., the number of Markov moves t) of solving CVP by MCMC is

$$C_{\text{CVP}} \triangleq \frac{t_{mix}}{D_{\Lambda,\sigma,\mathbf{c}}(\widehat{\mathbf{x}})}. \tag{34}$$

According to (33) and (34), the decoding complexity of the proposed sliced sampling for CVP can be upper bounded by

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

$$= \log\left(\frac{1}{\epsilon}\right) \cdot \frac{\prod_{i=1}^{n} \rho_{\sigma_i}(\mathbb{Z})}{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\widehat{\mathbf{x}})}$$
$$= \log\left(\frac{1}{\epsilon}\right) \cdot C(\sigma), \tag{35}$$

where

$$C(\sigma) \triangleq \frac{\prod_{i=1}^{n} \rho_{\sigma_i}(\mathbb{Z})}{\rho_{\sigma,\mathbf{c}}(\mathbf{B}\widehat{\mathbf{x}})} = \left(\prod_{i=1}^{n} \sum_{x_i \in \mathbb{Z}} e^{-\frac{\|\widehat{\mathbf{b}}_i\|^2}{2\sigma^2} \cdot x_i^2}\right) \cdot e^{\frac{d^2(\Lambda,\mathbf{c})}{2\sigma^2}}. \tag{36}$$

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 [45]. Based on (36), further analysis is given to optimize the choice of the standard deviation σ in what follows, thus leading to a better decoding performance. In addition, the following derived results of σ is also applicable to the IMHK sampling decoding for the performance enhancement as it considers the same optimization problem to minimize $C(\sigma)$ in (36).

A. Optimization of σ

From the point of view of simulated annealing in statistics, σ functions as "temperature" to guide the Markov mixing, which also has an impact upon t_{mix} as well. Given the lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ shown in (3), although a small size σ offers a relatively large decoding sampling probability $D_{\Lambda,\sigma,\mathbf{c}}(\widehat{\mathbf{x}})$ for the target optimal decoding solution, it also incurs a "cold" Markov chain which tends to be trapped by the frozen status, and *vice versa* [56] Therefore, to balance this inherent trade-off for a better sampling decoding performance, necessary optimization about σ should be carried out carefully.

In [45], 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 (36) for the proposed sliced sampling, we have

$$C(\sigma_A) = e^{\frac{2\pi}{\min^2 \|\widehat{\mathbf{b}}_i\|} \cdot d^2(\Lambda,\mathbf{c})} \cdot \prod_{i=1}^{n} \vartheta_3\left(\frac{2\|\widehat{\mathbf{b}}_i\|^2}{\min^2 \|\widehat{\mathbf{b}}_i\|}\right) \tag{37}$$

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

$$C(\sigma_A) \le e^{\frac{2\pi}{\min_i \|\widehat{\mathbf{b}}_i\|^2} \cdot d^2(\Lambda,\mathbf{c})} \cdot \vartheta_3^n(2)$$
$$= 1.0039^n \cdot e^{\frac{2\pi}{\min_i \|\widehat{\mathbf{b}}_i\|^2} \cdot d^2(\Lambda,\mathbf{c})}, \tag{38}$$

where $\vartheta_3(2) = 1.0039$ was given in [45]. 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

$$\overline{C}(\sigma) \triangleq \left(\prod_{i=1}^{n} \int_{-\infty}^{\infty} e^{-\frac{\|\widehat{\mathbf{b}}_i\|^2}{2\sigma^2} \cdot x_i^2} dx_i\right) \cdot e^{\frac{d^2(\Lambda,\mathbf{c})}{2\sigma^2}}, \tag{39}$$

which is a continuous version of (36). According to the fact $\int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma^2}} dx = \sqrt{2\pi}\sigma$, it follows that

$$\overline{C}(\sigma) = \frac{(2\pi)^{\frac{n}{2}}}{|\det(\mathbf{B})|} \cdot \sigma^n \cdot e^{\frac{d^2(\Lambda,\mathbf{c})}{2\sigma^2}} \tag{40}$$

and the derivative of function $\overline{C}(\sigma)$ with respect to σ can be easily obtained. Furthermore, by letting $\frac{\partial \overline{C}(\sigma)}{\partial \sigma} = 0$, then we have

$$\sigma = \frac{d(\Lambda,\mathbf{c})}{\sqrt{n}}. \tag{41}$$

Here, for notational simplicity, we apply $\sigma_B = d(\Lambda,\mathbf{c})/\sqrt{n}$ as the choice for $C(\sigma)$ shown in (36), and it follows that

$$C(\sigma_B) = e^{\frac{n}{2}} \cdot \left(\prod_{i=1}^{n} \sum_{x_i \in \mathbb{Z}} e^{-\frac{n\|\widehat{\mathbf{b}}_i\|^2}{2d^2(\Lambda,\mathbf{c})} \cdot x_i^2}\right)$$
$$= e^{\frac{n}{2}} \cdot \prod_{i=1}^{n} \vartheta_3\left(\frac{n\|\widehat{\mathbf{b}}_i\|^2}{2\pi d^2(\Lambda,\mathbf{c})}\right). \tag{42}$$

