6. Gazdar, G. English as a context-free language. Manuscript, April 1979.
7. Gazdar, G., Pullum, G.K., and Sag, I. A context-free phrase-structure grammar for the English auxiliary system. To appear.
8. Grosz, B. Focusing and description in natural language dialogues. In *Elements of Discourse Understanding*, A.K. Joshi et al. (Eds.) Cambridge Univ. Press, Cambridge, England, 1980.
9. Grosz, B., and Hendrix, G. A computational perspective on indefinite reference. Tech. Note 181, SRI International, Menlo Park, CA., 1980.
10. Hobbs, J., and Robinson, J. Why ask. *Discourse Processes 2*, 4 (Oct.–Dec. 1979) 311–318.
11. Jackendoff, R. Constraints on phrase-structure rules. In Culicover, P.W., Wasow, T., and Akmajian, A. (Eds.), *Formal Syntax*, Academic Press, New York, 1977, 249–283.
12. Jackendoff, R. *X-Bar Syntax: A Study of Phrase Structure.* MIT Press, Cambridge, MA, 1977.
13. Joshi, A.K., and Levy, L.S. Phrase structure trees bear more fruit than you would have thought. A revised and expanded version of a paper presented at the 18th Annual Meeting of the Association for Computational Linguistics, Univ. of Pennsylvania, Philadelphia, June 1980.
14. Kaplan, R.M., and Bresnan, J.W. Lexical-functional grammar: A formal system for grammatical representation. To appear in J.W. Bresnan (Ed.), *The Mental Representation of Grammatical Relations.* MIT Press, Cambridge, MA, 1980.
15. Konolige, K. Capturing linguistic generalizations with metarules in an annotated phrase—structure grammars. In Proceedings of the 18th Annual Meeting of the Association for Computing Linguistics, University of Pennsylvania, Philadelphia, June 1980.
16. Langacker, R.W. The form and meaning of the English auxiliary. *Language 54* (Dec. 1978), 853–882.
17. Paxton, W.H. The language definition system. In *Understanding Spoken Language*, D.E. Walker (Ed.) North-Holland, New York, 1978.
18. Paxton, W.H. A framework for speech understanding. Ph.D. dissertation, Stanford Univ., Stanford, CA, June 1977.
19. Robinson, J. Dependency structures and transformational rules. *Language 46* (June 1970), 259–285.
20. Robinson, A. Interpreting natural-language utterances in dialogs about tasks. Tech. Note 210, SRI International, Menlo Park, CA, March 15, 1980. (To appear.)
21. Ross, J.R. Adjectives as noun phrases. In *Modern Studies in English*, D.A. Reibel and S.A. Schane (Eds.) Prentice-Hall, Englewood Cliffs, NJ, 1969.
22. Searle, J.R. A classification of illocutionary acts. In *Proc. Texas Conf. Performatives, Presuppositions and Implicatures*, Center for Applied Linguistics, Arlington, VA, 1977.
23. Tesniere, L. *Elements de Syntaxe Structurale*, 2nd ed. Editions Klincksieck, Paris, France, 1976.
24. Wasow, T. Remarks on processing, constraints, and the lexicon. Presented at *Theoretical Issues in Natural Language Processing—2 (TINLAP—2)*, University of Illinois at Urbana-Champaign, July 25–27, 1978.
25. Woods, W.A., Transition network grammars for natural language analysis. *Comm. ACM 13*, 10 (Oct. 1970), 591–606.

Applications:
Management Science
and Operations Research
Harvey Greenberg
Editor

Generating Gamma Variates by a Modified Rejection Technique

J.H. Ahrens
University of Kiel, West Germany
and
U. Dieter
Technical University, Graz, Austria

A suitable square root transformation of a gamma random variable with mean $a \geq 1$ yields a probability density close to the standard normal density. A modification of the rejection technique then begins by sampling from the normal distribution, being able to accept and transform the initial normal observation quickly at least 85 percent of the time (95 percent if $a \geq 4$). When used with efficient subroutines for sampling from the normal and exponential distributions, the resulting accurate method is significantly faster than competing algorithms.

CR Categories and Subject Descriptors: G.3 [**Probability and Statistics**]: *statistical computing*; I.6.m [**Simulation and Modeling**]: Miscellaneous.

General Term: Algorithms

Additional Key Words and Phrases: gamma distribution, random numbers, acceptance–rejection method

1. Introduction

A number of useful algorithms for sampling from the standard gamma distribution with parameter a (mean a) have been published in recent years. For an even better utility routine the following four properties, A, V, C, and F, are desirable:

(A)—The method should be theoretically correct, and all calculations should approximately maintain single-precision accuracy.

This research was supported by the Austrian Research Council (Fonds zur Förderung der wissenschaftlichen Forschung).

Authors' present address: J. H. Ahrens, Department of Mathematics, University of Kiel, Kiel, West Germany; U. Dieter, Institut für Statistik der Technischen Universität Graz, Hamerlinggasse 6, A 8010 Graz, Austria.

(V)—The algorithm should remain efficient even if the parameter a varies all the time.

(C)—The method should guarantee constant computation times for $a \rightarrow \infty$.

(F)—Functions like ln and exp should not occur in the main paths of the algorithm.

We shall carefully consider these four requirements:

(A)—Whenever a formula is likely to produce a substantial loss of accuracy if implemented directly, a numerically safer substitute expression is proposed.

(V)—This requirement rules out procedures which need parameter-dependent tables of coefficients like the Forsythe method in Atkinson and Pearce [7]. Ahrens and Kohrt [5] describe a general table-aided inversion routine. When applied to gamma distributions, their set-up algorithm produces about 350 coefficients for 99 approximating polynomials of average degree 2.4 to 2.6. The sampling routine is then almost as fast as a square root (SQRT) and faster than a logarithm (ALOG); but the set-up is slow, and the method is therefore suitable only for repeated sampling with fixed parameters.

(C)—The best existing algorithms lead to decreasing times which stabilize quickly as a increases. The requirement C rules out the old method of taking −ln of a product of uniform deviates, as well as the algorithms of Fishman [16] and Tadikamalla [23].

