AD-A044 566 SOUTHERN METHODIST UNIV DALLAS TEX DEPT OF INDUSTRIA--ETC F/G 12/1
REJECTION METHODS FOR BETA VARIATE GENERATION.(U)
JUL 77 B SCHMEISER, M A SHALABY N00014-77-C-0425
UNCLASSIFIED IEOR-77014 NL

1 OF 1
AD A044566

END
DATE
FILMED
10-77
DDC

SOUTHERN METHODIST UNIVERSITY

DEPARTMENT OF INDUSTRIAL ENGINEERING AND OPERATIONS RESEARCH
School of Engineering and Applied Science
Dallas, Texas 75275

Technical Report IEOR-77014

REJECTION METHODS FOR BETA VARIATE GENERATION

Bruce Schmeiser

and

Mohamed A. Shalaby

Department of Industrial Engineering and
Operations Research
Southern Methodist University
Dallas, Texas 75275

July 1977

This work supported in part by the Office of Naval Research under Contract N00014-77-C-0425

Comments and criticisms from interested readers are cordially invited.

ABSTRACT

The computer generation of pseudo-random variates from the beta distribution is discussed. Presented are three exact methods applicable for parameter values $p > 1$ and $q > 1$. All three methods use rejection from regions defined by the location of the points of inflexion. The methods differ in the amounts of set-up required, with higher set-up times generally resulting in lower marginal generation times. Generation times are compared to three methods in current use: those of Jöhnk, Fishman, and Ahrens and Dieter. Over a wide range of parameter values marginal generation times are reduced about 50%. Substantial set-up time savings and moderate marginal time savings are obtained compared to Forsythe's approach as recently developed by Atkinson and Pearce.

KEY WORDS

Beta Distribution

Simulation

Random Variate Generation

Rejection Methods

Monte Carlo

Distribution Sampling

1. INTRODUCTION

The pseudo-random generation of values from the beta distribution is considered in this paper. The standardized beta distribution has density function

$$f_\beta(x) = x^{p-1} (1-x)^{q-1}/\beta(p,q) \qquad 0 \leq x \leq 1$$

where the beta function is defined

$$\beta(p,q) = \int_0^1 u^{p-1} (1-u)^{q-1} du$$

Since the more general form of beta variates over the range [a,b] may be obtained by the simple transformation $y = a + x(b-a)$, only the standard form is considered here. This paper considers parameter values $p > 1$ and $q > 1$. For these parameter values $\lim_{x \to 0} f_\beta(x) = 0$, $\lim_{x \to 1} f_\beta(x) = 0$, and there is a unique mode at $x = (p-1)/(p+q-2)$.

1.1 Application

Over the interval [a,b] the beta distribution is used as a model to describe random phenomenon. A common example is to represent activity times in PERT networks (See, e.g., Clark (1962)). The beta distribution is well-suited to this purpose, as it can assume a wide variety of shapes. Schmeiser and Deutsch (1977) include beta generators in their discussion of general process generators, due to the distribution's ability to assume a wide range of skewness and kurtosis values. In particular, the Bernoulli trial and uniform distributions are special cases and the gamma and normal distributions are limiting cases.

The beta distribution also arises in a number of theoretical ways. The k^{th} order statistic of a sample of size n of uniform (0,1) values has density function $f_\beta(x; k, n-k+1)$. Ramberg and Tadikamalla (1977) use this result to generate order statistics from distributions having a closed-formed inverse cdf. Any method of generating from the beta distribution with fractional p and q is also a method for generating from other related distributions. In particular, if X is distributed $f_\beta(\frac{v}{2}, \frac{w}{2})$, then $Y = (w/v)X/(1-X)$ has an F distribution with v and w degrees of freedom (Patel, Kapadia and Owen [1976]). If X is beta (1/2,w/2), then $Y = s[wX/(1-X)]^{1/2}$ has a Student's t distribution with w degrees of freedom, where s is a random sign. The Pearson Type VI distribution

$$f(y) = y^w/((1+y)^v \beta(w+1, v-w-1)) \qquad y \geq 0$$

may be generated using $Y = X/(1-X)$ where X has a beta (w+1, v-w-1) distribution. These and other applications are discussed in Johnson and Kotz (1970).

