JOURNAL OF COMPUTATIONAL PHYSICS **121**, 238–245 (1995)

Improvement of Efficiency in Generating Random *U*(1) Variables with Boltzmann Distribution (von Mises Distribution Revisited)

TETSUYA HATTORI* AND HIDEO NAKAJIMA†

Faculty of Engineering, Utsunomiya University, Ishii, Utsunomiya 321, Japan

Received October 1, 1992; revised October 3, 1994

A method for generating random $U(1)$ variables with Boltzmann distribution (i.e., realizing von Mises distribution) is presented. Based on the rejection method, high efficiency is achieved for the whole range of temperatures or coupling parameters, which makes the present method especially suitable for (parallel or pipeline) vector processing machines. Results of computer runs are presented to illustrate the efficiency. An idea to find the algorithm is presented, which is applicable to other distributions of interest in Monte Carlo simulations. © 1995 Academic Press, Inc.

1. INTRODUCTION

In this paper we give an algorithm based on the rejection method, for realizing the von Mises distribution [8; 3, Section IX.7.3], or, equivalently, an algorithm for generating random $U(1)$ numbers to update a variable of a canonical ensemble in the Monte Carlo simulation of $U(1)$ spin systems or $U(1)$ lattice gauge theories. A method given by [1] has been known to be very efficient [3, Section IX.7.3]. Our method provides an improvement to this method.

Our method has a special feature that it has a very high acceptance rate *uniformly* in the parameter of the distribution. We also pay attention on having as few "if-branches" (conditional jumps) as possible. Interest on these points comes from the recent activity in the study of Monte Carlo simulation for the numerical study of quantum field theories in lattice formalism and statistical mechanics of spin systems. In Monte Carlo simulations, one generates random numbers with given probability distributions, to "update" a spin or a gauge variable. Probability distributions have parameters, which carry the information of the neighboring spin states. We have fluctuations in the neighboring spins, resulting in the changes of the parameters within a single program. One faces the problem of finding an algorithm which maintains uniformly high efficiency as one varies the parameters. As far as we know, currently available (parallel or pipeline) vector processors work efficiently when there are no "if-branches" in the program; hence we are also interested in reducing the "if-branches."

We aim at giving a practical method, with support of rigorous proofs. We set up the problem in Section 2, formulate our strategy and give a practical solution in Section 3 and Section 4, respectively, with proofs in Appendix A, and give discussions in Section 5. An example of a program for our proposal is in Appendix B.

2. RANDOM *U*(1) VARIABLE AND REJECTION METHOD

A random $U(1)$ variable (a realization of von Mises distribution) is a sequence of numbers (the angle variables)

$$\theta_1,\ \theta_2,\ \theta_3,\ \ldots, \tag{1}$$

whose distribution $P([\theta, \theta + d\theta]) = f_a(\theta)\, d\theta$ is given by the density function $f_a(\theta) = N_a \exp(a \cos(\theta - \theta_0))$, where N_a is a normalization constant. By a shift of variable $\theta' = \theta - \theta_0$ if $a > 0$, and $\theta' = \theta - \theta_0 - \pi$ if $a < 0$, we may assume withhout loss of generality that $\theta_0 = 0$ and $a \geq 0$. Hence

$$f_a(\theta) = N_a \exp(a \cos \theta), \quad -\pi \leq \theta < \pi, a > 0,$$
$$\frac{1}{N_a} = \int_{-\pi}^{\pi} \exp(a \cos \theta)\, d\theta = 2\pi\, I_0(a). \tag{2}$$

$I_0(a)$ is the zeroth modified Bessel function of the first kind.

The inversion method has been unsuccessful for generating the random $U(1)$ variable, and the rejection method [3, Section II.3; 10; 2] is adopted to solve the problem [3, Section IX.7.3]. Let $\tilde{f}(\theta)$ be some approximate, normalized, density function to $f_a(\theta)$. Suppose that there is a monotone function h which satisfies $h(0) = -\pi$, $h(1) = \pi$, and

$$\tilde{f}(h(x)) \frac{dh}{dx}(x) = 1, \quad 0 < x < 1. \tag{3}$$

(For the moment, we suppress possible parameter dependences of $\tilde{f}$ and h.) Define a function g by

* E-mail address: hattori@tansei.cc.u-tokyo.ac.jp.
† E-mail address: nakajima@kinu.infor.utsunomiya-u.ac.jp.

