Slice Sampling for Lattice Gaussian Distribution

Zheng Wang
College of Electronic and Information Engineering
Nanjing University of Aeronautics and Astronautics
Nanjing, 210000, China
Email: z.wang@ieee.org

Cong Ling
Department of EEE
Imperial College London
London, SW7 2AZ, United Kingdom
Email: cling@ieee.org

***Abstract*—Sampling from the lattice Gaussian distribution has emerged as a key problem in coding and cryptography. In this paper, the slice sampling from Markov chain Monte Carlo (MCMC) is adopted to lattice Gaussian sampling. Firstly, the slice-based sampling algorithm is proposed to sample from lattice Gaussian distribution. Then, we demonstrate that the Markov chain arising from it is uniformly ergodic, namely, it converges exponentially fast to the stationary distribution. Moveover, the convergence rate of the underlying Markov chain is investigated, and we show the proposed slice sampling algorithm entails a better convergence performance than the independent Metropolis-Hastings-Klein (IMHK) sampling algorithm. Finally, simulation results based on 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.**

I. Introduction

Recently, lattice Gaussian distribution has become a common theme in various research fields. In mathematics, Banaszczyk used it to prove the transference theorems for lattices [1]. In coding, it was applied to achieve the full shaping gain for lattice coding [2], and to achieve the capacity of the Gaussian channel and the secrecy capacity of the Gaussian wiretap channel, respectively [3], [4]. 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 [5]. Meanwhile, it also has underpinned the fully-homomorphic encryption for cloud computing [6]. Algorithmically, lattice Gaussian sampling with a suitable variance allows to solve the shortest vector problem (SVP) and the closest vector problem (CVP); for example, it has led to efficient lattice decoding for multi-input multi-output (MIMO) systems [7].

Due to the central role of the lattice Gaussian distribution playing in these fields, its sampling algorithms become an important computational problem. Unfortunately, compared to sampling from continuous Gaussian distributions, it is by no means trivial to perform the sampling even from a low-dimensional discrete Gaussian distribution. To address this issue, the Gibbs sampling from Markov chain Monte Carlo (MCMC) methods was firstly introduced to sample from lattice Gaussian distribution by the evolution of Markov moves [8]. Moreover, the geometric ergodicity of Gibbs sampling for lattice Gaussian distribution was also demonstrated, implying an exponential convergence decay [9]. On the other hand, in [10], the independent Metropolis-Hastings-Klein (IMHK) sampling algorithm is proposed, which not only experiences uniform ergodicity but also entails an accessible convergence rate.

As a foremost sampling scheme in MCMC, Metropolis-Hastings (MH) algorithm takes advantage of a proposal distribution which suggests a possible move and employs an acceptance-rejection rule to make the decision for the Markov move. In this paper, to further improve the convergence performance, the slice sampling is introduced to sample from lattice Gaussian distribution. In particular, based on IMHK, auxiliary variables are employed in slice sampling to speed up the convergence rate. Moreover, the Markov chain induced by the proposed slice sampling is demonstrated to be uniformly ergodic while its convergence rate is proven to be superior to that of IMHK, making it an advanced choice for lattice Gaussian sampling.

II. IMHK Sampling for 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

$$\Lambda = \{\mathbf{Bx} : \mathbf{x} \in \mathbb{Z}^n\}, \tag{1}$$

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

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

for all $\mathbf{x} \in \mathbb{Z}^n$, 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 [11], 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

$$P_{\text{Klein}}(\mathbf{x}) = \prod_{i=1}^{n} D_{\mathbb{Z},\sigma_i,\widetilde{x}_i}(x_i) = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{Bx})}{\prod_{i=1}^{n}\rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})}. \quad (4)$$

In [12], $P_{\text{Klein}}(\mathbf{x})$ has been demonstrated to be close to $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ within a negligible statistical distance if

$$\sigma \geq \omega(\sqrt{\log\, n}) \cdot \max_{1\leq i\leq n}\|\widehat{\mathbf{b}}_i\|. \quad (5)$$

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

On the other hand, 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 [10]. 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:

$$q(\mathbf{x},\mathbf{y}) = P_{\text{Klein}}(\mathbf{y}) = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{By})}{\prod_{i=1}^{n}\rho_{\sigma_i,\widetilde{y}_i}(\mathbb{Z})}, \quad (6)$$

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

$$\begin{aligned}\alpha(\mathbf{x},\mathbf{y}) &= \min\left\{1, \frac{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})}\right\} \\ &= \min\left\{1, \frac{\prod_{i=1}^{n}\rho_{\sigma_i,\widetilde{y}_i}(\mathbb{Z})}{\prod_{i=1}^{n}\rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})}\right\}, \quad (7)\end{aligned}$$

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.

