Markov Chain Monte Carlo Methods for Lattice Gaussian Sampling: Convergence Analysis and Enhancement

Zheng Wang, *Member, IEEE*

***Abstract*— Sampling from lattice Gaussian distribution has emerged as an important problem in coding, decoding, and cryptography. In this paper, the classic Gibbs algorithm from Markov chain Monte Carlo (MCMC) methods is demonstrated to be geometrically ergodic for lattice Gaussian sampling, which means that the Markov chain arising from it converges exponentially fast to the stationary distribution. Meanwhile, the exponential convergence rate of the Markov chain is also derived through the spectral radius of the forward operator. Then, a comprehensive analysis of the convergence rate is carried out, and two sampling schemes are proposed to further enhance the convergence performance. The first one, referred to as a Metropolis-within-Gibbs (MWG) algorithm, improves the convergence by refining the state space of the univariate sampling. The second is a blocked strategy of the Gibbs algorithm, which performs sampling over multivariates at each Markov move, and is shown to yield a better convergence rate than the traditional univariate sampling. In order to perform blocked sampling efficiently, the Gibbs–Klein (GK) algorithm is proposed, which samples block by block using the Kleins algorithm. Furthermore, the validity of the GK algorithm is demonstrated by showing its ergodicity. Simulation results based on MIMO detections are presented to confirm the convergence gain brought by the proposed Gibbs sampling schemes.**

***Index Terms*— Lattice coding and decoding, lattice Gaussian sampling, Gibbs sampler decoding, Markov chain Monte Carlo, MIMO detection.**

I. Introduction

NOWADAYS, lattice Gaussian distribution, which is supported over a multi-dimensional Euclidean lattice, has drawn a lot of attentions in various research fields. In mathematics, Banaszczyk first applied it to prove the transference theorems for lattices [1]. In coding, lattice Gaussian distribution was employed to obtain the full shaping gain for lattice coding [2]–[4], and to achieve the capacity of the Gaussian channel [5]. Meanwhile, it was also used to achieve information-theoretic security in Gaussian wiretap channel [6]–[8]. Furthermore, lattice Gaussian distribution has been adapted to bidirectional relay network under the compute-and-forward strategy for the physical layer security [9]. Additionally, it is also applied to realize the probabilistic shaping for optical communication systems [10], [11].

In cryptography, lattice Gaussian distribution has already become a central tool in the construction of many primitives. Specifically, Micciancio and Regev used it to propose lattice-based cryptosystems based on the worst-case hardness assumptions [12]. In [13], lattice Gaussian distribution is applied to create even more powerful cryptographic primitives, namely, hierarchical identity-based encryption and standard model signatures. In learning with errors (LWE) based encryption, sampling from lattice Gaussian distribution is highly demanded for the key generation [14]. Meanwhile, it also has underpinned the fully-homomorphic encryption for cloud computing [15]. Besides, there are various applications that require sampling over lattice Gaussian distribution as part of the “on-line” computation, where the most notable one among them is the secure lattice-based digital signature on a constrain device [16].

On the other hand, algorithmically, lattice Gaussian distribution with a suitable variance allows lattice decoding to solve the shortest vector problem (SVP) and the closest vector problem (CVP) [17], [18]. Intuitively, the formulation of it comes from a conceptually simple fact that each lattice point in the discrete Gaussian distribution entails a sampling probability scaled by the Euclidean distance from the query point [19]. The lattice points which are close to the center of the distribution naturally correspond to large sampling probabilities. Therefore, the desired closest lattice point or shortest lattice vector would conceivably be obtained due to the largest sampling probability. To this end, sampling over lattice Gaussian distribution has been widely applied in multi-input multi-output (MIMO) communications for signal detection [20]–[22]. Compared to the optimal sphere detection, it is not only much more efficient, but also can be realized

Manuscript received October 31, 2018; revised April 13, 2019 and June 19, 2019; accepted June 23, 2019. Date of publication July 2, 2019; date of current version October 16, 2019. This work was supported in part by the open research fund of Key Laboratory of Dynamic Cognitive System of Electromagnetic Spectrum Space (Nanjing Univ. Aeronaut. Astronaut.), Ministry of Industry and Information Technology, Nanjing, 211106, China (No. KF20181913), the open research fund of National Mobile Communications Research Laboratory, Southeast University (No. 2019D04), the National Natural Science Foundation of China under Grant 61801216, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20180420. This paper was presented in part at the IEEE Information Theory Workshop (ITW), Kaohsiung, Taiwan, November, in 2017, and in part at the IEEE International Symposium on Information Theory (ISIT), Honolulu, USA, June 2014. The associate editor coordinating the review of this paper and approving it for publication was G. Durisi.

The author is with the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing 211106, China (e-mail: z.wang@ieee.org).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCOMM.2019.2926470

flexibly to achieve the trade-off between decoding performance and complexity [23]. In addition, such a sampling decoding strategy can be easily extended to signal processing as an useful signal estimator or detector [24]–[28].

Because of the central role of lattice Gaussian distribution in these fields, its sampling algorithms become an important computational problem. However, different from the case of continuous Gaussian density, sampling from the discrete lattice Gaussian distribution is by no means trivial even for a low-dimensional system. For this reason, Markov chain Monte Carlo (MCMC) methods were introduced as an alternative way for lattice Gaussian sampling, which attempts to sample from the target distribution by building a Markov chain [29], [30]. Typically, after a burn-in stage, which is normally measured by the *mixing time* in total variance distance, the Markov chain will step into a stationary distribution, where samples from the target distribution can be successfully obtained thereafter. Specifically, in [30], Gibbs algorithm was introduced for lattice Gaussian sampling by showing its ergodicity, which employs conditional univariate sampling to build the Markov chain. Nevertheless, ergodicity only guarantees the convergence while the way of convergence (e.g., polynomial convergence, geometric convergence and so on) as well as the related convergence rate are unclear, resulting in an untractable Markov chain. In fact, compared to Gibbs algorithm for lattice Gaussian distribution, a better progress has been made with respect to Metropolis-Hastings (MH) algorithm, which is well known as another important sampling scheme in MCMC. For example, the independent Metropolis-Hastings-Klein (IMHK) algorithm given in [29] is not only uniformly ergodic for lattice Gaussian sampling, but also has an accessible convergence rate.

This paper was partially presented in [30] and [31] while further investigation and extensions are given as follows. On one hand, with respect to the geometric ergodicity of Gibbs algorithm for lattice Gaussian sampling [31], a prospective way for convergence diagnosis by means of the spectral radius of the forward operator is offered. Inspired by it, convergence analysis is carried out in this paper, where the corresponding enhancement scheme named as Metropolis-within-Gibbs (MWG) algorithm is proposed for univariate Gibbs sampling. More importantly, the superiority of MWG over Gibbs algorithm in terms of convergence rate is demonstrated, and further improvement can be realized by the parallel tempering technique. On the other hand, different from the work in [30] which only concerns the efficient implementation for the blocked strategy of Gibbs algorithm regardless of the convergence behavior, the blocked strategy by sampling over multivariate is demonstrated to enable a faster convergence rate than the univariate sampling. Moreover, the geometric ergodicity of the proposed Gibbs-Klein (GK) algorithm is also given, which removes the approximation errors by resorting to the rejection sampling. Hence, a whole framework of Gibbs-based algorithms for lattice Gaussian sampling is established.

The rest of this paper is organized as follows. Section II introduces the lattice Gaussian distribution and briefly reviews the basics of MCMC methods. In Section III, Gibbs algorithm is introduced for lattice Gaussian sampling, and its geometric ergodicity is demonstrated. In Section IV, the MWG algorithm is proposed to strengthen the convergence performance in terms of the univariate sampling, followed by the proof to show the convergence enhancement. In Section V, blocked strategy is adopted to Gibbs algorithm to achieve a better convergence rate. In order to realize efficient blocked sampling, GK algorithm is proposed and the proof of its validity is also given. Simulations through MIMO systems are presented in Section VI to illustrate the convergence gain of these two proposed algorithms. At the end, Section VII concludes the paper.

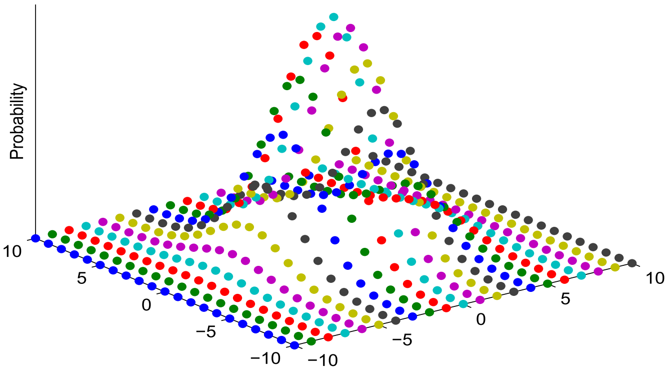

Fig. 1. Illustration of a two-dimensional lattice Gaussian distribution.

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 $\mathbf{B}$, $\widehat{\mathbf{b}}_i$ for the ith Gram-Schmidt vector of the matrix $\mathbf{B}$, $b_{i,j}$ for the entry in the ith row and jth column of the matrix $\mathbf{B}$. We use the standard *small omega* notation $\omega(\cdot)$, i.e., $f(n) = \omega(g(n))$ if for any $k > 0$, the inequality $|f(n)| > k \cdot |g(n)|$ holds for all sufficiently large n. Finally, $h \in L_0^2(\pi)$ and $L_0^2(\pi)$ denote the set of all mean zero and finite variance functions with respect to the target distribution π, i.e., $E_\pi[h(\mathbf{x})] = 0$ and $\text{var}_\pi[h(\mathbf{x})] = v < \infty$.

II. Preliminaries

In this section, the background and mathematical tools needed to describe and analyze the Gibbs algorithm for lattice Gaussian sampling are introduced.

A. Lattice Gaussian Distribution

Let $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_n] \subset \mathbb{R}^n$ consist of n linearly independent 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 the full rank matrix $\mathbf{B} \in \mathbb{R}^{n\times n}$ is called the lattice basis. The *Gaussian function* centered at $\mathbf{c} \in \mathbb{R}^n$ with standard deviation $\sigma > 0$ is defined as

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

Algorithm 1 Klein's Algorithm

```
Input: $\mathbf{B}, \sigma, \mathbf{c}$
Output: $\mathbf{Bx} \in \Lambda$
let $\mathbf{B} = \mathbf{QR}$ and $\mathbf{c}' = \mathbf{Q}^T\mathbf{c}$
for $i = n, \ldots, 1$ do
    let $\sigma_i = \frac{\sigma}{|r_{i,i}|}$ and $\widetilde{x}_i = \frac{c'_i - \sum_{j=i+1}^{n} r_{i,j}x_j}{r_{i,i}}$
    sample $x_i$ from $D_{\mathbb{Z},\sigma_i,\widetilde{x}_i}$
end for
return $\mathbf{Bx}$
```

for all $\mathbf{z} \in \mathbb{R}^n$. When $\mathbf{c}$ or σ are not specified, they are assumed to be $\mathbf{0}$ 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 and $\sigma > 0$ represents the standard deviation.

B. Klein's Algorithm

At present, the default sampling algorithm for lattice Gaussian distribution is due to Klein, which was originally proposed for bounded-distance decoding (BDD) in lattices [19]. As shown in Algorithm 1, the operation of Klein's algorithm has polynomial complexity $O(n^2)$ excluding QR decomposition. More precisely, 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 forward order from x_1 to x_n is fine as well), the Gaussian-like distribution arising from Klein's algorithm is given by

$$\begin{aligned} P_{\text{Klein}}(\mathbf{x}) &= \prod_{i=1}^{n} D_{\mathbb{Z},\sigma_i,\widetilde{x}_i}(x_i) = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{Bx})}{\prod_{i=1}^{n} \rho_{\sigma_i,\widetilde{x}_i}(\mathbb{Z})} \\ &= \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{Bx}-\mathbf{c}\|^2}}{\prod_{i=1}^{n} \sum_{\widetilde{x}_i\in\mathbb{Z}} e^{-\frac{1}{2\sigma_i^2}\|x_i-\widetilde{x}_i\|^2}}, \end{aligned} \tag{4}$$

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 $\mathbf{B} = \mathbf{QR}$ and $\widehat{\mathbf{b}}_i$'s are the Gram-Schmidt vectors of $\mathbf{B}$ with $\|\widehat{\mathbf{b}}_i\| = |r_{i,i}|$.