0021-9991/95 $12.00
Copyright © 1995 by Academic Press, Inc.
All rights of reproduction in any form reserved.

$$g(x) = R(a)\frac{f_a(h(x))}{\tilde{f}(h(x))} = R(a)f_a(h(x))\frac{dh}{dx}(x), \quad 0 \le x < 1, \tag{4}$$

$$R(a) = \min_{-\pi\le\theta<\pi}\left\{\frac{\tilde{f}(\theta)}{f_a(\theta)}\right\}. \tag{5}$$

Let ω_j and ω_j' with $j = 1, 2, 3, \ldots$, be two sequences of independent uniform random variables with the probability distribution $P([\omega, \omega + d\omega]) = d\omega$, $0 \le \omega < 1$. Define a subsequence $\tilde{\omega}_i = \omega_{j_i}$, $i = 1, 2, 3, \ldots$, of the sequence $\{\omega_j\}$ by selecting the numbers $j = j_i$ that satisfy $\omega_j' \le g(\omega_j)$. Then the sequence $h(\tilde{\omega}_1)$, $h(\tilde{\omega}_2)$, $h(\tilde{\omega}_3)$, ..., is the random $U(1)$ variable. We call the rate of picking up $\tilde{\omega}_i$ out of ω_j the acceptance rate, which is equal to $R(a)$ in Eq. (5) [3, Section II.3.1]. $\tilde{f}$ and f_a are non-negative and normalized; hence $0 \le R(a) \le 1$.

To illustrate the problem, recall first the simplest choice of $\tilde{f}$, the uniform distribution [11]; $\tilde{f}(\theta) = 1/2\pi$, $h(x) = (2x - 1)\pi$. Hereafter, we refer to this choice as the "direct" method. The acceptance rate is $R(a) = \exp(-a)I_0(a)$, which is small for $a \gg 1$ [1]; $R(a) \approx (2\pi a)^{-1/2}$. In particular, $\inf_{a>0} R(a) = 0$: For large a the original distribution f_a has a large peak at $\theta = 0$, and the uniform distribution $\tilde{f}$ is not a good approximation [1]. We need to find a $\tilde{f}$ which is a good approximation to the original distribution f_a for all values of a, or equivalently, such $\tilde{f}$ that g is almost flat. In a sense, this is to find a family of distributions which interpolates the uniform distribution ($a = 0$) and the delta function distribution ($a = \infty$) expressible as a simple computer program.

A standard criterion for the choice of $\tilde{f}$ is [1; 3, Section II.3.1] that (i) it is easy to calculate h; (ii) $R(a)$ is large; and (iii) it is easy to calculate $g(x)$. As stated in Section 1, we here add that (iv) $\inf_{a>0} R(a)$ is large (uniformly high efficiency) and (v) "if-branches" are avoided. Also, we take into account the recent trends that many computers are equipped with co-processors which quickly calculate elementary functions such as $\tan x$, $\exp x$, and their inverse functions. This last point seems to be a standard assumption [3, Section I.1, Assumption 3].

3. APPROXIMATE DISTRIBUTIONS AND THE OPTIMIZATION OF THE ACCEPTANCE RATE

The density function f_a is an even function which has a sharp peak at $\theta = 0$ for large a. We require that $\tilde{f}$ is an even function which has two free parameters, one to adjust the sharp peak at $\theta \approx 0$ and one to adjust off-peak behavior, and that the corresponding function h in Eq. (3) has an analytic expression. The simplest choice satisfying these conditions is

$$\tilde{f}_{\alpha,\beta}(\theta) = \frac{\tilde{N}_{\alpha,\beta}}{2\cosh(\alpha\theta) + 2\beta}, \quad \alpha > 0, \beta > -1,$$

where $\tilde{N}_{\alpha,\beta}$ is a normalization constant;

$$\tilde{N}_{\alpha,\beta} = \begin{cases} \alpha\dfrac{\sqrt{\beta^2-1}}{2\,\mathrm{arc\,tanh}(AB)}, & \beta > 1, \\ \alpha\dfrac{1}{A}, & \beta = 1, \\ \alpha\dfrac{\sqrt{1-\beta^2}}{2\,\mathrm{arc\,tan}(AB)}, & -1 < \beta < 1, \end{cases} \tag{6}$$

with $A = \tanh(\pi\alpha/2)$, and $B = \sqrt{|\beta - 1|/(\beta + 1)}$. The corresponding function h in Eq. (3) is

$$h_{\alpha,\beta}(x) = \begin{cases} 2\alpha^{-1}\,\mathrm{arc\,tanh}(B^{-1}\tanh((2x-1)\,\mathrm{arc\,tanh}(AB))), & \beta > 1, \\ 2\alpha^{-1}\,\mathrm{arc\,tanh}((2x-1)A), & \beta = 1, \\ 2\alpha^{-1}\,\mathrm{arc\,tanh}(B^{-1}\tan((2x-1)\,\mathrm{arc\,tan}(AB))), & -1 < \beta < 1. \end{cases} \tag{7}$$

The next step is to choose $\alpha = \alpha(a)$ and $\beta = \beta(a)$ as functions of a. In principle, they should be chosen so as to optimize the acceptance rate $R = R(a)$. Here, we search for a solution that satisfies a condition that the minimum in the definition of $R(a)$ (i.e., in the right-hand side of Eq. (5)) is achieved at $\theta = 0$. We impose this condition to avoid "if-branches" in the resulting computer program. We have an argument that the optimal solution under this condition, which we shall refer to as the "optimized cosh" method, is given by choosing $\alpha = \alpha(a)$ and $\beta = \beta(a)$ in Eq. (7) to satisfy

$$\alpha(a) = \sqrt{3a - 1}, \qquad \beta(a) = 2 - \frac{1}{a}, \quad \text{if } a \ge a^0, \tag{8}$$

$$\frac{\cosh(\pi\alpha(a)) - 1}{\alpha(a)^2} = \frac{\exp(2a) - 1}{a}, \qquad \beta(a) = \frac{\alpha(a)^2}{a} - 1, \quad \text{if } a^0 > a \ge a^*. \tag{9}$$

