Universal Generators for Correlation Induction

Wolfgang Hörmann and Gerhard Derflinger

Institut f. Statistik Wirtschaftsuniversität Wien
Augasse 2-6 A-1090 WIEN

Keywords. Simulation, non-uniform random number generation, rejection method, universal generators, correlation induction

1 Introduction

Compared with algorithms specialized for a single distribution universal (also called automatic or black-box) algorithms for continuous distributions were relatively seldom discussed. But they have important advantages for the user: One algorithm coded and tested only once can do the same or even more than a whole library of standard routines. It is only necessary to have a program available that can evaluate the density of the distribution up to a multiplicative factor. Black box algorithms suggested in literature fall into two groups. Simple and short algorithms with almost no setup (eg. [4]) but only moderate speed and very fast table-methods which need a long and complicated setup (eg. [2] and [1]). In [7] and [6] we introduced new universal generators based on transformed density rejection which lie between these two groups. The suggested algorithms are quite simple and need only a moderate setup time whereas the marginal execution time for standard distributions (like the gamma- or the beta-family) is about the same as for specialized algorithms. In [8] we demonstrated how transformed density rejection can be used to construct very fast table methods as well. In this paper we show that transformed density rejection is well suited to construct universal algorithms suitable for correlation induction which is important for variance reduction in simulation.

2 Transformed density rejection

Rejection is the most flexible method for generating non-uniform random variates. To generate random variates with density function $f(x)$ we need a "hat" function $h(x)$ and a real number α with $f(x) \leq \alpha h(x)$. As the first step a random variate X with density $h(x)$ and a uniform ($U(0,1)$) random number U is generated. If $\alpha h(X)U \leq f(X)$ then X is accepted as a random variate of the desired distribution otherwise X is rejected and we have to try again. It is obvious that α is the expected number of replications to get one random number and that $1/\alpha$ is the probability of acceptance. Therefore $h(x)$ should be chosen in a way that α is close to one and that the generation of random numbers with density $h(x)$ is simple.

Transformed density rejection uses a new and general method for constructing hat functions. To do so we need a transformation $T(x)$ with the property that $T(f(x))$ is concave on the support of $f(x)$ (i.e. $\{x|f(x) > 0\}^-$), and arbitrary but fixed design points x_i, $i = 0, \cdots, n-1$. Then we define $lin(x; x_0, \ldots, x_{n-1}) = \min_i\{f(x_i) + f'(x_i)(x - x_i)\}$ which is obviously greater or equal $T(f(x))$. Thus $T^{-1}(lin(x; x_0, \ldots, x_{n-1})) = h(x)$ is a hat function for $f(x)$. The lines, connecting neighbouring points of contact, transformed with T^{-1} can be used as simple squeezes to avoid the evaluation of $f(x)$ in most cases. Figure 1 illustrates transformed density rejection for the normal distribution with $x_0 = -1.665$, $x_1 = 0$, $x_2 = 1.665$ and $T(x) = -1/\sqrt{x}$; [left hand side: $T(f(x))$ (thick line) and $lin(x)$ (thin line); right hand side: $f(x)$ (thick line) and $T^{-1}(lin(x))$ (thin line); the dashed lines are the squeezes.]

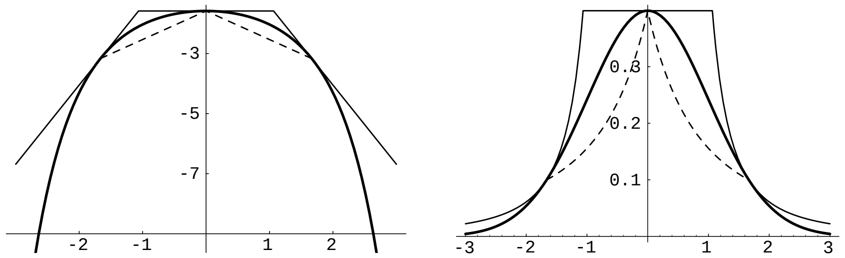

Fig. 1

If we take $T(x) = \log(x)$ (this special case was investigated in [5]) or $T(x) = -1/\sqrt{x}$ the generation of random variates with density $h(x)$ can be easily done by inversion. For the case $T(x) = \log(x)$ transformed density rejection is applicable to the well-known class of log-concave distributions. This class consists of distributions with unimodal densities that have subexponential tails and is defined by the property $\log(f(x)$ is concave or equivalently $f(x)f''(x) - f'(x)^2 \leq 0$ on the support of f. Examples are the normal the gamma (with $a \geq 1$) and the beta (with $a, b \geq 1$) distributions. For $T(x) = -1/\sqrt{x}$ we call the densities T-concave. This class of distributions consists of unimodal densities with subquadratic tails and is defined by $f(x)f''(x) - 1.5f'(x)^2 \leq 0$. Obviously T-concave is more general than log-concave. The most prominent distribution which is T-concave but not log-concave is the t-family with $\nu \geq 1$.

3 Universal algorithms

It is not difficult to use the transformed density rejection method to construct a hat function and thus a rejection algorithm for an arbitrary T-concave density that is – up to a multiplicative factor – computable by a given subprogram (black box). First we give a sketch of the universal algorithm for given design points x_i which are sorted in ascending order. (C-code is available on request.)