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

$$C(\sigma_B) \le e^{\frac{n}{2}} \cdot \left(\prod_{i=1}^{n} \int_{-\infty}^{\infty} e^{-\frac{n\|\widehat{\mathbf{b}}_i\|^2}{2d^2(\Lambda,\mathbf{c})} \cdot x_i^2} dx_i\right)$$
$$= e^{\frac{n}{2}} \cdot \prod_{i=1}^{n} \left(\sqrt{2\pi} \cdot \frac{d(\Lambda,\mathbf{c})}{\sqrt{n}\|\widehat{\mathbf{b}}_i\|}\right)$$
$$= \left(\frac{2\pi e}{n}\right)^{\frac{n}{2}} \cdot \frac{d^n(\Lambda,\mathbf{c})}{|\det(\mathbf{B})|}. \tag{43}$$

Intuitively, when $n > 2\pi e$ (i.e., $n \ge 18$), $C(\sigma_B)$ is mainly dominated by the relationship between $d(\Lambda,\mathbf{c})$ and $|\det(\mathbf{B})|$. More precisely, according to (43), the complexity of $O(e^{\frac{n}{2}})$ for solving CVP can be achieved by $C(\sigma_B)$ if

$$d(\Lambda,\mathbf{c}) \le \sqrt{\frac{n}{2\pi}} \cdot |\det(\mathbf{B})|^{\frac{1}{n}}. \tag{44}$$

Furthermore, by substituting (43) into (35), the complexity of solving CVP via the proposed sliced sampling decoding can be derived as

$$C_{\text{CVP}} \le \log\left(\frac{1}{\epsilon}\right) \cdot \left(\frac{2\pi e}{n}\right)^{\frac{n}{2}} \cdot \frac{d^n(\Lambda,\mathbf{c})}{|\det(\mathbf{B})|} \tag{45}$$

and its decoding radius in terms of bounded distance decoding (BDD) follows that

$$d_{\sigma_{\text{B}}}(\Lambda,\mathbf{c}) = \sqrt{\frac{n}{2\pi e}} \cdot \left(\frac{C_{\text{CVP}}}{\log(\frac{1}{\epsilon})}\right)^{\frac{1}{n}} \cdot |\det(\mathbf{B})|^{\frac{1}{n}}, \tag{46}$$

where the decoding radius based on σ_A is

$$d_{\sigma_{\text{A}}}(\Lambda,\mathbf{c}) = \sqrt{\frac{1}{2\pi} \cdot \ln\frac{C_{\text{CVP}}}{\log\left(\frac{1}{\epsilon}\right)}} \cdot \min_{1 \le i \le n} \|\widehat{\mathbf{b}}_i\|. \tag{47}$$

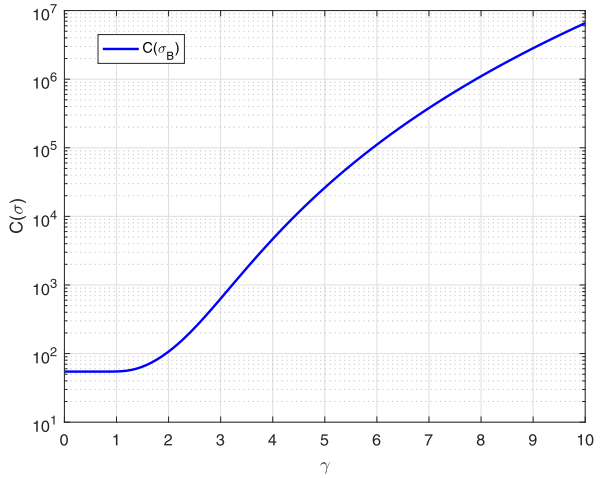

Fig. 2. $C(\sigma_{\mathrm{B}})$ versus $\gamma = d(\Lambda,\mathbf{c})/\left(\frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}\right)$ for the lattice basis $\mathbf{B}\in\mathbb{R}^{8\times 8}$ with Gaussian coefficients.

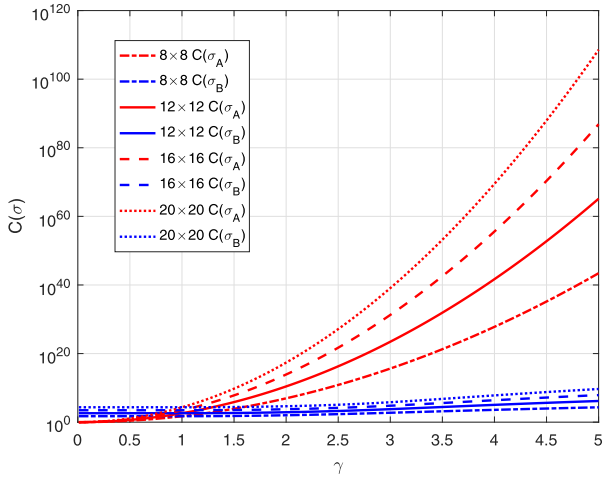

Fig. 3. $C(\sigma)$ versus $\gamma = d(\Lambda,\mathbf{c})/\left(\frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}\right)$ for 8×8, 12×12, 16×16 and 20×20 lattice basis $\mathbf{B}$ with Gaussian coefficients.