(F)—Standard functions can be permitted only in steps of low probability. There seems to be no way of avoiding one square root whenever the parameter a changes. Otherwise we aim at the speed of one single logarithm, and this precludes frequent use of ln, exp, and ln Γ. The methods in Atkinson [6], Best [8, 9], Cheng [10], Cheng and Feast [11], Tadikamalla [22], and Ahrens and Dieter [2, Algorithm GC] require at least one call of ln or exp.

However, we assume that fast subprograms for sampling from the normal and exponential distributions are available, such as the ones in Marsaglia's Super-Duper set (software package at McGill University, Montreal). For our experiments we incorporated assembler programs of Algorithm FL_5 in Ahrens and Dieter [3] and Algorithm SA in Ahrens and Dieter [1], and naturally we did not mind the two-day job of debugging our final gamma method GD in assembler code. In high-level languages, procedures which violate property F can still appear to be fully competitive, particularly if something like the polar method for the normal distribution is employed, as in Cheng [10].

The literature contains three methods which conform to all requirements A, V, C, and F. The first two (GO and MS) use von Neumann's [21] rejection technique. For this a function which is proportional to some density and which majorizes the given or transformed gamma density is constructed. We use the words *hat* or *cover* for such majorizing functions.

GO is Algorithm GO in Ahrens and Dieter [2], which is restricted to $a > 2.5328$. A normal hat is constructed, but this does not majorize the right-hand tail of the gamma distribution. Therefore an additional exponential cover is needed in about 1 percent of all cases.

MS is Greenwood's [17] application of the Wilson–Hilferty transformation, as improved by Marsaglia's "squeeze" method in Marsaglia [20] ($a > \frac{1}{3}$). The transformation $x \leftarrow a(1 - \frac{1}{9}a + t/\sqrt{9a})^3$ changes the gamma probability density $\gamma(x)$ into a new function $g(t)$ which is much closer to a normal density $f(t)$ and which can be covered completely by a normal hat proportional to $f(t)$.

CF, the third Algorithm GKM 3 in Cheng and Feast [12], uses Kinderman and Monahan's [18] quotients of uniform deviates ($a > 1$); it does not employ a normal cover.

In between GO (linear transformation) and MS (third-order transformation) lies the as yet unexplored possibility of a quadratic transformation such as Fisher's $x \leftarrow (\sqrt{a - \frac{1}{4}} + t/2)^2$. This is our basic approach, but we take $x \leftarrow (\sqrt{a - \frac{1}{2}} + t/2)^2$ instead. The resulting transformed function $g(t)$ is not as close to the standard normal density $f(t)$ as the one obtained from Fisher's transformation, and it is certainly a poorer approximation when compared with the Wilson–Hilferty formula. Moreover, there is no scaling factor α such that a normal cover $\alpha f(t)$ could majorize $g(t)$, and this is true no matter which quadratic transformation is tried.

Still, our $g(t)$ has some other features which are proved in Sec. 2. The mode of $g(t)$ is at $t = 0$, but $g(0)$ is a little larger than $f(0) = 1/\sqrt{2\pi}$. Also, $g(t)$ intersects the standard normal density $f(t)$ only once at some $t = \tau(a) < 0$. Consequently, $g(t) \geq f(t)$ for all $t \geq 0$ (Figure 1 displays the case $a = 2$.) This calls for the following modification of von Neumann's [21] acceptance-rejection technique:

Generate a standard normal deviate T. If $T \geq 0$, accept $X \leftarrow (\sqrt{a - \frac{1}{2}} + T/2)^2$ as a gamma(a) sample. For $T < \tau(a)$, where $f(t)$ majorizes $g(t)$, the ratio $r(T) = g(T)/f(T)$ can be compared with a (0, 1)-uniform deviate U for an ordinary rejection test. (For simplicity this test is also applied when $\tau(a) < T < 0$. In this case $r(t) > 1$ and T is always accepted.) Obviously rejection occurs with probability

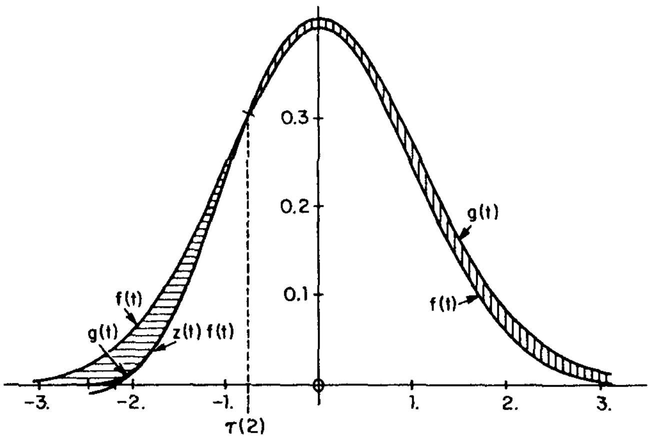


Fig. 1. Comparison of Standard Normal Density $f(t)$ with Transformed Gamma (2) Density $g(t)$.

$$P(H) = \int_{-\infty}^{\tau} (f(t) - g(t))\,dt = \int_{\tau}^{\infty} (g(t) - f(t))\,dt.$$

Hence, whenever a negative T is rejected, it must be replaced with a new $T \geq \tau(a)$, and this has to be a sample from the difference distribution whose probability density function is proportional to $g(t) - f(t)$ in $[\tau, \infty)$. Sampling from this difference is tricky, but it can be done by means of a double-exponential hat, whose proper construction requires some analysis. The hat case will also burden the final sampling Algorithm GD with some additional calculations. Therefore we shall not attempt to include parameters $0.5 < a < 1$, which our transformation would still permit. For $a > 1$, the probability of shifting the excess area on the left ($t < \tau$) over to the right ($t > \tau$) dwindles away fairly quickly ($O(1/\sqrt{a})$).

With a probability of 0.5 we have nothing but a transformed sample from the standard normal distribution, and the transformation is easier to calculate than the one in MS. (This probability could be increased by also accepting samples T between $\tau(a)$ and O immediately. However, since the calculation of $\tau(a)$ is difficult we decided against this possibility after some experimentation with timing.)