Algorithm UNIV

1: [Set-up]

1.0 Compute $y_i = T(f(x_i))$, $a_i = T(f(x_i))'$ and $as_i = (y_{i+1} - y_i)/(x_{i+1} - x_i)$ for $i = 0 \ldots n-1$. Compute the first coordinates of the points of intersection b_i between tangent i and $i-1$. b_0 and b_n are the left and right border of the domain and can be ∞.

1.1 (We define $F(x) = \int T^{-1}(x)\, dx$) Compute $lf_i = F(y_i + a_i(b_i - x_i))/a_i$ and $area_i = F(y_i + a_i(b_{i+1} - x_i))/a_i - lf_i$, the area between the x-axis and the hat-function between b_i and b_{i+1}. Compute the cumulated areas $areac_i = \sum_{j=0}^{i} area_j$.

2.1 Generate two uniform random numbers U and V.

2.3 Set $U \to U * areac_{n-1}$. Let I be the index with $areac_{I-1} \leq U \leq areac_I$. ($I$ is a random deviate from the discrete distribution with probabilities proportional to $area_I$. If n is large use a guide table for acceleration). Set $U \leftarrow U - areac_{I-1} + lf_I$. U is a uniform random number in the interval $(lf_I, lf_I + area_I)$.

2.4 (Generation of a variate from the dominating distribution)
Compute $X = x_I - y_I/a_I + F^{-1}(a_I U)/a_I$
and $lx = T^{-1}(y_I + a_I(X - x_I))$.

2.5 (Squeeze acceptance)
If $X \geq x_I$ and $I < n-1$ and $V * lx \leq T^{-1}(y_I + as_I(X - x_I))$ return X.
If $X < x_I$ and $I > 0$ and $V * lx \leq T^{-1}(y_I + as_{I-1}(X - x_I))$ return X.

2.6 (Final acceptance)
If $V * lx \leq f(x)$ return X, else go to 2.1.

It is not necessary to know the derivative of the density function since we can replace the tangent of $T(f(x))$ in the point x_i by the "pseudo-tangent" through the point $(x_i, T(f(x_i + \Delta)))$ with slope $(T(f(x_i + \Delta)) - T(f(x_i)))/\Delta$ which is always greater than $T(f(x))$ if we chose $\Delta > 0$ for x_i on the left-hand side of the mode and $\Delta < 0$ for x_i on the right-hand side.

To use Algorithm UNIV we have to chose T, n and the x_i. Our tests showed that $T = -1/\sqrt{x}$ (then $T^{-1}(x) = 1/x^2$ and $F(x) = -1/x$) is not only more general but also considerably faster than the logarithm. By taking the number of touching points n low ($n = 3$ is a good choice) or large (for example $n = 33$ or $n = 65$) we can switch between faster setup but slower marginal execution time and short marginal execution time but slow setup. The question how to select the points of contact to minimize the expected number of iterations or the expected number of evaluations of f is not so simple. For $n = 3$, f with unbounded support and x_1 = the mode, Theorem 1 of [6] gives the answer: Take x_0 and x_2 such that the equation $f(x_i) = T^{-1}\left(T(f(x_1)) - F(T(f(x_1)))/f(x_1)\right)$ (i.e. $f(x_i) = f(x_1)/4$ for $T(x) = -1/\sqrt{x}$) is fulfilled. x_i that are close to the solution can be easily found by numerical search. For n large the problem is more difficult: After several attempts we obtained the best results with a formula that is asymptotically optimal in both respects: Choose the location of the points in a way that

$$\frac{(\text{length of interval}) \times \sqrt[3]{T(f(x))''}}{T(f(x))} = \text{constant}.$$

This can be established by integrating $\sqrt[3]{T(f(x))''}/T(f(x))$ numerically. Then, by interpolation, the inverse function of this indefinite integral is approximately evaluated for equidistant arguments. Points chosen by this method are very close to optimal. For details see [8]. In that paper we also describe an algorithm (UNIVD) based on the same idea as UNIV but utilizing the idea of decomposition. UNIVD is more complicated than UNIV but the expected number of uniforms required is reduced almost to one.

4 Correlation induction

In many simulation experiments variance reduction can be obtained by inducing positive (common random numbers) or negative (antithetic variates) correlation between the random deviates generated (see simulation text books e.g. [3]). The highest (or lowest) correlation possible can always be obtained with the inversion method which is very slow for most distributions. In [9] and [10] it is demonstrated that it is also possible to obtain correlation induction with rejection methods. Transformed density rejection is especially well suited to install correlation induction facilities as it is so close to the inversion method. In algorithm UNIV monotonicity is already at hand and it is only necessary to establish synchronization by using two random number streams: The first one for the two random numbers necessary for the first acceptance-rejection experiment, the second stream if the first pair was rejected. As the decomposition logic destroys monotonicity in Algorithm UNIVD we changed algorithm UNIV following the guidelines of [10] but using only two random numbers from the first stream instead of four for one random deviate. The argument *ianti* (either 1 or -1) controls the orientation of the induced correlation: For positive correlation take *ianti* = 1 (or -1) for both sequences. For negative correlation take the opposite sign for the second sequence. It is necessary to have to independent streams of random numbers available, usually by starting the same generator with two different starting values.