Here a^0 and a^* are positive constants satisfying $a^0 > a^*$, uniquely determined by

$$\frac{\exp(2a^0) - 1}{a^0} = \frac{\cosh(\pi\sqrt{3a^0 - 1}) - 1}{3a^0 - 1}, \tag{10}$$

$$\frac{\exp(2a^*) - 1}{a^*} = \frac{\pi^2}{2}. \tag{11}$$

Numerically, $a^* \approx 0.79895368608398$ and $a^0 \approx 5.0422719051807$. The function $g = g_a$ in Eq. (4) is, for $a \ge a^*$,

$$g_a(x) = \exp(-aG_a(h_{\alpha(a),\beta(a)}(x))) \tag{12}$$

with

$$G_a(\theta) = 1 - \cos\theta - \frac{1}{a}\log\left(1 + \frac{1}{1+\beta(a)}(\cosh(\alpha(a)\theta) - 1)\right). \tag{13}$$

For the parameter range of $0 < a < a^*$, we have to take a limit $\alpha \downarrow 0$ with $\alpha(a)^2/(1 + \beta(a))$ fixed to $2\pi^{-2}(\exp(2a) - 1)$. We have, in place of Eq. (7),

$$h_\gamma(x) = \frac{1}{\gamma}\tan((2x-1)\arctan(\pi\gamma)), \tag{14}$$

where $\gamma = \gamma(a)$ is

$$\gamma(a) = \pi^{-1}\sqrt{\exp(2a) - 1}, \quad \text{if } 0 < a < a^*. \tag{15}$$

The function $g = g_a$ in Eq. (4) is

$$g_a(x) = \exp(-aG_a(h_{\gamma(a)}(x))) \tag{16}$$

with

$$G_a(\theta) = 1 - \cos\theta - \frac{1}{a}\log(1 + \gamma(a)^2\theta^2). \tag{17}$$

The distribution is reduced to the Cauchy distribution:

$$\tilde{f}_\gamma(\theta) = \frac{\tilde{N}_\gamma}{1+\gamma^2\theta^2}, \quad \tilde{N}_\gamma = \frac{\gamma}{2\arctan(\pi\gamma)}. \tag{18}$$

See Appendix A for the proof that these formulae correctly generate a random $U(1)$ variable, and arguments for our choice of the parameters.

The acceptance rate $R = R(a)$ for the "optimized cosh" method is given by

$$\begin{aligned} R(a) &= \frac{\tilde{N}_{\alpha(a),\beta(a)}}{2N_a\exp(a)(1+\beta(a))}, & a \geq a^*, \\ &= \frac{\gamma(a)}{2N_a\exp(a)\arctan(\pi\gamma(a))}, & a^* > a > 0, \end{aligned}$$

where N_a and $\tilde{N}_{\alpha,\beta}$ are defined in Eq. (2) and Eq. (6), respectively. Note the high acceptance rate for both small a and large a:

$$\begin{aligned} R(a) &\approx 1 - \frac{1}{3}a + \frac{71}{180}a^2, & a \ll 1, \\ R(a) &\to \frac{\sqrt{2\pi}}{2\log(2+\sqrt{3})} \approx 0.95167365657, & a \to \infty. \end{aligned}$$

The acceptance rate for the "optimized cosh" method keeps more than 0.9 for all values of a (Fig. 1). The infimum of $R(a)$ is attained at $a \approx 1.95292714301$ with the value $\inf_{a>0} R(a) \approx 0.905563958$.

4. PROPOSED ALGORITHM

The acceptance rate for the "optimized cosh" method is high, but to obtain the parameter $\alpha(a)$ for $a^* \leq a < a^0$, one has to solve a transcendent equation Eq. (9). Also, one has to use different formulae for $0 < a < a^*$, $a^* \leq a < a^0$, and $a^0 < a$, which will cause "if-branches" that will lower the efficiency when using with vectorized processors.