When $t < 0$, the evaluation of $r(t) = g(t)/f(t)$ can usually be replaced with a simpler function $z(t)$ which is a lower bound of $r(t)$. Such functions have been dubbed *squeezes* by Marsaglia [20]. Their faster evaluation leads to a quicker acceptance with high probability. Our squeeze $z(t)$ does not need standard functions, thus satisfying requirement F.

In order to conform to requirement A, some expressions involving small differences of large quantities will be replaced with economized polynomials. After all, there is no point in correcting the normal distribution to a transformed gamma distribution if this adjustment becomes meaningless on account of truncation errors. In fact, MS would *also* need such a device (or slow double-precision calculations).

The final Algorithm GD looks more complicated than CF, MS, and even GO. A tested Fortran version is contained in Ahrens and Dieter [4]; it may also be requested from the authors. The new method is meant to be part of a package consisting of fast machine-code routines for sampling from the most common statistical distributions. (In assembler code GD is not much more complex than GO.)

A formal statement of GD in the style of Knuth [19] is contained in Sec. 3, and computational experience is reported in Sec. 4. The new algorithm is significantly faster than its competitors for all $a > 1$. Trials with continually changing parameters yielded computation times between less than two ALOG times (large a) and about three ALOG times at $a = 1$ and approximately *one* ALOG time for $a > 10$. There the generation of one gamma deviate took only twice as long as one sample from the standard normal distribution.

2. The Method

The standard gamma(a) probability density function

$$\gamma(x) = \frac{1}{\Gamma(a)} x^{a-1} \exp(-x), \qquad a \geq 1, x \geq 0 \tag{1}$$

is transformed into a density $g(t)$ by the substitution

$$x = \left(s + \frac{t}{2}\right)^2; \qquad dx = \left(s + \frac{t}{2}\right) dt \tag{2}$$

$$\text{where } s = \sqrt{a - \frac{1}{2}},$$

$$g(t) = \frac{1}{\Gamma(a)} \left(s + \frac{t}{2}\right)^{2s^2} \exp\left(-\left(s + \frac{t}{2}\right)^2\right), \qquad a \geq 1, \quad t \geq -2s. \tag{3}$$

This $g(t)$ is close to the standard normal density

$$f(t) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}t^2\right), \qquad -\infty < t < \infty. \tag{4}$$

Figure 1 displays the case $a = 2$. To the left of $\tau(2) = -0.67988$ we have $g(t) < f(t)$. The area in between (horizontal shades) is equal to the area on the right of τ (vertical shades) where $g(t) > f(t)$. The narrow black area on the left illustrates the so-called squeeze $z(t)f(t)$ which is explained later by Eq. (11).

We now prove that $g(t)$ and $f(t)$ always intersect only once at some negative $\tau(a)$. For this the following quantities are calculated:

$$r(t) = \frac{g(t)}{f(t)} = \frac{\sqrt{2\pi}}{\Gamma(a)} \left(s + \frac{t}{2}\right)^{2s^2} \exp\left(-s^2 - st + \frac{t^2}{4}\right) \tag{5}$$

$$\begin{aligned} q(t) = \ln r(t) &= \ln \sqrt{2\pi} - \ln \Gamma(a) \\ &\quad - s^2 - st + \frac{t^2}{4} + 2s^2 \ln\left(s + \frac{t}{2}\right) \\ &= q(0) - st + \frac{t^2}{4} + 2s^2 \ln\left(1 + \frac{t}{2s}\right). \end{aligned} \tag{6}$$

$q(0)$ is analyzed by means of the Stirling approximation

$$\begin{aligned} \ln \Gamma(a) &= \ln \sqrt{2\pi} + \left(a - \frac{1}{2}\right) \ln a - a \\ &\quad + \frac{1}{12a} - \frac{1}{360a^3} + \frac{1}{1260a^5} + O(a^{-5}) \end{aligned}$$

$$\begin{aligned} q(0) &= \ln \sqrt{2\pi} - \ln \Gamma(a) - s^2 + s^2 \ln s^2 \\ &= \left(a - \frac{1}{2}\right) \ln\left(1 - \frac{1}{2a}\right) + \frac{1}{2} - \frac{1}{12a} \\ &\quad + \frac{1}{360a^3} - \frac{1}{1260a^5} + O(a^{-5}) \\ &= \frac{1}{24a} + \frac{1}{48a^2} + \frac{23}{2880a^3} \\ &\quad + \frac{1}{640a^4} - \frac{11}{40320a^5} + O(a^{-5}). \end{aligned} \tag{7}$$

The error in Eq. (7) is below 1 percent even for $a = 1$, and it fades away quickly as a increases. At any rate, we have $q(0) > 0$ and hence $r(0) > 1$, implying $g(0) > f(0)$.

Differentiating $q(t)$ in Eq. (6), we obtain

$$q'(t) = r'(t)/r(t) = \frac{t^2}{4}\Big/\left(s + \frac{t}{2}\right) \tag{8}$$

which shows that $q(t)$ and $r(t)$ are monotonically increasing for $t > -2s$. Because $g(-2s) = 0$ implies $r(-2s) = 0$ and because $r(0) > 1$, there is exactly one

$$t = \tau \quad \text{for which} \quad g(\tau) = f(\tau) \quad \text{and} \quad -2s < \tau < 0. \tag{9}$$

Hence

$$g(t) < f(t) \quad \text{if} \quad t < \tau, g(t) > f(t) \quad \text{if} \quad t > \tau. \tag{10}$$

In particular, we see that $g(t) > f(t)$ for all $t \geq 0$.