Theorem 1 ([10]). *Given the invariant lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}$, the Markov chain established by the IMHK algorithm is uniformly ergodic:*

$$\|P^t(\mathbf{x},\cdot) - D_{\Lambda,\sigma,\mathbf{c}}(\cdot)\|_{TV} \leq (1-\delta)^t \quad (8)$$

with

$$\delta = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{\Lambda})}{\prod_{i=1}^{n}\rho_{\sigma_i}(\mathbb{Z})} \quad (9)$$

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 $\gamma = 1-|\lambda_{\max}|$, where $|\lambda_{\max}| \neq 1$ denotes the largest eigenvalue of the transition matrix [13].

III. Slice Sampling for Lattice Gaussian Distribution

In this section, we present the conventional slice sampling in MCMC and give the proposed slice-based sampling algorithm for lattice Gaussian distribution. Note that the Markov chain that we are concerned with here has a countably infinite state space, i.e., the lattice Λ with $\mathbf{x} \in \mathbb{Z}^n$.

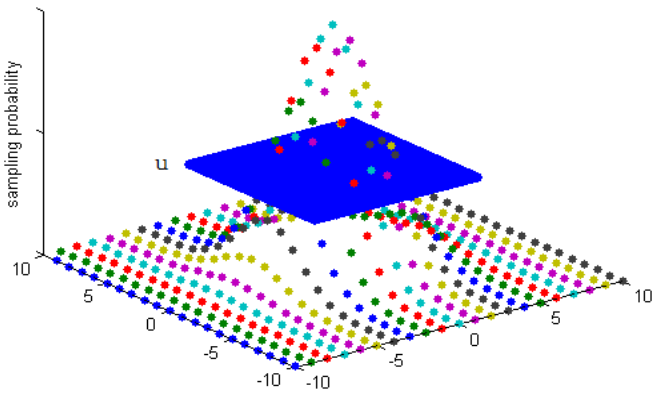

Fig. 1. Illustration of a two-dimensional lattice Gaussian distribution and a slice (blue plane) with $u \geq 0$ over it.

A. Slice Sampling

The classical slice sampling was generalized by Neal in [14]. 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. To summarize, it follows that

1) *Sample u_t from the conditional distribution*

$$p(u_t \mid \mathbf{x}_{t-1}) \sim Uni(0,\pi(\mathbf{x}_{t-1})). \quad (10)$$

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

$$p(\mathbf{x}_t \mid u_t) \sim Uni(S_u), \quad (11)$$

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 (16) 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. Proposed Sampling Algorithm

We now present the proposed 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

$$\pi(\mathbf{x}) = D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x}) = P_{\text{Klein}}(\mathbf{x}) \cdot l(\mathbf{x}) \tag{12}$$

with

$$l(\mathbf{x}) \triangleq \frac{\prod_{i=1}^{n} \rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})}{\rho_{\sigma,\mathbf{c}}(\Lambda)}, \tag{13}$$

we can establish the joint distribution as

$$\Pi(\mathbf{x}, u) = P_{\text{Klein}}(\mathbf{x}) \cdot I_{u<l(\mathbf{x})}(\mathbf{x}), \tag{14}$$

where $I_A(\mathbf{x})$ is the indicator function of the set A. More precisely, 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*

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

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

$$p(\mathbf{x}_t \mid u_t) \sim P_{\text{Klein}}^{A_{u_t}}(\mathbf{x}), \tag{16}$$

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

Intuitively, 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 $\mathbf{x}_t$. Interestingly, the numerator in (13) has already calculated by Klein's algorithm during the sampling, which means a low computational cost by incurring rejection sampling. Note that different from IMHK, the acceptance-rejection mechanism is absent in the proposed slice sampling, which naturally leads to complexity reduction and efficiency enhancement. We also 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, which could be further investigated in future.

Here, the factorization based on Klein's probability is applied, which offers a feasible way to merge $l(\mathbf{x})$ together with the uniform sampling of u. Clearly, Klein's distribution $P_{\text{Klein}}(\cdot)$ is unimodal, making it easy in tackling with the slice interval problem. Meanwhile, compared to the target distribution $D_{\Lambda,\sigma,\mathbf{c}}(\cdot)$, $P_{\text{Klein}}(\cdot)$ is heavier-tailed. This is helpful to avoid getting stuck in the tails for long periods since $l(\mathbf{x}_t)$ will be small and set A_{u_t} will have large probability, thus resulting in a small rejection rate. Also, for convenience, the systematic update scheme that updates $\mathbf{x}$ and u sequentially is considered through the context.