One may, for example, use the Newton method to solve equations numerically, but here instead we approximate the function $\alpha(a)$ in Eq. (9) directly by a function which is explicitly expressible on computer programs without using if-branches for all a and is designed to keep high acceptance rate. The if-branches are avoided in such a way that we have $-1 < \beta < 1$ for all $a \in (0, \infty)$. We give an example of a practical algorithm:

1. Define $\alpha(a)$ by

$$\begin{aligned} \alpha^2(a) = \min\{&a(2-\varepsilon), \\ &\max\{\varepsilon a, 0.4162(a-a^*)^2 + 1.5056(a-a^*)\}\}, \end{aligned}$$

where $\varepsilon = 10^{-3}$ and a^* is the value given below Eq. (11), and also define $\beta(a)$ by

$$\beta(a) = \frac{\alpha(a)^2}{a}\max\left\{1, \frac{2aQ_b}{\exp(2a)-1}\right\} - 1,$$

where

$$Q_b = \frac{\cosh(\pi\sqrt{\varepsilon_a}) - 1}{2\varepsilon_a}, \quad \varepsilon_a = a^*\varepsilon(1+\varepsilon).$$

2. Define functions $h_{\alpha,\beta}$ and g_a by

$$\begin{aligned} \tanh(\alpha h_{\alpha,\beta}(x)/2) = &\sqrt{(1+\beta)/(1-\beta)} \\ &\tan\left((2x-1)\arctan\left(\sqrt{(1-\beta)/(1+\beta)}\tanh\frac{\pi\alpha}{2}\right)\right) \end{aligned}$$

and

$$g_a(x) = \exp(a\cos h_{\alpha(a),\beta(a)}(x) - a) \frac{1 + \dfrac{1-\beta(a)}{1+\beta(a)}\tanh^2\left(\dfrac{\alpha(a)}{2}h_{\alpha(a),\beta(a)}(x)\right)}{1 - \tanh^2\left(\dfrac{\alpha(a)}{2}h_{\alpha(a),\beta(a)}(x)\right)}.$$

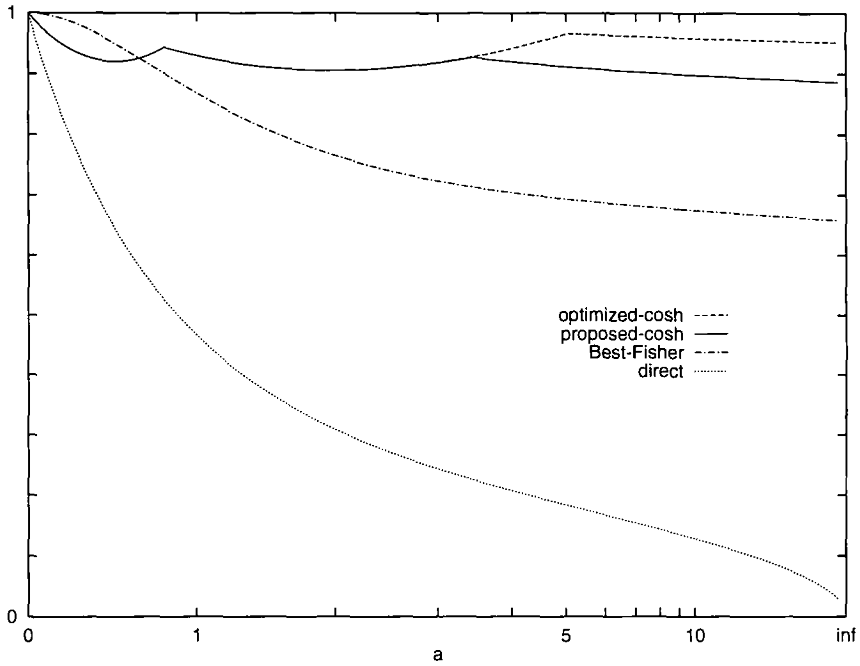

FIG. 1. Acceptance rate $R(a)$.

3. Let ω_j and ω_j' with $j = 1, 2, 3, \ldots$, be two sequences of independent random variables uniformly distributed in $[0, 1)$. Define a subsequence

$$\tilde{\omega}_i = \omega_j, \quad i = 1, 2, 3, \ldots,$$

of the sequence $\{\omega_j\}$ by selecting the numbers $j = j_i$ that satisfy $\omega_j' \leq g_a(\omega_j)$.

The sequence $h_{\alpha(a),\beta(a)}(\tilde{\omega}_1)$, $h_{\alpha(a),\beta(a)}(\tilde{\omega}_2)$, $h_{\alpha(a),\beta(a)}(\tilde{\omega}_3)$, ..., is the random $U(1)$ variable.

We will refer to this choice as the "proposed cosh" method. One can explicitly check that the conditions of the Proposition in Appendix A, or the alternative conditions in the remark to the proposition, hold with this choice of parameters, which implies that the "proposed cosh" method correctly generates random $U(1)$ variables. The acceptance rate $R(a)$ is greater than 0.9 for $a \leq 8.5$, decreasing for larger a, and the infimum is attained at $a = \infty$ (Fig. 1). We have

$$\inf_{a>0} R(a) = R(\infty) = \frac{\sqrt{2\pi\varepsilon}}{4 \arctan\sqrt{\varepsilon/(2-\varepsilon)}} \approx 0.8861530627.$$