As mentioned before, the rejection test will be speeded up considerably if a simple lower bound $z(t)$ of the quotient $r(t) = g(t)/f(t)$ can be established for all negative t. Tables of $g(t)$ and $f(t)$ indicate an approximation $r(t) \approx 1 + t^3/C$, and a theory of Dieter [13] yields the same type of bound. Therefore, we tried $z(t) = 1 + t^3/C$ and determined the greatest best constant C:

$$z(t) = 1 + \frac{t^3}{12s - 4\sqrt{2}} < r(t) \quad \text{if} \quad t \leq 0. \tag{11}$$

Proof. Differentiating $z(t)/r(t)$ and using Eq. (8) yields

$$\frac{d}{dt}\left(\frac{z(t)}{r(t)}\right)' = \frac{z}{r}\left(\frac{z'}{z} - \frac{r'}{r}\right) = \frac{z}{r}\left(\frac{3t^2}{C + t^3} - \frac{t^2}{4s + 2t}\right)$$
$$= \frac{t^2(12s - C + 6t - t^3)}{r(t)C(4s + 2t)}.$$

The last denominator is positive for all $t > -2s$. The expression $n(t) = 12s - C + 6t - t^3$ in the numerator is positive if $t < 0$ and $|t|$ large. $n(t)$ decreases for $t < -\sqrt{2}$ and increases for $-\sqrt{2} < t < \sqrt{2}$. Thereafter it decreases again. At the local minimum $t = -\sqrt{2}$ we require $n(-\sqrt{2}) = 12s - C - 4\sqrt{2} \geq 0$ or $C \leq 12\text{s} - 4\sqrt{2}$. The greatest feasible value is $C = 12s - 4\sqrt{2}$, and with this we still have

$$n(t) = 12s - C + 6t - t^3 = 4\sqrt{2} + 6t - t^3$$
$$= (t + \sqrt{2})^2(2\sqrt{2} - t) > 0$$

for all $t < 2\sqrt{2}$. Hence $(z(t)/r(t))' \geq 0$ for all $t \leq 0$ and therefore $z(t)/r(t) \leq z(0)/r(0) = 1/r(0) < 1$.

We can now state the basic sampling method.

(I) Take a sample T from the standard normal distribution; calculate $X \leftarrow s + T/2$; if $T \geq 0$, return X^2 as a sample from the standard gamma(a) distribution. Obviously, the probability of this "immediate acceptance" is $P(I) = 0.5$

(S) If $T < 0$, generate a (0, 1)-uniform deviate U and return X^2 if $1 - U \leq z(T)$ [see Eq. (11)], that is, if $(4\sqrt{2} - 12s)U \leq T^3$. The probability of this "squeeze acceptance" is

$$P(S) = \int_u^0 f(t)z(t)\,dt$$
$$= \int_u^0 \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{t^2}{2}\right)\left(1 + \frac{t^3}{12s - 4\sqrt{2}}\right) dt$$
$$= \frac{1}{2} - \phi(u) - \frac{1}{\sqrt{2\pi}}\frac{1}{12s - 4\sqrt{2}}$$
$$\left(2 - (2 + u^2)\exp\left(-\frac{u^2}{2}\right)\right) \tag{12}$$

where u is determined from $z(u) = 1 + u^3/(12s - 4\sqrt{2}) = 0$ as $u = -(12s - 4\sqrt{2})^{1/3}$, and $\phi(t)$ is the standard normal distribution function.

(Q) If (S) fails, calculate $Q = q(t)$ from Eqs. (6) and (7) and accept X if $\ln(1 - U) \leq Q$. This "quotient acceptance" is the rarest case; its probability is

$$P(Q) = 1 - P(I) - P(S) - P(H)$$
$$= 0.5 - \text{P(S)} - \text{P(H)} \tag{13}$$

where $P(H)$ pertains to the next case.

(H) If (Q) also leads to rejection, a *new* sample T from the difference distribution with density proportional to $g(t) - f(t)$ in (τ, ∞) is taken and $(s + T/2)^2$ is returned. For this a hat function $h(t) \geq g(') - f(t)$ is constructed, and von Neumann's [21] acceptance–rejection procedure is used. The probability of this "hat acceptance"is

$$P(H) = \int_{-\infty}^{\tau} f(t)\,dt - \int_{-2s}^{\tau} g(t)\,dt$$
$$= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\tau} \exp\left(-\frac{t^2}{2}\right) dt$$
$$- \int_{-2s}^{\tau} \frac{1}{\Gamma(a)}\left(s + \frac{t}{2}\right)^{2s^2}$$
$$\exp\left(-\left(s + \frac{t}{2}\right)^2\right) dt$$
$$= \phi(\tau) - \Gamma\left(a, \left(s + \frac{\tau}{2}\right)^2\right) \tag{14}$$

where $\Gamma(a, x)$ is the incomplete gamma function $\Gamma(a, x) = (1/\Gamma(a)) \int_0^x t^{a-1}\exp(-t)\,dt$.

The method will be complete once a suitable hat function $h(t)$ has been selected. Convenient choices are the gamma(2), the Cauchy, or the t distribution with 2 degrees of freedom, but after some experimentation we chose a double-exponential (Laplace) hat of the form

$$h(t) = \frac{c}{\sqrt{2\pi}} \exp\left(-\frac{|t-b|}{\sigma} \right), \qquad -\infty < t < \infty. \tag{15}$$

The Laplace density has the factor $1/2\sigma$ and the exponent $-|t|/\sigma$. Hence $h(t)$ is proportional to a double-exponential density shifted to the right by b. Our notation $c/\sqrt{2\pi}$ for the factor has a technical reason: it speeds up the test in Step 11 of the Final Algorithm GD in Sec. 3.

The constants c, b, and σ have to be determined in such a way that the area below the hat $h(t)$ is as small as possible. For optimal c, b, and σ the hat function $h(t)$ will touch $g(t) - f(t)$ at least at two points. The situation is explained best by Figure 2.

(*i*) If $a \le 3.686$, the optimal $h(t)$ touches $g(t) - f(t)$ at L' and R, where $-0.29 < L' < -0.22$ and $2.18 < R < 2.22$. There is a third point L'' for which the difference $h(t) - (g(t) - f(t))$ has a local minimum.
(*ii*) If $3.686 \le a \le 13.022$, the hat contacts at L', R, *and* L'' ($-0.23 < L' < -0.21$, $1.08 < L'' < 1.20$, $2.21 < R < 2.59$).
(*iii*) If $13.022 \le a$, the optimal $h(t)$ touches $g(t) - f(t)$ at L'' and R, where $1.08 < L'' < 1.17$, $2.57 < R < 2.59$, and the local minimum of $h(t) - (g(t) - f(t))$ at L' is again positive.

