Journal of Computational and Applied Mathematics 31 (1990) 181–189
North-Holland

The ratio of uniforms approach for generating discrete random variates

Ernst STADLOBER
Institute of Statistics, Technical University of Graz, Lessingstraße 27, A-8010 Graz, Austria

Received 10 March 1989

Abstract: The most efficient algorithms for sampling from classical discrete distributions are based upon the acceptance/rejection principle. They are complicated and not easy to understand. By adapting the ratio of uniforms method to unimodal discrete distributions, sampling procedures can be established, which are both simple and fast. Algorithms for the hypergeometric distribution are developed and compared with competing methods.

Keywords: Discrete random variate generation, ratio of uniforms, simulation, hypergeometric distribution.

1. Introduction

Existing fast algorithms for generating Poisson, binomial and hypergeometric variates are rather complicated whereas simple procedures are usually slow when the means μ are large. *Straightforward inversion* via sequential search from the bottom and procedures which are based upon *special distributional properties* are of simple structure, but their execution times are not uniformly bounded over the whole set of parameter values (see [5]). On the other hand, sophisticated table methods like the *guide table method* of Chen and Asau [3] and the *alias method* of Walker [14] are the fastest methods if many variates for fixed parameters are needed. Note that every change of parameters demands the construction of new tables which can be done in $O(n)$-time, where n is the number of mass points of the discrete distribution in hand. The *acceptance / rejection approach* of Von Neumann [13] leads to uniformly fast algorithms, i.e., algorithms with bounded execution time over the defined parameter range. However, efficient competitors are involved and overburdened with case distinctions (see [1,6,7]).

Therefore we were looking for simple and uniformly fast methods. Recently, Ahrens and Dieter [2], and Stadlober [11] proposed successful procedures for Poisson and binomial distributions, respectively by applying the *ratio of uniforms method* of Kinderman and Monahan [8]. The generalization to any unimodal discrete distribution is based on Theorem 1, given in Section 2. It requires that the standardized histogram function $f(x)$ of the target distribution is majorized by a table mountain hat $h(x) = \min(1, s^2/(x-a)^2)$ with suitable chosen location parameters a and scale parameters s. Then sampling from $f(x)$ is very easy:

Generate a pair (U, V) uniformly distributed over the rectangle $R = [0, 1] \times [-1, 1]$, set $X \leftarrow sV/U + a$ and return $K \leftarrow \lfloor X \rfloor$ as a sample from $f(x)$ whenever $U^2 \leqslant f(X)$ is fulfilled. Otherwise reject X and try again.

Details of the sampling method are discussed in Section 2. Section 3 is devoted to the choice of the hat parameters a and s in case of Poisson, binomial and hypergeometric distributions. Implementations of hypergeometric generators are suggested in Section 4. Computational experience is reported in Section 5.

2. The method

The ratio of uniforms method was invented by Kinderman and Monahan [8] for continuous distributions with rescaled densities $f(x)$. They utilized it for Cauchy, normal and exponential generators. Algorithms for parametric densities were also constructed by Kinderman and Monahan [9] (Student-t ($a \geqslant 1$), gamma ($a \geqslant 1$)), by Cheng and Feast [4] (gamma ($a > \frac{1}{4}$)), and by Monahan [10] (chi ($a \geqslant 1$)).

A specialization and reformulation of the original procedure allows to extend the method to discrete distributions. We start with (U, V) uniformly distributed over the standardized rectangle $R = \{(u, v) \mid 0 \leqslant u \leqslant 1, -1 \leqslant v \leqslant 1\}$, transform (U, V) to $(X, Y) = (a + sV/U, U^2)$ and cover the domain $T(C) = \{(x, y) \mid -\infty < x < \infty, 0 \leqslant y \leqslant f(x)\}$ of the target distribution by $T(R) = \{(x, y) \mid -\infty < x < \infty, 0 \leqslant y \leqslant \min(1, s^2/(x-a)^2)\}$ such that acceptance/rejection is possible. This is illustrated in Fig. 1 for $a = 0$, $s = 1$ in the case of the Cauchy distribution. The half unit circle $C = \{(u, v) \mid 0 \leqslant u \leqslant \sqrt{f(v/u)}\} = \{(u, v) \mid 0 \leqslant u \leqslant 1, u^2 + v^2 \leqslant 1\}$ is enclosed in R and the rescaled density $f(x) = 1/(1+x^2)$ is covered by the *table mountain hat* $h(x) = \min(1, 1/x^2)$. Thus ratio of uniforms with rectangles is nothing but acceptance/rejection with table mountains.