5. EFFICIENCY TEST AND CONCLUDING REMARKS

The Best–Fisher's method [1] uses the rejection method, where the approximate distribution $\tilde{f}$ is the wrapped Cauchy distribution:

$$\tilde{f}(\theta) = \frac{1}{2\pi} \frac{\sqrt{1 + 4\gamma(a)^2}}{1 + 2\gamma(a)^2(1 - \cos\theta)},$$

$$\gamma(a)^2 = 4^{-1}(-1 + 2a + \sqrt{1 + 4a^2}).$$

(We changed the parametrization from [1] to show correspondence with Eq. (18).) The acceptance rate $R(a)$ is

$$R(a) = \frac{1}{\exp(a)N_a} \frac{ea}{4\pi\gamma(a)^2} \frac{\sqrt{1 + 4\gamma(a)^2}}{\exp(-a/(2\gamma(a)^2))},$$

where N_a is as in Eq. (2). In particular, $R(\infty) = \sqrt{e/(2\pi)} \approx 0.657744623$. This method is faster than the "direct" method for all $a > 0$ [1].

Before going into the comparison with our method, we would like to refer to two other methods in the recent literature, both of which are based on the rejection method, but use different approximate distributions $\tilde{f}$.

The approximate distribution adopted in [9] is

$$\tilde{f}(\theta) = \tilde{N}_a \exp\left(a\left(1 - \frac{2}{\pi}|\theta|\right)\right),$$

where $\tilde{N}_a$ is a normalization constant. The acceptance rate $R(a)$ for this choice is

$$R(a) = \frac{1}{N_a \exp(a)} \frac{a \exp(-ca)}{\pi(1 - \exp(-2a))},$$

$$c = \frac{2}{\pi} \text{arc sin} \left(\frac{2}{\pi}\right) + \sqrt{1 - (2/\pi)^2} - 1,$$

where N_a is as in Eq. (2). $R(a)$ is high for $a \leq 1$, but for large a, we have $R(a) \approx \sqrt{2a/\pi} \exp(-ca)$, which rapidly approaches 0 as $a \to \infty$. Therefore this choice suffers from the same problem as "direct" method that the time necessary to generate a random number indefinitely increases as $a \to \infty$. (In fact, we checked that the "proposed cosh" method is faster for $a > 3$.)

The approximate distribution adopted in the study of [4] is the Gaussian distribution

$$\tilde{f}(\theta) = \tilde{N}_a \exp(-\alpha(a)\theta^2),$$

where $\alpha(a) = 2\pi^{-2} \max(a, 4^{-1})$. (To be precise, the "direct" method is adopted for $a < 1.5$ and the Gaussian distribution for $a > 1.5$ in [4]. We focus our attention on $a > 1.5$, where the "direct" method is not effective.) For the algorithm of generating the Gaussian random variable, [4] quotes [7]. The original distribution $f_a(\theta)$ is now considered as a distribution on $\theta \in \mathbf{R}$, with $f_a(\theta) = 0$ if $|\theta| > \pi$. The acceptance rate $R(a)$ is [4]

$$R(a) = \frac{1}{N_a \exp(a)} \sqrt{2a/\pi^3}.$$

The acceptance rate for the Best–Fisher method is larger than this value for all $a > 0$. The method of [4] is very efficient, but by comparing the actual speed in producing one random number, we found that the Best–Fisher method is 1.3 times faster than the method of [4] for all $a > 0$. The long known Best–Fisher method is very efficient because it has relatively high efficiency with simple program.

Let us now compare our method with the Best–Fisher method. As noted in the introduction, our interest is the efficiency when used on vector processors, and without "if-branches" (conditional jumps). In the rejection method, one usually repeats the acceptance trial until acceptance occurs where one obtains a random variable. This procedure causes "if-branches."

In the Monte Carlo simulation of statistical systems (where one performs multi-dimensional integration numerically) combined with the rejection method [10], one may make a fixed number n of rejection trials for each spin variable (each degree of freedom) and go to next variable whether or not acceptance has occurred [5]. Let us call this set of n trials, an update. Trying an update with n trials effectively improves the acceptance rate R to $R_n = 1 - (1 - R)^n$. One is then interested in the speed to keep R_n above some fixed high value, say $R_n > 0.9$. Note that in this approach "if-branches" are avoided. This approach has been widely adopted in Monte-Carlo simulations of statistical systems.

Based on this approach, we performed efficiency test comparison of the Best–Fisher and the "proposed-cosh" method, with HITACH S3800 in Computer Centre of University of Tokyo which uses a pipelined vector processor. We measured the efficiency by the average VPU (vector processing unit) time consumption for an update, with iteration number n so chosen that for given b, $R_n > 0.9$ in the range $1 < a < b$. The average is taken over 4×10^6 updates. The measured acceptance rates for each a were in good agreement with the theoretical predictions in Section 3 and in Fig. 1 (within 0.1% accuracy). We found that for uniformly distributed a in the range $1 < a < 8$, the "proposed cosh" method is 1.2 times faster than the Best–Fisher method. The acceptance rate for the "proposed cosh" method is very high, uniformly in a, which makes this method efficient.