In [32], it has been demonstrated that $P_{\text{Klein}}(\mathbf{x})$ is close to $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ within a negligible statistical distance if

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

However, even with the help of lattice reduction,[1] the requirement of standard deviation $\omega(\sqrt{\log n}) \cdot \max_{1\leq i\leq n} \|\widehat{\mathbf{b}}_i\|$ is too large to be useful in practice, rendering Klein's algorithm inapplicable to many cases of interest.

C. MCMC Methods

By establishing a Markov chain that randomly generates the next state, MCMC is capable of sampling from the target distribution of interest. As an important parameter which measures the time (i.e., number of Markov moves) required by a Markov chain to get close to its stationary distribution, the *mixing time* $t_{\text{mix}}(\epsilon)$ is defined as [35]

$$t_{\text{mix}}(\epsilon) = \min\{t : \max \|P^t(\mathbf{x}, \cdot) - \pi(\cdot)\|_{TV} \leq \epsilon\}, \tag{6}$$

where the integer $t \geq 1$ denotes the index of Markov moves, $\|\cdot\|_{TV}$ represents the total variation distance, π is the target invariant distribution, $P^t(\mathbf{x}; \cdot)$ indicates a row of the transition matrix $\mathbf{P}$ after t Markov moves with the initial state $\mathbf{x}$.[2] We now give the descriptions of the basic properties that are important to Markov chains, where the state space of the Markov chain is denoted by $\mathbf{x} \in \Omega$.

1) *Irreducible*: For any two states $s_i, s_j \in \Omega$, there exists a positive integer k such that $P(\mathbf{X}^{t+k} = s_j|\mathbf{X}^t = s_i) > 0$.
2) *Aperiodic*: For any two states $s_i, s_j \in \Omega$, the Markov chain is not forced into any cycle with fixed period between them. In other words, the period of any two states that communicate is the same, e.g., gcd$\{k : P(\mathbf{X}^{t+k} = s_j|\mathbf{X}^t = s_i) > 0\} = 1$, where "gcd" represents the greatest common divisor.
3) *Reversible*: For any two states $s_i, s_j \in \Omega$, $\pi(s_i)P(\mathbf{X}^{t+1} = s_j|\mathbf{X}^t = s_i) = \pi(s_j)P(\mathbf{X}^{t+1} = s_i|\mathbf{X}^t = s_j)$ always holds.

Thanks to the celebrated *coupling technique*, for any Markov chain with finite state space, exponentially fast convergence can be demonstrated if the underlying Markov chain is irreducible and aperiodic with an invariant distribution π [35]. However, in the case of lattice Gaussian sampling, the countably infinite state space $\mathbf{x} \in \mathbb{Z}^n$ naturally imposes a challenge, making us consider the convergence analysis from the ergodicity.

Definition 1 ([36]): Let $\mathbf{P}$ *be an irreducible and aperiodic transition matrix for a Markov chain. If the chain is reversible (hence positive recurrent), then the Markov chain is said to be ergodic, which is defined as* $\lim_{t\to\infty} \|P^t(\mathbf{x}, \cdot) - \pi\|_{TV} = 0$ *for all* $\mathbf{x} \in \Omega$.

Note that the state space Ω in the above definition can be countably infinite, which offers a valid way to verify the ergodicity. Unless stated otherwise, the state space we are concerned with throughout the context is the countably infinite $\Omega = \mathbb{Z}^n$. Although *ergodicity* implies asymptotic convergence to stationarity, it does not entail the way of convergence, resulting in an intractable Markov chain [37]. Among kinds of ergodicity in literature, *geometric ergodicity* which converges exponentially is defined as:

Definition 2 ([35]): A Markov chain with stationary distribution π *is geometrically ergodic if there exists* $0 < \varrho < 1$ *and* $0 < M(\mathbf{x}) < \infty$ *such that* $\|P^t(\mathbf{x}, \cdot) - \pi(\cdot)\|_{TV} \leq M(\mathbf{x})\varrho^t$ *for all* $\mathbf{x}$ *with* $t \geq 1$*, where function* $M(\mathbf{x})$ *is parameterized by the initial state* $\mathbf{x}$.

[1] It is well known that lattice reduction such as the Lenstra-Lenstra-Lovász (LLL) algorithm is able to significantly improve $\min_i \|\widehat{\mathbf{b}}_i\|$ while reducing $\max_i \|\widehat{\mathbf{b}}_i\|$ at the same time [33], [34].

[2] The (i, j)-th entry $P(i; j)$ of transition matrix $\mathbf{P}$ represents the probability of transferring to state j from the previous state i

Clearly, coefficient ϱ is the convergence rate of the Markov chain. In comparison, a Markov chain is said to be uniformly ergodic if it is geometrically ergodic and $M(\mathbf{x})$ is a constant M independent of $\mathbf{x}$ [36].

III. Gibbs Algorithm for Lattice Gaussian Sampling

In this section, Gibbs algorithm is introduced for lattice Gaussian sampling, which establishes the Markov chain through univariate sampling.

A. Ergodicity

Typically, as for lattice Gaussian sampling by Gibbs algorithm, each coordinate of $\mathbf{x}$ is sampled from the following 1-dimensional conditional distribution

$$P(x_i|\mathbf{x}_{[-i]})=D_{\Lambda,\sigma,\mathbf{c}}(x_i|\mathbf{x}_{[-i]})=\frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{Bx}-\mathbf{c}\|^2}}{\sum_{x_i\in\mathbb{Z}}e^{-\frac{1}{2\sigma^2}\|\mathbf{Bx}-\mathbf{c}\|^2}} \tag{7}$$

with $\sigma > 0$. Here $1 \leq i \leq n$ denotes the coordinate index of $\mathbf{x}$, $\mathbf{x}_{[-i]} \triangleq [x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n]^T$. During this univariate sampling, the other $n-1$ variables contained in $\mathbf{x}_{[-i]}$ are keeping fixed. By repeating such a procedure with a certain scan scheme, a Markov chain $\{\mathbf{X}^0, \mathbf{X}^1, \ldots\}$ is established. Here, for simplicity, we use κ_i to denote the lower bound of the variance of each random variable x_i

$$\text{var}_P[x_i|\mathbf{x}_{[-i]}] \geq \kappa_i > 0^3 \tag{8}$$

for $1 \leq i \leq n$. This is easy to understand since x_i is no longer a random variable if it is completely determined by $\mathbf{x}_{[-i]}$ once $\text{var}[x_i|\mathbf{x}_{[-i]}] = 0$, which is also contradictory with the default setting of $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$.[4] Therefore, all the sampling candidates of $x_i \in \mathbb{Z}$ are possible to be sampled, leading to an irreducible chain that is free to communicate between any of two states. Meanwhile, it is straightforward to verify that every state is aperiodic, which results in a aperiodic Markov chain.

On the other hand, in Gibbs algorithm, there are various scan schemes to proceed the component updating. Among them, random scan is the basic one. Typically, under random scan, the coordinate index i is randomly chosen from a set of selection probabilities $[\beta_1, \ldots, \beta_n]$, where $\sum_{i=1}^n \beta_i = 1$ and $\beta_i > 0$. The extension to other scan strategies is possible. Without loss of generality, the random scan scheme is considered for Gibbs algorithm throughout the context and flexible implementation based on it can be easily carried out in practice. Therefore, the transition probability $P(\mathbf{X}^t, \mathbf{X}^{t+1})$ of Gibbs algorithm for lattice Gaussian sampling is

$$P(\mathbf{X}^t = \mathbf{x}, \mathbf{X}^{t+1} = \mathbf{y}) = P(x_i^{t+1}|\mathbf{x}^t_{[-i]}) = D_{\Lambda,\sigma,\mathbf{c}}(x_i^{t+1}|\mathbf{x}^t_{[-i]}), \tag{9}$$

for $t \geq 1$, where random variable i follows from the distribution $[\beta_1, \ldots, \beta_n]$. Clearly, every two adjacent states $\mathbf{X}^t = \mathbf{x} = [x_1^t, \ldots, x_i^t, \ldots, x_n^t]^T$ and $\mathbf{X}^{t+1} = \mathbf{y} = [x_1^t, \ldots, x_i^{t+1}, \ldots, x_n^t]^T$ differ from each other by at most one coordinate x_i.

Algorithm 2 Gibbs Algorithm for Lattice Gaussian Sampling

```
Input: $\mathbf{B}, \sigma, \mathbf{c}, \mathbf{X}^0, \beta_i$'s, $t_{\text{mix}}(\epsilon)$
Output: $\mathbf{x} \sim \pi$, $\pi$ is within statistical distance of $\epsilon$ to $D_{\Lambda,\sigma,\mathbf{c}}$
for $t$ =1,2, ... do
    let $\mathbf{x}$ denote the state of $\mathbf{X}^{t-1}$
    randomly choose index $i$ by distribution $[\beta_1, \ldots, \beta_n]$
    sample $x_i$ from $P(x_i|\mathbf{x}_{[-i]})$ shown in (7)
    update $\mathbf{x}$ with the sampled $x_i$ and let $\mathbf{X}^t = \mathbf{x}$
    if $t \geq t_{\text{mix}}(\epsilon)$ then
        output the state of $\mathbf{X}^t$
    end if
end for
```

With the transition probabilities (9), we may form the infinite transition matrix $\mathbf{P}$. Then, according to Definition 1, besides irreducible and aperiodic property, it is also easy to verify that the underlying Markov chain is reversible by $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{X}^t)P(\mathbf{X}^t, \mathbf{X}^{t+1}) = D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{X}^{t+1})P(\mathbf{X}^{t+1}, \mathbf{X}^t)$ so as to the following Theorem about ergodicity, where the proof is omitted due to simplicity.

Theorem 1: Given the invariant distribution $D_{\Lambda,\sigma,\mathbf{c}}$, the Markov chain induced by the Gibbs algorithm is ergodic as $\lim_{t\to\infty} \|P^t(\mathbf{x}, \cdot) - D_{\Lambda,\sigma,\mathbf{c}}\|_{TV} = 0$ for all states $\mathbf{x} \in \mathbb{Z}^n$.

According to Theorem 1, if time permits to reach the stationary distribution, Gibbs algorithm will draw samples from $D_{\Lambda,\sigma,\mathbf{c}}$ no matter what value $\sigma > 0$ is, which means the obstacle encountered by Klein's algorithm is overcome. To summarize, Algorithm 2 illustrates the operation of Gibbs algorithm for lattice Gaussian sampling. The initial Markov state $\mathbf{X}^0$ can be chosen from $\mathbb{Z}^n$ arbitrarily or from the output of a suboptimal algorithm.

B. Geometric Ergodicity

To analyze convergence, we introduce the notion of spectral gap, formally defined as $\gamma = 1 - \text{spec}(\mathbf{F})$.[5] Here, $\text{spec}(\cdot)$ denotes the spectral radius and $\mathbf{F}$ represents the forward operator of the Markov chain defined as [39]

$$\mathbf{F}h(\mathbf{X}^t) \triangleq \sum_{\mathbf{X}^{t+1}\in\Omega} h(\mathbf{X}^{t+1})P(\mathbf{X}^t, \mathbf{X}^{t+1}) = E_P[h(\mathbf{X}^{t+1})|\mathbf{X}^t] \tag{10}$$

with induced operator norm

$$\|\mathbf{F}\| = \sup_{h\in L_0^2(\pi), \text{var}(h)=1} \|\mathbf{F}h\|. \tag{11}$$

Therefore, $\text{spec}(\mathbf{F})$ is closely related with the norm of $\mathbf{F}$ as [40], [41]

$$\text{spec}(\mathbf{F}) = \lim_{t\to\infty} \|\mathbf{F}^t\|^{1/t}. \tag{12}$$

Here, $E_\pi(\cdot)$ and $\text{var}_\pi(\cdot)$ denote the expectation taken under the probability measure π, $E_P(\cdot)$ and $\text{var}_P(\cdot)$ denote the

[3] $\text{var}_P[x_i|\mathbf{x}_{[-i]}]$ is slight affected by $\mathbf{x}_{[-i]}$ but the impact of $\mathbf{x}_{[-i]}$ upon it is periodic with respect to $\mathbb{Z}$, thus leading to the positive lower bound κ_i.