Theorem 1. *Let R be the rectangle*

$$R = \{(u, v) \mid 0 \leqslant u \leqslant 1, -1 \leqslant v \leqslant 1\}.$$

If (U, V) is uniformly distributed over R, then

(a) *$X = sV/U + a$, $s > 0$, $-\infty < a < \infty$, has the density*

$$g(x) = g(x; a, s) = \begin{cases} \dfrac{1}{4s}, & a - s \leqslant x \leqslant a + s, \\ \dfrac{s}{4(x-a)^2}, & \text{elsewhere}. \end{cases} \tag{1}$$

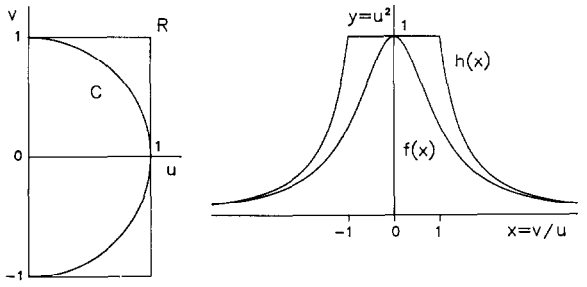

Fig. 1. Ratio of uniforms as rejection with table mountain hat.

(b) *The conditional density of* $Y = U^2$, *given* $X = x$, *is calculated as*

$$g(y \mid x) = \begin{cases} 1 & \text{for all } x\colon\ a - s \leqslant x \leqslant a + s,\ 0 \leqslant y \leqslant 1, \\ \dfrac{(x-a)^2}{s^2} & \text{for all other } x,\ 0 \leqslant y \leqslant \dfrac{s^2}{(x-a)^2}, \end{cases} \tag{2}$$

i.e., $g(y \mid x)$ *is the density of the* $(0, 4s\, g(x))$*-uniform distribution.*

Proof. See [11]. □

Before turning Theorem 1 into profit, some features of the ratio of uniforms method should be pointed out.

(a) For the sake of simplicity, Theorem 1 is stated only for symmetric table mountains with scale parameter s on both sides. On the other hand, we allow for a shift a, which is not part of the original version of the method.

(b) Restriction to $0 \leqslant u \leqslant 1$ requires a standardization of the histogram function

$$f_\cdot(x) = \begin{cases} p_j, & j \leqslant x < j+1,\ j = 0, 1, 2, \ldots, \\ 0, & \text{elsewhere}, \end{cases} \tag{3}$$

where p_j are the probabilities at the mass points j. Then the standardization reads

$$f(x) = \frac{f_\cdot(x)}{p_m}, \quad \text{where } p_m = \max_x f_\cdot(x). \tag{4}$$

(c) In view of Theorem 1(b) use

$$h(x) = 4s\, g(x) = \begin{cases} 1, & a - s \leqslant x \leqslant a + s, \\ \dfrac{s^2}{(x-a)^2}, & \text{elsewhere}, \end{cases} \tag{5}$$

as majorizing hat function of $f(x)$.

The acceptance/rejection approach demands that $h(x)$ dominates $f(x)$. Obviously this is no problem in the flat center between $a - s$ and $a + s$. But in the tails one has to check for candidates a and s, whether $f(x) \leqslant h(x)$ holds for $x \leqslant a - s$ and whether $\lim_{\epsilon \downarrow 0} f(x - \epsilon) \leqslant h(x)$ is true if $x \geqslant a + s$.

(d) The best possible hat parameters a and s in (5) are the ones that lead to the smallest efficiencies

$$\alpha = \frac{\int h(x)\, \mathrm{d}x}{\int f(x)\, \mathrm{d}x} = 4sp_m. \tag{6}$$

We have arrived at the following special case of acceptance/rejection based on Theorem 1.

Procedure RUD

(1) Generate U, $V \sim U(0, 1)$ and set $X \leftarrow a + s(2V - 1)/U$, $K \leftarrow \lfloor X \rfloor$. (*Generate* X *with density* $g(x)$, *set* $K \leftarrow \lfloor X \rfloor$).

(2) Set $Y \leftarrow U^2$. (*Take Y from* $U(0, 4s\, g(X))$.
(3) If $Y \leqslant f(X)$ $(= p_K)$, return K as a sample from $\{p_j\}$. Otherwise go to (1).