1.2 Survey of the Literature

Two approaches for generating pseudo-random beta variates have appeared in the literature: those based on specific properties of the beta distribution and those based on rejection from the density function. They are discussed in turn.

Three methods based on specific properties of the beta distribution have been proposed and used. When p and q are integer the first method, due to Fox (1963), returns the p^{th} order statistic of a sample of size q+p-1 from the uniform distribution as a beta (p,q) observation. Rather than sorting to determine the order statistics, the order statistics may be generated more efficiently by using the technique of Schucany (1972) if p is close to q+p-1 or the technique of Lurie and Hartley (1972) if p is close to one. The second method, due to Jöhnk, is based on $X = Y^{1/p}/(Y^{1/p} + Z^{1/q})$ being beta (p,q) if $Y^{1/p} + Z^{1/q} \leq 1$, where Y and Z are independent uniform (0,1) random variables. Both Jöhnk's method and the third method are valid for all $p > 0$ and $q > 0$. The third method is based on the result that $X = Y / (Y + Z)$ has a beta distribution if Y and Z are independent gamma random variables with shape parameters $\alpha_1 = p$ and $\alpha_2 = q$, respectively. While any exact method for generating the gamma variates could be used, the method presented by Fishman (1973) is considered here: the product of a U-shaped beta variate ($p, q < 1$) and an exponential variate are added to an Erlang variate to obtain the appropriate gamma variate. (Other techniques for generating gamma random variates may be found in McGrath and Irving (1973), Ahrens and Dieter (1974), Wallace (1974), Whittaker (1974), and Fishman (1976)).

Three other methods for beta generation based on rejection via comparison with the density function have been proposed. The first method, applicable when $p \geq 1$ and $q \geq 1$, is the general rejection technique consisting of the generation of uniformly distributed points (x,v) in the rectangle formed by (0,0), (0, f(p-1, p+q-2)), (1, f(p-1, p+q-2)), and (1,0) and the acceptance of x as the beta variate if $v \leq f(x)$. (See, e.g., Lewis (1975)). A better fitting region than the rectangle was proposed by Ahrens and Dieter (1974). Their algorithm BN generates uniformly distributed points (x,v) in a region based on the normal distribution. For the parameter values $p > 1$ and $q > 1$, the probability of acceptance is increased considerably compared to the first rejection method. Recently Atkinson and Pearce (1976) proposed a method, but did not specify an algorithm, based on the work of Forsythe (1972). An elaborate set-up step, involving numerical integration, was used to allow efficient generation in terms of marginal generation time per variate. A fourth method, efficient when at least one parameter is less than one, is the "switching" algorithm given by Atkinson and Whittaker (1976). This rejection method, which is based on decomposing the density function into a product of two functions, is not efficient when both parameter values are greater than one.

1.3 Motivation and Direction of the Research

This paper is motivated by the need for techniques capable of generating pseudo-random beta variates faster than existing techniques allow. The commonly encountered case of $p > 1$ and $q > 1$ is considered. In many situations requiring computer generated beta variates, many observations are obtained for fixed values of p and q. It is in these situations that the three techniques presented here will be of greatest benefit. By spending some additional time in initialization, the marginal time per variate is reduced.

Two tactics are employed to reduce the number of time consuming operations necessary to generate each variate. The composition-rejection method (see, e.g., Jansson (1966)), using only triangular and rectangular regions from which it is easy to generate random points, is combined with a preliminary rejection and/or acceptance step which reduces the number of times the beta density function must be evaluated. Variations of these tactics, combined with some standard practices, produce three techniques which have various initialization, marginal time, and memory requirements. The three methods are developed in section 2. In section 3 they are compared to existing methods.

2. DEVELOPMENT OF THE TECHNIQUES

Developed in this section are three related techniques for generating beta variates for $p > 1$ and $q > 1$. All three techniques generate a point (x,v) distributed uniformly over a region defined by a function $t(x)$ which envelopes $f(x)$ (i.e., $t(x) \geq f(x)$ for all $x \varepsilon [0,1]$). The value x is delivered as the next observation if $v \leq f(x)$. Tocher (1963) proves that such an approach is valid.