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 given.

Proposition 3: Given $C(\sigma_A)$ and $C(\sigma_B)$ in (37) and (42) 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

$$\sigma = \begin{cases} \sigma_A = \min\limits_i \|\widehat{\mathbf{b}}_i\|/2\sqrt{\pi} & \text{if } d(\Lambda,\mathbf{c}) \le \dfrac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}; \\ \sigma_B = d(\Lambda,\mathbf{c})/\sqrt{n} & \text{if } d(\Lambda,\mathbf{c}) > \dfrac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}. \end{cases} \tag{48}$$

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 $d(\Lambda,\mathbf{c}) > \frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$. Additionally, in order to present a clear complexity comparison, the complexity of upper bounds of solving CVP (i.e., the maximum number of Markov moves required to solve CVP) of various sampling decoding are given in Table I. Because the computational complexities of Gibbs, IMHK and the proposed sliced sampling algorithms have the same complexity order $O(n^2)$ within one single Markov move, the corresponding computational costs (i.e., the maximum number of Markov moves times the computational complexity per move) for each sampling decoding scheme can be easily derived as $O(C\cdot n^2)$. Intuitively, the computational complexity costs are different from each other mainly due to the different numbers of the required Markov moves, making the number of Markov moves more critical than the computational complexity of each move.[3]

Another point that should be emphasized is the application of lattice reduction techniques. 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 $\max_i\|\widehat{\mathbf{b}}_i\|$ [57]. 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 [58]. Note that as shown in (42) and (46), LLL reduction does not alter the volume of lattice, which corresponds to a constant $|\det(\mathbf{B})|$. Since $C(\sigma_A)$ is better

[3]In fact, the computational complexities of random and derandomized sampling algorithms within one single sampling are $O(n^2)$ as well, so that their computational complexities are mainly determined by the number of samples, i.e., $O(C\cdot n^2)$.

TABLE I
COMPLEXITY COMPARISON OF VARIOUS DECODING SCHEMES

	Complexity Upper Bound of Solving CVP
Klein Sampling [38]	$C_{\text{klein}} \leq e^2 \cdot n^{\frac{d^2(\Lambda,\mathbf{c})}{\min_i^2 \lvert r_{i,i}\rvert}}$
Randomized Sampling [26]	$C_{\text{random}} \leq e^{\frac{2n}{\varrho}} \cdot \varrho^{\frac{d^2(\Lambda,\mathbf{c})}{\min_i^2 \lvert r_{i,i}\rvert}},\ \varrho > 1$
Derandomized Sampling [27]	$C_{\text{derandom}} \leq \frac{1}{2} \cdot e^{\frac{2n}{\varrho}} \cdot \varrho^{\frac{d^2(\Lambda,\mathbf{c})}{\min_i^2 \lvert r_{i,i}\rvert}},\ \varrho > 1$
IMHK Sampling [45]	$C_{\text{imhk}} \leq \log(1/\epsilon) \cdot e^{\frac{2\pi d^2(\Lambda,\mathbf{c})}{\min_i^2 \lvert r_{i,i}\rvert}}$
Sliced Sampling	$C_{\text{slice}} \leq \log(1/\epsilon) \cdot \min\left\{ e^{\frac{2\pi d^2(\Lambda,\mathbf{c})}{\min_i^2 \lvert r_{i,i}\rvert}}, (\frac{2\pi e}{n})^{\frac{n}{2}} \cdot \frac{d^n(\Lambda,\mathbf{c})}{\lvert\det(\mathbf{B})\rvert} \right\}$

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.

C. Dynamic Update of σ

However, given the choices of σ in (48), an estimation with respect to $d(\Lambda,\mathbf{c})$ is required. For this reason, a natural question shown below raises, which should be considered carefully:

- The premise $d(\Lambda,\mathbf{c})$ corresponds to solving CVP, which is difficult to obtain at the beginning.

Therefore, we try to answer it 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.[4] 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, $\mathbf{x}_{\text{sic-lll}}$ can also be applied as the initial starting state of the Markov chain, which is helpful to the Markov mixing [59].

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. 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

$$\sigma_{\text{dynamic}} = \frac{d_{\text{update}}(\Lambda,\mathbf{c})}{\sqrt{n}} \triangleq \frac{\min_{\mathbf{x}\in S}\|\mathbf{B}\mathbf{x}-\mathbf{c}\|}{\sqrt{n}} \cdot \vartheta_3^n(2), \quad (49)$$

where set S contains all the samples of $\mathbf{x}$ already collected during the sampling. Hence, along with the sampling, $d_{\text{update}}(\Lambda,\mathbf{c})$ shrinks gradually, leading to a better estimation of σ to σ_B.

