Computational Statistics and Data Analysis 71 (2014) 1001–1010

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Discretization-based direct random sample generation

Liqun Wang [a,b,*], Chel Hee Lee [c]

[a] Department of Statistics, University of Manitoba, Winnipeg, Manitoba, Canada R3T 2N2

[b] School of Science, Beijing Jiaotong University, Beijing, China

[c] The Collaborative Biostatistics Program, University of Saskatchewan, Saskatoon, Saskatchewan, Canada

ARTICLE INFO

Article history:
Received 1 July 2012
Received in revised form 5 June 2013
Accepted 6 June 2013
Available online 25 June 2013

Keywords:
Direct sampling
Discretization
Monte Carlo sampling
Multivariate random variate generation
R package
Visualization

ABSTRACT

An efficient Monte Carlo method for random sample generation from high dimensional distributions of complex structures is developed. The method is based on random discretization of the sample space and direct inversion of the discretized cumulative distribution function. It requires only the knowledge of the target density function up to a multiplicative constant and applies to standard distributions as well as high-dimensional distributions arising from real data applications. Numerical examples and real data applications are used for illustration. The algorithms are implemented in statistical software R and a package `dsample` has been developed and is available online.

1. Introduction

Multivariate random sample generation plays an important role in computational statistics and data analysis. It is a central part of solutions to many high-dimensional integration and optimization problems as well as Bayesian data analysis (e.g., Sotto et al., 2011). Many numerical methods and algorithms have been developed in the literature, most notably the Markov chain Monte Carlo (MCMC) methods. However, in real applications the MCMC have practical difficulties when dealing with high-dimensional, multi-modal distributions, as well as other technical problems such as convergence (e.g., O'Hagan et al., 2012).

To overcome difficulties encountered in many MCMC algorithms, Fu and Wang (2002) proposed a direct method for random sample generation based on the discretization of the sample space and the target distribution. This method is easy to implement and fairly efficient in generating a large sample from a relatively high dimensional distribution. It was subsequently applied to Bayesian finite mixture models with a varying number of components by Xue et al. (2005) and Wang and Fu (2007). It was also extended to a global optimization algorithm by Wang et al. (2004) that has been used in many engineering design and optimization problems. Discretization-based methods are also studied by other researchers, e.g., Malefaki and Iliopoulos (2009), and Punzo and Zini (2012). The direct sampling approach is also used by, e.g., An and Bentler (2012).

However, when dealing with distributions with long tails the Fu–Wang algorithm suffers computational efficiency loss due to the large number of strata (contours) created in very low probability regions. In this paper, we propose a different way to stratify the sample space and modify the Fu–Wang method accordingly. This modification leads to significant improvement of computational efficiency of the algorithm. We also implement both algorithms in the statistical software R (R Core Team, 2013).

* Corresponding author at: Department of Statistics, University of Manitoba, Winnipeg, Manitoba, Canada R3T 2N2. Tel.: +1 204 474 6270; fax: +1 204 474 7621.

E-mail addresses: liqun.wang@umanitoba.ca (L. Wang), chl948@mail.usask.ca (C.H. Lee).

0167-9473/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.csda.2013.06.011

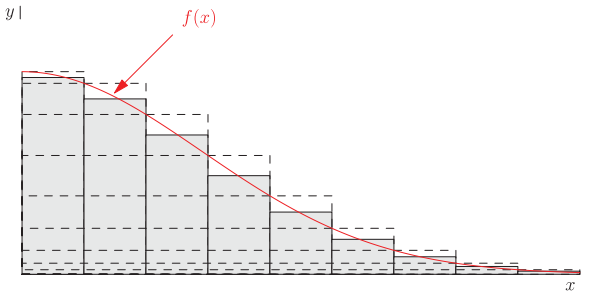

Fig. 1. Fu–Wang algorithm: in each contour the dashed line is the maximum, the solid line is the mid-height and the shaded area is the probability.

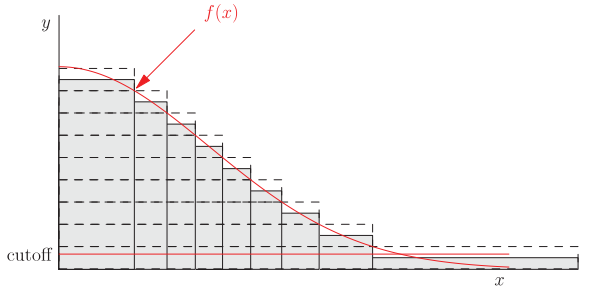

Fig. 2. Wang–Lee algorithm: in each contour the dashed line is the maximum, the solid line is the mid-height and the shaded area is the probability.

This paper is organized as follows. Section 2 introduces the proposed method. Section 3 develops the algorithms and their R implementations, and illustrates the method through numerical examples. Section 4 contains real data applications and further demonstrations of the proposed method. Finally, Section 5 contains conclusions and discussions.

2. Methodology

Suppose we are given a d-dimensional probability density function $f(x), x \in \mathbb{R}^d$ up to a multiplicative constant. The support of $f(x)$, $S^f \subset \mathbb{R}^d$, can be either bounded or unbounded. Our goal is to generate a random sample of size $m \geq 1$ from $f(x)$. As in Fu and Wang (2002) and Wang and Fu (2007), the proposed method consists of the following components: initialization and discretization of the sample space, contourization and construction of the discretized probability distribution, two-stage Monte Carlo sample generation, and visualization and update. These components are described in details below.