[4] The lattice basis $\mathbf{B}$ is a full rank matrix while $\mathbf{b}_i$'s are linear independent of each other.

[5] The geometric ergodicity of Markov chains can be also verified by other ways, i.e., *drift condition* in [29], [38].

expectation taken under the probability measure $P(\mathbf{X}^t, \mathbf{X}^{t+1})$ shown in (9). $L^2(\pi)$ is the Hilbert space of square integrable functions with respect to π so that $L_0^2(\pi) \triangleq \{h(\mathbf{x}) : E[h(\mathbf{x})] = 0, \text{var}[h(\mathbf{x})] < \infty\}$ denotes the subspace of $L^2(\pi)$ consisting of functions with zero mean relative to π. More precisely, for $h(\cdot), g(\cdot) \in L_0^2(\pi)$, the inner product defined by the space is

$$\langle h(\mathbf{x}), g(\mathbf{x})\rangle = E_\pi[h(\mathbf{x})g(\mathbf{x})] \tag{13}$$

with variance

$$\text{var}_\pi[h(\mathbf{x})] = \langle h(\mathbf{x}), h(\mathbf{x})\rangle = \|h(\mathbf{x})\|^2. \tag{14}$$

Theorem 2 ([41]): Given the invariant distribution π, a reversible, irreducible and aperiodic Markov chain with spectral gap $\gamma = 1 - spec(\mathbf{F}) > 0$ is geometrically ergodic $\|P^t(\mathbf{x}, \cdot) - \pi(\cdot)\|_{TV} \leq M(\mathbf{x})(1-\gamma)^t$ with $t \geq 1$.

Clearly, from Theorem 2, the convergence rate of the Markov chain is exactly characterized by the spectral radius of $\mathbf{F}$, i.e., $\varrho = \text{spec}(\mathbf{F})$. Based on it, we then arrive at the following Corollary to show the geometric ergodicity.

Corollary 1: Given the target lattice Gaussian distribution $\pi = D_{\Lambda,\sigma,\mathbf{c}}$, the Markov chain induced by Gibbs algorithm is geometrically ergodic $\|P^t(\mathbf{x}, \cdot) - \pi\|_{TV} \leq M(\mathbf{x})\varrho^t$ with convergence rate $\varrho = spec(\mathbf{F}) < 1$.

Proof: First of all, based on (12), because reversibility corresponds to a self-adjoint operator $\mathbf{F}$ with [42]

$$\|\mathbf{F}^t\| = \|\mathbf{F}\|^t, \tag{15}$$

it follows that

$$\text{spec}(\mathbf{F}) = \|\mathbf{F}\|. \tag{16}$$

Subsequently, according to (11) and (14), the spectral radius of $\mathbf{F}$ is further expressed as

$$\begin{aligned}
&\text{spec}(\mathbf{F})\\
&= \sup_{h\in L_0^2(\pi),\text{var}(h)=1} \|\mathbf{F}h\|\\
&= \sup_{h\in L_0^2(\pi),\text{var}(h)=1} \{\text{var}_\pi[E_P[h(\mathbf{X}^{t+1})|\mathbf{X}^t]]\}^{\frac{1}{2}} \qquad (17)\\
&\overset{(a)}{=} \sup_{h\in L_0^2(\pi),\text{var}(h)=1} \{\text{var}_\pi[h(\mathbf{X}^{t+1})] - E_\pi[\text{var}_P[h(\mathbf{X}^{t+1})|\mathbf{X}^t]]\}^{\frac{1}{2}}\\
&= \left[\sup_{h\in L_0^2(\pi),\text{var}(h)=1} \{\text{var}_\pi[h(\mathbf{X}^{t+1})] - E_\pi[\text{var}_P[h(\mathbf{X}^{t+1})|\mathbf{X}^t]]\}\right]^{\frac{1}{2}}\\
&= \left[1 - \inf_{h\in L_0^2(\pi),\text{var}(h)=1} \{E_\pi[\text{var}_P[h(\mathbf{X}^{t+1})|\mathbf{X}^t]]\}\right]^{\frac{1}{2}}\\
&\overset{(b)}{=} \left[1 - \inf_{h\in L_0^2(\pi),\text{var}(h)=1} \left\{\sum_{i=1}^n \beta_i E_\pi[\text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]]\right\}\right]^{\frac{1}{2}}\\
&= \Bigg[1 - \inf_{h\in L_0^2(\pi),\text{var}(h)=1}\\
&\qquad \left\{\sum_{i=1}^n \beta_i \sum_{\mathbf{x}_{[-i]}} \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]P_\pi(\mathbf{x}_{[-i]})\right\}\Bigg]^{\frac{1}{2}}. \qquad (18)
\end{aligned}$$

Here, (a) follows the *law of total variance* of random variable in statistics, i.e., $\text{var}(\mathbf{A}) = E[\text{var}(\mathbf{A}|\mathbf{B})] + \text{var}[E(\mathbf{A}|\mathbf{B})]$. (b) comes from the fact that $\mathbf{X}^t$ and $\mathbf{X}^{t+1}$ differs by only one component x_i and the index i obeys the distribution β_i's as a random variable, and $P_\pi(\mathbf{x}_{[-i]})$ is the marginal distribution of multivariate $\mathbf{x}_{[-i]}$ with respect to π.

On the other hand, since $h(\cdot) \in L_0^2(\pi)$ is a square integrable function (i.e., $\sum_{\mathbf{x}\in\mathbb{Z}^n} |h(\mathbf{x})| < \infty$) satisfying $L_0^2(\pi) \triangleq \{h(\mathbf{x}) : E_\pi[h(\mathbf{x})] = 0, \text{var}_\pi[h(\mathbf{x})] < \infty\}$, the variance of $\text{var}_\pi[h(\mathbf{x})]$ is not determined by the specific values of the multivariate $\mathbf{x}$, otherwise both $\sum_{\mathbf{x}\in\mathbb{Z}^n} |h(\mathbf{x})|$ and $\text{var}_\pi[h(\mathbf{x})]$ would be infinite due to the countably infinite state space of $\mathbf{x}$. Meanwhile, given the facts that $\text{var}_\pi[h(\mathbf{x})] = 1$ and $\text{var}_P[x_i|\mathbf{x}_{[-i]}] \geq \kappa_i$ shown in (8), it follows that $\sum_{i=1}^n \text{var}[h(\mathbf{x})|\mathbf{x}_{[-i]}] > 0$. Note that P in (9) is the conditional distribution of the target distribution π. More precisely, once $\sum_{i=1}^n \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}] = 0$, it means $h(\cdot)$ has no impact on every single random variable of $\mathbf{x}$, which naturally leads to $\text{var}_\pi[h(\mathbf{x})] = 0$ rather than $\text{var}_\pi[h(\mathbf{x})] > 0$.

Next, according to the proof by contradiction, let us focus on the lower bound of the summation term $\sum_{i=1}^n \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]$ given the requirement of $\text{var}_\pi[h(\mathbf{x})] = 1$. To start with, if $\text{var}_\pi[h(\mathbf{x})] > 0$, the value of the summation $\sum_{i=1}^n \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]$ could be arbitrarily small (i.e., $\inf_{h\in L_0^2(\pi)} \sum_{i=1}^n \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}] = 0$) since $h(\cdot)$ freely comes from the Hilbert space with $L_0^2(\pi) \triangleq \{h(\mathbf{x}) : E_\pi[h(\mathbf{x})] = 0, \text{var}_\pi[h(\mathbf{x})] < \infty\}$. However, here $h(\cdot)$ is required as $\text{var}_\pi[h(\mathbf{x})] = 1$, which not only makes $\sum_{i=1}^n \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}] > 0$ hold but also brings a latent lower bound $\kappa^\dagger > 0$ of it, namely,

$$\sum_{i=1}^n \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}] \geq \kappa^\dagger > 0. \tag{19}$$

This can be verified by contradiction. In particular, if $\inf_{h\in L_0^2(\pi),\text{var}(h)=1} \sum_{i=1}^n \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}] = 0$, then the requirement of $\text{var}_\pi[h(\mathbf{x})] = 1$ would be violated. More specifically, consider an extreme case that $\inf_{h\in L_0^2(\pi),\text{var}(h)=1} \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}] = 0$ for $1 \leq i \leq n-1$, then to meet the requirement of $\text{var}_\pi[h(\mathbf{x})] = 1$, there must be a positive constant for the term of $\inf_{h\in L_0^2(\pi),\text{var}(h)=1} \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-n]}]$, which indicates a latent lower bound with respect to the summation $\sum_{i=1}^n \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]$ no matter what $h(\cdot)$ is. Therefore, by simple induction, the following infimum will be lower bounded as

$$\begin{aligned}
\inf_{h\in L_0^2(\pi),\text{var}(h)=1} &\left\{\sum_{i=1}^n \beta_i \sum_{\mathbf{x}_{[-i]}} \text{var}_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]P(\mathbf{x}_{[-i]})\right\}\\
&= \kappa^\ddagger > 0, \qquad (20)
\end{aligned}$$

leading to

$$\text{spec}(\mathbf{F}) = (1-\kappa^\ddagger)^{\frac{1}{2}} < 1. \tag{21}$$

Hence, by invoking Theorem 2, the proof is completed with $\gamma = 1 - \text{spec}(\mathbf{F}) > 0$. ■

To summarize, the Markov chain converges exponentially fast to the lattice Gaussian distribution, where the exponential convergence rate $\varrho = \text{spec}(\mathbf{F})$ is derived in (18). Although it is difficult to calculate ϱ explicitly, comprehensive convergence analysis still can be carried out, which targets at a smaller ϱ for the convergence enhancement.

IV. Convergence Enhancement for Univariate Sampling

In this section, Metropolis-within-Gibbs algorithm is proposed for lattice Gaussian sampling. By refining the state space of each univariate sampling, the sampler turns out to be more efficient by a faster convergence rate.

A. Classic MH Algorithms

In particular, let us consider a target invariant distribution π together with a proposal distribution $q(\mathbf{x}, \mathbf{y})$ [43]. Given the current state $\mathbf{X}^t = \mathbf{x}$ for Markov chain, a state candidate $\mathbf{y}$ for the next Markov move $\mathbf{X}^{t+1}$ is generated from the proposal distribution $q(\mathbf{x}, \mathbf{y})$. After that, the acceptance ratio α is computed by

$$\alpha(\mathbf{x}, \mathbf{y}) = \min\left\{1, \frac{\pi(\mathbf{y})q(\mathbf{y}, \mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x}, \mathbf{y})}\right\}, \tag{22}$$

and $\mathbf{y}$ will be accepted by $\mathbf{X}^{t+1}$ with probability α. Otherwise, $\mathbf{x}$ will be retained by $\mathbf{X}^{t+1}$. In this way, a Markov chain $\{\mathbf{X}^0, \mathbf{X}^1, \ldots\}$ is established with the transition probability $P(\mathbf{X}^t, \mathbf{X}^{t+1})$ as follows:

$$\begin{aligned} &P(\mathbf{X}^t = \mathbf{x}, \mathbf{X}^{t+1} = \mathbf{y}) \\ &\quad = \begin{cases} q(\mathbf{x}, \mathbf{y})\alpha(\mathbf{x}, \mathbf{y}) & \text{if } \mathbf{y} \neq \mathbf{x}, \\ 1 - \sum_{\mathbf{z} \neq \mathbf{x}} q(\mathbf{x}, \mathbf{z})\alpha(\mathbf{x}, \mathbf{z}) & \text{if } \mathbf{y} = \mathbf{x}. \end{cases} \end{aligned} \tag{23}$$

Note that the proposal distribution $q(\mathbf{x}, \mathbf{y})$ in MH algorithm can be any fixed distribution from which we can conveniently draw samples. In principle, Gibbs algorithm is a special case of MH sampling by letting the proposal distribution be the univariate conditional distribution, i.e.,

$$q(\mathbf{x}, \mathbf{y}) = \pi(x_i | \mathbf{x}_{[-i]}). \tag{24}$$

Interestingly, with the above proposal distribution, it is easy to verify that the acceptance ratio α of Gibbs algorithm is always 1, making the acceptance of Markov $\mathbf{X}^{t+1} = \mathbf{y}$ without uncertainty.