Proposition 4: 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*

$$d_{\text{initial}}(\Lambda,\mathbf{c}) \lessapprox |\det(\mathbf{B})|^{\frac{1}{n}} \cdot \sqrt{\frac{n}{2\pi e}} \cdot e^{\frac{2\pi\cdot d^2(\Lambda,\mathbf{c})}{n\cdot\min^2\|\widehat{\mathbf{b}}_i\|}}, \quad (50)$$

where $\lessapprox$ denotes approximately less than.

Proof: First of all, by substituting $\sigma_{\text{initial}} = \frac{d_{\text{initial}}(\Lambda,\mathbf{c})}{\sqrt{n}}$ into (43), we have

$$C(\sigma_{\text{initial}}) \leq \left(\frac{2\pi e}{n}\right)^{\frac{n}{2}} \cdot \frac{d^n_{\text{initial}}(\Lambda,\mathbf{c})}{|\det(\mathbf{B})|}. \quad (51)$$

Then, given (37), in order to make sure $C(\sigma_{\text{initial}}) < C(\sigma_A)$, it follows that

$$\left(\frac{2\pi e}{n}\right)^{\frac{n}{2}} \cdot \frac{d^n_{\text{initial}}(\Lambda,\mathbf{c})}{|\det(\mathbf{B})|} < e^{\frac{2\pi}{\min^2\|\widehat{\mathbf{b}}_i\|}\cdot d^2(\Lambda,\mathbf{c})} \cdot \prod_{i=1}^{n} \vartheta_3\left(\frac{2\|\widehat{\mathbf{b}}_i\|^2}{\min^2\|\widehat{\mathbf{b}}_i\|}\right) \quad (52)$$

so as to

$$\begin{aligned} & d_{\text{initial}}(\Lambda,\mathbf{c}) \\ & < |\det(\mathbf{B})|^{\frac{1}{n}} \cdot \sqrt{\frac{n}{2\pi e}} \cdot e^{\frac{2\pi\cdot d^2(\Lambda,\mathbf{c})}{n\cdot\min^2\|\widehat{\mathbf{b}}_i\|}} \cdot \prod_{i=1}^{n} \vartheta_3\left(\frac{2\|\widehat{\mathbf{b}}_i\|^2}{\min^2\|\widehat{\mathbf{b}}_i\|}\right) \\ & \leq |\det(\mathbf{B})|^{\frac{1}{n}} \cdot \sqrt{\frac{n}{2\pi e}} \cdot e^{\frac{2\pi\cdot d^2(\Lambda,\mathbf{c})}{n\cdot\min^2\|\widehat{\mathbf{b}}_i\|}} \cdot \vartheta_3^n(2) \\ & \approx |\det(\mathbf{B})|^{\frac{1}{n}} \cdot \sqrt{\frac{n}{2\pi e}} \cdot e^{\frac{2\pi\cdot d^2(\Lambda,\mathbf{c})}{n\cdot\min^2\|\widehat{\mathbf{b}}_i\|}}, \end{aligned} \quad (53)$$

completing the proof. ∎

From Proposition 4, 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 (49) has the risk to be excessively large. To prevent such a problem, it is necessary to set an upper bound for σ as

$$\sigma_{\text{dynamic}} = \min\left\{\frac{d_{\text{update}}(\Lambda,\mathbf{c})}{\sqrt{n}}, \gamma\cdot\frac{\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}\right\}, \quad (54)$$

where $d(\Lambda,\mathbf{c}) = \gamma\cdot\frac{\sqrt{n}\min\|\widehat{\mathbf{b}}_i\|}{2\sqrt{\pi}}$. Here, the coefficient γ is suggested to choose from the range $[1.2, 1.4]$ experimentally, which could be further optimized in practice. We emphasize the significance of the above upper bound, which is important

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

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 [60]–[62] 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 [56].

VI. Numerical Studies

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

$$\mathbf{c} = \mathbf{H}\mathbf{x} + \mathbf{w}. \tag{55}$$

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. Let E_b represent the average power per bit at the receiver, then the signal-to-noise ratio (SNR) $E_b/N_0 = n/(\log_2(M)\sigma_w^2)$ where M is the modulation level and σ_w^2 is the noise variance. 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

$$P_{\mathcal{L}(\mathbf{H}),\sigma,\mathbf{c}}(\mathbf{x}) = \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{H}\mathbf{x}-\mathbf{c}\|^2}}{\sum_{\mathbf{x}\in\mathcal{X}^n} e^{-\frac{1}{2\sigma^2}\|\mathbf{H}\mathbf{x}-\mathbf{c}\|^2}}, \tag{56}$$

and the closest lattice point $\mathbf{H}\mathbf{x}$ will be returned with the highest probability. Hence, after multiple samplings, the solution of CVP is the most likely to be returned.