1. *Initialization:* First choose an initial compact set $C_0 \subset \mathbb{R}^d$ that contains the significant region of $f(x)$. If $f(x)$ has a bounded support S^f, then take $C_0 = S^f$. If $f(x)$ has an unbounded support, then choose $C_0 = S^f \cap [a, b]^d$, where $-\infty < a < b < \infty$ are chosen so that C_0 covers the significant region of $f(x)$. In practice, one may start with a reasonable guess of the interval based on the properties of the target density function and modify it in step 5 if necessary.

2. *Discretization:* Within C_0 generate a large number of random points to form a discrete base set $S_n^f = \{x_j \in C_0, j = 1, 2, \ldots, n\}$. The sequence may be deterministic (such as low discrepancy sequence) or stochastic (such as independent and uniformly distributed random vectors). Further, the points in S_n^f are reordered such that $f(x_i) \geq f(x_j)$, if $i < j$.

3. *Contourization:* Let $f_{\max} = \max_{x \in S_n^f} f(x)$ and $f_{\min} = \min_{x \in S_n^f} f(x)$. For a given integer $k \geq 1$, partition S_n^f into k contours $E_i = \{x \in S_n^f : (i-1)h \leq f_{\max} - f(x) < ih\}$, $i = 1, 2, \ldots, k$, where $h = (f_{\max} - f_{\min})/k$. Further define

$$P_i = \frac{h_i n_i}{\sum_{i=1}^{k} h_i n_i}, \quad i = 1, 2, \ldots, k,$$

where n_i is the number of base points in E_i, $h_i = f_{\max} - (i - 0.5)h$ is the mid-height of contour E_i, $i = 1, 2, \ldots, k-1$, and $h_k = w[f_{\max} - (k - 0.5)h]$. Here, the weight parameter w is used to adjust the tail probability of the lowest contour and the default value is $w = 0.5$.

4. *Sampling:* First, randomly sample m contours with replacement from $\{E_i\}_{i=1}^k$ according to probabilities $\{P_i\}_{i=1}^k$. Denote by m_i the number of occurrence of E_i in the m draws, where $\sum_{i=1}^k m_i = m$. Then for each $1 \leq i \leq k$, randomly sample m_i points with replacement from the contour E_i. All points thus drawn form the desired sample.

5. *Visualization and update:* First, using the sample generated in Step 4 to produce marginal histograms for all dimensions, which represent the marginal distributions of $f(x)$. These histograms allow one to visualize the significant region and the

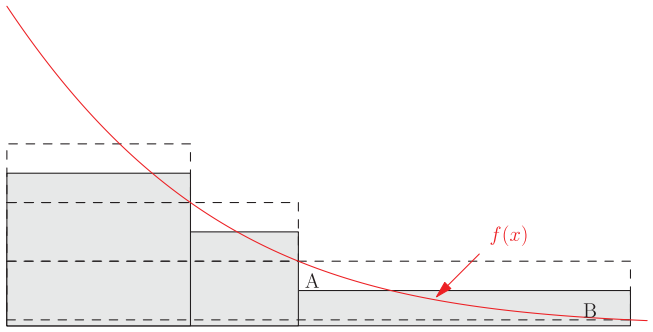

Fig. 3. Adjusting probability of the last contour.

negligible region of the sample space of $f(x)$. Let C_1 be the compact significant region thus identified. If $C_1 = C_0$ then stop the procedure and accept the sample from Step 4; otherwise replace C_0 with C_1 and go back to Step 2.

The main difference between the above procedure and the method of Fu and Wang (2002) and Wang and Fu (2007) is the contourization step 3. While in the Fu–Wang method the discrete sample space S_n^f is partitioned into contours of equal size (Fig. 1), here the range of functional values $(f_{\min}, f_{\max})$ is divided into equal sized intervals so that the contours are of unequal sizes (Fig. 2). Hence the sample space is partitioned in a "vertical" instead of a "horizontal" way, which reduces the number of redundant contours of very low probabilities and thus improves the computational efficiency.

Another novelty of the proposed method is the introduction of the weight parameter w that is used to adjust the probability of the lowest contour. Typically the last contour probability is over-estimated by as much as the difference between the area A and shaded area B in Fig. 3. This is especially the case when the target distribution has a long and thin tail. In this case the problem can be effectively avoided by adjusting the value of w. In general, if the distribution has a short tail, then w should be close to one. On the other hand, if the distribution has a long tail, then w should be 0.5 or lower. And the longer the tail, the smaller the value of w should be. This point will be illustrated through the examples in Sections 3 and 4.

Finally, since in step 2 the discrete points in S_n^f are ordered according to descending functional value, the first points in the first contour can be used to estimate the modes of $f(x)$. However, the accuracy of these estimates depends on the sparsity of the discrete points in the first contour.

3. Numerical examples and R package `dsample`

The algorithms for the methods of this paper and Fu and Wang (2002) have been developed using the statistical software language R (R Core Team, 2013). A package called `dsample` can be downloaded from the CRAN website http://cran.r-project.org/web/packages/dsample/ or from the R-Forge project site http://r-forge.r-project.org/projects/wanglee/. In this section we demonstrate the usage of the package using two benchmark examples in the literature. Each example has been specifically selected for the purpose of demonstrating certain features of the algorithm. Unless otherwise stated, in all examples the default parameter values are: the number of discrete base points $n = 10^7$, the number of contours $k = 10^5$, the weight of the last contour probability $w = 0.5$, and the target sample size $m = 10^3$.