Algorithm 1 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 (13)
    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 (4)
        calculate l(x_t) according to (13)
        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

A. Uniform Ergodicity

Consider the marginal distribution $\pi(\mathbf{x}) = D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ with respect to the mixed type distribution $\Pi(\mathbf{x}, u)$, clearly, such a marginal chain $\{\mathbf{X}_1, \mathbf{X}_2, ...\}$ regarding to $\mathbf{x}$ is still a valid Markov chain, which is reversible (also known as detail balance) due to

$$\begin{aligned}
\pi(\mathbf{x}_t)P(\mathbf{x}_t, \mathbf{x}_{t+1}) &= \pi(\mathbf{x}_t) \int \Pi(u_{t+1}|\mathbf{x}_t)\Pi(\mathbf{x}_{t+1}|u_{t+1})du_{t+1} \\
&= \int \Pi(\mathbf{x}_t|u_{t+1})\Pi(u_{t+1}|\mathbf{x}_t)\Pi(\mathbf{x}_{t+1}|u_{t+1})du_{t+1} \\
&= \pi(\mathbf{x}_{t+1}) \int \Pi(u_{t+1}|\mathbf{x}_t)\Pi(\mathbf{x}_t|u_{t+1})du_{t+1} \\
&= \pi(\mathbf{x}_{t+1}) \int \Pi(u_t|\mathbf{x}_{t+1})\Pi(\mathbf{x}_t|u_t)du_t \\
&= \pi(\mathbf{x}_{t+1})P(\mathbf{x}_{t+1}, \mathbf{x}_t).
\end{aligned} \tag{17}$$

Based on the sub-Markov chain $\{\mathbf{X}_1, \mathbf{X}_2, ...\}$, its transition probability can be derived as

$$\begin{aligned}
P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) &= \int p(\mathbf{x}_{t+1}|u_{t+1})p(u_{t+1}|\mathbf{x}_t)du_{t+1} \\
&= \int P_{\text{Klein}}^{A_{u_{t+1}}}(\mathbf{x}_{t+1})p(u_{t+1}|\mathbf{x}_t)du_{t+1} \\
&= \frac{1}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)} P_{\text{Klein}}^{A_{u_{t+1}}}(\mathbf{x}_{t+1})du_{t+1} \\
&\overset{(a)}{=} \frac{1}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)} \frac{P_{\text{Klein}}(\mathbf{x}_{t+1})I_{u_{t+1}<l(\mathbf{x}_{t+1})}(\mathbf{x}_{t+1})}{P_{\text{Klein}}(A_{u_{t+1}})}du_{t+1} \\
&= \frac{P_{\text{Klein}}(\mathbf{x}_{t+1})}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)\wedge l(\mathbf{x}_{t+1})} \frac{1}{P_{\text{Klein}}(A_{u_{t+1}})}du_{t+1} \\
&= \frac{P_{\text{Klein}}(\mathbf{x}_{t+1})}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)\wedge l(\mathbf{x}_{t+1})} \beta du_{t+1}
\end{aligned} \tag{18}$$

where $\beta \triangleq 1/P_{\text{Klein}}(A_{u_{t+1}})$, (a) recalls *Bayes' theorem* and "$\wedge$" yields the smaller choice between two terms.

Insight into β, it can be further expressed as

$$\beta = \frac{1}{\sum_{\mathbf{x}\in\{\mathbf{x}:l(\mathbf{x})>u_{t+1}\}} P_{\text{Klein}}(\mathbf{x})}. \tag{19}$$

Intuitively, $\beta = 1$ happens if and only if u_{t+1} is selected to be 0. However, since the term $l(\mathbf{x}_t) \wedge l(\mathbf{x}_{t+1})$ in the integration in (18) is lower bounded by

$$l(\mathbf{x}) \geq \frac{\prod_{i=1}^{n} \rho_{\sigma_i,1/2}(\mathbb{Z})}{\rho_{\sigma,\mathbf{c}}(\Lambda)}, \tag{20}$$

for all $\mathbf{x} \in \mathbb{Z}^n$, the following relationship holds

$$\int_0^{l(\mathbf{x}_t)\wedge l(\mathbf{x}_{t+1})} \beta du_{t+1} > \int_0^{l(\mathbf{x}_t)\wedge l(\mathbf{x}_{t+1})} du_{t+1}. \tag{21}$$