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, we use K to denote the number of Markov moves of MCMC sampling, and 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

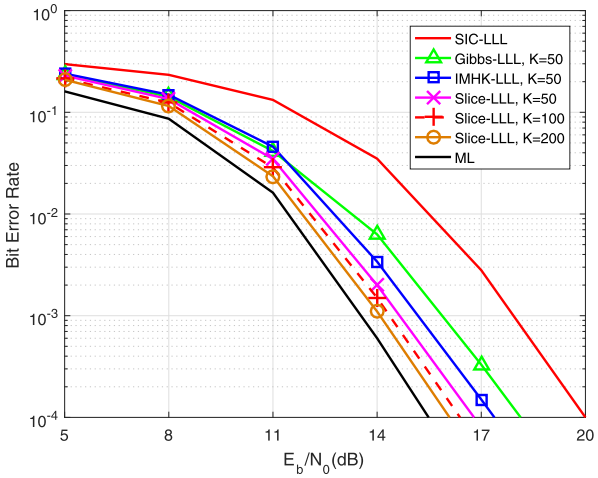

Fig. 4. Bit error rate versus average SNR per bit for the uncoded 8×8 MIMO system using 16-QAM.

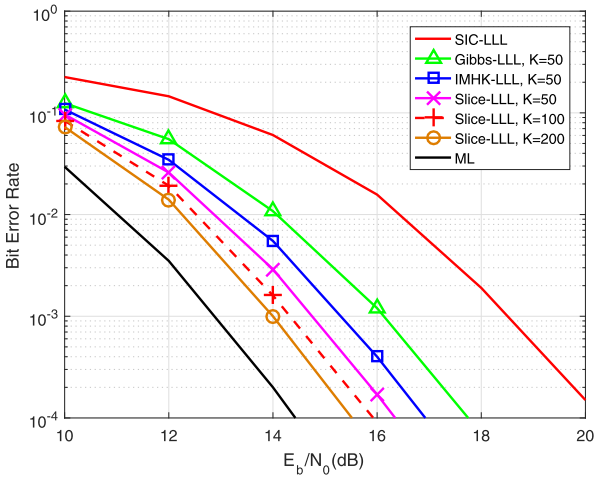

Fig. 5. Bit error rate versus average SNR per bit for the uncoded 12×12 MIMO system using 16-QAM.

number of Markov moves (i.e., $K = 50$). On the other hand, the performance of ML decoding, which is realized by sphere decoding (SD) to solve the closest vector problem (CVP) in (5) by enumerations is also given as a baseline. Clearly, 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, the Gibbs sampling from MCMC is also added for the decoding comparison, where the setting of standard deviation σ_{distance} comes from [63].

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 superior to IMHK sampling in terms of decoding performance, thus implying a better convergence performance. 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

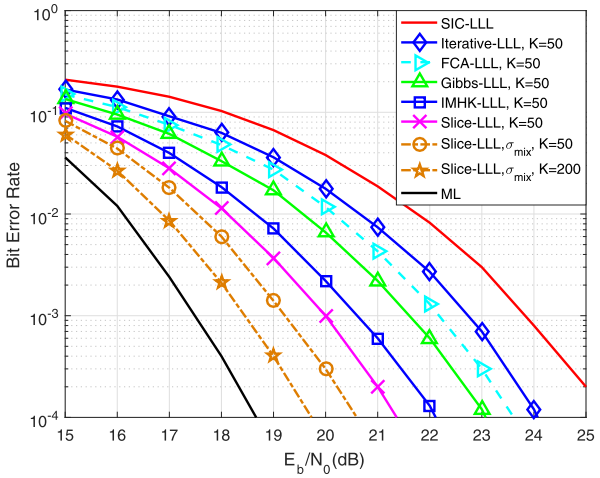

Fig. 6. Bit error rate versus average SNR per bit for the uncoded 16×16 MIMO system using 64-QAM.

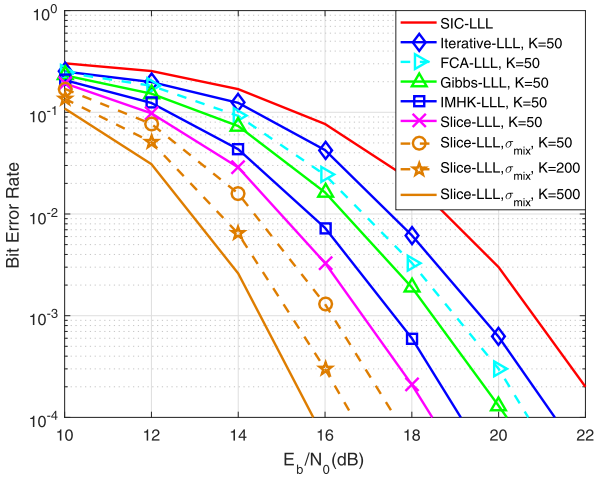

Fig. 7. Bit error rate versus average SNR per bit for the uncoded 24×24 MIMO system using 16-QAM.

the 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 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_{mix} in a 16×16 uncoded MIMO system with 64-QAM. As shown in Proposition 3, σ_A is only advantageous in a bounded 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 (54) 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 [64] and iterative list decoding in [65] 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. The similar observations can be found in Fig. 8, where the BER performance comparison is illustrated in a 48×48 uncoded MIMO system with 4-QAM. As expected, the decoding gain of the choice σ_{mix} over σ_A still can be observed for both