The consideration given in Section 3 implies that the use of the "cosh" distribution will be efficient for generating random variables taking values in a finite interval (e.g., $[-\pi, \pi]$) and whose distribution $f_a(\theta)$ is an even function with maximum at $\theta = 0$, and behaves like $f_a(\theta) \approx \text{const} - a\theta^2$ near $\theta = 0$, with a parameter $a > 0$ that controls the sharpness of the peak. This is a common feature of weight functions for statistical mechanical systems with one component spins and link variables. We have seen that the method is particularly suitable for vector processors, equipped with coprocessors for floating-point calculations.

Note, however, that the speed of an algorithm may depend on the situation, whether the parameter a is fixed or varying, range or value of a, and what kind of processor is used. For example, our method, as well as all the other methods discussed above, considers the case where a varies. For fixed a, there may be faster methods.

APPENDIX A

In this appendix, we give a proof that the "optimal cosh" method in Section 3 correctly gives the random $U(1)$-variables, and also we give the argument for the choice of the parameters.

We consider the case $a > a^*$. The proof for the case $0 < a \leq a^*$ is similar. By explicit calculation, one sees that $h_{\alpha,\beta}$ of Eq. (7) satisfies Eq. (3). Therefore it suffices to show that Eq. (12) satisfies Eq. (4) with Eq. (5). Define,

$$G(\theta) = \frac{1}{a} \log \left\{ \frac{\tilde{f}(\theta)}{f_a(\theta)} \frac{f_a(0)}{\tilde{f}(0)} \right\} \tag{19}$$

$$= 1 - \cos\theta - \frac{1}{a} \log \left(\frac{1}{1+\beta} (\cosh(\alpha\theta) + \beta) \right). \tag{20}$$

(G depends on three free parameters a, α, and β. We suppress the parameter dependences for the moment.) Then Eq. (4) and Eq. (5) imply

$$g(x) = \exp\left\{-a(G(h(x)) - \min_{\theta\in[-\pi,\pi]} G(\theta))\right\} \tag{21}$$

and

$$R(a) = \frac{\tilde{f}(0)}{N_a \exp(a)} \exp\left\{a \min_{\theta\in[-\pi,\pi]} G(\theta)\right\}.$$

Comparing Eq. (21) with Eq. (12), one sees that the results in Section 3 are correct if

$$\min_{\theta\in[-\pi,\pi]} G(\theta) = 0. \tag{22}$$

PROPOSITION. *Let $a > 0$, $\alpha > 0$, and $\beta > -1$. If G satisfies the three conditions*

$$G''(0) = 1 - \frac{1}{a}\frac{\alpha^2}{1+\beta} \geq 0, \tag{23}$$

$$G^{(4)}(0) = -1 - \frac{1}{a}\frac{\alpha^4(\beta-2)}{(1+\beta)^2} \geq 0, \tag{24}$$

$$G(\pi) = 2 - \frac{1}{a}\log\left(\frac{\cosh(\pi\alpha)+\beta}{1+\beta}\right) \geq 0, \tag{25}$$

then G satisfies

$$G(x) \geq 0, \quad -\pi \leq x \leq \pi. \tag{26}$$

Assume for the moment that this proposition is true. It is easy to see by explicit calculations that $\alpha = \alpha(a)$ and $\beta = \beta(a)$ defined by Eq. (8) or Eq. (9) satisfy the conditions (23), (24), and (25), and $\alpha(a) > 0$ and $\beta(a) > -1$, for all $a > a^*$. Since $G(0) = 0$, the proposition implies that Eq. (22) is satisfied for all $a > a^*$.

It remains to prove the proposition. Since G is an even function, it is sufficient to prove $G(x) \geq 0$ for $0 \leq x \leq \pi$.

The conditions (23) and (24) imply that $\alpha^2(2-\beta) \geq \beta + 1$. The equality holds if and only if $a(\beta+1) = \alpha^2$ and $\alpha^2 = 3a - 1$, which, with (25) implies $(3a-1)(\exp(2a)-1) \geq a(\cosh(\pi\sqrt{3a-1})-1)$. This is equivalent to $a > a^0$ ($>\frac{3}{2}$), where a^0 is defined by Eq. (10). Therefore, if we define a set D by

$$D = D_1 \cup D_2,$$
$$D_1 \equiv \{(a,\alpha,\beta) \in (0,\infty)^2 \times (-1,\infty) \mid \alpha^2(2-\beta) > \beta + 1 \geq \alpha^2/a\},$$
$$D_2 \equiv \{(a,\alpha,\beta) \in (0,\infty)^2 \times (-1,\infty) \mid \alpha^2 = 3a - 1 = a(\beta+1), a > 3/2\},$$

it is sufficient to prove that for all $(a,\alpha,\beta) \in D$ and $0 \leq x \leq \pi$, (25) implies $G(x) \geq 0$.