3.1. A bimodal distribution

We first consider a bimodal distribution that has been used by West (1993) and Wang and Fu (2007). The density function up to a normalizing constant is given by

$$f(x_1, x_2) = (x_1(1-x_2))^5 (x_2(1-x_1))^3 (1 - x_1(1-x_2) - x_2(1-x_1))^{37}, \quad 0 < x_1, x_2 < 1. \tag{3.1}$$

This distribution has compact support $S^f = [0, 1]^2$ and two modes located in the corners (0, 0) and (1, 1) respectively. In this case it is straightforward to generate samples using the standard form of our algorithm with weight $w = 1$ since the distribution has a short tail. Using the package `dsample`, we first create the discretized sample space and distribution by the following commands where the seed number for random number generation is set for the purpose of reproducing the numerical results.

```
> set.seed(584479233)
> ndp <- 1e7
> x1 <- runif(ndp)
> x2 <- runif(ndp)
> val <- (x1*(1-x2))^5 * (x2*(1-x1))^3 * (1-x1*(1-x2)-x2*(1-x1))^37
> support <- as.data.frame(cbind(val, x1, x2))
```

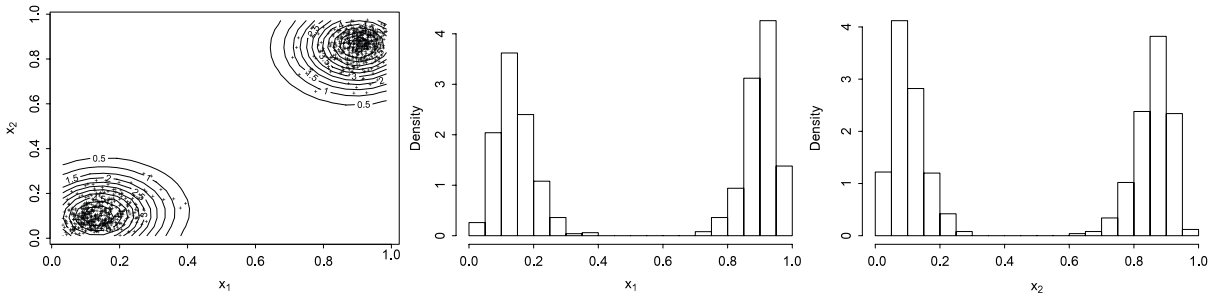

Fig. 4. Sample scatter and contour plots, and marginal histograms from density (3.1).

Table 1
Sample means, standard deviations and modes from density (3.1).

	Fu–Wang		Wang–Lee	
Mean	0.5107	0.4672	0.5220	0.4815
SD	0.3805	0.3825	0.3819	0.3810
	0.9258	0.8804	0.1191	0.0741
Modes	0.9236	0.8777	0.9225	0.8803
	0.1178	0.0764	0.9243	0.8823

Then we generate a random sample from (3.1) using the Wang–Lee algorithm by

```
> smpl <- sample.wl(X=support, nc=1e5, n=1e3, wconst=1)
```

The contour and scatter plots, and two marginal histograms of the sample are shown in Fig. 4.

We also produce the numerical output of the summary statistics by

```
> summary(smpl)
```

For comparison we also draw a sample using the Fu–Wang algorithm and produce the summary statistics by

```
> summary(sample.fw(X=support, nc=1e5, n=1e3))
```

The summary statistics of both samples are shown in Table 1, where we included the first 3 discrete points in the first contour as estimates of the functional modes.

3.2. A mixture distribution

The second example is a mixture of three normal distributions used by Liang et al. (2007):

$$f(x_1, x_2) = \frac{1}{3}N\left[\begin{pmatrix}-8\\-8\end{pmatrix}, \begin{pmatrix}1 & 0.9\\0.9 & 1\end{pmatrix}\right] + \frac{1}{3}N\left[\begin{pmatrix}6\\6\end{pmatrix}, \begin{pmatrix}1 & -0.9\\-0.9 & 1\end{pmatrix}\right] + \frac{1}{3}N\left[\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}1 & 0\\0 & 1\end{pmatrix}\right]. \tag{3.2}$$

This distribution has three separate modes and unbounded support. Since the two extreme component distributions are located at $(-8, -8)$ and $(6, 6)$ respectively and have unit variance, it is reasonable to start with the initial compact set $C_0 = [-11, 9]^2$. However, an initial sample showed that the marginal histograms were cut-off on both ends, indicating that this region was too narrow. Therefore the initial set C_0 was enlarged to $C_1 = [-12, 11]^2$, from which the discretized sample space and distribution are created through

```
> set.seed(584479233)
> require(mnormt)
> ndp <- 1e7
> x1 <- runif(ndp, -12, 11)
> x2 <- runif(ndp, -12, 11)
> x <- cbind(x1,x2)
> val <- 1/3 * dmnorm(x, mean=c(-8,-8), varcov=matrix(c(1,0.9,0.9,1), ncol=2)) + 1/3 * dmnorm(x, mean=c(6,6), varcov=
    matrix(c(1,-0.9,-0.9,1), ncol=2)) + 1/3 * dmnorm(x, mean=c(0,0), varcov=matrix(c(1,0,0,1), ncol=2))
```

and a sample is drawn using the Wang–Lee algorithm as

```
> support <- as.data.frame(cbind(val, x1, x2))
> smpl1 <- sample.wl(X=support, nc=1e5, n=1e3, wconst=0.5)
```

The scatter and contour plots, and two marginal histograms are shown in Fig. 5.