Algorithm UNIVCI
(Add or replace the following steps in algorithm UNIV)

2.0 Generate two uniform random numbers U, V from the first stream. Go to step 2.2.
2.1 Generate two uniform random numbers U, V from the second stream.
2.2 If ianti= -1 set $U \to 1 - U$.

But there are more reasons than simplicity to implement correlation induction for transformed density rejection: In many simulation applications, especially when comparing the performance of different systems, it is important to induce correlation between random numbers of different distributions. This is best done by a universal algorithm that uses the same algorithm for both cases. A second advantage is the high acceptance probability of transformed rejection if we take n large. For $n = 33$ our experiments with several distributions showed that the correlation induced by UNIVCI is almost the same as for inversion, in no case it was more than 0.015 away. As an example Table 1 shows the negative correlation obtained with UNIVCI between beta and gamma distributions compared with the correlation induced by inversion. The strongest negative correlation varies due to the different shapes of the distributions whereas the difference of the correlations induced by the two methods remains close to 0.01.

Table 1 Negative correlation induced by UNIVCI and by inversion

	Beta			
	100,100	10,100	2,100	2,2
Gamma 2	-0.932/-0.946	-0.887/-0.897	-0.925/-0.932	-0.925/-0.932
Gamma 10	-0.979/-0.989	-0.950/-0.961	-0.883/-0.892	-0.970/-0.979
Gamma 100	-0.990/-0.999	-0.974/-0.984	-0.920/-0.934	-0.979/-0.990
Beta 2,2	-0.980/-0.992	-0.973/-0.983	-0.928/-0.937	-1.000/-1.000
Beta 2,100	-0.961/-0.971	-0.892/-0.901	-0.800/-0.805	
Beta 10,100	-0.982/-0.992	-0.955/-0.967		
Beta 100,100	-0.999/-1.000			

5 Comparison of Algorithms

To be able to judge the speed of Algorithm UNIVCI we compared our C-implementation with Algorithm UNIVD which is at least as fast as the fastest known specialized algorithms for the gamma- and beta distributions (see [8]) and with an inversion algorithm. It is based on the "regula falsi" with a large auxiliary table for good starting values. The results show that the speed of Algorithm UNIVCI is well comparable with fast specialized algorithms. For the case that correlation induction is needed the gain in speed is dramatic. UNIVCI is between 20 and 80 times faster than inversion which makes the

Table 2 Execution times in μ-seconds

	UNIVD(n=33)	UNIVCI(n=33)	inversion
Gamma 2	4.1	7.0	160
Gamma 10	4.1	7.0	240
Gamma 100	4.1	7.0	550
Beta 2,2	4.1	7.0	190
Beta 2,100	4.1	7.0	320
Beta 10,100	4.1	7.0	410
Beta 100,100	4.1	7.0	440

savings substantial even for simulations where only a small portion of computing time is used for random variate generation.

We are convinced that for many applications universal algorithms based on transformed density rejection have important advantages compared with standard random variate procedures. They are – coded and debugged only once – applicable to a wide range of distributions. They have good speed for all of these distributions almost independent of the speed of the evaluation of f and they have correlation induction capabilities practically equal to (very slow) inversion algorithms. We hope that this paper will therefore increase the future use of those algorithms in practice.

References

[1] J. H. Ahrens. Sampling from general distributions by suboptimal division of domains. Grazer Mathematische Berichte Nr. 319, Graz, Austria, 1993.

[2] J.H. Ahrens and K.D. Kohrt. Computer methods for sampling from largely arbitrary statistical distributions. *Computing*, 26:19–31, 1981.

[3] P. Bratley, B. L. Fox, and L. E. Schrage. *A Guide to Simulation*. Springer-Verlag, New York, 1983.

[4] L. Devroye. *Non-Uniform Random Variate Generation*. Springer-Verlag, New York, 1986.

[5] W. R. Gilks and P. Wild. Adaptive rejection sampling for gibbs sampling. *Applied Statistics*, 41:337–348, 1992.

[6] W. Hörmann. A rejection technique for sampling from t-concave distributions. *ACM Transactions on Mathematical Software*,(to appear), 1994.

[7] W. Hörmann. A universal generator for discrete log-concave distributions. *Computing*, 51, 1993.

[8] W. Hörmann and G. Derflinger. Universal sampling algorithms for a large class of continuous distributions. Technical report, Institut f. Statistik, Wirtschaftsuniversität Wien, 1994.

[9] V. Kachitvichyanukul, S. J. Cheng, and B. W. Schmeiser. Fast poisson and binomial algorithms for correlation induction. *Journal of Statistical Computation and Simulation*, 29:17–33, 1988.

[10] B. Schmeiser and V. Kachitvichyanukul. Noninverse correlation induction: guidelines for algorithm development. *Journal of Computational and Applied Mathematics*, 31:173–180, 1990.