This rejection approach is modified in all three techniques presented here. A function $b(x)$ is defined such that $b(x) \leq f(x)$ for all $x \varepsilon [0,1]$. Then x is delivered as the next observation if $v \leq b(x)$ or if $v \leq f(x)$. By defining $b(x)$ as a piece-wise linear function, substantial computation is saved by the capability of accepting x based on comparisons with $b(x)$ rather than $f(x)$, since $f(x)$ involves time consuming operations. Note that the use of $b(x)$ reduces the computation required to accept or reject x, but does not change the probability of acceptance or rejection.

It should be noted that rejection techniques do not need to consider the normalizing constant; in this case $\beta(p,q)^{-1}$. However, simply ignoring $\beta(p,q)$ causes scaling problems. Therefore, following Ahrens and Dieter (1974), define $P = p-1$, $Q = q-1$, $R = P + Q$, and

$$f(x) = (x/P)^P((1-x)/Q)^Q R^R \qquad 0 \leq x \leq 1 \quad (1)$$

which is proportional to $f_\beta(x)$, does not involve $\beta(p,q)$ and has no scaling problems since evaluation at the mode always yields a value of one.

All three techniques presented in this paper use the result that the points of inflexion of the beta distribution lie at

$$[P \pm (PQ/(R-1))^{1/2}]/R$$

if the values lie between zero and one and are real. The density function is concave between the points of inflexion and convex at all other points in [0,1]. In addition for $p > 1$ and $q > 1$, the mode is located at P/R which lies between the points of inflexion. (See, e.g., Johnson and Kotz (1970)).

For any $P > 0$ and $Q > 0$, define the constants

$$x_1 = \begin{cases} 0 & \text{if } x_2 = 0 \\ x_2 - x_2(1-x_2)/(P-Rx_2) & \text{otherwise} \end{cases}$$

$$x_2 = \begin{cases} (P - (PQ/(R-1))^{1/2})/R & \text{if real and in } [0,1] \\ 0 & \text{otherwise} \end{cases}$$

$$x_3 = P/R \tag{2}$$

$$x_4 = \begin{cases} (P + (PQ/(R-1))^{1/2})/R & \text{if real and in } [0,1] \\ 1 & \text{otherwise} \end{cases}$$

$$x_5 = \begin{cases} 1 & \text{if } x_4 = 1 \\ x_4 + x_4(1-x_4)/(P-Rx_4) & \text{otherwise} \end{cases}$$

The mode is represented by x_3. If there are inflexion points, they are represented by x_2 and x_4. If there are no points of inflexion, then $x_2 = 0$ and $x_4 = 1$. x_1 and x_5 are the points at which the lines tangent to the density function at x_2 and x_4 cross the X axis. If $x_2 = 0$, then $x_1 = 0$ and if $x_4 = 1$, then $x_5 = 1$. Figure A illustrates the location of these points.

Figure A About Here

2.1 Modified Ahrens and Dieter Technique

The Ahrens and Dieter (1974) algorithm BN is repeated here for convenience.

1. Set $P \leftarrow p - 1$, $Q \leftarrow q - 1$, $R \leftarrow P + Q$, $L \leftarrow R \ln R$,
   $\mu = P/R$, $\sigma = 0.5/R^{1/2}$
2. Take an observation s from the standard normal distribution and form $x \leftarrow s\sigma + \mu$.
3. If $x < 0$ or $x > 1$ go to step 2.
4. Generate $u \sim U(0,1)$.
5. If $\ln u > P \ln(x/P) + Q \ln((1-x)/Q) + L + .5s^2$ go to step 2.
   Otherwise, deliver x as the beta (p,q) observation.

Here the enveloping function is $t(x) = \exp(-s^2/2)$ where $s = 2(x-P/R)R^{1/2}$. The modification proposed is to sometimes bypass the time consuming step 5 through the use of the preliminary acceptance function

$$b_1(x) = \begin{cases} 0 & 0 \le x \le x_1 \\ (x-x_1)/(x_3-x_1) & x_1 < x \le x_3 \\ (x_5-x)/(x_5-x_3) & x_3 < x \le x_5 \\ 0 & x_5 < x \le 1 \end{cases}$$

as illustrated in Figure B. Proposition 1 forms the basis for the modified Ahrens and Dieter technique BNM proposed here.