B. Metropolis-Within-Gibbs Algorithm

Inspired by the flexible choice of $q(\mathbf{x}, \mathbf{y})$ in MH algorithm, we now present the proposed Metropolis-within-Gibbs (MWG) algorithm for lattice Gaussian sampling to further exploit the convergence potential of the univariate sampling.

Specifically, following the instruction of classic MH algorithm, given the Markov state $\mathbf{X}^t = \mathbf{x} = [x_1^t, \ldots, x_i^t, \ldots, x_n^t]^T$, a state candidate $\mathbf{y} = [x_1^t, \ldots, x_i^*, \ldots, x_n^t]^T$ for $\mathbf{X}^{t+1}$ is obtained through the proposal distribution

$$q(\mathbf{x}, \mathbf{y}) = q(x_i | \mathbf{x}_{-i}) = \frac{D_{\Lambda,\sigma,\mathbf{c}}(x_i | \mathbf{x}_{[-i]})}{1 - D_{\Lambda,\sigma,\mathbf{c}}(x_i^t | \mathbf{x}_{[-i]})}, \quad x_i \in \overline{\mathbb{Z}} \tag{25}$$

and x_i^t from the ith coordinate of $\mathbf{x}$ is eliminated from the state space $\mathbb{Z}$ in sampling x_i^*, which results in a reduced state space $\overline{\mathbb{Z}}$ with

$$\overline{\mathbb{Z}} \cup x_i^t = \mathbb{Z} \text{ and } \overline{\mathbb{Z}} \cap x_i^t = \emptyset. \tag{26}$$

In other words, the sample x_i^* in $\mathbf{y}$ is obtained according to the sampling from (25), namely,

$$x_i^* \sim q(x_i | \mathbf{x}_{-i}). \tag{27}$$

Once x_i^* in $\mathbf{y}$ is obtained, then the acceptance ratio α in (22) is calculated, and the decision about whether to accept it as $\mathbf{X}^{t+1} = \mathbf{y}$ is performed thereafter. Note that in conventional Gibbs algorithm $\mathbf{y}$ will be accepted by $\mathbf{X}^{t+1}$ without uncertainty. This is the core difference between the proposed MWG and Gibbs algorithms since the uncertainty in the judgment of $\mathbf{X}^{t+1}$ to choose $\mathbf{y}$ or not is retained. Hence, the proposed MWG algorithm can be summarized as the following three main procedures.

1) *Sample from the following univariate proposal distribution in (25) to obtain the candidate sample x_i^*.*

2) *From* (22), *calculate the acceptance ratio $\alpha(\mathbf{x}, \mathbf{y})$*

$$\alpha(\mathbf{x}, \mathbf{y}) = \min\left\{1, \frac{1 - D_{\Lambda,\sigma,\mathbf{c}}(x_i^t | \mathbf{x}_{[-i]})}{1 - D_{\Lambda,\sigma,\mathbf{c}}(x_i^* | \mathbf{x}_{[-i]})}\right\}. \tag{28}$$

3) *Make a decision for $\mathbf{X}^{t+1}$ based on $\alpha(\mathbf{x}, \mathbf{y})$ to accept $\mathbf{y} = [x_1^t, \ldots, x_i^*, \ldots, x_n^t]^T$ or not (i.e., $\mathbf{X}^{t+1} = \mathbf{x}$).*

Here, we emphasize that different from Gibbs algorithm who always accepts the sampling candidate $\mathbf{y}$ as the state of Markov move X^{t+1}, a salient feature of MWG algorithm is that the uncertainty arising from the sample acceptance is retained [44] as acceptance ratio $\alpha(\mathbf{x}, \mathbf{y}) \leq 1$. Put it in another way, the sampling candidate x_i^* obtained by $\mathbf{y}$ can be rejected in the proposed MWG algorithm. To conclude, the proposed MWG algorithm for lattice Gaussian sampling is presented in Algorithm 3.

Algorithm 3 Metropolis-Within-Gibbs Algorithm for Lattice Gaussian Sampling

```
Input: $\mathbf{B}, \sigma, \mathbf{c}, \mathbf{X}^0, \beta_i$'s, $t_{\text{mix}}(\epsilon)$
Output: $\mathbf{x} \sim \pi$, $\pi$ is within statistical distance of $\epsilon$ to $D_{\Lambda,\sigma,\mathbf{c}}$
for $t$ =1,2, ... do
    let $\mathbf{x}$ denote the state of $\mathbf{X}^{t-1}$
    randomly choose index $i$ by distribution $[\beta_1, \ldots, \beta_n]$
    sample $x_i^*$ by proposal distribution $q(x_i|\mathbf{x}_{[-i]})$ in (27)
    calculate the acceptance quantity $\alpha$ shown in (28)
    generate a sample $u \sim U[0, 1]$
    if $u \leq \alpha$ then
        get $\mathbf{y}$ with the sampled $x_i$ and let $\mathbf{X}^t = \mathbf{y}$
    else let $\mathbf{X}^t = \mathbf{x}$
    end if
    if $t \geq t_{\text{mix}}(\epsilon)$ then
        output the state of $\mathbf{X}^t$
    end if
end for
```

C. Convergence Rate Analysis

Theorem 3: Given the invariant lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}$, the proposed Metropolis-within-Gibbs algorithm achieves a better exponential convergence performance than Gibbs algorithm by $\varrho_{MWG} < \varrho$.

Proof: First of all, according to (23), the transition probability of MWG sampling algorithm is derived as

$$\begin{aligned}P_{\text{MWG}}(&\mathbf{X}^t\\ &= \mathbf{x}, \mathbf{X}^{t+1} = \mathbf{y}) = q(\mathbf{x},\mathbf{y}) \cdot \alpha(\mathbf{x},\mathbf{y})\\ &= \min\left\{\frac{D_{\Lambda,\sigma,\mathbf{c}}(x_i^*|\mathbf{x}_{[-i]})}{1-D_{\Lambda,\sigma,\mathbf{c}}(x_i^t|\mathbf{x}_{[-i]})}, \frac{D_{\Lambda,\sigma,\mathbf{c}}(x_i^*|\mathbf{x}_{[-i]})}{1-D_{\Lambda,\sigma,\mathbf{c}}(x_i^*|\mathbf{x}_{[-i]})}\right\}. \quad (29)\end{aligned}$$

Compared to the transition probability of Gibbs algorithm given in (9), it is straightforward to see that

$$\begin{aligned}P_{\text{MWG}}(\mathbf{X}^t = \mathbf{x}, \mathbf{X}^{t+1} = \mathbf{y}) &> P_{\text{Gibbs}}(\mathbf{X}^t = \mathbf{x}, \mathbf{X}^{t+1} = \mathbf{y})\\ &= D_{\Lambda,\sigma,\mathbf{c}}(x_i^{t+1}|\mathbf{x}_{[-i]})\\ &\overset{(c)}{=} D_{\Lambda,\sigma,\mathbf{c}}(x_i^*|\mathbf{x}_{[-i]}) \quad (30)\end{aligned}$$

for cases of $\mathbf{x} \neq \mathbf{y}$. Here equality (c) holds because in Gibbs sampling the sample x_i^* in $\mathbf{y}$ is always accepted, i.e., $x_i^* = x_i^{t+1}$ and $\mathbf{X}^{t+1} = \mathbf{y} = [x_1^t, \ldots, x_i^{t+1}, \ldots, x_n^t]^T$. Therefore, it means that each off-diagonal element in transition matrix $\mathbf{P}_{\text{MWG}}$ is always larger than that of $\mathbf{P}_{\text{Gibbs}}$. From literatures of MCMC, such a case is known as *Peskun ordering* written by

$$P_{\text{MWG}}(\mathbf{X}^t, \mathbf{X}^{t+1}) \succeq P_{\text{Gibbs}}(\mathbf{X}^t, \mathbf{X}^{t+1}). \quad (31)$$

Now, we invoke the following Lemma to reveal the relation between Peskun ordering and convergence rate.

Lemma 1 ([45]): Given reversible Markov chains P and Q with stationary distribution π, if $P \succeq Q$, then their convergence rates satisfy $\varrho_P \leq \varrho_Q$.

Note that the definition of Peskun ordering $P(\mathbf{X}^t, \mathbf{X}^{t+1}) \succeq Q(\mathbf{X}^t, \mathbf{X}^{t+1})$ given in [45] is based on the inequality $P(\mathbf{X}^t, \mathbf{X}^{t+1}) \geq Q(\mathbf{X}^t, \mathbf{X}^{t+1})$, where the equality $\varrho_P = \varrho_Q$ holds only if $P(\mathbf{X}^t, \mathbf{X}^{t+1}) = Q(\mathbf{X}^t, \mathbf{X}^{t+1})$. Here, because the case of equality is not included, according to (31) and Lemma 1, we can immediately obtain that $\varrho_{\text{MWG}} < \varrho$, completing the proof. ■

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 decrease the rejection probability of the proposed moves.

D. Parallel Tempering

Now, the parallel tempering technique is adopted to the proposed Metropolis-within-Gibbs algorithm to alleviate the possible risks associated with slow mixing Markov chains, which may get stuck during the convergence.

Theoretically, parallel tempering is a generic MCMC sampling method which allows a better convergence. The inspiration of it comes from the idea that a temperature parameter could be used to flatten out the target distribution, thus making the random walk chain for that temperature more likely to mix quickly [46]. Therefore, according to parallel tempering, the Markov chain induced by the Metropolis-within-Gibbs algorithm for lattice Gaussian sampling can be strengthened as follows.

1) *Define a set of target lattice Gaussian distributions $\pi_{t_1}, \ldots, \pi_{t_k}$*

$$\pi_{t_j} = D_{\Lambda,t_j\sigma,\mathbf{c}}(\mathbf{x}), \quad 1 \leq j \leq k \quad (32)$$

where $t_k > \ldots > t_1 = 1$ represent different temperature parameters respectively.

2) *Run k Markov chains in parallel with the MWG transition probability*

$$\begin{aligned}P_{\text{MWG}}^j(&\mathbf{X}_j^t = \mathbf{x}, \mathbf{X}_j^{t+1} = \mathbf{y})\\ &= \min\left\{\frac{\pi_{t_j}(x_i^*|\mathbf{x}_{[-i]})}{1-\pi_{t_j}(x_i^t|\mathbf{x}_{[-i]})}, \frac{\pi_{t_j}(x_i^*|\mathbf{x}_{[-i]})}{1-\pi_{t_j}(x_i^*|\mathbf{x}_{[-i]})}\right\} \quad (33)\end{aligned}$$

for $1 \leq j \leq k$.

3) *After t_{swap} Markov moves on each Markov chain, consecutively select chain pairs between two neighboring temperatures t_j and t_{j+1}, $1 \leq j \leq k-1$, then attempt to swap their states with probability*

$$\alpha_{\text{swap}} = \min\left\{1, \frac{\pi_{t_j}(\mathbf{X}_{t_{j+1}}^{t+1})\pi_{t_{j+1}}(\mathbf{X}_{t_j}^{t+1})}{\pi_{t_{j+1}}(\mathbf{X}_{t_{j+1}}^{t+1})\pi_{t_j}(\mathbf{X}_{t_j}^{t+1})}\right\}, \quad (34)$$

otherwise the swap over $\mathbf{X}_j^{t+1}$ and $\mathbf{X}_{j+1}^{t+1}$ is canceled.