The optimal b, c, and σ were calculated as follows. If $g(t) - f(t)$ is covered by a Laplace hat of smallest area touching at L and R, then

$$2\sigma = R - L \quad \text{provided that} \quad L < b < R. \tag{16}$$

This follows from Dieter [13]. L and R are determined by

$$\begin{aligned} g(L) - f(L) &= h(L); \\ g'(L) - f'(L) &= h'(L) = \frac{1}{\sigma} h(L) \\ g(R) - f(R) &= h(R); \\ g'(R) - f'(R) &= h'(R) = -\frac{1}{\sigma} h(L) \end{aligned}$$

which can be combined into

$$g'(L) - f'(L) = \frac{1}{\sigma}(g(L) - f(L));$$

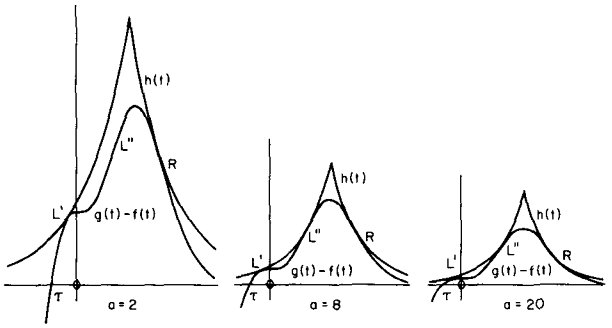
$$g'(R) - f'(R) = \frac{1}{\sigma}(f(R) - g(R)). \tag{17}$$

Equations (16) and (17) are sufficient to determine L, R, and σ. Thereafter b and c are obtained from

$$g(L) - f(L) = \frac{c}{\sqrt{2\pi}} \exp\left(\frac{L-b}{\sigma} \right);$$

$$g(R) - f(R) = \frac{c}{\sqrt{2\pi}} \exp\left(\frac{b-R}{\sigma} \right)$$

Fig. 2. Difference functions $g(t) - f(t)$ and their "Hats" $h(t)$; Examples (*i*) $a = 2$, (*ii*) $a = 8$, and (*iii*) $a = 20$.



as

$$2b = L + R + \sigma \ln \frac{g(R) - f(R)}{g(L) - f(L)};$$

$$c^2 = 2\pi e^2 (g(R) - f(R))(g(L) - f(L)). \tag{18}$$

This solves the problem in the case (*i*), where $L = L'$, and (*iii*) where $L = L''$. Numerical calculations show that there is a third point $M < b$ [$M = L''$ in (*i*), $M = L'$ in (*iii*)] for which $h(t) - [g(t) - f(t)]$ has a local positive minimum. However, this difference is negative whenever a lies between 3.686 and 13.022. Consequently, in case (*ii*) the Laplace hat of minimum area touches $g(t) - f(t)$ at three points $L' < L'' < b < R$. L' and L'' are again determined by the left half and R by the right half of Eq. (17). σ is now obtained as follows:

$$g(L') - f(L') = \frac{c}{\sqrt{2\pi}} \exp\left(\frac{L'-b}{\sigma} \right);$$

$$g(L'') - f(L'') = \frac{c}{\sqrt{2\pi}} \exp\left(\frac{L''-b}{\sigma} \right)$$

$$\sigma = \frac{\mathrm{L}'' - \mathrm{L}'}{\ln(g(L'') - f(L'')) - \ln(g(L') - f(L'))}. \tag{19}$$

In this way L', L'', R, and σ are calculated simultaneously (for instance, by Newton iteration), and b and c are worked out as before.

In the algorithm the true optimal b, c, and σ cannot be used. The tedious recalculation of these parameters would wreck the performance of the method in the case of shifting means a. Instead we insert reasonable approximations in the three cases of Figure 2; they are stated in Step 4 of Algorithm GD in Sec. 3. If $a < 3.686$, the hat case is critical enough to justify four multiplications/divisions. This number is three in the intermediate case, and for $a > 13.022$ we fix b and σ; only c still requires one division. Once the theoretically best values of b and σ are approximated, one can always make sure that c is large enough such that $h(t) > g(t) - f(t)$ holds for all t and all $a \ge 1$, and we have amassed enough evidence to be absolutely certain about this.

The theoretically best hats minimize the expected number of trials α until a sample from $h(t)$ is accepted as a sample from $g(t) - f(t)$:

Table I. Intersections τ, Probabilities of the Cases I, S, Q, H and Hat Parameters.

a	τ	$P(I)$	$P(S)$	$P(Q)$	$P(H)$	b	σ	cs	α	$\hat{\alpha}$
1	−0.7187	0.5	0.3468094	0.0309360	0.1222546	1.0813	1.2357	0.1469	1.6750	1.6772
1.5	−0.7035	0.5	0.4036618	0.0156981	0.0806401	1.2864	1.2342	0.1314	1.6050	1.6130
2	−0.6800	0.5	0.4250712	0.0117060	0.0632228	1.4211	1.2304	0.1215	1.5409	1.5557
3	−0.6424	0.5	0.4449496	0.0081460	0.0469044	1.5960	1.2234	0.1094	1.4396	1.4628
4	−0.6153	0.5	0.4549244	0.0063205	0.0387551	1.6847	1.1724	0.1058	1.3656	1.3799
5	−.0.5945	0.5	0.4611236	0.0051745	0.0337019	1.6988	1.0608	0.1117	1.3221	1.3445
7	−0.5641	0.5	0.4686248	0.0037992	0.0275760	1.7174	0.9274	0.1218	1.2822	1.3057
10	−0.5332	0.5	0.4747796	0.0027123	0.0225081	1.7351	0.8172	0.1348	1.2669	1.2824
15	−0.4997	0.5	0.4801523	0.0018311	0.0180165	1.7502	0.7470	0.1470	1.2771	1.3215
20	−0.4771	0.5	0.4831778	0.0013794	0.0154428	1.7551	0.7414	0.1484	1.2875	1.3294
50	−0.4112	0.5	0.4898723	0.0005537	0.0095740	1.7654	0.7286	0.1521	1.3123	1.3459
100	−0.3670	0.5	0.4930037	0.0002776	0.0067187	1.7699	0.7222	0.1540	1.3245	1.3527
10^3	−0.2508	0.5	0.4978650	0.0000285	0.0021065	1.7764	0.7119	0.1576	1.3438	1.3613
10^4	−0.1710	0.5	0.4993319	0.0000029	0.0006651	1.7782	0.7086	0.1587	1.3494	1.3630
10^5	−0.1165	0.5	0.4997894	0.0000003	0.0002103	1.7788	0.7076	0.1591	1.3510	1.3634
10^6	−0.0794	0.5	0.4999335	0.0000000	0.0000665	1.7789	0.7073	0.1592	1.3515	1.3635
∞	−0.0000	0.5	0.5000000	0.0000000	0.0000000	1.7790	0.7071	0.1593	1.3517	1.3635