Figure B About Here

Proposition 1: For x_1, x_3 and x_5 defined as in equation (2), $b_1(x) \leq f(x)$ for all $x \epsilon [0,1]$.

Proof: The proof is trivial for $0 \leq x \leq x_1$ and $x_5 < x \leq 1$, since $b_1(x) = 0 \leq f(x)$ over these intervals. Now consider the interval $[x_1, x_3]$. Let $y(x) = f(x_2) + f'(x_2)(x-x_2)$, the tangent to $f(x)$ at the inflexion point x_2. Define $g(x)$ as

$$g(x) = \begin{cases} y(x), & x \epsilon [x_1, x_2] \\ f(x), & x \epsilon (x_2, x_3]. \end{cases}$$

Then $g(x)$ is differentiable everywhere on $[x_1, x_3]$ since

$$\lim_{h \to 0^+} \frac{g(x_2+h)-g(x_2)}{h} = \lim_{h \to 0^-} \frac{g(x_2+h)-g(x_2)}{h} = f'(x_2).$$

Also $g'(x)$ is monotone non-increasing on $[x_1, x_3]$, first on $[x_1, x_2]$ by linearity of $y(x)$, and second on $(x_2, x_3]$ by concavity of $f(x)$ (Roberts and Varberg (1973)). Hence $g(x)$ is concave on $[x_1, x_3]$. Moreover, $y(x) \leq f(x)$ on $[x_1, x_2]$ since $f(x)$ is convex in this region and $y(x)$ is the line of support at x_2. It follows that $g(x) \leq f(x)$ for all $x \epsilon [x_1, x_3]$. Now for $x \epsilon [x_1, x_3]$, let $x = \lambda x_1 + (1-\lambda) x_3$ for $0 \leq \lambda \leq 1$. Then

$$\begin{aligned} f(x) \geq g(x) &\geq \lambda g(x_1) + (1-\lambda)\, g(x_3) \\ &= \lambda b_1(x_1) + (1-\lambda)\, b_1(x_3) \\ &= b_1(x) \qquad \text{by linearity of } b_1(\cdot). \end{aligned}$$

Similar logic is applicable to the interval $[x_3, x_5]$, completing the proof of Proposition 1.

The Ahrens and Dieter technique may now be modified as follows:

a. Add to step 1 the calculation of x_1 and x_5 (which require x_2 and x_4 as intermediate values) and note that $x_3 = \mu$.

b. Between steps 4 and 5 add step 4b as follows:

If $u \exp(-s^2/2) \le b_1(x)$, deliver x.

Otherwise go to step 5.

No other changes are necessary.

2.2 The "Two-Point" Technique

A second method for generating beta variates is developed in this section. The normal enveloping function of the Ahren's and Dieter approach is replaced in this section with a piece-wise linear function $t_1(x)$. Calculations necessary to determine $t_1(x)$ allow the use of a better fitting preliminary acceptance function $b_2(x)$. The functions $t_1(x)$ and $b_2(x)$ require $f(x)$ to be evaluated at x_2 and x_4 in step 1. This method will be referred to as the two-point technique 2P. It is illustrated in Figure C.

Figure C About Here

Consider the enveloping function $t_1(x)$ defined as

$$t_1(x) = \begin{cases} x\, f(x_2)/x_2 & 0 \le x \le x_2 \\ 1 & x_2 < x \le x_3 \\ 1 & x_3 < x \le x_4 \\ (1-x)\, f(x_4)/(1-x_4) & x_4 < x \le 1 \end{cases}$$

Proposition 2: For x_1, x_3, and x_5 defined x in equations (2), $t_1(x) \ge f(x)$ for all $x \varepsilon [0,1]$.