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

$$\begin{aligned} P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) &> \frac{P_{\text{Klein}}(\mathbf{x}_{t+1})}{l(\mathbf{x}_t)} \int_0^{l(\mathbf{x}_t)\wedge l(\mathbf{x}_{t+1})} du_{t+1} \\ &= P_{\text{Klein}}(\mathbf{x}_{t+1}) \left[1 \wedge \frac{l(\mathbf{x}_{t+1})}{l(\mathbf{x}_t)}\right] \\ &= \left[P_{\text{Klein}}(\mathbf{x}_{t+1}) \wedge \frac{\pi(\mathbf{x}_{t+1}) P_{\text{Klein}}(\mathbf{x}_t)}{\pi(\mathbf{x}_t)}\right] \\ &= P_{\text{Klein}}(\mathbf{x}_{t+1}) \cdot \alpha(\mathbf{x}_t, \mathbf{x}_{t+1}) \\ &= P_{\text{IMHK}}(\mathbf{x}_t, \mathbf{x}_{t+1}) \\ &\overset{(b)}{\geq} \delta \cdot \pi(\mathbf{x}_{t+1}), \end{aligned} \tag{22}$$

where the inequality (b) follows the fact that [5]

$$\frac{P_{\text{Klein}}(\mathbf{x})}{\pi(\mathbf{x})} = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{\Lambda})}{\prod_{i=1}^{n} \rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})} \geq \delta \tag{23}$$

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

Actually, $P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) > \delta \cdot \pi(\mathbf{x}_{t+1})$ for all the Markov state is accordance with the definition of *small set* in literatures of MCMC [13]. Furthermore, given (22), for a reversible Markov chain, it is straightforward to demonstrate its *uniform ergodicity* of the underlying Markov chain through *coupling technique*. Here, for simplicity, the related proof is omitted while more details about the proof can be found in [10], [15].

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 slice sampling algorithm is uniformly ergodic as:*

$$\|P^t(\mathbf{x}, \cdot) - D_{\Lambda,\sigma,\mathbf{c}}(\cdot)\|_{TV} < (1-\delta)^t \tag{24}$$

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

B. Convergence Improvement

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

$$P_{\text{Slice}}(\mathbf{x}_t, \mathbf{x}_{t+1}) > P_{\text{IMHK}}(\mathbf{x}_t, \mathbf{x}_{t+1}). \tag{25}$$

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

$$P_{\text{Slice}}(\mathbf{X}_t{=}\mathbf{x},\mathbf{X}_{t+1}{=}\mathbf{y}) > P_{\text{IMHK}}(\mathbf{X}_t = \mathbf{x},\mathbf{X}_{t+1} = \mathbf{y}) \tag{26}$$

for $\mathbf{x} \neq \mathbf{y}$, which means each off-diagonal element in transition matrix $\mathbf{P}_{\text{Slice}}$ is always larger than that of $\mathbf{P}_{\text{IMHK}}$. From literatures of MCMC, such a case is known as *Peskun ordering* written by

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

We then invoke the following Theorem to show the convergence performance from Peskun ordering.

Theorem 3 ([16])**.** *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*

$$v(f, \mathbf{P}_1) \geq v(f, \mathbf{P}_2). \tag{28}$$

Here, $L^2(\pi)$ denotes 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 $\mathbf{X}_0, \ldots, \mathbf{X}_t$ establish the corresponding Markov chain. Clearly, from Theorem 3, the proposed slice 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 [17], Mira shows that if two transition matrices are Peskun ordered, then the corresponding eigenvalues are also ordered, i.e.,

$$|\lambda_{\text{max},1}| \geq |\lambda_{\text{max},2}|, \tag{30}$$

where convergence rate in uniform ergodicity is exactly characterized by the largest eigenvalue $|\lambda_{\text{max}}| \neq 1$. Therefore, we can easily arrive at the following result to show the convergence gain of the proposed slice sampling.

Corollary 1. *The proposed slice sampling algorithm is more efficient than the IMHK algorithm due to a better convergence rate by*

$$|\lambda_{\text{max}}|_{\text{Slice}} \leq |\lambda_{\text{max}}|_{\text{IMHK}} \leq 1 - \delta \tag{31}$$