$$\alpha = \frac{\displaystyle\int_{-\infty}^{\infty} h(t)\,dt}{\displaystyle\int_{\tau}^{\infty} (g(t) - f(t))\,dt} = \frac{2c\sigma}{(P(H)\,\sqrt{2\pi})}. \tag{20}$$

In the last two columns of Table I the theoretically possible expected numbers α are compared with the slightly larger values $\hat{\alpha}$ resulting from our approximations of b, c, and σ. The true optimum values of b, σ, and cs in Table I can be compared easily with their approximations using the simple formulas in Step 4 of Algorithm GD. The left-hand side of Table I contains the intersections $\tau[f(\tau) = g(\tau)]$ and the probabilities $P(I)$, $P(S)$, $P(Q)$, and $P(H)$ of occurrence of the four cases.

For large a, the following approximations hold:

$$\tau \to -(2s)^{-1/3};\ P(H) \to (6s\sqrt{2\pi})^{-1};$$

$$b \to \frac{1}{4}\sqrt{14} + \frac{3}{4}\sqrt{2}\ln\left(4 + \frac{\sqrt{7}}{3}\right);$$

$$c \to \frac{\sqrt{3}}{4es};\quad \sigma \to \frac{1}{\sqrt{2}};\quad \alpha \to \frac{3\sqrt{6}}{2e}.$$

In Algorithm GD we fix b and σ for $a > 13.022$. In order to be sure that $(g(t) - f(t))/h(t) < 1$ remains true for $a \to \infty$, we proved that for $\sigma = 0.75$ this quotient is extreme at $(\sqrt{31} \pm 2)/3 = t_1$ and t_2. With $b = 1.77$ the maximum permissible cs works out to $t^3 \exp(|t - 1.77|/0.75 - t^2/2)/12$, which is 0.1514764645 at t_1 and 0.1499036494 at t_2. Hence our approximation $0.1515/s$ for c is safe even if a is large.

3. The Algorithm

In the formal statement of Algorithm GD below the triggers a' and a'' ensure that the quantities s_2, s, d, $q(0)$, b, σ, and c are recalculated in Steps 1 and 4 only if this is demanded by a change of parameter a.

Steps 2 and 3 express (I) (immediate acceptance) and (S) (squeeze acceptance), and in most cases the algorithm will exit here.

The evaluation of $q(0)$ in Step 4 should *never* be done from its definition directly, using slow and badly written system routines for $\ln \Gamma(a)$ and losing precision to an intolerable degree: both $\ln \Gamma(a)$ and $s^2 \ln s^2 - s^2$ grow faster than a. Instead the approximation (7) can be modified so as to become single-precision accurate even for small $a \geq 1$. In Table II we supply three sets of coefficients q_k for 7 to 10 decimal-digit accuracy of $q(0) = \sum q_k a^{-k}$. They are obtained from our general routine for producing Chebychev-economized polynomials. Step

Table II. Coefficients of Approximating Polynomials.

$\lvert\epsilon\rvert < 3.2 \times 10^{-8}$		$\lvert\epsilon\rvert < 1.4 \times 10^{-7}$		$\lvert\epsilon\rvert < 1.1 \times 10^{-7}$	
q_1	0.04166669	a_1	0.3333333	e_1	1.0000000
q_2	0.02083148	a_2	−0.2500030	e_2	0.4999897
q_3	0.00801191	a_3	0.2000062	e_3	0.1668290
q_4	0.00144121	a_4	−0.1662921	e_4	0.0407753
q_5	−0.00007388	a_5	0.1423657	e_5	0.0102930
q_6	0.00024511	a_6	−0.1367177		
q_7	0.00024240	a_7	0.1233795		
$\lvert\epsilon\rvert < 5.5 \times 10^{-9}$		$\lvert\epsilon\rvert < 1.5 \times 10^{-8}$		$\lvert\epsilon\rvert < 2.0 \times 10^{-9}$	
q_1	0.041666661	a_1	0.33333332	e_1	1.00000000
q_2	0.020834040	a_2	−0.24999995	e_2	0.50000027
q_3	0.007970958	a_3	0.20000622	e_3	0.16666050
q_4	0.001686911	a_4	−0.16667748	e_4	0.04171864
q_5	−0.000775882	a_5	0.14236572	e_5	0.00813673
q_6	0.001274709	a_6	−0.12438558	e_6	0.00172501
q_7	−0.000506403	a_7	0.12337954		
q_8	0.000213943	a_8	−0.11275089		
$\lvert\epsilon\rvert < 2.6 \times 10^{-10}$		$\lvert\epsilon\rvert < 2.1 \times 10^{-9}$		$\lvert\epsilon\rvert < 3.1 \times 10^{-11}$	
q_1	0.0416666664	a_1	0.333333333	e_1	1.000000000
q_2	0.0208333723	a_2	−0.249999949	e_2	0.499999994
q_3	0.0079849875	a_3	0.199999867	e_3	0.166666848
q_4	0.0015746717	a_4	−0.166677482	e_4	0.041664508
q_5	−0.0003349403	a_5	0.142873973	e_5	0.008345522
q_6	0.0003340332	a_6	−0.124385581	e_6	0.001353826
q_7	0.0006053049	a_7	0.110368310	e_7	0.000247453
q_8	−0.0004701849	a_8	−0.112750886		
q_9	0.0001710320	a_9	0.104089866		