Step 1. Fix $(a,\alpha,\beta) \in D$. Put $g(x) \equiv G'(x)$. (This g has nothing to do with g in Eq. (21).) Then we have

$$\begin{aligned} f(x) &\equiv g(x) + g''(x) \\ &= \frac{\alpha^5 \sinh(\alpha x)}{a(\beta + \cosh(\alpha x))^3} h(\alpha^{-2}(\cosh(\alpha x) - 1)), \end{aligned} \tag{27}$$

where

$$h(y) \equiv -y^2 + (\beta - 2\alpha^{-2}(\beta+1))y + \alpha^{-4}(\beta+1)(\alpha^2(2-\beta) - (\beta+1)).$$

Since $(a,\alpha,\beta) \in D$, we see that there exists one and only one positive root $y = y_0$ of $h(y) = 0$ and that $h(y) > 0$, if $0 < y < y_0$, and $h(y) < 0$, if $y > y_0$. Therefore if we let $x = x_0$ to be the unique positive solution to the equation $\alpha^{-2}(\cosh(\alpha x) - 1) = y_0$, we have

$$f(x) > 0, \quad \text{if } 0 < x < x_0, \tag{28}$$

$$f(x) < 0, \quad \text{if } x > x_0. \tag{29}$$

Note that $g(0) = 0$ and $g'(0) \geq 0$ if $(a,\alpha,\beta) \in D$. With Eq. (27) and Eq. (28) we conclude that

$$g(x) > 0, \quad \text{if } 0 < x \leq x_0; 0 < x < \pi. \tag{30}$$

(The conclusion may be easily understood if one notes that Eq. (27) is an equation of motion of harmonic oscillation with external force f.)

Equations (27), (29), (30) imply

$$G'(x) = g(x) > 0, \quad \text{if } 0 < x \leq x_0; 0 < x < \pi. \tag{31}$$

$$g(x) + g''(x) < 0, \quad \text{if } x > x_0. \tag{32}$$

Step 2. Fix $\alpha > 0$ and $t \equiv a\alpha^{-2}(\beta + 1) \geq 1$, and let a vary with the restriction $(a, \alpha, \beta) \in D$. The allowed region of a differs by the values of t and α:

1. $t > 1$, or $t = 1$ and $\alpha \leq \sqrt{7/2}$. In this case, $(a, \alpha, \beta) \in D$ is equivalent to $a > (\alpha^2 + 1)/3$.

2. $t = 1$ and $\alpha > \sqrt{7/2}$. In this case, $(a, \alpha, \beta) \in D$ is equivalent to $a \geq (\alpha^2 + 1)/3$.

Note that $g(x) = g_a(x)$ is continuous (uniformly continuous on compact sets in $(0, \pi]$ w.r.t. x) and increasing in a, and $x_0 = x_0(a)$ is continuous in a. Also, $\lim_{a\to\infty} g(x) = \sin x$, uniformly on compact sets in $(0, \pi]$.

We claim that for every a (such that $(a, \alpha, \beta) \in D$), and for any x_1 and x_3 satisfying $g_a(x_1) > 0$, $g_a(x_3) > 0$, and $0 < x_1 < x_3 \leq \pi$, we have $g_a(x) > 0$, $x \in [x_1\ x_3]$. Assume this is wrong; assume that for $a = a_0$ and $0 < x_1 < x_2 < x_3 \leq \pi$ we have $g_{a_0}(x_1) > M$, $g_{a_0}(x_2) \leq 0$, and $g_{a_0}(x_3) > M$, where M is a positive constant. Since $g_a(x)$ is increasing in a, we have

$$g_a(x_1) > M, \quad g_a(x_3) > M, \quad a \geq a_0.$$

Put

$$q(a) \equiv \min_{x_1 \leq x \leq x_3} g_a(x).$$

Then $q(a)$ is continuous in a and $\lim_{a\to\infty} q(a) > 0$. Therefore there exists $a_1 \geq a_0$ such that $q(a_1) = 0$, which further implies that $g_{a_1}(x) \geq 0$, $x_1 \leq x \leq x_3$, and that there exists x_4 satisfying $x_1 < x_4 < x_3$ and $g_{a_1}(x_4) = 0$. In particular, $g_{a_1}(x_4) = 0$ and $g''_{a_1}(x_4) \geq 0$ hold, which contradicts Eq. (31) and Eq. (32). Hence the claim is proved.

Step 3. Fix $(a, \alpha, \beta) \in D$. The claim and Eq. (31) imply that either $g(x) = G'(x) > 0$ for $0 < x < \pi$, or there exists x' such that $0 < x' < \pi$, and $g(x) > 0$ for $0 < x < x'$, and $g(x) < 0$ for $x' < x \leq \pi$. Hence $G(x)$ is either increasing in $0 < x < \pi$ or has just one peak and no valley. Since $G(0) = 0$ and $G(\pi) \geq 0$, we have $G(x) > 0$, $0 < x < \pi$. This completes the proof.

Remark. It is easy to see that the above proof holds also if the conditions (23) and (24) are replaced by $G''(0) > 0$ and $\beta < 0$.