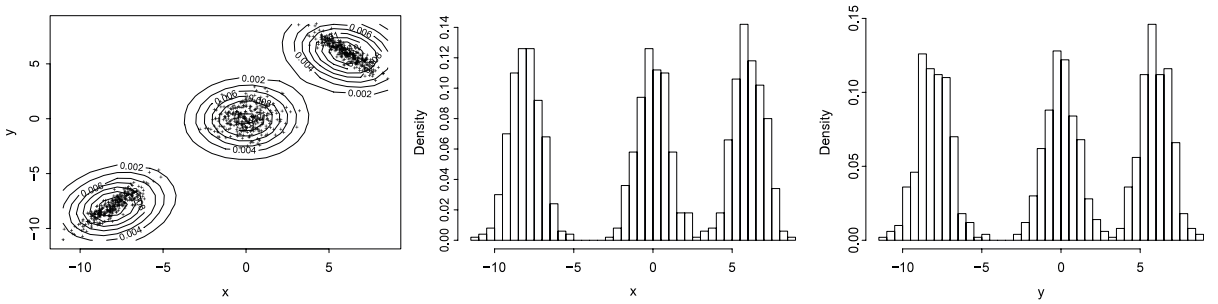

Fig. 5. Sample scatter and contour plots, and histograms from density (3.2).

Table 2
Sample means and standard deviations from density (3.2).

	Fu–Wang		Wang–Lee	
Mean	−0.7010	−0.7831	−0.6237	−0.6386
SD	5.8425	5.7940	5.8929	5.8664
	−8.0378	−8.0457	−7.9717	−7.9559
Modes	6.0460	5.9658	−7.9484	−7.9611
	−8.0572	−8.0483	6.0130	5.9598

We can see that the histograms are fairly smooth and die down without apparent cut-off, indicating that the support C_1 is adequate. Therefore we accept the sample and produce the summary statistics with

```
> summary(smpl1)
```

Further, we draw another sample using the Fu–Wang algorithm as

```
> smpl2 <- sample.fw(X=support, nc=1e5, n=1e3)
> summary(smpl2)
```

The summary statistics from both samples are shown in Table 2.

4. Applications

In this section we apply the proposed method to some real data applications and further demonstrate the usage of our method and algorithm. We also compare our method with other methods proposed in the literature. Since in most cases both Fu–Wang and Wang–Lee algorithms give very similar numerical results, except for the last example we only present the results calculated using the later algorithm.

4.1. Beetles dataset

This data set has been used in many dose–response studies of the effect of gaseous carbon disulfide (CS2) on the death risk of adult flour beetles. Prentice (1976) analyzed this data set using the generalized logit model

$$P(\text{death}|x) \equiv h(x) = \left[1 + \exp\left(-\frac{x-\mu}{\sigma}\right)\right]^{-\nu}$$

and obtained the maximum likelihood estimates of unknown parameters μ, σ^2 and $\nu > 0$. Later Carlin and Gelfand (1991) proposed a Gibbs sampler and Carlin and Louis (1996) proposed a Metropolis–Hasting algorithm to calculate the posterior distribution of the unknown parameters. They used the prior distributions $\nu \sim \mathcal{G}(a_0, b_0)$, $\mu \sim \mathcal{N}(c_0, d_0)$, $\sigma^2 \sim \mathcal{IG}(e_0, f_0)$, so that the full posterior density is given by

$$p(\mu, \sigma^2, \nu|\mathbf{x}, \mathbf{y}) \propto \left\{\prod_{i=1}^{k}[h(x_i)]^{y_i}[1-h(x_i)]^{n_i-y_i}\right\} \frac{\nu^{a_0-1}}{\sigma^{2(e_0+1)}} \exp\left[-\frac{1}{2}\left(\frac{\mu-c_0}{d_0}\right)^2 - \frac{\nu}{b_0} - \frac{1}{f_0\sigma^2}\right], \tag{4.1}$$

where $a_0 = 0.25$, $b_0 = 4$, $c_0 = 2$, $d_0 = 10$, $e_0 = 2.000004$ and $f_0 = 1000$. After initial trials the compact support was identified as $1.76 \le \mu \le 1.84$, $0.01 \le \sigma \le 0.03$ and $0.13 \le \nu \le 1.22$. In order to accelerate the convergence of their MCMC procedures and to facilitate the convergence diagnostics, Carlin and Gelfand (1991) and Carlin and Louis (1996) used the reparameterization $(\mu, \log(\sigma), \log(\nu))$. In contrast, our method can be applied straightforwardly without any parameter transformation. The final sample obtained by the Wang–Lee algorithm is shown in Fig. 6.

The posterior means, standard deviations and modes of μ, σ and ν are computed using the final sample and are shown in Table 3. For comparison, we also included the MLE of Prentice (1976), the mode estimates of Carlin and Gelfand (1991) and the percentile estimates of Carlin and Louis (1996), transformed to the original scales.