Proof: Consider $0 \le x \le x_2$. Since $f(x)$ is convex over the interval,

$$f((1-\lambda)(0)+\lambda x_2) \le (1-\lambda)f(0) + \lambda f(x_2)$$

or

$$f(\lambda x_2) \leq \lambda f(x_2)$$

Since $x = \lambda x_2$ in this interval,

$$f(x) \leq (x/x_2)f(x_2) = t_2(x) \qquad \text{for all } x\varepsilon[0,x_2]$$

proving the result for all $x\varepsilon[0,x_2]$. The proof follows similarly for $x\varepsilon[x_4, 1]$. For $x\varepsilon[x_2, x_4]$, $t_2(x) = 1 = f(x_3) \geq f(x)$ for all $x\varepsilon[x_2, x_4]$ since x_3 is the mode of the distribution. Thus $t_1(x)$ is seen to be an enveloping function for $f(x)$ and Proposition 2 is proved.

A better preliminary acceptance function than $b_1(x)$ may also be defined based on $f(x_2)$ and $f(x_4)$. Let

$$b_2(x) = \begin{cases} 0 & 0 \leq x \leq x_1 \\ (x-x_1)f(x_2)/(x_2-x_1) & x_1 < x \leq x_2 \\ f(x_2) + (x-x_2)(1-f(x_2))/(x_3-x_2) & x_2 < x \leq x_3 \\ f(x_4) + (x_4-x)(1-f(x_4))/(x_4-x_3) & x_3 < x \leq x_4 \\ (x_5-x)f(x_4)/(x_5-x_4) & x_4 < x \leq x_5 \\ 0 & x_5 < x \leq 1 \end{cases}$$

Based on Proposition 3, $b_2(x)$ may be used to deliver x in some instances without evaluating $f(x)$.

Proposition 3. For x_1, x_2, x_3, x_4, and x_5 defined as in equations (2), $b_2(x) \leq f(x)$ for all $x\varepsilon[0,1]$.

Proof: For $x\varepsilon[0,x_1]$ and $x\varepsilon[x_5,1]$ the proposition follows trivially, since $b_2(x) = 0$, and $f(x) \geq 0$. For $x\varepsilon[x_1,x_2]$, $b_2(x)$ is the tangent to f(x) at x_2, hence $b_2(x) \leq f(x)$ by convexity of f(x). For $x \varepsilon [x_2,x_3]$, $b_2(x)$ is a cord to f(x) and $b_2(x) \leq f(x)$ by concavity of f(x) in this interval. It follows that $b_2(x) \leq f(x)$ for all $x \varepsilon [x_1,x_3]$. Similar logic applies to prove that $b_2(x) \leq f(x)$ for all $x \varepsilon [x_3,x_5]$, thus proving the proposition.

Just as the function $b_1(x)$ can be used to accept x without comparison to f(x), the normal density function can be used to reject x without comparison to f(x). Define $\phi(x) = \exp(-2R(x-x_3)^2)$. The point (x,v) may be rejected if $v > \phi(x)$ since $\phi(x) \geq f(x)$ for all $x \varepsilon [0,1]$, as proved by Ahrens and Dieter (1974).

The two-point technique proceeds as follows:

1. Calculate x_1, x_2, x_3, x_4, and x_5 using equations (2) and $f(x_2)$ and $f(x_4)$ using equation (1).
2. Generate a value x with probabilities proportional to $t_1(x)$.
3. Generate a value v distributed $U(0, t_1(x))$.
4. If $v \leq b_2(x)$ deliver x.
5. If $v \geq \phi(x)$ go to step 2.
6. If $v \leq f(x)$ deliver x, otherwise go to step 2.

Actual implementation of the algorithm proceeds best by considering each interval separately, due to the fragmented definitions of $t_1(x)$ and $b_2(x)$.

2.3 The "Four Point" Technique

The third technique for beta variate generation is given here. Based on evaluating f(x) at x_1, x_2, x_4, and x_5, the "four point" technique 4P uses the better fitting enveloping function

$$t_2(x) = \begin{cases} xf(x_1)/x_1 & 0 \leq x \leq x_1 \\ f(x_1)+(x-x_1)[f(x_2)-f(x_1)]/(x_2-x_1) & x_1 < x \leq x_2 \\ 1 & x_2 < x \leq x_3 \\ 1 & x_3 < x \leq x_4 \\ f(x_5)+(x_5-x)[f(x_4)-f(x_5)]/(x_5-x_4) & x_4 < x \leq x_5 \\ (1-x)f(x_5)/(1-x_5) & x_5 < x \leq 1 \end{cases}$$

as illustrated in Figure D. Proposition 4 states that $t_2(x)$ is indeed an enveloping function.