To summarize, this modification essentially allows two types of update. The first one draws samples from distributions $D_{\Lambda,t_j\sigma,\mathbf{c}}(\mathbf{x})$, and the second one is based on a proposal generated from the potential swapping of states between Markov chains. Here, the acceptance probability shown in (34) mainly ensures that the second type of update preserves the stationary distribution [47]. Note that only pairs between neighboring temperatures are considered for swapping, where the chances of accepting an exchange are more likely to be higher.

Clearly, with the increase of temperature parameter t_j, the lattice Gaussian distribution $D_{\Lambda,t_j\sigma,\mathbf{c}}(\mathbf{x})$ becomes ‘warm’, which would correspond to a uniform distribution over the entire state space. More specifically, the warm distribution mix progressively more rapidly than the cold one which is of primary interest. By allowing the Markov chains to swap states, the convergence performance of the ‘cold’ chain is improved since the state space is traversed more rapidly. Note that such an operation also requires multiple chains in parallel, and only the output from one is used as a basis for inference.

V. Convergence Enhancement for Multivariate Sampling

To further improve the convergence performance, the blocked strategy, which performs the sampling over multiple components of $\mathbf{x}$ within a block, is investigated. Then, Gibbs-Klein (GK) algorithm is proposed for the efficient implementation of blocked Gibbs algorithm.

A. Convergence Analysis of Blocked Sampling

The idea of blocked strategy in Gibbs algorithm is to perform multivariate sampling over multiple components at

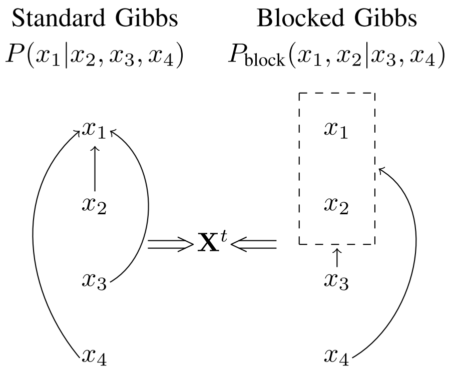

Fig. 2. Illustration of standard and blocked Gibbs sampling strategies. Components within the dashed block are sampled as a whole by blocked Gibbs algorithm, where the components emitting the arrows are being conditioned during the univariate sampling or blocked sampling.

each Markov move. Compared to the standard Gibbs algorithm in (7), the blocked sampling for lattice Gaussian distribution can be expressed as

$$\begin{aligned} P(\mathbf{x}_{\text{block}}|\mathbf{x}_{[-\text{block}]}) &= D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x}_{\text{block}}|\mathbf{x}_{[-\text{block}]}) \\ &= \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{Bx}-\mathbf{c}\|^2}}{\sum_{\mathbf{x}_{\text{block}}\in\mathbb{Z}^m} e^{-\frac{1}{2\sigma^2}\|\mathbf{Bx}-\mathbf{c}\|^2}}, \end{aligned} \tag{35}$$

where each block $\mathbf{x}_{\text{block}} = [x_i, x_j, \ldots]^T$ contains multiple components of $\mathbf{x}$ and $\mathbf{x}_{[-\text{block}]}$ denotes the other components of $\mathbf{x}$ except $\mathbf{x}_{\text{block}}$. In this way, the transition probability $P_{\text{block}}(\mathbf{X}^t, \mathbf{X}^{t+1})$ of the blocked Gibbs algorithm for lattice Gaussian sampling is

$$P_{\text{block}}(\mathbf{X}^t = \mathbf{x}, \mathbf{X}^{t+1} = \mathbf{y}) = D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x}_{\text{block}}|\mathbf{x}_{[-\text{block}]}). \tag{36}$$

For a better illustration, a two-component blocked sampling strategy is depicted in Fig. 2. Compared to univariate sampling, by sampling multiple components together, the slow componentwise moves will be replaced by the fast moves incorporating the information about dependence between components.

Lemma 2: Given the invariant lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}$, the blocked Gibbs algorithm achieves a faster convergence rate than the standard one as $\varrho_{block} \leq \varrho$.

Proof: First of all, by taking the random index i at each Markov move into account, the term shown in (17) can be described as

$$\text{var}_\pi[E_P[h(\mathbf{X}^{t+1})|\mathbf{X}^t]] = \sum_{i=1}^{n} \beta_i \text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]] \tag{37}$$

and subsequently, we have

$$\begin{aligned} \varrho &= \sup_{h\in L_0^2(\pi),\text{var}(h)=1} \left[\sum_{i=1}^{n} \beta_i \text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]]\right]^{\frac{1}{2}} \\ &= \left[\sup_{h\in L_0^2(\pi),\text{var}(h)=1} \sum_{i=1}^{n} \beta_i \text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]]\right]^{\frac{1}{2}}. \end{aligned} \tag{38}$$

For ease of presentation, a two-component blocked sampling scenario is firstly concerned. Typically, suppose components x_i and x_j of $\mathbf{x}$ can be sampled together as a block, then consider the fact that

$$E_P[h(\mathbf{x})|\mathbf{x}_{[-i,-j]}] = E_P[E_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]|\mathbf{x}_{[-j]}], \tag{39}$$

we can immediately get

$$\text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-i,-j]}]] \leq \text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]] \tag{40}$$

and

$$\text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-i,-j]}]] \leq \text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-j]}]] \tag{41}$$

by the law of total variance $\text{var}(\mathbf{A}) = E[\text{var}(\mathbf{A}|\mathbf{B})] + \text{var}[E(\mathbf{A}|\mathbf{B})]$. Therefore, given the index selection probabilities β_i and β_j, we have

$$\begin{aligned} (\beta_i + \beta_j)&\text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-i,-j]}]] \\ &\leq \beta_i \text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-i]}]] \\ &\quad + \beta_j \text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-j]}]]. \end{aligned} \tag{42}$$

where the equality in (42) holds if and only if components of x_i and x_j are independent of each other. From (38), this indicates a more efficient convergence rate ρ for the blocked sampling over x_i and x_j.

Inductively, this two-component blocked sampling over coordinates i and j can be easily extended to any larger size blocked sampling. Hence, according to (38) and (42), it follows that

$$\varrho_{\text{block}} \leq \varrho, \tag{43}$$

completing the proof. ■

Note that the equality $\varrho_{\text{block}} = \varrho$ holds if and only if the components of $\mathbf{x}$ are independent of each other, which corresponds to the lattice basis $\mathbf{B}$ in the lattice Gaussian distribution is an orthogonal matrix. From (41), it is straightforward to check that the convergence performance also improves gradually by grouping more elements into the block

$$\text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-\text{block},-j]}]] \leq \text{var}_\pi[E_P[h(\mathbf{x})|\mathbf{x}_{[-\text{block}]}]] \tag{44}$$

since a larger block size allows moves in more general directions. If all the components forming a single block could be sampled directly, there would be no need for MCMC sampling. In this regard, blocked strategy is strongly recommended if sampling over multivariate can be efficiently performed.

B. Efficient Blocked Sampling by Gibbs-Klein Algorithm

Although blocked sampling achieves a better convergence rate than univariate one, sampling over a block is generally more costly than componentwise sampling as its sampling space increases exponentially with the block size. Because of this, we propose to use Klein's algorithm for multi-component sampling, which leads to the Gibbs-Klein algorithm.

At each step of Markov chain, the proposed Gibbs-Klein algorithm randomly picks up m components of $\mathbf{x}$ to update. For a better illustration of the proposed sampling, here we establish another new scheme but equivalent to the foregoing one, which resorts to the help of permutation matrices. In particular, an $n \times n$ permutation matrix $\mathbf{E}$ is applied to sort the updating order within the blocked sampling

$$D_{\mathcal{L}(\mathbf{B}),\sigma,\mathbf{c}}(\mathbf{x}) = \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{Bx}-\mathbf{c}\|^2}}{\sum_{\mathbf{x}\in\mathbb{Z}^n} e^{-\frac{1}{2\sigma^2}\|\mathbf{Bx}-\mathbf{c}\|^2}}$$

$$
\begin{aligned}
&= \frac{e^{-\frac{1}{2\sigma^2}\|\widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c}\|^2}}{\sum_{\mathbf{z}\in\mathbb{Z}^n} e^{-\frac{1}{2\sigma^2}\|\widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c}\|^2}}\\
&= D_{\mathcal{L}(\widetilde{\mathbf{B}}),\sigma,\mathbf{c}}(\mathbf{z}). \qquad (45)
\end{aligned}
$$

where $\mathbf{z} = \mathbf{E^{-1}x}$ and $\widetilde{\mathbf{B}} = \mathbf{BE}$. Intuitively, if $\mathbf{E}$ is randomly generated, then Gibbs-Klein algorithm on m randomly chosen components will be equivalent to sample m consecutive components of $\mathbf{z}$ in a fixed order. Therefore, here we always consider the block formed by the first m components of $\mathbf{z}$, namely $\mathbf{z}_{\text{block}} = [z_1, \ldots, z_m]^T$, which simply corresponds to $\mathbf{x}_{\text{block}}$ containing m components of $\mathbf{x}$

$$
\begin{aligned}
&D_{\mathcal{L}(\mathbf{B}),\sigma,\mathbf{c}}(\mathbf{x}_{\text{block}}|\mathbf{x}_{[-\text{block}]})\\
&\quad= \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{Bx}-\mathbf{c}\|^2}}{\sum_{\mathbf{x}_{\text{block}}\in\mathbb{Z}^m} e^{-\frac{1}{2\sigma^2}\|\mathbf{Bx}-\mathbf{c}\|^2}}\\
&\quad= \frac{e^{-\frac{1}{2\sigma^2}\|\widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c}\|^2}}{\sum_{\mathbf{z}_{\text{block}}\in\mathbb{Z}^m} e^{-\frac{1}{2\sigma^2}\|\widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c}\|^2}}\\
&\quad= D_{\mathcal{L}(\widetilde{\mathbf{B}}),\sigma,\mathbf{c}}(\mathbf{z}_{\text{block}}|\mathbf{z}_{[-\text{block}]}). \qquad (46)
\end{aligned}
$$

Here, we apply QR-decomposition to $\widetilde{\mathbf{B}} = \mathbf{QR}$ so that $\| \widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c} \| = \| \mathbf{Rz}-\mathbf{c}' \|$ with $\mathbf{c}' = \mathbf{Q}^T\mathbf{c}$. This is unnecessary in fact but could provide a more straightforward way to illustrate and analyze the proposed Gibbs-Klein algorithm due to the form of the upper triangular matrix $\mathbf{R}$. Then, as for the blocked sampling over $\mathbf{z}_{\text{block}}$, we propose to sample each component z_i within it from the following 1-dimensional distribution in the backward order from z_m to z_1:

$$
P_i(z_i|\overline{\mathbf{z}}_{[-i]}) = D_{\mathbb{Z},\sigma_i,\widetilde{z}_i}(z_i), \quad 1 \le i \le m, \qquad (47)
$$

where $\sigma_i = \frac{\sigma}{|r_{i,i}|}$, $\overline{\mathbf{z}}_{[-i]} = [z_{i+1}, \ldots, z_m, z_{m+1}, \ldots, z_n]^T$ and $\widetilde{z}_i = \frac{c_i' - \sum_{j=i+1}^m r_{i,j}z_j - \sum_{j'=m+1}^n r_{i,j'}z_{j'}}{r_{i,i}}$.[6] Clearly, from (47), all the $n-m$ components of $\mathbf{z}$ out of the block (i.e., $\mathbf{z}_{[-\text{block}]} = [z_{m+1}, \ldots, z_n]^T$) are taken into account by each element within the block. To summarize, Algorithm 4 gives the proposed Gibbs-Klein algorithm, where the extension to other scan strategies is possible.