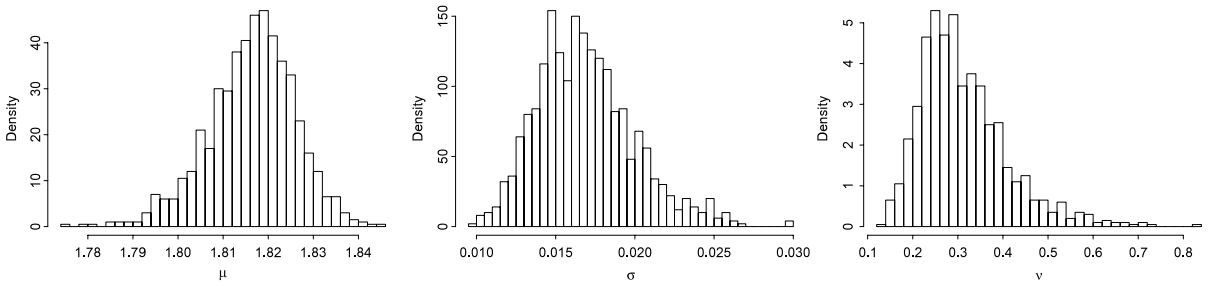

Fig. 6. Posterior marginal histograms of μ, σ, ν from (4.1).

Table 3
Posterior means, standard deviations, and modes of μ, σ, ν in (4.1).

	Wang–Lee			Other algorithms		
	μ	σ	ν	μ	σ	ν
Mean	1.8163	0.0168	0.3128	1.818	0.016	0.279
SD	0.0096	0.0030	0.0977	–	–	–
Mode	1.8172	0.0163	0.2915	1.81	0.0175	0.3362
2.5%	1.7951	0.0117	0.1743	1.78	0.013	0.199
50%	1.8170	0.0165	0.2939	1.81	0.019	0.374
97.5%	1.8341	0.0241	0.5706	1.83	0.27	0.779

4.2. Dugongs dataset

This is one of the benchmark examples in OpenBUGS (Spiegelhalter et al., 2007) and has been used in many studies of Gibbs sampler and MCMC algorithms. It contains the measurements of the length (y) and age (x) of 27 dugongs. Carlin and Gelfand (1991) studied the growth curve model $y|x \sim N(\alpha - \beta\gamma^x, \tau^{-1})$ with unknown parameters $\alpha, \beta, \tau > 0$ and $0 < \gamma < 1$. Malefaki and Iliopoulos (2009) proposed the prior distributions $\alpha \sim N(0, \tau_\alpha^{-1})I(\alpha > 0)$, $\beta \sim N(0, \tau_\beta^{-1})I(\beta > 0)$, $\gamma \sim U(0, 1)$ and $\tau \sim G(\kappa, \kappa)$ with $\tau_\alpha = \tau_\beta = 10^{-4}$ and $\kappa = 10^{-3}$. Therefore the likelihood function of $(\alpha, \beta, \gamma, \tau)$ is

$$L(\alpha, \beta, \gamma, \tau|\mathbf{x}, \mathbf{y}) \propto \tau^{\frac{n}{2}+a-1} \exp\left\{-a\tau - \frac{\tau}{2}\sum_{i=1}^{n}(y_i - \alpha + \beta\gamma^{x_i})^2\right\} \tag{4.2}$$

and the corresponding posterior density is

$$p(\theta|\mathbf{x}, \mathbf{y}) \propto L(\theta|\mathbf{x}, \mathbf{y}) \exp\left(-\tau k - \frac{\tau_\alpha\alpha^2}{2} - \frac{\tau_\beta\beta^2}{2}\right) \cdot I(\alpha > 0, \beta > 0, \tau > 0, 0 < \gamma < 1). \tag{4.3}$$

We generated a sample that is shown in Fig. 7, while the corresponding summary statistics are given in Table 4. These results are consistent with mean estimates obtained by Malefaki and Iliopoulos (2009) and OpenBUGS, and the mode estimates (transformed to the original scales) of Carlin and Gelfand (1991) using informative priors.

4.3. British coalmining data

Now consider a mixed continuous–discrete distribution and the data from British coalmining disasters from 1851 to 1962. Carlin et al. (1992) proposed a model where $y_i \sim \text{Poisson}(\theta t_i)$, $i = 1, 2, \ldots, \kappa$ and $y_i \sim \text{Poisson}(\lambda t_i)$, $t = \kappa + 1, \ldots, n$, with priors $\kappa \sim \mathcal{U}(1, N)$, $\theta \sim \mathcal{G}(a_1, b_1)$ and $\lambda \sim \mathcal{G}(a_2, b_2)$ and the hyperparameters $b_1 \sim \text{IG}(c_1, d_1)$ and $b_2 \sim \text{IG}(c_2, d_2)$ with $a_1 = a_2 = 0.5$, $c_1 = c_2 = 0$ and $d_1 = d_2 = 1$. Therefore the log-likelihood and the log-posterior densities are respectively (Fu and Wang, 2002)

$$\mathcal{L}(\kappa, \theta, \lambda|\mathbf{y}) = \left(\sum_{i=1}^{\kappa} y_i - 1/2\right)\log\theta + \left(\sum_{i=\kappa+1}^{n} y_i - 1/2\right)\log\lambda - \kappa\theta - (n - \kappa)\lambda \tag{4.4}$$

and

$$p(\kappa, \theta, \lambda, \alpha, \beta|\mathbf{y}) = l(\kappa, \theta, \lambda|\mathbf{y}) + 1.5\log\alpha + 1.5\log\beta - (\theta + 1)\alpha - (\lambda + 1)\beta. \tag{4.5}$$

Again after some trials the initial compact support was identified as $\kappa \in (30 : 50)$, $\theta \in [2.2, 4]$, $\lambda \in [0.6, 1.4]$, $\alpha \in [0, 2]$ and $\beta \in [0, 4]$. The marginal posterior distributions are shown in Fig. 8 that shows clearly that the change occurred around 1890 as supported by the estimation of the posterior mode at $\kappa = 41$. The approximate MLE and other summary statistics are given in Table 5 which also includes the mode estimates of Carlin et al. (1992) using the Gibbs sampler.