Procedure RUD leads to very simple and efficient generators for special discrete distributions. Algorithms for the hypergeometric distribution are developed and analyzed in Section 4.

3. Choice of the hat parameters

The hat function $h(x)$ (5) will touch the standardized histogram $f(x)$ (4) at one outer corner on the left and at one outer corner on the right, if the choice of the parameters a and s is optimal. For the Poisson and binomial distributions we calculated such optima by numerical search methods for different means μ within the range $1 \leqslant \mu \leqslant 1000$ (see [11]). It appeared that the tightest hats $h(x)$ touch $f(x)$ at two points $L \approx \mu - \sqrt{2}\,\sigma$ and $R \approx \mu + \sqrt{2}\,\sigma$ where σ is the standard deviation of the distribution. Nevertheless, for efficient algorithms simple choices of a and s are needed. In case of the Poisson distribution ($P(\mu)$) with histogram function

$$f_P(x) = f_P(x;\, \mu) = \frac{\mu^{\lfloor x \rfloor}}{\lfloor x \rfloor !} e^{-\mu}, \quad 0 \leqslant x < \infty, \quad \mu \geqslant 1, \tag{7}$$

and mode at $m = \lfloor \mu \rfloor$, Ahrens and Dieter [2] fix a at $\mu + \frac{1}{2}$ anticipating extensions to the binomial and hypergeometric distributions: $a = \mu + \frac{1}{2}$ is the best choice for symmetric histograms. Then they approximate the best possible parameter s^* for fixed $a = \mu + \frac{1}{2}$ by the simple upper bound

$$\hat{s} = \sqrt{\frac{2}{e}\left(\sigma^2 + \frac{1}{2}\right)} + \frac{3}{2} - \sqrt{\frac{3}{e}}\,, \tag{8}$$

which is justified by numerical verification.

Analysis of the binomial distribution ($B(n,\, p)$) with histogram function

$$f_B(x) = f_B(x;\, n,\, p) = \binom{n}{\lfloor x \rfloor} p^{\lfloor x \rfloor} (1-p)^{n - \lfloor x \rfloor}, \quad 0 \leqslant x < n+1, \quad \mu = np \geqslant 1, \tag{9}$$

and one mode at $m = \lfloor (n+1)p \rfloor$, can be restricted to $p \leqslant \frac{1}{2}$, because of $f_B(x;\, n,\, p) = f_B(n - x;\, n,\, 1-p)$. For the hypergeometric distribution ($H(N,\, M,\, n)$) with histogram function

$$f_H(x) = f_H(x;\, N,\, M,\, n) = \frac{\binom{M}{\lfloor x \rfloor}\binom{N-M}{n - \lfloor x \rfloor}}{\binom{N}{n}},$$

$$\max(0,\, n - N + M) \leqslant x < \min(n,\, M) + 1, \quad \mu = n\frac{M}{N} \geqslant 1, \tag{10}$$

and one mode at $m = \lfloor (n+1)(m+1)/(N+2) \rfloor$, it suffices to consider parameters $1 \leqslant n \leqslant \frac{1}{2}N$, and $1 \leqslant M \leqslant \frac{1}{2}N$ on account of the properties

$$\begin{aligned} f_H(x;\, N,\, M,\, n) &= f_H(n - x;\, N,\, N - M,\, n) = f_H(M - x;\, N,\, M,\, N - n) \\ &= f_H(n - N + M + x;\, N,\, N - M,\, N - n). \end{aligned} \tag{11}$$

The approximation $\hat{s}$ (8) of s^* proved also to be appropriate for the binomial (9) and hypergeometric (10) histograms [12], but it is even possible to construct table mountains with the true optima s^*. The important result which delivers a simple rule for determining s^* is the following.

Theorem 2. (a) *The hat function*

$$h(x) = \begin{cases} 1, & a - s^* \leqslant x \leqslant a + s^*, \\ \dfrac{s^{*2}}{(x-a)^2}, & \textit{elsewhere}, \end{cases}$$

with smallest possible scale parameter s^*, *given that* $a = \mu + \frac{1}{2}$ $(\mu \geqslant 1)$, *contacts the standardized histogram* $f(x) = f.(x)/p_m$ *at*

$$k^* = \lfloor z \rfloor \quad \textit{or} \quad k^* = \lfloor z \rfloor + 1, \tag{12}$$

where

(i) $z = a - \sqrt{2a}$, *if* $f.(x) = f_P(x;\ \mu)$,

(ii) $z = a - \sqrt{2a(1-p)}$, *if* $f.(x) = f_B(x;\ n,\ p)$ $(p \leqslant \frac{1}{2})$,

(iii) $z = a - \sqrt{2a\left(1 - \frac{M}{N}\right)\left(1 - \frac{n}{N}\right)}$, *if* $f.(x) = f_H(x;\ N,\ M,\ n)$ $(n \leqslant \frac{1}{2}N,\ M \leqslant \frac{1}{2}N)$.