Algorithm 4 Gibbs-Klein Algorithm for Lattice Gaussian Sampling

```
Input: $\mathbf{B}, \sigma, \mathbf{c}, \mathbf{X}^0, t_{\text{mix}}(\epsilon)$;
Output: $\mathbf{x} \sim \pi$, $\pi$ is within statistical distance of $\epsilon$ to $D_{\Lambda,\sigma,\mathbf{c}}$
for $t$ =1,2 … do
    let $\mathbf{x}$ denote the state of $\mathbf{X}^{t-1}$
    randomly generate a permutation matrix $\mathbf{E}$
    let $\widetilde{\mathbf{B}} = \mathbf{BE}$ and $\mathbf{z} = \mathbf{E}^{-1}\mathbf{x}$
    let $\widetilde{\mathbf{B}} = \mathbf{QR}$ and $\mathbf{c}' = \mathbf{Q}^T\mathbf{c}$
    for $k = 1$, … do
        for $i = m$, …, 1 do
            let $\sigma_i = \frac{\sigma}{|r_{i,i}|}$
            let $\widetilde{z}_i = \frac{c_i' - \sum_{j=i+1}^m r_{i,j}z_j - \sum_{j'=m+1}^n r_{i,j'}z_{j'}}{r_{i,i}}$
            sample $z_i$ from $D_{\mathbb{Z},\beta_i,\widetilde{z}_i}$
        end for
        calculate the acceptance ratio $\alpha_{\text{accept}}$ shown in (53)
        generate a sample $u \sim U[0,1]$
        if $u \le \alpha_{\text{accept}}$ then
            output $\mathbf{z}_{\text{block}}$ as the exact sample from $D_{\mathcal{L}(\overline{\mathbf{r}}),\sigma,\overline{\mathbf{c}}}$
            Break
        end if
    end for
    update $\mathbf{z} = [\mathbf{z}_{\text{block}}; \mathbf{z}_{[-\text{block}]}]^T$
    return $\mathbf{x} = \mathbf{Ez}$ and let $\mathbf{X}^t = \mathbf{x}$
    if $t \ge t_{\text{mix}}(\epsilon)$ then
        output the state of $\mathbf{X}^t$
    end if
end for
```

C. Validity of Gibbs-Klein Algorithm

Now, the validity of Gibbs-Klein algorithm is verified by showing its ergodicity, where rejection sampling is resorted to make sure the generated distribution by Gibbs-Klein is exact lattice Gaussian distribution.

Lemma 3: For a given invariant lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}$, if $\sigma = \omega(\sqrt{\log m}) \cdot \max_{1\le i\le m}|r_{i,i}|$, under the help of rejection sampling, the proposed Gibbs-Klein algorithm is able to sample from the following distribution

$$
D_{\mathcal{L}(\widetilde{\mathbf{B}}),\sigma,\mathbf{c}}(\mathbf{z}_{\text{block}}|\mathbf{z}_{[-\text{block}]}) = \frac{e^{-\frac{1}{2\sigma^2}\|\widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c}\|^2}}{\sum_{\mathbf{z}_{\text{block}}\in\mathbb{Z}^m} e^{-\frac{1}{2\sigma^2}\|\widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c}\|^2}}, \qquad (48)
$$

where $\mathbf{z} = [\mathbf{z}_{\text{block}}; \mathbf{z}_{[-\text{block}]}]$.

$$
\begin{aligned}
P(\mathbf{z}_{\text{block}} \mid \mathbf{z}_{[-\text{block}]}) &= \prod_{i=1}^m D_{\mathbb{Z},\sigma_{m+1-i},\widetilde{z}_{m+1-i}}(z_{m+1-i})\\
&= \frac{e^{-\frac{1}{2\sigma^2}\sum_{i=1}^m\left(\overline{c}_{m+1-i}-\sum_{j=m+1-i}^m r_{m+1-i,j}z_j\right)^2}}{\prod_{i=1}^m\sum_{z_{m+1-i}\in\mathbb{Z}} e^{-\frac{1}{2\sigma^2}\left(\overline{c}_{m+1-i}-\sum_{j=m+1-i}^m r_{m+1-i,j}z_j\right)^2}}\\
&= \frac{e^{-\frac{1}{2\sigma^2}\|\overline{\mathbf{c}}-\overline{\mathbf{R}}\mathbf{z}_{\text{block}}\|^2}}{\prod_{i=1}^m\sum_{z_{m+1-i}\in\mathbb{Z}} e^{-\frac{1}{2\sigma^2}\left(r_{m+1-i,m+1-i}z_{m+1-i}-\overline{c}_{m+1-i}+\sum_{j=m+2-i}^m r_{m+1-i,j}z_j\right)^2}}\\
&= \frac{\rho_{\mathcal{L}(\overline{\mathbf{R}}),\sigma,\overline{\mathbf{c}}}(\mathbf{z}_{\text{block}})}{\prod_{i=1}^m \rho_\sigma(r_{m+1-i,m+1-i}\mathbb{Z}+\xi_i)}, \qquad (50)
\end{aligned}
$$

[6] Determining $\widetilde{z}_i$ without QR-decomposition is similar as $\widetilde{z}_i = \mathbf{D}(i,:)(\mathbf{c} - \sum_{j=i+1}^m \widetilde{\mathbf{b}}_j z_j - \sum_{j'=m+1}^n \widetilde{\mathbf{b}}_{j'} z_{j'})$, where $\mathbf{D}(i,:)$ denotes the ith row of $\mathbf{D} = \widetilde{\mathbf{B}}^\dagger$.

Proof: From (47) and by induction, the blocked sampling probability $\mathbf{z}_{\text{block}}$ conditioned on $\mathbf{z}_{[-\text{block}]}$ is given by

$$P(\mathbf{z}_{\text{block}} \mid \mathbf{z}_{[-\text{block}]}) = \prod_{i=1}^{m} P(z_{m+1-i} | \overline{\mathbf{z}}_{[-(m+1-i)]}). \tag{49}$$

Then, according to (47) and (49), we have the following derivation in (50), as shown at the bottom of previous page, where $\overline{c}_i = c'_i - \sum_{j'=m+1}^{n} r_{i,j'} z_{j'}$, $\overline{\mathbf{c}} = [\overline{c}_1, \ldots, \overline{c}_m]^T$, $\xi_i = \sum_{j=m+2-i}^{m} r_{m+1-i,j} z_j - \overline{c}_{m-i+i}$ and $\overline{\mathbf{R}}$ is the $m \times m$ segment of $\mathbf{R}$ with $r_{1,1}$ to $r_{m,m}$ in diagonal. Clearly, the effect of the subvector $\mathbf{z}_{[-\text{block}]}$ is hidden in $\overline{c}_i$.

In [48], it has been demonstrated that if $\sigma > \eta_\varepsilon(\mathcal{L}(\overline{\mathbf{R}}))$, then

$$\frac{\prod_{i=1}^{m} \rho_\sigma(r_{i,i}\mathbb{Z} + \xi_i)}{\prod_{i=1}^{m} \rho_\sigma(r_{i,i}\mathbb{Z})} \in \left(\left(\frac{1-\varepsilon}{1+\varepsilon}\right)^m, 1\right] \tag{51}$$

which means $\prod_{i=1}^{m} \rho_\sigma(r_{i,i}\mathbb{Z} + \xi_i)$ can be substituted by $\prod_{i=1}^{m} \rho_\sigma(r_{i,i}\mathbb{Z})$ within negligible errors when ε is sufficiently small. As shown in [32], $\eta_\varepsilon(\Lambda)$ with negligible ε is upper bounded as $\eta_\varepsilon(\Lambda) \leq \omega(\sqrt{\log n}) \cdot \max_{1 \leq i \leq n} \|\widehat{\mathbf{b}}_i\|$. Therefore, if $\sigma = \omega(\sqrt{\log m}) \cdot \max_{1 \leq i \leq m} \|r_{i,i}\|$, $P(\mathbf{z}_{\text{block}} \mid \mathbf{z}_{[-\text{block}]})$ shown in (50) can be rewritten as

$$P(\mathbf{z}_{\text{block}} \mid \mathbf{z}_{[-\text{block}]}) \simeq \frac{\rho_{\mathcal{L}(\overline{\mathbf{r}}),\sigma,\overline{\mathbf{c}}}(\mathbf{z}_{\text{block}})}{\prod_{i=1}^{m} \rho_\sigma(r_{i,i}\mathbb{Z})}, \tag{52}$$

where "$\simeq$" represents equality up to a negligible error. Moreover, in order to remove the latent negligible bias shown above, the classic rejection sampling can be applied to yield an exact sample (see [49] for more details). Specifically, the candidate of $\mathbf{z}_{\text{block}}$ is outputted with probability

$$\alpha_{\text{accept}} = \frac{\prod_{i=1}^{m} \rho_\sigma(r_{i,i}\mathbb{Z} + \xi_i)}{\prod_{i=1}^{m} \rho_\sigma(r_{i,i}\mathbb{Z})} \tag{53}$$

and this probability can be efficiently computed (achieve any desired t bits of accuracy in time $poly(t)$, t denotes the number of iterations), as shown in [49]. Therefore, under the help of rejection sampling, it follows that

$$P(\mathbf{z}_{\text{block}} \mid \mathbf{z}_{[-\text{block}]}) = \frac{\rho_{\mathcal{L}(\overline{\mathbf{R}}),\sigma,\overline{\mathbf{c}}}(\mathbf{z}_{\text{block}})}{\prod_{i=1}^{m} \rho_\sigma(r_{i,i}\mathbb{Z})}. \tag{54}$$

where the denominator is a constant since it is independent of $\mathbf{z}_{\text{block}}$, $\mathbf{z}_{[-\text{block}]}$ and $\mathbf{c}$. In this condition, we obtain that in any given iteration, the output is distributed according to the desired distribution

$$D_{\mathcal{L}(\overline{\mathbf{R}}),\sigma,\overline{\mathbf{c}}}(\mathbf{z}_{\text{block}}) = \frac{\rho_{\mathcal{L}(\overline{\mathbf{R}}),\sigma,\overline{\mathbf{c}}}(\mathbf{z}_{\text{block}})}{\sum_{\mathbf{z}_{\text{block}} \in \mathbb{Z}^m} \rho_{\mathcal{L}(\overline{\mathbf{R}}),\sigma,\overline{\mathbf{c}}}(\mathbf{z}_{\text{block}})} \tag{55}$$

where the denominator in (55) is also a constant by serving as a scalar. Therefore this is also the overall output distribution of the blocked target sampler

$$\begin{aligned} D_{\mathcal{L}(\widetilde{\mathbf{B}}),\sigma,\mathbf{c}}(\mathbf{z}_{\text{block}} | \mathbf{z}_{[-\text{block}]}) &= \frac{e^{-\frac{1}{2\sigma^2}\|\widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c}\|^2}}{\sum_{\mathbf{z}_{\text{block}} \in \mathbb{Z}^m} e^{-\frac{1}{2\sigma^2}\|\widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c}\|^2}} \\ &= D_{\mathcal{L}(\overline{\mathbf{R}}),\sigma,\overline{\mathbf{c}}}(\mathbf{z}_{\text{block}}). \end{aligned} \tag{56}$$

■

Furthermore, because $D_{\mathcal{L}(\widetilde{\mathbf{B}}),\sigma,\mathbf{c}}(\mathbf{z})$ and $D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x})$ describe the same lattice Gaussian distribution, namely,

$$\frac{e^{-\frac{1}{2\sigma^2}\|\widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c}\|^2}}{\sum_{\mathbf{z}_{\text{block}} \in \mathbb{Z}^m} e^{-\frac{1}{2\sigma^2}\|\widetilde{\mathbf{B}}\mathbf{z}-\mathbf{c}\|^2}} \triangleq \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}-\mathbf{c}\|^2}}{\sum_{\mathbf{x}_{\text{block}} \in \mathbb{Z}^m} e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}-\mathbf{c}\|^2}} \tag{57}$$

$D_{\mathcal{L}(\widetilde{\mathbf{B}}),\sigma,\mathbf{c}}(\mathbf{z}_{\text{block}} | \mathbf{z}_{[-\text{block}]})$ is essentially equivalent to $D_{\mathcal{L}(\mathbf{B}),\sigma,\mathbf{c}}(\mathbf{x}_{\text{block}} | \mathbf{x}_{[-\text{block}]})$. Therefore, according to Lemma 3, Gibbs-Klein algorithm is capable to sample multiple components of $\mathbf{x}$ at each Markov move. We then arrive at the following Theorem.