Figure D About Here

Proposition 4: For x_1, x_2, x_3, x_4, and x_5 defined in equations (2), $t_2(x) \geq f(x)$ for all $x \varepsilon [0,1]$.

Proof: The proof follows the logic of the proof of Proposition 2, but considers the intervals $[0,x_1]$, $[x_1,x_2]$, $[x_4,x_5]$ and $[x_5,1]$ individually. The logic is not repeated here.

For the implementation of the 4P algorithm $t_2(x)$ involves trapezoidal regions for the intervals $[x_1,x_2]$ and $[x_4,x_5]$. Values of x can be generated from each trapezoid through generating from a rectangle and a triangle with probabilities proportional to their areas. However, since each rectangle is entirely below f(x) (see fig. D) all the points (x,v) generated with the rectangle probabilities will be accepted and no rejection test need be performed in this case. Likewise, for all the points (x,v) generated with probabilities proportional to the inner rectangle areas for the $[x_2,x_3]$ and $[x_3,x_4]$ regions no rejection test is needed. Thus the implementation of the four point technique, also given in the Appendix, differs from the two point technique in that four regions of zero rejection probability are created.

3. COMPUTATIONAL EXPERIENCE

In this section the three methods developed in section 2; designated BNM, 2P and 4P; are compared to the Ahrens and Dieter method BN, the ratio of gamma random variates method RG, and Jöhnk's method JK. Section 3.1 discusses the implementation of the comparison, and section 3.2 discusses the results of the comparisons.

3.1 Implementation

All timing comparisons were made on the SMU CDC CYBER 72 using the KRONOS 2.1.2-A operating system. Each of the six algorithms were implemented in FORTRAN using the FTN compiler at optimization level 0. The U(0,1) values were obtained from RANF, the relatively fast generator built into FTN.

Each method was coded as efficiently as is reasonable. The Ahrens and Dieter methods, both original and modified, were implemented using the Kinderman and Ramage (1976) algorithm for generating normal deviates, as this appears to be the fastest method available. (The initial use of another generator resulted in two to three times inflation of computation time.) The ratio method RG was implemented using the approach described in Fishman (1973). The source listings for all programs used to obtain the computational results are available from the authors.

For each combination of p, q and method, four samples of size 1000 were generated. Never is the ratio of standard deviation of average time to average time greater than .01. The least significant digit shown is correct to within one or two units.

3.2 Computational Time Comparisons

Symmetric distributions lead to the fastest generation times for all six methods. Table 1 shows marginal generation times for the six methods over a wide range of parameter values $p = q$, corresponding to the symmetric case. Method JK deteriorates rapidly as $p = q$ increases, due to the increasing probability of rejection. Method RG is quite competitive for small integer values, but requires two beta generations using JK for fractional $p = q$ and requires $p + q$ uniform observations for all values of p and q, making RG inefficient for large and/or fractional parameter values. Methods BN and BNM perform best for large parameter values, due to the asymptotic normality of the beta distribution.

It is clear from Table 1 that the three methods presented in this paper reduce marginal computation time substantially. Method BNM reduces the time of BN by 28% for $p = q \geq 100$, with less reduction for smaller parameter values. For p and q less than two, the 4P reduces to the logic 2P and hence the times are very similar. For larger values of p and q, 4P is substantially faster.

Table 1 can be used to compare the three methods of this paper with Forsythe's method. As applied to the beta distribution by Atkinson and Pearce (1976), Forsythe's method requires 63% to 85% of the marginal time needed by the BN method for parameter values ranging from two to seven. Thus, over this range, Forsythe's method appears to be about as fast as BNM, and somewhat slower than 2P and 4P. However, the Forsythe method requires a set-up step equal to about 1000 marginal generation times. This is considerably more than required by any of the six methods compared in Table 1.