We now turn to the argument for the choice of the parameters. We want to choose the parameters so that the acceptance rate is large. As stated at the end of Section 2, we want a flat $g(x)$; hence we require $G''(0) = 0$. We impose the condition that the minimum of $R(a)$ is achieved at $\theta = 0$, which is equivalent to assuming Eq. (26). As a necessary condition, we have $G^{(4)}(0) \geq 0$ and $G(\pi) \geq 0$. (By the proposition, we know that these are sufficient to ensure Eq. (26).) As we want to have flat $G(x)$, it should be best to have either $G^{(4)}(0) = 0$ or $G(\pi) = 0$. If one draws a graph of these three conditions, in (α, β)-plane, one easily sees that the choice given in the Section 3 is the one that we are looking for.

APPENDIX B

```
      SUBROUTINE U1RND(A,H)
C A sample FORTRAN program for generating a random U(1) variable
C using the proposed cosh method.
C The first argument A is the parameter in the distribution.
C The second argument H returns a random U(1) variable when
C U1RND is called.  A uniform [0,1) source (random variable) RND
C is assumed and called twice.
      REAL A,AS,ALPH2A,ALPHA,B1,EPS,G,H,H1,P1,P2,PI,QB
      PARAMETER (PI=SNGL(3.14159265358979323846))
      PARAMETER (AS=SNGL(0.798953686083986))
      PARAMETER (P1=0.4162, P2=1.5056)
      PARAMETER (EPS=0.001)
C Correspondence of variables with the text (see Section 4):
C AS=a*, P1=p1, P2=p2, EPS=epsilon,
C ALPH2A=alpha**2/a, B1=sqrt((1-beta)/(1+beta)), ALPHA=alpha,
C H1=tanh(h(omega)*alpha/2), H=h(omega)=random U(1) variable,
C G=g_a(omega),  RND=omega and omega'.
C QB is a constant which should actually be calculated at
C the beginning of the program, not in the subroutine.
C COSH, SQRT, EXP, MIN, MAX, ATAN, TANH, TAN, LOG, COS are
C standard intrinsic functions.
C (The program used to test the efficiency in Section 5 is
C slightly involved than this sample program, to make it
C efficient for vector processors.)
      QB=(COSH(SQRT(AS*(1+EPS)*EPS)*PI)-1d0)/AS/(1+EPS)/EPS/2
      ALPH2A=MIN(2-EPS, MAX(EPS, (P1*(A-AS)+P2)*(A-AS)/A))
      B1=SQRT(MIN(2, (EXP(2*A)-1)/A/QB)/ALPH2A-1)
      ALPHA=SQRT(ALPH2A*A)
 1    CONTINUE
      H1=TAN((2*RND-1)*ATAN(TANH(PI*ALPHA/2)*B1))/B1
      H=LOG((1+H1)/(1-H1))/ALPHA
      G=EXP(A*(COS(H)-1))*(1+(B1*H1)**2)/(1-H1**2)
      IF (G .LT. RND) GOTO 1
      RETURN
      END
```

ACKNOWLEDGMENTS

We thank Professor Y. Oyanagi for very helpful discussions and encouragements. We also greatly thank Professor M. Sibuya, Professor A. D. Sokal, and the first referee, for comments which were instructive, especially in revising Section 5. The references have been updated (which also led to revising Sections 1, 2, and 5) with the kind help of Professor M. Fushimi, Professor H. Sakasegawa, Professor M. Sibuya, Professor A. D. Sokal, and the two referees, whom we also thank. The research of T. Hattori is supported in part by a Grant-in-Aid for General Scientific Research from the Ministry of Education, Science and Culture.

REFERENCES

1. D. J. Best and N. I. Fisher, *Appl. Statist.* **28,** 152 (1979).
2. J. W. Butler, in *Symposium on Monte Carlo Methods,* edited by H. A. Meyer (Wiley, New York, 1956).
3. L. Devroye, *Non-uniform Random Variate Generation* (Springer-Verlag, New York, 1986).
4. R. G. Edwards, J. Goodman, and A. D. Sokal, *Nucl. Phys. B* **354,** 289 (1991).

5. K. Fredenhagen and M. Marcu, *Phys. Lett. B* **193,** 486 (1987).
6. M. Fushimi, *Random Variables* (Tokyo Univ. Press, Tokyo, 1989). [Japanese]
7. D. E. Knuth, *The art of computer programming,* Vol. II, 2nd ed. (Addison–Wesley, Reading, MA, 1981).
8. K. V. Mardia, *J. R. Statist. Soc. B* **37,** 349 (1975).
9. K. J. Moriarty, *Phys. Rev. D* **25,** 2185 (1982).
10. J. von Neumann, *Nat. Bur. Stand. Appl. Math.* **12,** 36 (1951).
11. M. J. Seigerstetter, discussion (pp. 411–412) of D. G. Kendall, *J. R. Statist. Soc. B* **36,** 365 (1974).
12. T. Tsuda, *Monte Carlo Methods and Simulation* (Baihuukan, Tokyo, 1969). [Japanese]