Table III. Siemens 7760 Assembler Times (in μs) for Fixed and Variable Parameter a.

							a							
Algorithm	1	1.5	2	3	5	7	10	15	20	30	50	100	10^3	10^4
							Fixed parameter a							
GD	96	80	74	67	61	59	56	54	53	52	51	50	48	48
MS	108	101	98	95	93	92	91	91	91	90	90	90	91	91
GO	—	—	—	162	152	145	137	127	119	111	101	92	77	73
							Variable Parameter a							
GD	144	132	123	110	102	101	99	96	92	90	90	90	89	89
MS	171	164	161	158	157	156	155	153	152	152	152	153	154	154
GO	—	—	—	225	215	208	201	191	184	175	165	159	141	138

4 also contains our approximations of b, σ, and c as explained in Sec. 2.

If $X \le 0$ in Step 5, then $T \le -2s$, $g(t)$ is undefined [Eq. (3)], and the hat case applies (Step 8). Otherwise Q is evaluated in Step 6 according to Eq. (6). However, severe accuracy problems can arise if $V = T/2s$ is small. So, whenever $|V| \le 1/4$ we substitute the economized expression $Q = q(0) + (T^2/2) \sum a_k V^k$ with coefficients a_k also listed in Table II. Note that $|V| \le 1/4$ is practically always true if a is large.

Step 7 expresses (Q) (quotient acceptance). In Step 8 the hat case (H) is entered: the new T is a sample from the Laplace distribution with parameters b and σ. If $T \le \tau_1 = \tau(1)$ in Step 9, then $T \le \tau(a)$ and $g(t) - f(t) \le 0$. Therefore we can reject T immediately (an omission of Step 9 would occasionally lead to overflow in Step 11.) Otherwise a new $Q(T)$ is calculated in Step 10.

In Step 11 rejection is indicated whenever $|U| > [g(T) - f(T)]/h(T)$; that is, if $|U| > [r(T) - 1]f(T)/h(T)$ [see Eq. (5)] or, using Eqs. (4), (6), (15), and $E = |T - b|/\sigma$ (Step 8), whenever $c|U| > [\exp(Q) - 1]\exp(E - T^2/2)$. If $Q \le 1/2$, the factor $\exp(Q) - 1$ is calculated as $\sum e_k Q^k$ with coefficients e_k from Table II. If T is not rejected, the new $X = (s + T/2)^2$ is returned in Step 12 (H) (hat acceptance).

Algorithm GD would look simpler without the polynomial approximations, but a direct calculation of $q(0)$, Q and $\exp(Q) - 1$ for $a = 100$ yielded errors of up to 20 percent in the test function of Step 11. For larger a, the results became completely meaningless, whereas using Table II everything works out even more accurately when a is large. On our Siemens 7760 computer with its 24-bit mantissa the top blocks in Table II were used: all truncation errors $|\epsilon|$ are below 1.3×10^{-7}. The other two sets of coefficients q_k, a_k, e_k are for computers with a better single-precision accuracy.

Algorithm GD

```
0. Preset $a' \leftarrow 0$ and $a'' \leftarrow 0$ (at compilation time).
1. If $a \ne a'$ set $a' \leftarrow a$, $s_2 \leftarrow a - 1/2$, $s \leftarrow \sqrt{s_2}$, and $d \leftarrow 4\sqrt{2} - 12s$
   $= 5.656\,854\,249\,492\,38 - 12s$.
2. Generate $T$ (standard normal deviate). Set $X \leftarrow s + T/2$. If $T \ge 0$
   return $X^2$.
3. Generate $U$ [(0, 1)-uniform deviate]. If $dU \le T^3$ return $X^2$.
4. If $a \ne a''$ set $a'' \leftarrow a$ and calculate $q0$, $b$, $\sigma$, and $c$ as follows.
   $q0 \leftarrow \sum q_k a^{-k}$ (instead of $\ln\sqrt{2\pi} - \ln\Gamma(a) - s_2 + s_2 \ln s_2$); $1 \le a$
   $\le 3.686$: $b \leftarrow 0.463 + s + 0.178 s_2$, $\sigma \leftarrow 1.235$, $c \leftarrow 0.195/s - 0.079$
   $+ 0.16s$; $3.686 < a \le 13.022$: $b \leftarrow 1.654 + 0.0076 s_2$, $\sigma \leftarrow 1.68/s$
   $+ 0.275$, $c \leftarrow 0.062/s + 0.024$; $13.022 < a < \infty$: $b \leftarrow 1.77$, $\sigma \leftarrow 0.75$,
   $c \leftarrow 0.1515/s$.
5. If $X \le 0$ go to Step 8.
6. Set $V \leftarrow T/(s + s)$ and calculate $Q$ as follows.
   If $|V| > 1/4$: $Q \leftarrow q0 - sT + T^2/4 + (s_2 + s_2)\ln(1 + V)$.
   If $|V| \le 1/4$: $Q \leftarrow q0 + (T^2/2) \sum a_k V^k$.
7. If $\ln(1 - U) \le Q$ return $X^2$.
8. Generate $E$ (standard exponential deviate) and $U$ [(0, 1)-uniform
   deviate]. Set $U \leftarrow U + U - 1$ and $T \leftarrow b + E\sigma$ sign $U$.
9. If $T \le \tau_1 = -0.718\,744\,837\,717\,19$ go to Step 8.
10. Set $V \leftarrow T/(s + s)$ and calculate $Q$ as in Step 6.
11. If $Q \le 0$ or if $c|U| > (\exp Q - 1)\exp(E - T^2/2)$ go to Step 8.
    (If $Q \le 1/2$ the factor $\exp Q - 1$ is calculated as $\sum e_k Q^k$.)
12. Set $X \leftarrow s + T/2$ and return $X^2$.
```