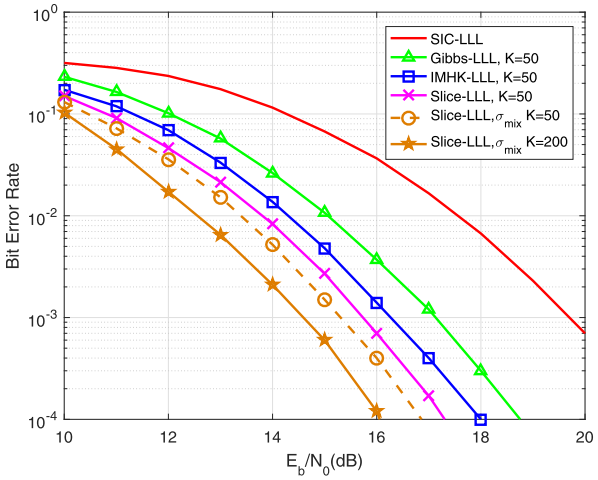

Fig. 8. Bit error rate versus average SNR per bit for the uncoded 48×48 MIMO system using 4-QAM.

IMHK sampling decoding and sliced sampling decoding, which further improves 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. 9 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

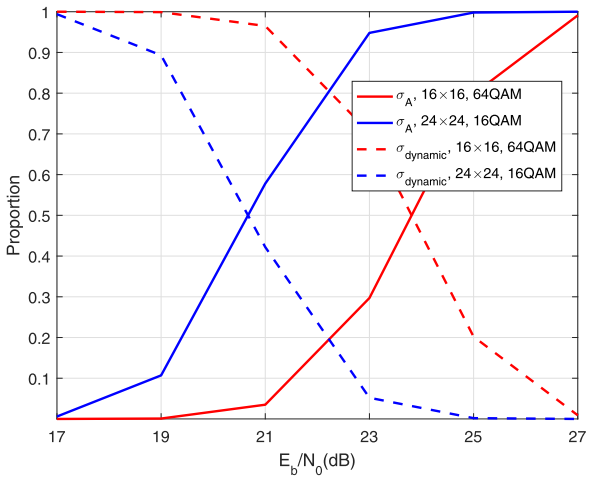

Fig. 9. Choice proportion of σ_{mix} versus average SNR per bit for the uncoded MIMO systems.

TABLE II

AVERAGE RUNNING TIMES OF $K = 50$ MARKOV MOVES (I.E., MEASURED IN SECONDS) IN MIMO DETECTION WITH VARIOUS DIMENSIONS AND E_b/N_0 UNDER 16-QAM

	E_b/N_0=11	E_b/N_0=13	E_b/N_0=15	E_b/N_0=17
16×16 IMHK	0.0371	0.0373	0.0366	0.0368
16×16 Slice	0.0397	0.0391	0.0387	0.0382
24×24 IMHK	0.0509	0.0517	0.0548	0.0503
24×24 Slice	0.0561	0.0554	0.0545	0.0538

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.

Table II is given to show the complexity comparison in average elapsed running times for $K = 50$ Markov moves in both sliced sampling decoding and IMHK sampling decoding. Typically, the simulation is conducted by MATLAB R2019a on a single computer, with an Intel Core i7 processor at 2.3GHz, a RAM of 8GB and Windows 10 Enterprise Service Pack operating system. As can be seen clearly, the average elapsed running times of the sliced sampling decoding are comparable to these of IMHK. This is accordance with the derived computational complexity $O(C{\cdot}n^2)$, where the number of Markov moves is the key. Clearly, under the same number of Markov moves, sliced sampling algorithm achieves a better decoding performance than IMHK.

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 uniform ergodicity to converge in an exponential way, 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 $\sigma > 0$ 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. Finally, simulation results based on the large-scale MIMO detection are presented to confirm the performance gain by the convergence enhancement and the parameter optimization of the proposed sliced lattice Gaussian sampling algorithm.

REFERENCES

[1] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.
[2] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.
[3] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, 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 Trans. Commun.*, vol. 61, no. 4, pp. 1436–1449, Apr. 2013.
[5] L. Babai, "On Lovász' lattice reduction and the nearest lattice point problem," *Combinatorica*, vol. 6, no. 1, pp. 1–13, Mar. 1986.
[6] M. Taherzadeh, A. Mobasher, and A. K. Khandani, "LLL reduction achieves the receive diversity in MIMO decoding," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4801–4805, Dec. 2007.
[7] 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. Inf. Theory*, vol. 56, no. 10, pp. 4765–4780, Oct. 2010.
[8] C. Ling, "On the proximity factors of lattice reduction-aided decoding," *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2795–2808, Jun. 2011.
[9] R. Kannan, "Improved algorithms for integer programming and related lattice problems," in *Proc. ACM Symp. Theory Comput.*, Boston, MA, USA, Apr. 1983, pp. 193–206.
[10] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inf. Theory*, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.
[11] B. Hassibi and H. Vikalo, "On the sphere-decoding algorithm I. Expected complexity," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2806–2818, Aug. 2005.
[12] M. O. Damen, H. El Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2389–2402, Oct. 2003.
[13] J. Jalden and B. Ottersten, "On the complexity of sphere decoding in digital communications," *IEEE Trans. Signal Process.*, vol. 53, no. 4, pp. 1474–1484, Apr. 2005.
[14] A. D. Murugan, H. El Gamal, M. O. Damen, and G. Caire, "A unified framework for tree search decoding: Rediscovering the sequential decoder," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 933–953, Mar. 2006.
[15] 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.
[16] J. Jalden, L. G. Barbero, B. Ottersten, and J. S. Thompson, "The error probability of the fixed-complexity sphere decoder," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2711–2720, Jul. 2009.