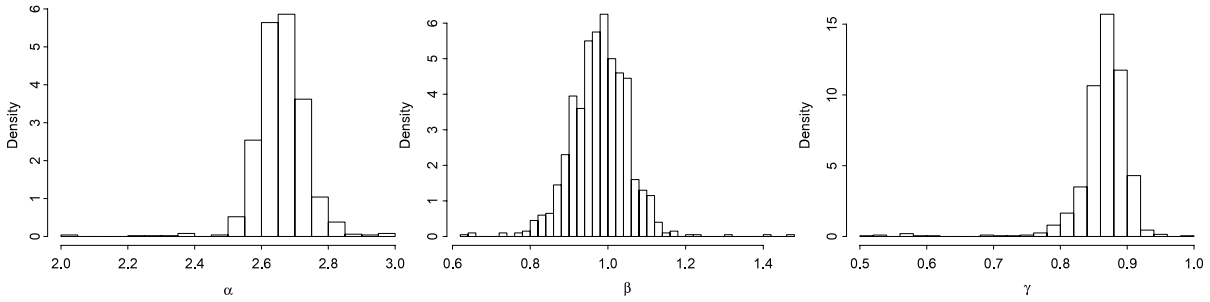

Fig. 7. Histograms of α, β, γ from (4.3).

Table 4
Sample means and modes of α, β, γ from (4.3).

	Wang–Lee			Other algorithms		
	α	β	γ	α	β	γ
Mean	2.6601	0.9791	0.8642	2.652	0.9729	0.8623
Mode	2.6710	0.9668	0.8770	2.6512	0.9861	0.8701

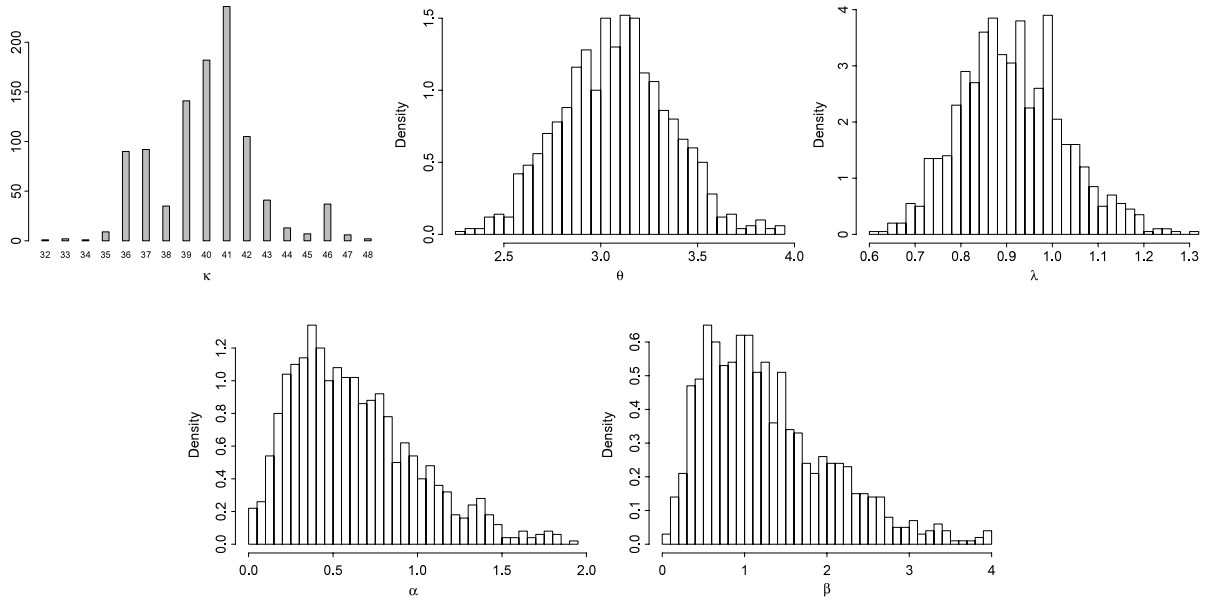

Fig. 8. Histograms of the sample from posterior (4.5).

Table 5
AMLE, sample means, standard deviations, and modes from (4.5).

	κ	θ	λ	α	β
	Wang–Lee				
AMLE	41	3.0765	0.9118	–	–
Mean	40.0160	3.0765	0.9118	0.6250	1.2815
SD	2.4778	0.2826	0.1133	0.3667	0.7608
Mode	41	2.9957	0.8995	0.3905	0.7629
	Gibbs sampler				
Mode	41	3.06	0.89	–	–

4.4. Nuclear power failures data

This dataset was first analyzed by Gaver and O'Muircheartaigh (1987) and later by many other researchers. A popular model is $y_i \sim \text{Poisson}(\lambda_i t_i)$, $i = 1, 2, \ldots, 10$ and priors are $\lambda_i \sim \text{Gamma}(\alpha, \beta)$ and $\beta \sim \text{Gamma}(\gamma, \delta)$. The full posterior is

Table 6
Rates, sample modes, means, and standard deviations from (4.6).