(b) *The optimal parameter* s^* *is calculated as*

$$s^* = (a - k^*)\sqrt{f(k^*)}\,. \tag{13}$$

Proof. See [12]. □

In Fig. 2 the rescaled hypergeometric histogram $f(x) = f_H(x)/p_m$ for $N = 256$, $M = 64$, $n = 16$ $(\mu = 4)$ is compared with two different table mountain hats. The hat with optimal parameter s^* (13)—displayed in (a)—touches $f(x)$ at the point $k^* = 2$, whereas the simple approximation $\hat{s}$ (8) leads to a slightly looser hat shown in (b).

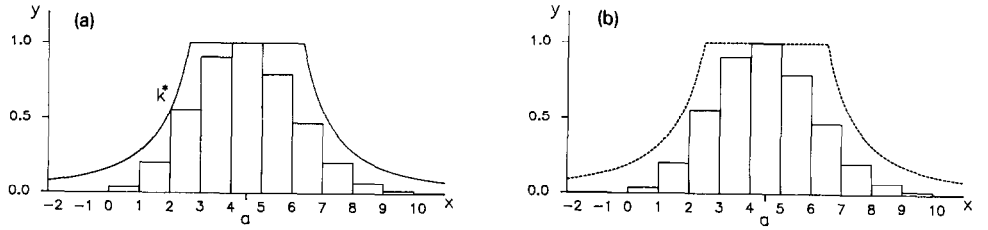

Fig. 2. Standardized $H(256, 64, 16)$-histogram $f(x)$ with two hats $h(x)$. (a) Simple choice $a = 4.5$, optimum $s^* = 1.87$, $\alpha_{s^*} = 1.74$, one touching point $k^* = \lfloor 1.98 \rfloor + 1 = 2$. (b) Simple choices $a = 4.5$ and $\hat{s} = 2.01$, $\alpha_{\hat{s}} = 1.87$.

4. Algorithms

Two algorithmic versions for the hypergeometric distribution with hat parameters (13) are developed. The first procedure HRUE utilizes an external function for evaluating $\gamma_k = \ln k!$. For $k = 0, \ldots, 9$, γ_k is stored in a table and for $k \geqslant 10$ the Stirling approximation

$$\gamma_k = \ln \sqrt{2\pi} + \left(k + \tfrac{1}{2}\right) \ln k - k + \frac{1}{12k} - \frac{1}{360k^3},$$

truncation error ϵ: $|\epsilon| < 7.9 \times 10^{-9}$, is implemented. In the second version HRUE′ all values of $\gamma_k = \ln k!$ are tabulated.

Algorithm HRUE ($\mu \geqslant 1$, $2 \leqslant n \leqslant \frac{1}{2}N$, $2 \leqslant M \leqslant \frac{1}{2}N$)

(0) [*Set-up*]. External double precision function $\gamma_k = \ln k!$,
double precision function $\delta_k = \gamma_k + \gamma_{M-k} + \gamma_{n-k} + \gamma_{N-M-n+k}$.
Constants $\ln 2 = 0.693147181$, $B = 5$ (for 9 decimal digits precision.)
Pre-set $p \leftarrow M/N$, $q \leftarrow 1 - p$, $a \leftarrow np + \frac{1}{2}$, $c \leftarrow \sqrt{2aq(1 - n/N)}$,
$m \leftarrow \lfloor (n+1)(M+1)/(N+2) \rfloor$, $g \leftarrow \delta_m$, $k \leftarrow \lfloor a - c \rfloor$, $x \leftarrow (a - k - 1)/(a - k)$.
If $(n-k)(p - k/N)x^2 > (k+1)(q - (n-k-1)/N)$, set $k \leftarrow k + 1$.
Set $h \leftarrow (a-k)\exp(\frac{1}{2}(g - \delta_k) + \ln 2)$,
$b \leftarrow \min(\min(n, M) + 1, \lfloor a + Bc \rfloor)$.