[17] S. Chen, T. Zhang, and Y. Xin, "Relaxed K-best MIMO signal detector design and VLSI implementation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 15, no. 3, pp. 328–337, Mar. 2007.

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

[19] 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 Trans. Commun.*, vol. 59, no. 11, pp. 2955–2963, Nov. 2011.

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

[21] A.-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 Trans. Commun.*, vol. 64, no. 2, pp. 586–600, Feb. 2016.

[22] 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 J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 902–915, Oct. 2014.

[23] 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 J. Sel. Topics Signal Process.*, vol. 5, no. 8, pp. 1497–1511, Dec. 2011.

[24] B. Hassibi, M. Hansen, A. G. Dimakis, H. A. J. 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 Trans. Signal Process.*, vol. 62, no. 17, pp. 4436–4450, Sep. 2014.

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

[26] S. Liu, C. Ling, and D. Stehle, "Decoding by sampling: A randomized lattice algorithm for bounded distance decoding," *IEEE Trans. Inf. Theory*, vol. 57, no. 9, pp. 5933–5945, Sep. 2011.

[27] 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.

[28] G. D. Forney, "Multidimensional constellations. II. Voronoi constellations," *IEEE J. Sel. Areas Commun.*, vol. SAC-7, no. 6, pp. 941–958, Aug. 1989.

[29] F. R. Kschischang and S. Pasupathy, "Optimal nonuniform signaling for Gaussian channels," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 913–929, May 1993.

[30] L. Liu and C. Ling, "Polar codes and polar lattices for independent fading channels," *IEEE Trans. Commun.*, vol. 64, no. 12, pp. 4923–4935, Dec. 2016.

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

[32] L. Liu, Y. Yan, and C. Ling, "Achieving secrecy capacity of the Gaussian wiretap channel with polar lattices," *IEEE Trans. Inf. Theory*, vol. 64, no. 3, pp. 1647–1665, Mar. 2018.

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

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

[35] D. Micciancio and O. Regev, "Worst-case to average-case reductions based on Gaussian measures," in *Proc. 45th Annu. IEEE Symp. Found. Comput. Sci.*, Rome, Italy, Oct. 2004, pp. 372–381.

[36] O. Regev, "On lattice, learning with errors, random linear codes, and cryptography," *J. ACM*, vol. 56, no. 6, pp. 1–40, Sep. 2009.

[37] T. Fehenberger, D. Lavery, R. Maher, A. Alvarado, P. Bayvel, and N. Hanik, "Sensitivity gains by mismatched probabilistic shaping for optical communication systems," *IEEE Photon. Technol. Lett.*, vol. 28, no. 7, pp. 786–789, Apr. 1, 2016.

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

[39] Z. Wang, C. Ling, and G. Hanrot, "Markov chain Monte Carlo algorithms for lattice Gaussian sampling," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Honolulu, HI, USA, Jun. 2014, pp. 1489–1493.

[40] Z. Wang, "Markov chain Monte Carlo methods for lattice Gaussian sampling: Convergence analysis and enhancement," *IEEE Trans. Commun.*, vol. 67, no. 10, pp. 6711–6724, Oct. 2019.

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

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

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

[44] Z. Wang and C. Ling, "On the geometric ergodicity of metropolis-Hastings algorithms for lattice Gaussian sampling," *IEEE Trans. Inf. Theory*, vol. 64, no. 2, pp. 738–751, Feb. 2018.

[45] Z. Wang and C. Ling, "Lattice Gaussian sampling by Markov chain Monte Carlo: Bounded distance decoding and trapdoor sampling," *IEEE Trans. Inf. Theory*, vol. 65, no. 6, pp. 3630–3645, Jun. 2019.

[46] N. Stephens-Davidowitz, "Discrete Gaussian sampling reduces to CVP and SVP," in *Proc. 37th Annu. ACM-SIAM Symp. Discrete Algorithms*, Jan. 2016, pp. 1748–1764.

[47] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proc. 14th Annu. ACM Symp. Theory Comput.*, Victoria, BC, Canada, 2008, pp. 197–206.

[48] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains Mixing Time*. Providence, RI, USA: AMS, 2008.

[49] R. M. Neal, "Slice sampling," *Ann. Statist.*, vol. 31, no. 3, pp. 705–767, Jun. 2003.

[50] A. Mira and L. Tierney, "Efficiency and convergence properties of slice samplers," *Scandin. J. Statist.*, vol. 29, no. 1, pp. 1–12, Mar. 2002.