Pump	Rate	Wang–Lee			Dagpunar	OpenBUGS
		Mode	Mean	SD	Mean	Mean
λ_1	0.0530	0.0586	0.0577	0.0237	0.0581	0.0598
λ_2	0.0636	0.0645	0.0884	0.0681	0.0920	0.1015
λ_3	0.0795	0.0775	0.0862	0.0356	0.0867	0.0889
λ_4	0.1113	0.1147	0.1132	0.0286	0.114	0.1156
λ_5	0.5725	0.4185	0.5819	0.3072	0.566	0.6043
λ_6	0.6043	0.5318	0.6028	0.1344	0.602	0.6121
λ_7	0.9523	0.2370	0.6525	0.4682	0.764	0.899
λ_8	0.9523	0.1199	0.6623	0.4662	0.764	0.9095
λ_9	1.9047	2.0381	1.4163	0.6704	1.470	1.587
λ_{10}	2.0992	2.0105	1.9658	0.4269	1.958	1.995

(Robert and Casella, 2004, pp. 385–387)

$$p(\lambda_1, \ldots, \lambda_{10}, \beta|\mathbf{t}, \mathbf{y}) \propto \prod_{i=1}^{10}\{(\lambda_i t_i)^{y_i} e^{-(t_i+\beta)\lambda_i}\}\beta^{10\alpha+\gamma-1}e^{-\delta\beta}. \tag{4.6}$$

Using our algorithm, the sample is drawn and the sample mode, mean and SD are given in Table 6. For comparison we also included the numerical results of Dagpunar (2007) and from OpenBUGS (Spiegelhalter et al., 2007).

4.5. A genetic dataset

Dudley et al. (1991) studied a dataset of 190 individual measurements of red blood cell sodium–lithium counter transport (SLC). Wang and Fu (2007) analyzed this dataset using a Bayesian finite mixture model with varying number of components. They used a finite normal mixture model

$$\prod_{i=1}^{N} f(y_i|\mu^K, \sigma_K^2, w^K, K) = \prod_{i=1}^{N}\sum_{j=1}^{K} \frac{w_{Kj}}{\sqrt{2\pi\sigma_K^2}} \exp -\frac{(y_i-\mu_{Kj})^2}{2\sigma_K^2}, \tag{4.7}$$

with priors $\mu^K \sim \mathcal{N}(\mu_0, \sigma_0^2)$, $\sigma_K^2 \sim \mathcal{IG}(\alpha, \beta)$, $w^K \sim \mathcal{D}(\gamma)$ and parameter values $\alpha = 2$, $\beta = (R_y/6)^2$ and $\gamma = 1$, where $w^K = (w_{K1}, w_{K2}, \ldots, w_{KK})$ and $\mu^K = (\mu_{K1}, \mu_{K2}, \ldots, \mu_{KK})$. The full posterior is

$$\prod_{i=1}^{N} f(y_i|\mu^K, \sigma_K^2, w^K, K)p(\mu^K|K)p(\sigma_K^2|K)p(w^K|K)p(K). \tag{4.8}$$

Assuming the maximum number of components $K_{\max} = 4$, this distribution has dimensions $d = 21$. The significant region is identified as $\mu^K \in [0, 12]^K$, $\sigma_K^2 \in [0.1, 5]^K$ and $w^K \in [0, 1]^K$ (see also Wang and Fu, 2007), and the weight is set to $w = 0.01$ to reflect the fact that this mixture model has a large proportion of low probability area within its support. We estimate this model using our proposed method and compare our results with those of Wang and Fu (2007). The results in Table 7 indicate clearly that the $K = 3$ has the highest posterior probability. The corresponding posterior statistics are reported in Table 8.

4.6. Computing times

In all examples, there is little difference between Fu–Wang and Wang–Lee algorithms in terms of computing times when the dimension of the distribution is lower than or equal to five. In higher dimensional cases, the difference becomes more significant. For comparison, the computing times for several examples are reported in Table 9. As pointed out by a referee, some lengthy running times in the table could be due to the calls to `mapply` and `hist` and could be significantly reduced by delegating these functions to C codes.

Table 7
Prior and posterior for K in (4.8).

K	1	2	3	4
Prior	0.2	0.3	0.3	0.2
Fu–Wang	0	0.262	0.406	0.332
Wang–Lee	0.005	0.311	0.408	0.276

Table 8
Posterior means and standard deviations of μ^K and w^K for $K = 3$.

	Fu–Wang		Wang–Lee	
	Mean	SD	Mean	SD
μ_1	2.2474	0.1340	2.2503	0.2335
μ_2	3.5720	0.5100	3.6173	0.4878
μ_3	5.3284	0.5836	5.4520	0.6664
w_1	0.7164	0.1760	0.7304	0.1663
w_2	0.2315	0.1557	0.2159	0.1466
w_3	0.0520	0.0401	0.0535	0.0648
σ_3^2	0.4168	0.0862	0.4163	0.0914

Table 9
proc.time() in various examples.