(1) Generate U, U^* and set $X \leftarrow a + h(U^* - \frac{1}{2})/U$.

(2) If $X < 0$ or $X \geqslant b$ go to (1). Otherwise set $K \leftarrow \lfloor X \rfloor$.

(3) [*Test for appropriate method of evaluating* $f_k = p_k/p_m$].
If $m \leqslant 500$, go to (4).
(3.0) Set $T \leftarrow g - \delta_K$.
(3.1) If $U(4 - U) - 3 \leqslant T$, return K. (*Fast acceptance*)
(3.2) If $U(U - T) \geqslant 1$, go to (1). (*Fast rejection*)
(3.3) If $2 \ln U \leqslant T$, return K. Otherwise go to (1).

(4) [*Evaluate* f_k *via* $f_k = (M-k+1)(n-k+1)/(k(N-M-n+k))f_{k-1}$ *starting at* m].
Set $f \leftarrow 1.0$. If $m < K$, set $i \leftarrow m$ and repeat $i \leftarrow i + 1$,
$f \leftarrow f(M-i+1)(n-i+1)/(i(N-M-n+i))$ until $i = K$.
Otherwise, if $m > K$, set $i \leftarrow K$ and repeat $i \leftarrow i + 1$,
$f \leftarrow f\, i(N-M-n+i)/((M-I+1)(n-i+1))$ until $i = m$.
If $U^2 \leqslant f$, return K. Otherwise go to (1).

In step (0) of HRUE $h = 2s^*$ is calculated in the following manner. As a consequence of Theorem 2(iii) the quotient $q_k = f(k)/h(k)$ attains its maximum 1 either at $k = \lfloor z \rfloor = \lfloor a - \sqrt{2aq(1 - n/N)} \rfloor$ or at $\lfloor z \rfloor + 1$. Hence the touching point is at k whenever the ratio

$$\frac{q_{k+1}}{q_k} = \left(\frac{a-k-1}{a-k}\right)^2 \left(\frac{n-k}{k+1}\right)\left(\frac{p - k/N}{q - (n-k-1)/N}\right)$$

is *less equal* 1, otherwise the touching point is at $k + 1$. The constant b is a convenient safety bound for the deviates X in step (2). γ_k, δ_k and g are calculated in double precision to circumvent severe loss of accuracy.

For mode $m \leqslant 500$ $f(K)$ is evaluated recursively (step (4)) in order to accelerate the algorithm, and the deviate $K = \lfloor X \rfloor$ with density $g(x) = h(x)/4s^*$ can be accepted whenever $U^2 \leqslant f(K)$. If $m > 500$, the quantity $T = \ln f(K)$ is only rarely compared with $2 \ln U$ in step (3.3). More often the squeeze tests in steps (3.1) and (3.2) based on the inequalities $u - 1/u \leqslant 2 \ln u \leqslant -3 + 4u - u^2$, will decide.

The simpler version HRUE$'$ uses a table with double-precision values γ_k, $k = 0, \ldots, N$, and it needs only steps (0)–(2), (3.0)–(3.3) of HRUE. (The recursive evaluation of $f(K)$ contained in step (4) of HRUE is not necessary.)

Similar algorithms can be constructed using the simple choice $\hat{s}$ (8) instead of s^* (13) (see Algorithms HRUA and HRUA$'$ in [12]).

5. Computational experience

FORTRAN functions of HRUE and HRUE$'$, implemented on a Univac 1100/81 mainframe computer with 9 decimal digits accuracy, were extensively compared with the competitor H2PE

Table 1
Execution times [μsec/variate] UNIVAC 1100/81

N	40	100	400	1000	4000	10000	Algorithm
M	20	50	100	100	200	500	
$\mu = 10$	165	165	179	188	193	194	HRUE
	150	161	173	175	177	178	H2PE
	132	128	126	127	128	128	HRUE$'$
	139	136	139	134	134	134	H2PE$'$
	M	50	200	200	400	500	
	$\mu = 20$	180	196	214	221	226	HRUE
		167	178	195	203	205	H2PE
		125	126	126	125	125	HRUE$'$
		136	134	134	132	132	H2PE$'$
		M	200	500	1000	1000	
		$\mu = 100$	273	306	348	370	HRUE
			454	428	409	404	H2PE
			125	124	123	123	HRUE$'$
			132	131	130	130	H2PE$'$
			M	500	2000	2000	
			$\mu = 200$	367	408	475	HRUE
				423	408	398	H2PE
				123	123	123	HRUE$'$
				131	130	130	H2PE$'$
				M	2000	5000	
				$\mu = 500$	565	564	HRUE
					397	393	H2PE
					122	122	HRUE$'$
					130	129	H2PE$'$