[51] C. Dubois, A. Korattikara, M. Welling, and P. Smyth, "Approximate slice sampling for Bayesian posterior inference," in *Proc. 17th Int. Conf. Artif. Intell. Statist. (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 Conf. Neural Inf. Process. Syst. (NIPS)*, Barcelona, Spain, 2016, pp. 1741–1749.

[53] J. Besag and P. J. Green, "Spatial statistics and Bayesian computation," *J. Roy. Stat. Soc. B, Methodol.*, vol. 55, no. 1, pp. 25–37, 1993.

[54] J. S. Liu, *Monte Carlo Strategies in Scientific Computing*. New York, NY, USA: Springer-Verlag, 2001.

[55] P. H. Peskun, "Optimal Monte Carlo sampling using Markov chains," *Biometrika*, vol. 60, no. 3, pp. 607–612, 1973.

[56] C. J. Geyer and E. A. Thompson, "Annealing Markov chain Monte Carlo with applications to ancestral inference," *J. Amer. Stat. Assoc.*, vol. 90, no. 431, pp. 909–920, Sep. 1995.

[57] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," *Mathematische Annalen*, vol. 261, no. 4, pp. 515–534, Dec. 1982.

[58] C. Ling and N. Howgrave-Graham, "Effective LLL reduction for lattice decoding," in *Proc. IEEE Int. Symp. Inf. Theory*, Nice, France, Jun. 2007, pp. 196–200.

[59] J. S. Rosenthal, "Minorization conditions and convergence rates for Markov chain Monte Carlo," *J. Amer. Stat. Assoc.*, vol. 90, no. 430, pp. 558–566, Jun. 1995.

[60] K. Łatuszyński, G. O. Roberts, and J. S. Rosenthal, "Adaptive gibbs samplers and related MCMC methods," *Ann. Appl. Probab.*, vol. 23, no. 1, pp. 66–98, Feb. 2013.

[61] C. Andrieu and J. Thoms, "On the ergodicity properties of some adaptive MCMC algorithms," *Ann. Appl. Probab.*, vol. 16, no. 3, pp. 1462–1505, 2006.

[62] L. Martino, J. Read, and D. Luengo, "Independent doubly adaptive rejection metropolis sampling within gibbs sampling," *IEEE Trans. Signal Process.*, vol. 63, no. 12, pp. 3123–3138, Jun. 2015.

[63] Z. Wang, Y. Huang, and S. Lyu, "Lattice-reduction-aided gibbs algorithm for lattice Gaussian sampling: Convergence enhancement and decoding optimization," *IEEE Trans. Signal Process.*, vol. 67, no. 16, pp. 4342–4356, Aug. 2019.

[64] 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.

[65] T. Shimokawa and T. Fujino, "Iterative lattice reduction aided MMSE list detection in MIMO system," in *Proc. Int. Conf. Adv. Technol. Commun.*, Oct. 2008, pp. 50–54.

Zheng Wang (Member, IEEE) received the B.S. degree in electronic and information engineering from the Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China, in 2009, the M.S. degree in communications from the Department of Electrical and Electronic Engineering, University of Manchester, Manchester, U.K., in 2010, and the Ph.D. degree in communication engineering from Imperial College London, U.K., in 2015. From 2015 to 2016 he served as a Research Associate at Imperial College London. From 2016 to 2017, he was a Senior Engineer with the Radio Access Network Research and Development Division, Huawei Technologies Company. Since 2017, he has been an Associate Professor with the College of Electronic and Information Engineering, NUAA. His current research interests include MIMO systems, machine learning and data analytics over wireless networks, and lattice theory for wireless communications.

Ling Liu (Member, IEEE) received the B.S. degree from Nanjing University, Nanjing, China, in 2008, the M.S. degree from Peking University, Beijing, China, in 2012, and the Ph.D. degree from the Electrical and Electronic Engineering Department, Imperial College London, in 2016. From 2017 to 2019, he serves as a Senior Engineer with the CT Lab, Huawei Technologies. He is currently an Assistant Professor with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His research interests include classical and quantum information theory, coding theory, physical-layer security, and lattices.

Cong Ling (Member, IEEE) received the bachelor's and master's degrees from the Nanjing Institute of Communications Engineering, China, in 1995 and 1997, respectively, and the Ph.D. degree from Nanyang Technological University, Singapore, in 2005.

He is currently a Reader (equivalent to Professor/Associate Professor) with the Electrical and Electronic Engineering Department, Imperial College London. He is also a member of the Academic Centre of Excellence in Cyber Security Research, Imperial College, and an affiliated member of the Institute of Security Science and Technology, Imperial College. Before joining Imperial College, he had been on the faculties of the Nanjing Institute of Communications Engineering and Kings College. He visited The Hong Kong University of Science and Technology as a Hong Kong Telecom Institute of Information Technology (HKTIIT) Fellow in 2009. He has been an Associate Editor (in multiterminal communications and lattice coding) of IEEE TRANSACTIONS ON COMMUNICATIONS, and an Associate Editor of IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY and on the program committees of several international conferences, including IEEE Information Theory Workshop, GLOBECOM, and ICC.