Density	Dim	Fu–Wang			Wang–Lee		
		User	System	Total	User	System	Total
Eq. (4.1)	3	128.070	19.420	147.497	124.140	19.880	144.017
Eq. (4.3)	3	119.260	21.980	141.252	101.290	18.980	120.306
Eq. (4.5)	5	77.540	13.430	91.013	66.740	10.510	77.255
Eq. (4.6)	10	101.650	23.440	125.155	82.420	16.240	98.666
Eq. (4.8)	21	30281.730	180.330	30462.520	3750.520	20.540	3778.012

5. Conclusions and discussions

We developed a Monte Carlo method for multivariate random sample generation based on discretization of the sample space and direct inversion of the discretized distribution function. Different from the algorithm of Fu and Wang (2002), in this method the sample space is stratified according to the level of the density function. This new contourization scheme reduces the total number of contours and therefore leads to efficiency gain in computation. The efficiency gain can be significant especially when the mass of the target distribution concentrates on a small region of the sample space.

In the proposed method, the tail probability is controlled by a weight parameter, which can be done intuitively according to the tail property of the target distribution. However, a more rigorous rule for adjusting the weight parameter needs to be developed in the future. Also for distributions with certain complex structures, such as distributions with high spikes over small areas, the proposed method would require much more discretization points than the Fu–Wang algorithm in order for the high-level contours to have enough number of base points. Roughly speaking, the proposed algorithm should perform well as long as each contour contains at least ten to twenty base points. Therefore the proposed algorithm in this paper should be taken as an alternative rather than a replacement of the Fu–Wang algorithm. As suggested by a referee, a hybrid algorithm that switches between the two algorithms according to the shape of the density to minimize the computational cost would be interesting and worth investigating. For example, one simple switching rule would be to ensure that every contour contains at least ten to twenty base points. However, a more effective rule would depend on the spikiness of the density which is not easy to quantify in high dimensional cases.

Acknowledgments

The authors are grateful to the Editor, the Associate Editor and two referees for their constructive comments and suggestions. They also thank Benjamin Wang for proof reading this paper. The research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

An, X., Bentler, P.M., 2012. Efficient direct sampling MCEM algorithm for latent variable models with binary responses. Computational Statistics and Data Analysis 56, 231–244.
Carlin, B.P., Gelfand, A.E., 1991. An iterative Monte Carlo method for nonconjugate Bayesian analysis. Statistics and Computing 1, 119–128.
Carlin, B.P., Gelfand, A.E., Smith, A.F.M., 1992. Hierarchical Bayesian analysis of changepoint problems. Applied Statistics 41, 389–405.
Carlin, B.P., Louis, T.A., 1996. Bayes and Empirical Bayes Methods for Data Analysis. Chapman & Hall.
Dagpunar, J.S., 2007. Simulation and Monte Carlo: With Applications in Finance and MCMC. Wiley, New York.
Dudley, C.R.K., Giuffra, L.A., Raine, A.E.G., Reeders, S.T., 1991. Assessing the role of APNH, a gene encoding for a human amiloride-sensitive Na + /H+ antiporter, on the interindividual variation in red cell Na + /Li+ countertransport. Journal of the American Society of Nephrology 2, 937–943.
Fu, J., Wang, L., 2002. A random-discretization based Monte Carlo sampling method and its applications. Methodology and Computing in Applied Probability 4, 5–25.
Gaver, D.P., O'Muircheartaigh, I.G., 1987. Robust empirical Bayes analyses of event rates. Technometrics 29, 1–15.
Liang, F., Liu, C., Carroll, R.J., 2007. Stochastic approximation in Monte Carlo computation. Journal of the American Statistical Association 102, 305–320.
Malefaki, S., Iliopoulos, G., 2009. Simulation from a target distribution based on discretization and weighting. Communications in Statistics—Simulation and Computation 38, 829–845.

O'Hagan, A., Murphy, T.B., Gormley, I.C., 2012. Computational aspects of fitting mixture models via the expectation–maximization algoritm. Computational Statistics and Data Analysis 56, 3843–3864.
Prentice, R.L., 1976. A generalization of the Probit and Logit methods for dose response curves. Biometrics 32, 761–768.
Punzo, A., Zini, A., 2012. Discrete approximations of continuous and mixed measures on a compact interval. Statistical Papers 53, 563–575.
R Core Team, 2013. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. URL: http://www.R-project.org/.
Robert, C., Casella, G., 2004. Monte Carlo Statistical Methods. Springer, New York.
Sotto, C., Beunckens, C., Molenberghs, G., Kenward, M.G., 2011. MCMC-based estimation methods for continuous longitudinal data with non-random (non)-monotone missingness. Communications in Statistics—Simulation and Computation 55, 301–311.
Spiegelhalter, D., Thomas, A., Best, N., Lunn, D., 2007. OpenBUGS User Manual, Version 3.0.2., OpenBUGS.
Wang, L., Fu, J., 2007. A practical sampling approach for a Bayesian mixture model with unknown number of components. Statistical Papers 48, 631–653.
Wang, L., Shan, S., Wang, G.G., 2004. Mode-pursuing sampling method for global optimization on expensive black-bo functions. Engineering Optimization 36, 419–438.
West, M., 1993. Approximating posterior distributions by mixture. Journal of the Royal Statistical Society—B 55, 409–422.
Xue, L., Fu, J.C., Wang, F., Wang, L., 2005. A mixture model approach to analyzing major element chemistry data of the Changjiang (Yangtze River). Environmetrics 16, 305–318.