Table 2
Initialization times [μsec], words, lines of code

Algorithm	Initialization times	Words of code	Lines of code	Supporting functions
HRUE	680–1250	507	49	DLFAC [a]
H2PE	1470–1955	867	86	DLFAC
HRUE′	180	344	36	table γ_k [b]
H2PE′	320	443	45	table γ_k

[a] DLFAC: double precision function for ln $k!$: 94 words, 15 lines.
[b] $\gamma_k = \ln k!$: table of double precision values for $k = 0, \ldots, N$.

(valid for $m \geqslant 10$) by Kachitvichyanukul and Schmeiser [6] and its adapted table version H2PE′. H2PE is the only uniformly fast hypergeometric generator known from the literature; it is a composition/rejection method with uniform (center) and exponential (tails) envelopes. Uniform deviates were generated by the multiplicative congruential generator URAND (factor = 5 308 871 541, modulus = 2^{35}), coded in Assembler (time/deviate ≈ 8 μsec). The execution times in Table 1 for different combinations of N, M and n demonstrate that H2PE is a little faster than HRUE (unless $100 \leqslant \mu \leqslant 200$). On the other hand, it occupies more space (867 words versus 507 words, third column of Table 2) than HRUE, indicating a significantly higher complexity of H2PE. Initialization times are of interest, if only a few deviates are needed for a fixed set of parameters N, M and n. This performance measure favors also HRUE (second column of Table 2). Hence, HRUE would be a good choice for a *uniformly fast* and *short* sampling routine, supported by a double precision function for ln $k!$, and complemented by simple inversion for small means μ (say $\mu \leqslant 3$).

The table-supplied ratio of uniforms method HRUE′ is both faster and simpler (344 words versus 443 words) than H2PE′, additionally it needs also less set-up time (180 μsec) than H2PE′ (320 μsec). Consequently, HRUE′ can be recommended, if *speed* and *simplicity* are important, provided that the user is prepared to store the values of ln $k!$ in a long double-precision table.

References

[1] J.H. Ahrens and U. Dieter, Computer generation of Poisson deviates from modified normal distributions, *ACM Trans. Math. Software* **8** (1982) 163–179.
[2] J.H. Ahrens and U. Dieter, A convenient sampling method with bounded computation times for Poisson distributions, *Amer. J. Math. Management Sci.*, to appear.
[3] H.C. Chen and Y. Asau, On generating random variates from an empirical distribution, *AIIE Trans.* **6** (1974) 163–166.
[4] R.C.H. Cheng and G.M. Feast, Gamma variate generators with increased shape parameter range, *Comm. ACM* **23** (1980) 389–394.
[5] L. Devroye, *Non-uniform Random Variate Generation* (Springer, New York, 1986).
[6] V. Kachitvichyanukul and B.W. Schmeiser, Computer generation of hypergeometric random variates, *J. Statist. Comput. Simulation* **22** (1985) 127–145.
[7] V. Kachitvichyanukul and B.W. Schmeiser, Binomial random variate generation, *Comm. ACM* **31** (1988) 216–222.
[8] A.J. Kinderman and J.F. Monahan, Computer generation of random variables using the ratio of uniform deviates, *ACM Trans. Math. Software* **3** (1977) 257–260.

[9] A.J. Kinderman and J.F. Monahan, New methods for generating Student's t and gamma variables, *Computing* **25** (1980) 369–377.

[10] J.F. Monahan, An algorithm for generating chi random variables, *ACM Trans. Math. Software* **13** (1987) 168–172.

[11] E. Stadlober, Binomial random variate generation: A method based on ratio of uniforms, *Amer. J. Math. Management Sci.*, to appear.

[12] E. Stadlober, Sampling from Poisson, binomial and hypergeometric distributions: Ratio of uniforms as a simple and fast alternative, Math. Statist. Sektion 303, Forschungsgesellschaft Joanneum, Graz, 1989.

[13] J. Von Neumann, Various techniques used in connection with random digits, in: A.S. Householder et al., *The Monte Carlo Method*, Nat. Bur. Standards Appl. Math. Ser. (1951) 36–38.

[14] A.J. Walker, An efficient method for generating discrete random variables with general distributions, *ACM Trans. Math. Software* **3** (1977) 253–256.