1. Marginal Generation Times (in μ Seconds) for Symmetric Beta Distributions

	METHOD					
p=q	JK	RG	BN	BNM	2P	4P
1.01	1.2	2.0	1.8	1.8	.37	.39
1.2	1.4	2.2	.85	.80	.39	.41
1.5	1.9	2.3	.77	.69	.42	.43
2	3.4	.56	.77	.64	.45	.46
3	11.5	.62	.74	.60	.31	.27
5	*	.77	.72	.55	.35	.26
10	*	1.1	.69	.51	.44	.28
100	*	7.1	.67	.48	1.2	.45
1000	*	*	.67	.50	3.6	.97

* Not calculated due to excessive time requirement.

The set-up times for the six methods vary widely. Both JK and RG require no set-up time, since neither makes use of constants based on p or q. The BN method requires calculation of a natural logarithm and a square root. The modified version, BNM, requires an additional square root. The 2P and 4P methods require varying amounts of set-up time ranging from no higher order operations to the evaluation of f(x) two and four times, respectively.

Table 2 shows generation times for the six methods when the set-up logic is performed for each observation generated. Methods JK and RG are unchanged, since they require no set-up. For parameter values less than two, 2P and 4P are fastest for both set-up and marginal generation times. For parameter values greater than two, the largest change from Table 1 is for 4P, with 2P, BNM and BN requiring decreasing set-up time. For these cases symmetry was not used as a computational advantage. Therefore the times shown are also indicative of the results for asymmetric distributions.

It can be seen from Tables 1 and 2 that the additional set-up time of 4P usually results in lower marginal times than BN or BNM. The number of values which must be generated to break-even ranges from less than one for small values of p and q to about five for $p = q = 10$ and to about 60 for $p = q = 100$. Thus 4P is fastest except for large values of p and q combined with small sample sizes.

Table 3 shows marginal execution times for all six methods for a range of parameter values corresponding to asymmetric distributions. Several points can be made concerning Table 3. First

2. Generation Times (in μ Seconds)
Including One Time Set-Up For
Symmetric Beta Distributions

	METHOD					
p=q	JK	RG	BN	BNM	2P	4P
1.01	1.2	2.0	2.2	2.0	.69	.83
1.2	1.4	2.2	1.1	.99	.69	.84
1.5	1.9	2.3	1.0	.86	.68	.81
2	3.4	.56	1.0	.96	.87	1.0
3	11.5	.62	1.0	1.0	1.7	2.7
5	*	.77	.99	.98	1.7	2.6
10	*	1.1	.99	.95	1.8	2.7
100	*	7.1	.98	.91	2.6	3.1
1000	*	*	.99	.94	5.1	3.6

* Not calculated due to excessive time requirement.

3. Marginal Generation Times (in μ Seconds) for Asymmetric Beta Distributions

		Method					
p	q	JK	RG	BN	BNM	2P	4P
1.01	1.5	1.4	2.2	.91	.82	.44	.44
	2	1.7	1.3	.93	.74	.48	.47
	5	3.4	1.4	1.2	1.1	.91	.44
	10	6.0	1.5	1.6	1.5	1.8	.77
	100	*.	4.5	4.6	4.7	16.1	5.7
1.5	2	2.5	1.5	.78	.69	.44	.46
	5	6.4	1.5	.82	.70	.47	.33
	10	*	1.7	.97	.87	.81	.42
	100	*	4.7	2.5	2.6	6.7	2.3
2	5	12.0	.67	.78	.63	.43	.35
	10	*	.83	.89	.76	.68	.40
	100	*	3.8	2.0	2.0	4.9	1.6
5	10	*	.93	.74	.58	.45	.28
	100	*	3.9	1.4	1.3	2.7	.84
10	100	*	4.0	1.1	1.0	1.8	.62

* Not calculated due to excessive time requirement.

note that RG is fast for small integer parameters and is fairly slow for fractional parameter values. Second, note that BNM saves 10-20% compared to BN, except for large parameter values. Finally, note that 4P dominates the other methods for all parameter values shown, except for p = 1.01 and q = 100 where RG, BN, and BNM are all faster.