4. Computational Experience

Two computers, a Siemens 7760 and a Univac 1100/81, were used to check out the accuracy of all calculations and the correct fit of the hat function $h(t)$. The exits in Steps 2, 3, 7, and 12 of Algorithm GD were also counted, and the counts corresponded closely to the expected values $P(I)$, $P(S)$, $P(Q)$, and $P(H)$ in Table I. The observed Siemens 7760 computation times in Table III were based on 10,000 samples for each choice of a. In the case of variable parameters the calculations in Step 1 were carried out for every sample, and Step 4 was performed whenever the exits in Step 2 and 3 could not be taken. Corresponding measures were applied to competing Algorithms MS (Marsaglia [20]) and GO (Ahrens and Dieter [2]). We did not program CF (GKM 3 in Cheng and Feast [12]) in Assembler language, but its times should be close to those of MS.

Fortran versions of GD, MS, and GO produced the same sets of samples as the corresponding Assembler routines because they used the same Assembler subprograms for sampling from the normal and exponential distributions (Algorithms FL_5 and SA). Their computation times (for $a > 3$) were larger than the figures in Table III by the following amounts:

GD: fixed a: 35–40 μs; variable a: 35–40 μs.
MS: fixed a: 85–88 μs; variable a: 130–133 μs.
GO: fixed a: 90–115 μs; variable a: 105–130 μs.

In order to predict the performance of the new algorithm on other computers, the times in Table III should be compared with observed Siemens 7760 times for the following statements:

Y = SQRT(X): 31-32 μs;
Y = ALOG(X): 47-51 μs;
Y = EXP(X): 48-51 μs;
Y = SUNIF(IR) [(0, 1)-uniform deviate, multiplicative-congruential generator]: 10 μs;
Y = SEXPO(IR) (exponential deviate, Algorithm SA): 20 μs;
Y = SNORM(R) (normal deviate, Algorithm FL_5): 24 μs.

The claims at the end of the introduction are based on these comparisons.

Acknowledgment. The authors wish to express their appreciation and thanks to the reviewers of this article for their constructive and valuable comments. In addition, they acknowledge stimulating conversations with K.D. Kohrt, E. Stadlober, and E. Unterluggauer.

Received 12/80; revised 4/81; accepted 6/81

References

1. Ahrens, J.H., and Dieter, U. Computer methods for sampling from the exponential and normal distributions. *Comm. ACM 15*, 10 (Oct. 1972), 873-882.
2. Ahrens, J.H., and Dieter, U. Computer methods for sampling from gamma, beta, Poisson, and binomial distributions. *Computing 12* (1974), 223-246.
3. Ahrens, J.H., and Dieter, U. Extensions of Forsythe's method for random sampling from the normal distribution. *Math. Comput. 27, 124* (Oct. 1973), 927-937.
4. Ahrens, J.H., and Dieter, U. Sampling from standard gamma distributions. Submitted to *ACM Trans. Math. Softw.*
5. Ahrens, J.H., and Kohrt, K.D. Computer methods for efficient sampling from largely arbitrary statistical distributions. *Computing 26* (1981), 19-31.
6. Atkinson, A.C. An easily programmed algorithm for generating gamma random variables. *J. Roy. Stat. Soc. A 140* (1977), 232-234.
7. Atkinson, A.C., and Pearce, M.C. The computer generation of beta, gamma and normal random variables. *J. Roy. Stat Soc. A 139* (1976), 431-461.
8. Best, D.J. A new method for the computer generation of gamma or chi-squared pseudo-random variables. To appear in *J. Amer. Stat. Assoc.*
9. Best, D.J. Letter to the editors. *Appl. Stat. 27* (1978), 181.
10. Cheng, R.C.H. The generation of gamma variables with nonintegral shape parameter. *Appl. Stat. 26* (1977), 71-75.
11. Cheng, R.C.H., and Feast, G.M. Gamma variate generators with increased shape parameter range. *Comm. ACM 23*, (July 1980), 389-394.
12. Cheng, R.C.H., and Feast, G.M. Some simple gamma variate generators. *Appl. Stat. 28* (1979), 290-295.
13. Dieter, U. Optimal acceptance-rejection envelopes for sampling from various distributions. Submitted to *Math. Comput.* (1981).
14. Dieter, U., and Ahrens, J.H. Acceptance-rejection techniques for sampling from the gamma and beta distributions. Tech. Rep. 83, Dep. Statistics, Stanford University, Stanford, CA, 1974.
15. Fishman, G.S. *Principles of Discrete Event Simulation.* Wiley-Interscience, New York, 1978.
16. Fishman, G.S. Sampling from the gamma distribution on a computer. *Comm. ACM 19* (July 1976), 407-409.
17. Greenwood, A.J. A fast generator for gamma distributed random variables. *COMPSTAT* (1974), 19-27.
18. Kinderman, A.J., and Monahan, J.F. Computer generation of random variables using the ratio of uniform deviates. *ACM Trans. Math. Sofw. 3*, (3 Sept. 1977), 257-260.
19. Knuth, D.E. *The Art of Computer Programming, Vol. 2: Seminumerical Algorithms*, 2nd ed. Addison-Wesley, Reading, MA, 1981.
20. Marsaglia, G. The squeeze method for generating gamma variates. *Comput. Math. Appl. 3* (1977), 321-325.
21. von Neumann, J. Various techniques used in connection with random digits. *Collected Works, Vol. 5, 768-770.* Pergamon Press, New York, 1963.
22. Tadikamalla, P.R. A survey of methods for sampling from the gamma distribution. *Proc. Winter Simulation Conf. 1978* Miami, Florida, Dec. 4-6, 1979, 131-134.
23. Tadikamalla, P.R. Computer generation of gamma random variables I and II. *Comm. ACM 21*, 5 (May 1978), 419-422; (Nov. 1978), 925-928.
24. Vaduva, I. On computer generation of gamma random variables by rejection and composition procedures. *Math. Oper.-forschung und Stat., Ser. Stat. 4* (1977), 545-576.