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

Hence, the mixing time of the Markov chain induced by

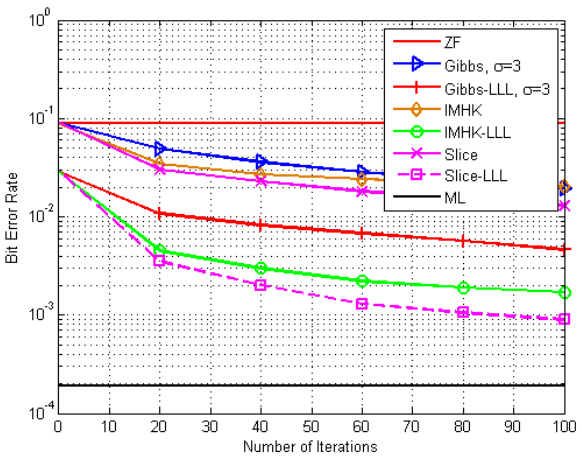

Fig. 2. Bit error rate versus the number of Markov moves for the uncoded 8×8 MIMO system using 16-QAM.

slice sampling is given by

$$t_{\text{mix}}(\epsilon) = \frac{\ln \epsilon}{\ln |\lambda_{\max}|_{\text{Slice}}} < (-\ln \epsilon) \cdot \left(\frac{1}{1 - |\lambda_{\max}|_{\text{Slice}}} \right), \quad \epsilon < 1 \tag{32}$$

where we use the bound $\ln c < c - 1$ for $0 < c < 1$.

From (18), the superiority of the proposed slice sampling over IMHK is partially determined by $\beta > 1$. More specifically, 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.

V. Simulations

In this section, the performances of MCMC-based sampling schemes are exemplified in the context of MIMO decoding. By sampling from $D_{\Lambda,\sigma,\mathbf{c}}$, the closest lattice point $\mathbf{Hx}$ in MIMO detection problem

$$\mathbf{c} = \mathbf{Hx} + \mathbf{w}. \tag{33}$$

will be returned with the highest probability, which implies an effective approach to lattice decoding. More precisely, when MCMC method is applied for sampler decoding, its decoding performance can be evaluated by CVP decoding complexity (i.e., the number of Markov move), which is defined by [15]

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

where a smaller C_{cvp} naturally corresponds to a better decoding performance. Because of this, we examine the decoding error probabilities to assess the convergence rates. 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.

In Fig. 2, the 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. The SNR is fixed as $E_b/N_0 = 15$ dB. The standard deviation is set as $\sigma = \min_i \|\widehat{\mathbf{b}}_i\|/(2\sqrt{\pi})$ for both slice sampling and IMHK sampling while we apply $\sigma = 3$ for Gibbs sampling (The performance of $\sigma = 2$ for Gibbs sampling can be found in [15], which is not as good as that of $\sigma = 3$). Clearly, the performances of all the MCMC detectors improve with the number of Markov moves. With the increasing number of trial samples, better decoding performance can be obtained by the proposed slice sampling algorithm. Here, LLL reduction is applied to output the better initial state for Gibbs sampler. As for IMHK and slice samplers, LLL can be further adopted to the Markov moves for a better decoding performance.

Acknowledgment

This work was supported in part by the open research fund of National Mobile Communications Research Laboratory, Southeast University (No. 2019D04), National Natural Science Foundation of China (Grants No. 61801216), Natural Science Foundation of Jiangsu Province under Grant BK20180420.

References

[1] W. Banaszczyk, "New bounds in some transference theorems in the geometry of numbers," *Math. Ann.*, vol. 296, pp. 625–635, 1993.
[2] 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.
[3] 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.
[4] 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.
[5] 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.
[6] C. Gentry, "Fully homomorphic encryption using ideal lattices," *STOC*, pp. 169–178, 2009.
[7] 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.
[8] 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.
[9] Z. Wang and C. Ling, "On the geometric ergodicity of Gibbs algorithm for lattice Gaussian sampling," in *Proc. IEEE Information Theory Workshop (ITW)*, 2017, pp. 269–273.
[10] ——, "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.
[11] P. Klein, "Finding the closest lattice vector when it is unusually close," in *ACM-SIAM Symp. Discr. Algorithms*, 2000, pp. 937–941.
[12] 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.
[13] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Time*, American Mathematical Society, 2008.
[14] R. Neal, "Slice sampling," *Ann. Statist.*, vol. 31, pp. 705–767, 2003.
[15] Z. Wang and C. Ling, "Lattice Gaussian sampling by Markov chain Monte Carlo: Bounded distance decoding and trapdoor sampling," *IEEE Transactions on Information Theory*, pp. 1–1, 2019.
[16] P. H. Peskun, "Optimal Monte Carlo sampling using Markov chains," *Biometrika*, vol. 60, pp. 607–612, 1973.
[17] A. Mira and L. Tierney, "Efficiency and convergence properties of slice samplers," *Scandinavian Journal of Statistics*, vol. 29, pp. 1–12, 2002.