Theorem 4: For $\sigma = \omega(\sqrt{\log m}) \cdot \max_{1 \leq i \leq m} |r_{i,i}|$, the Markov chain induced by the Gibbs-Klein algorithm with block size m is ergodic with respect to the lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}$ as $\lim_{t \to \infty} \|P^t(\mathbf{x}, \cdot) - D_{\Lambda,\sigma,\mathbf{c}}\|_{TV} = 0$ for all states $\mathbf{x} \in \mathbb{Z}^n$.

Proof: Based on Definition 1, we now prove the ergodicity by verifying the reversibility, irreducibility and aperiodic of the underlying Markov chain.

To start with, let $\mathbf{x}_i$ and $\mathbf{x}_j$ be two adjacent states in Gibbs-Klein algorithm. For block size m, every two adjacent states in Gibbs-Klein algorithm differ from each other by at most m components. For convenience, we express them as

$$\mathbf{x}_i = [\mathbf{x}_{\text{block}(i)}, \mathbf{x}_{[-\text{block}]}] \quad \text{and} \quad \mathbf{x}_j = [\mathbf{x}_{\text{block}(j)}, \mathbf{x}_{[-\text{block}]}], \tag{58}$$

where $\mathbf{x}_{\text{block}(i)}$ and $\mathbf{x}_{\text{block}(j)}$ denote the m components belonging to $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively. Then, the transition probability of Gibbs-Klein algorithm is

$$\begin{aligned} P(\mathbf{X}^t = \mathbf{x}_i, \mathbf{X}^{t+1} = \mathbf{x}_j) &= P(\mathbf{x}_{\text{block}(i)} \to \mathbf{x}_{\text{block}(j)} | \mathbf{x}_{[-\text{block}]}) \\ &\overset{(d)}{=} P(\mathbf{x}_{\text{block}(j)} | \mathbf{x}_{[-\text{block}]}) \\ &= \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}_j-\mathbf{c}\|^2}}{\sum_{\mathbf{x}_{\text{block}} \in \mathbb{Z}^m} e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}-\mathbf{c}\|^2}}, \end{aligned} \tag{59}$$

where (d) is due to the fact that $\mathbf{x}_{\text{block}}$ is sampled only conditioned on $\mathbf{x}_{[-\text{block}]}$.

To show the Markov chain is irreducible, note that given a state $\mathbf{x}_i$ one can attain with positive probability in one step any state $\mathbf{x}_j$ which shares $>= (n-m)$ components with $\mathbf{x}_i$. Now, if $\mathbf{x}_i$ and $\mathbf{x}_j$ have, say, $d < n - m$ components in common, there is always a positive probability that after each step they get exactly one more component in common. So we can go in $n - d$ steps from one to the other. But as soon as $m >= 2$, we can assume that at the first step we get two more components in common, and then one at each further step, so we can go with positive probability in $n - d - 1$ steps.

On the other hand, it is clear to see that the number of steps required to move between any two states (can be the same state) is arbitrary without any limitation to be a multiple of some integer. Put another way, the chain is not forced into some cycle with fixed period between certain states. Therefore, the Markov chain is aperiodic.

As for reversibility, it is not hard to check that the following relationship holds

$$D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x}_i) P(\mathbf{x}_i, \mathbf{x}_j) = D_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x}_j) P(\mathbf{x}_j, \mathbf{x}_i) \tag{60}$$

with the same expression $\frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}_i-\mathbf{c}\|^2}}{\sum_{\mathbf{x}\in\mathbb{Z}^n} e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}-\mathbf{c}\|^2}}$ · $\frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}_j-\mathbf{c}\|^2}}{\sum_{\mathbf{x}_{\text{block}}\in\mathbb{Z}^m} e^{-\frac{1}{2\sigma^2}\|\mathbf{B}\mathbf{x}-\mathbf{c}\|^2}}$. Thus, the conclusion follows, completing the proof. ■

After showing the Markov chain induced by Gibbs-Klein algorithm is ergodic, it is straightforward to arrive at the geometric ergodicity by the same arguments of Corollary 1 and Lemma 2, and its proof is omitted here.

Corollary 2: Given the invariant lattice Gaussian distribution $D_{\Lambda,\sigma,\mathbf{c}}$, the Markov chain induced by Gibbs-Klein algorithm is geometrically ergodic $\|P^t(\mathbf{x},\cdot) - D_{\Lambda,\sigma,\mathbf{c}}\|_{TV} \leq M(\mathbf{x})\varrho_{block}^t$ with $\varrho_{block} \leq \varrho$.

VI. Simulation Results

In this section, the performance of Gibbs-based sampling schemes for lattice Gaussian distribution are exemplified in the context of MIMO detection.

Specifically, simulation results for an $n \times n$ MIMO system with a square channel matrix containing i.i.d. Gaussian entries are presented. The i-th entry of the transmitted signal $\mathbf{x}$, denoted as x_i, is a modulation symbol taken independently from a Q^2-QAM constellation $\mathcal{X} \in \mathbb{Z}$ with Gray mapping. Meanwhile, it is assumed a flat fading environment, where the channel matrix $\mathbf{H}$ contains uncorrelated complex Gaussian fading gains with unit variance and remains constant over each frame duration. Let E_b represents the average power per bit at the receiver, then $E_b/N_0 = n/(\log_2(M)\sigma_w^2)$ holds where M is the modulation level and σ_w^2 is the noise power.[7] Then, we can construct the system model as $\mathbf{c} = \mathbf{H}\mathbf{x} + \mathbf{w}$ and this decoding problem of $\widehat{\mathbf{x}} = \underset{\mathbf{x}\in\mathcal{X}^n}{\arg\min}\|\mathbf{c} - \mathbf{H}\mathbf{x}\|$ can be solved by sampling over the discrete Gaussian distribution

$$P_{\mathcal{L}(\mathbf{H}),\sigma,\mathbf{c}}(\mathbf{x}) = \frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{H}\mathbf{x}-\mathbf{c}\|^2}}{\sum_{\mathbf{x}\in\mathcal{X}^n} e^{-\frac{1}{2\sigma^2}\|\mathbf{H}\mathbf{x}-\mathbf{c}\|^2}} \tag{61}$$

because the optimal solution has the largest probability making it most likely be encountered by sampling (this complex decoding system is straightforward to be extended to the real-valued system [22], [52]). For this reason, we examine the decoding error probabilities to approximately compare the convergence rates of Markov chains. Meanwhile, given the definition of geometric ergodicity shown in Definition 2, the choice of the starting state $\mathbf{x}$ also has an impact upon the convergence performance. Here, Babai rounding algorithm (also known as zero-forcing decoding) is applied to output the suboptimal result for the initial Markov state [53].

Fig. 3 depicts the bit error rates (BER) of the different sampling schemes in a 6×6 uncoded MIMO system with 4-QAM. The SNR is fixed as $E_b/N_0 = 10$ dB. This corresponds to lattice dimension $n = 12$. The performance of zero-forcing (ZF) and maximum-likelihood (ML) decoding are shown as benchmarks. Meanwhile, the lattice-reduction-aided decoding scheme ZF-LLL is also provided for a better illustration. For a fair comparison, we follow Klein's choice of $\sigma = \min_{1\leq i\leq n}\|\widehat{\mathbf{b}}_i\|/\sqrt{\log n}$ and run the univariate sampling in both MWG and Gibbs algorithm for n times as a full iteration. Additionally, the parallel tempering technique that fastens the mixing by utilizing the tuning temperatures is also illustrated. For the consideration of computational complexity, only two Markov chains are applied for parallel tempering with $t_1 = 1$ and $t_2 = 2$ (i.e., $k = 2$), where the swap gap t_{swap} is set as 1. As shown in Fig. 3, the decoding performance improves with the number of Markov chain iterations. In particular, Klein's sampling is not as good as MCMC sampling schemes since it does not really produce Gaussian samples [32]. On the other hand, as demonstrated, the proposed MWG algorithm outperforms Gibbs algorithm under the same number of iterations, implying a better convergence performance. Meanwhile, parallel tempering is strongly recommended if parallel implementation is allowed. Note that parallel tempering is also applicable to Gibbs algorithm for performance improvement.

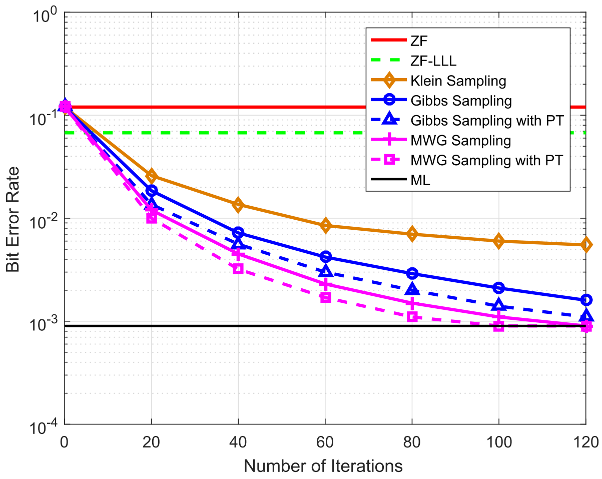

Fig. 3. Bit error rate versus iterations for the uncoded 6×6 MIMO system using 4-QAM with $E_b/N_0 = 10$ dB.

In Fig. 4 illustrates the BER decoding performance by Gibbs-based multivariate sampling over lattice Gaussian distribution, and its enhancement result by parallel tempering is also given. Specifically, in a 4×4 uncoded MIMO system with 16-QAM, which corresponds to lattice dimension $n = 8$, for a fair comparison, when the block size is m, we run block sampling for n/m times, and count this as a full iteration for Gibbs-Klein algorithm. Intuitively, this actually updates n components of $\mathbf{x}$ randomly in one iteration, which is comparable to the univariate sampling in standard Gibbs for n times. As can be seen clearly, with the same number of Markov chain iterations, the decoding performance improves with the block size, which indicates a faster convergence rate. These multiple updates are still determined by the conditional lattice Gaussian distribution, which takes the correlation structure into account. In this regard, block technique is worth trying for sampling decoding to enhance its performance.

Fig. 5 shows BER of different decoding schemes in a 8×8 uncoded MIMO system with 16-QAM. This corresponds to a lattice decoding scenario with dimension $n = 16$ while the

[7] In [50], the noise variance σ_w^2 is used as the sampling variance over the discrete Gaussian distribution, but this would lead to a stalling problem at high SNRs [51].

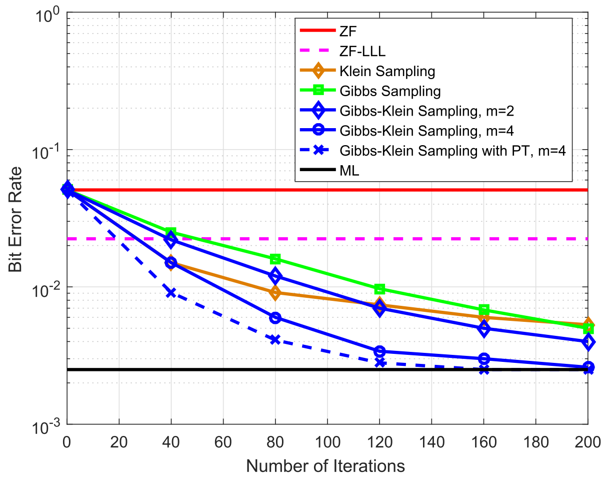

Fig. 4. Bit error rate versus iterations for the uncoded 4×4 MIMO system using 16-QAM with $E_b/N_0 = 15$ dB.

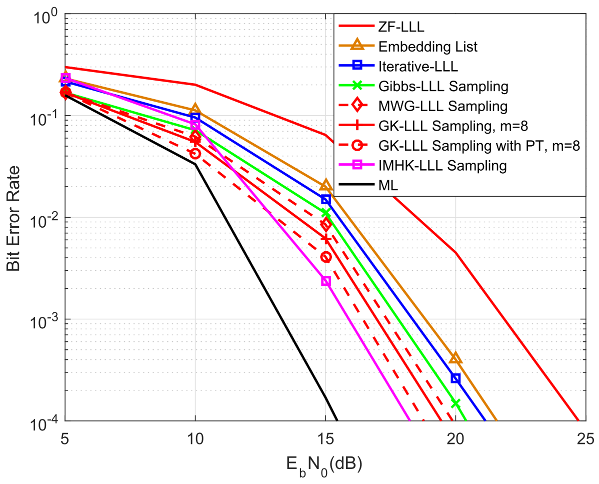

Fig. 5. Bit error rate versus E_b/N_0 for the uncoded 8×8 MIMO system using 16-QAM.

TABLE I

Average Complexity in Flops of Sampling Schemes for Uncoded MIMO System With 4-QAM

	n=8	n=12	n=16	n=20	n=24
Gibbs	240	757	1749	3372	5758
MWG	263	822	1896	3699	6302
MWG-PT$(k=2)$	542	1695	3987	7594	12956
GK$(m=4)$	291	927	2072	4056	7043
IMHK	303	961	2145	4176	7263

numbers of iterations of MCMC sampling schemes are set as $t = 50$. The SNR is fixed as $E_b/N_0 = 15$ dB. Besides MCMC decoding schemes, decoding schemes from MIMO detections like ZF-LLL, embedding list algorithm in [54] as well as iterative list decoding in [55] are also taken into account in the comparison. For a fair comparison, the list size of samples of embedding list algorithm and iterative list decoding are set as 50. Clearly, the proposed MWG and GK samplings outperform the standard Gibbs sampling due to the enhanced convergence rate, where further performance gain can be achieved under the help of parallel tempering. However, the decoding performance of Gibbs-based sampling algorithms are still less than that of independent Metropolis-Hastings-Klein (IMHK) in [29]. This can be interpreted by two reasons. Firstly, since the exact convergence rate of IMHK can be accessed, decoding optimization with respect to it can be carried out, which leads to a better decoding performance. For example, $\sigma = \min_i \|\widehat{\mathbf{b}}_i\|/(2\sqrt{\pi})$ was derived in [20] for IMHK while such a work is lacking for Gibbs-based sampling, which means the choice of $\sigma = 2$ that we use here is far away from the optimum. Secondly, it has shown in [20] that LLL reduction is well suited for IMHK to enhance the convergence rate. However, here for Gibbs sampling we only apply LLL as a preprocessing to yield the high quality initial point, and how to incorporate LLL into the operation of Gibbs smoothly should be further studied in future.

Table I shows the average complexity comparison in flops of the Gibbs-based sampling schemes with different system dimensions, where the flops evaluation scenario that we use comes from [56]. More specifically, the $n/2 \times n/2$ uncoded MIMO systems are applied with 4-QAM and the SNR is fixed as $E_b/N_0 = 15$ dB. Note that here we consider the average induced flops in every single Markov move (i.e., one iteration). Clearly, compared to standard Gibbs sampling, there is only a slight complexity increment with respect to the proposed MWG sampling, which is mainly due to the mechanism induced by acceptance ratio. Furthermore, the application of parallel tempering introduces more flops as another Markov chain is invoked for the exchange of Markov states. Additionally, as for Gibbs-Klein algorithm, more complexity will be consumed due to the proposed blocked strategy. Nevertheless, if the blocked sampling is performed by enumerating the state space of $\mathbf{x}_{\text{block}}$, exponential complexity increment would be introduced as the sampling space increases exponentially with the block size, which means significant complexity reduction is achieved by the proposed GK algorithm. On the other hand, the complexity cost of IMHK is more than that of Gibbs sampling due to the usage of proposal distribution.

VII. Conclusion

In this paper, the classic Gibbs algorithm for lattice Gaussian sampling is studied in full detail. By resorting to spectral radius of the forward operator, a comprehensive analysis is conducted to prove the geometric ergocidity of Gibbs algorithm for lattice Gaussian sampling, which means the underlying Markov chain converges to the lattice Gaussian distribution in an exponential way. Moreover, by showing the spectral radius of the forward operator exactly characterizes the convergence rate, analysis and optimization are performed to further enhance the convergence performance. Metropolis-within-Gibbs (MWG) and Gibbs-Klein (GK) algorithms for univariate and multivariate sampling are proposed respectively. Meanwhile, the validity of Gibbs-Klein algorithm for blocked sampling is confirmed by ergodicity. Therefore, blocked sampling can be efficiently performed with a flexible block size determined by the provided standard deviation.

Acknowledgment

The author would like to thank Dr. Cong Ling (Imperial College London, UK), Prof. Guillaume Hanrot and Prof. Damien Stehlé (École Normale Supérieure de Lyon, France) for their helpful discussions and insightful suggestions.

References

[1] W. Banaszczyk, "New bounds in some transference theorems in the geometry of numbers," *Math. Annalen*, vol. 296, no. 1, pp. 625–635, 1993.

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

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] H. Mirghasemi and J.-C. Belfiore, "Lattice code design criterion for MIMO wiretap channels," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Oct. 2015, pp. 277–281.

[9] 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.

[10] 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.

[11] T. Fehenberger, A. Alvarado, G. Böcherer, and N. Hanik, "On probabilistic shaping of quadrature amplitude modulation for the nonlinear fiber channel," *J. Lightw. Technol.*, vol. 34, no. 21, pp. 5063–5073, Nov. 15 2016.

[12] 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.

[13] D. Cash, D. Hofheinz, E. Kiltz, and C. Peikert, "Bonsai trees, Or how to delegate a lattice basis," in *Proc. EUROCRYPT*, May 2010, pp. 523–552.

[14] V. Lyubashevsky, C. Peikert, and O. Regev, "On ideal lattices and learning with errors over rings," in *Proc. EUROCRYPT*, May 2010, pp. 1–23.

[15] C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," in *Proc. CRYPTO*, Aug. 2013, pp. 75–92.

[16] T. Oder, T. Pöppelmann, and T. Güneysu, "Beyond ECDSA and RSA: Lattice-based digital signatures on constrained devices," in *Proc. 51st ACM/EDAC/IEEE Design Automat. Conf. (DAC)*, Jun. 2014, pp. 1–6.

[17] D. Aggarwal, D. Dadush, O. Regev, and N. Stephens-Davidowitz, "Solving the shortest vector problem in 2^n time via discrete Gaussian sampling," in *Proc. STOC*, Jun. 2015, pp. 733–742.

[18] D. Aggarwal, D. Dadush, and N. Stephens-Davidowitz, "Solving the closest vector problem in 2^n time—The discrete Gaussian strikes again!" in *Proc. IEEE 56th Annu. Symp. Found. Comput. Sci.*, Oct. 2015, pp. 563–582.

[19] P. Klein, " Finding the closest lattice vector when it's unusually close," in *Proc. 11th Annu. ACM-SIAM Symp. Discrete Algorithms*, Feb. 2000, pp. 937–941.

[20] 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.

[21] 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.

[22] 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.

[23] S. Yang and L. Hanzo, "Fifty years of MIMO detection: The road to large-scale MIMOs," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1941–1988, 4th Quart. 2015.

[24] K. Luo and A. Manikas, "Joint transmitter—Receiver optimization in multitarget MIMO radar," *IEEE Trans. Signal Process.*, vol. 65, no. 23, pp. 6292–6302, Dec. 2017.

[25] H. Cheng, Y. Xia, Y. Huang, L. Yang, and D. P. Mandic, "A normalized complex LMS based blind I/Q imbalance compensator for GFDM receivers and its full second-order performance analysis," *IEEE Trans. Signal Process.*, vol. 66, no. 17, pp. 4701–4712, Sep. 2018.

[26] Q. Wu, G. Ding, J. Wang, and Y. D. Yao, "Spatial-temporal opportunity detection for spectrum-heterogeneous cognitive radio networks: Two-dimensional sensing," *IEEE Trans. Wireless Commun.*, vol. 12, no. 2, pp. 516–526, Feb. 2013.

[27] J. Zhuang, H. Xiong, W. Wang, and Z. Chen, "Application of manifold separation to parametric localization for incoherently distributed sources," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2849–2860, Jun. 2018.

[28] M. Xiang, B. S. Dees, and D. P. Mandic, "Multiple-model adaptive estimation for 3-D and 4-D signals: A widely linear quaternion approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 1, pp. 72–84, Jan. 2019.

[29] 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.

[30] 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.

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

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

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

[34] S. Lyu and C. Ling, "Boosted KZ and LLL algorithms," *IEEE Trans. Signal Process.*, vol. 65, no. 18, pp. 4784–4796, Sep. 2017.

[35] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains Mixing Time*, Providence, RI, USA: American Mathematical Society, 2008.

[36] S. P. Meyn and R. L. Tweedie, *Markov Chains Stochastic Stability*. Cambridge, U.K.: Univ. Press, 2009.

[37] L. Tierney, "Markov chains for exploring posterior distributions," *Ann. Statist.*, vol. 22, pp. 1701–1728, Dec. 1994.

[38] G. O. Roberts, and J. S. Rosenthal, "General state space Markov chains and MCMC algorithms," *Probab. Surv.*, vol. 1, pp. 20–71, Apr. 2004.

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

[40] J. A. Fill, "Eigenvalue bounds on convergence to stationarity for nonreversible Markov chains, with an application to the exclusion process," *Ann. Appl. Probab.*, vol. 1, no. 1, pp. 62–87, 1991.

[41] I. Kontoyiannis and S. P. Meyn, "Geometric ergodicity and the spectral gap of non-reversible Markov chains," *Probab. Theory Rel. Fields*, vol. 154, pp. 327–339, Oct. 2012.

[42] J. S. Liu, W. H. Wong, and A. Kong, "Covariance structure and convergence rate of the Gibbs sampler with various scans," *J. Roy. Stat. Soc., Ser. Phys. Rev. B*, vol. 57, no. 1, pp. 157–169, Jan. 1995.

[43] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, Apr. 1970.

[44] J. S. Liu, "Peskun's theorem and a modified discrete-state Gibbs sampler," *Biometrika*, vol. 83, no. 3, pp. 681–682, 1996.

[45] A. Mira, "Ordering and improving the performance of Monte Carlo Markov chains," *Stat. Sci.*, vol. 16, no. 4, pp. 340–350, 2001.

[46] C. J. Geyer and E. A. Thompson, "Annealing Markov chain Monte Carlo with applications to ancestral inference," *J. Amer. Stat. Assoc.*, vol. 90, pp. 909–920, Jul. 1993.

[47] D. J. Earl and M. W. Deem, "Parallel tempering: Theory, applications, and new perspectives," *Phys. Chem. Chem. Phys.*, vol. 7, no. 23, pp. 3910–3916, 2005.

[48] O. Regev, "On lattice, learning with errors, random linear codes, and cryptography," *J. ACM*, vol. 56, no. 6, p. 34, Sep. 2009.

[49] Z. Brakerski, A. Langlois, C. Peikert, O. Regev, and D. Stehlé, "Classical hardness of learning with errors," in *Proc. 45th Annu. ACM Symp. Theory Comput.*, Jun. 2013, pp. 575–584.

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

[51] A. Kumar, S. Chandrasekaran, A. Chockalingam, and B. S. Rajan, "Near-optimal large-MIMO detection using randomized MCMC and randomized search algorithms," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2011, pp. 1–5.

[52] Y. Xia and D. P. Mandic, "Augmented performance bounds on strictly linear and widely linear estimators with complex data," *IEEE Trans. Signal Process.*, vol. 66, no. 2, pp. 507–514, Jan. 2018.

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

[54] 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.

[55] 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.

[56] H. Qian, "Counting the floating point operations (FLOPS)," *MATLAB Central File Exchange*, vol. 23, Jun. 2015, Art. no. 50608.

Zheng Wang 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, The 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 with Imperial College London. From 2016 to 2017, he was a Senior Engineer with the Radio Access Network Research and Development Division, Huawei Technologies Company, Ltd. He is currently an Assistant Professor with the College of Electronic and Information Engineering, NUAA. His current research interests include lattice methods for wireless communications, MIMO systems, and physical-layer security.