3.3 Other Comparisons

Only the speeds of the six methods were considered in section 3.2. Some comments concerning other criteria are given here. Since all are exact techniques no discussion of accuracy is given.

The storage requirements of the method varies widely. In order of ascending storage requirements the methods rank JK, RG, 2P, 4P, BN and BNM. This ranking includes the use of the Kinderman-Ramage normal generator. At the expense of computational speed, the use of another normal generator requiring less core could place BN and BNM before 2P and 4P.

Generality is the other important criterion for selection of generators. Fox's method, using order statistics, is valid only for integer parameter values. Only JK and RG are theoretically valid for any $p > 0$ and $q > 0$. The other methods apply only for $p > 1$ and $q > 1$.

4. SUMMARY AND CONCLUSIONS

The three rejection methods BNM, 2P and 4P for generating beta variates for $p > 1$ and $q > 1$ have been developed and compared to three existing methods JK, RG, and BN. As a group, the three new methods dominate the three existing methods in terms of generation time. In particular, the best overall performer of the existing methods, BN, is dominated by BNM for all parameter values in terms of both marginal time and set-up time. However, BNM does not dominate either 2P or 4P. Except for some insignificant cases, method 4P dominates method 2P, due to $t_2(\cdot)$ fitting the tails better than $t_1(\cdot)$.

In conclusion, method 4P is fastest except in the following three cases:

1) If parameter values are integer, $p + q \leq 20$, and the parameter values change for each observation generated, method RG is fastest. Note that in this case Fox's order statistics method is appropriate.
2) If the standard deviation is less than approximately .02 BNM is fastest, due to the good fit of the normal density function and the corresponding heavy tails. (A method 6P would be effective here.)
3) If p and q are greater than two and the parameter values change for each observation generated, then BNM is fastest due to its relatively fast set-up compared to 2P and 4P.

Methods 2P and 4P may be modified in a variety of ways. 2P could include two additional regions having zero probability of rejection. Both 2P and 4P could use various other $t(\cdot)$ and $b(\cdot)$ functions. In particular, $b_2(x)$ could be made to better fit $f(x)$ in 4P by making use of $f'(x_1)$ and $f'(x_5)$. Finally, methods corresponding to 6P, 8P, ... could easily be developed based on the results of section 2. While these modification would be very useful for large parameter values, the added complication would probably not be worthwhile for most users.

REFERENCES

Ahrens, J. H. and Dieter, U. (1974), "Computer Methods for Sampling From Gamma, Beta, Poisson and Binomial Distributions," Computing, 12, 223-246.

Atkinson, A. C., and Pearce, M. C. (1976), "The Computer Generation of Beta, Gamma and Normal Random Variables," Journal of the Royal Statistical Society, A, 139, 431-461.

Atkinson, A. C., and Whittaker, J. (1976), "A Switching Algorithm for the Generation of Beta Random Variables with at Least One Parameter Less Than 1," Journal of the Royal Statistical Society, A, 139, 462-467.

Clark, C. E. (1962), "The PERT Model for the Distribution of an Activity Time," Operations Research, 10, 405-406.

Fishman, G. S. (1973), Concepts and Methods in Discrete Event Digital Simulation, New York: John Wiley & Sons.

Fishman, G. S. (1976), "Sampling from the Gamma Distribution on a Computer," Communications of the ACM, 19, 407-409.

Forsythe, G. E. (1972), "Von Neumann's Comparison Method for Random Sampling from the Normal and Other Distributions," Mathematics of Computation, 26, 817-826.

Fox, B. L. (1963), "Generation of Random Samples from the Beta and F Distributions," Technometrics, 5, 269-270.

Jannson, B. (1966), Random Number Generators, Stockholm: Almquist and Wiksell.

Jöhnk, M. D. (1964), "Erzeugung von Betaverteilten und Gammaverteilten Zufallszahlen," Metrika, 8, 5-15.

Johnson, N. L., and Kotz, S. (1970), Continuous Univariate Distributions - 2, New York: John Wiley & Sons.

Kinderman, A. J., and Ramage, J. G. (1976), "Computer Generation of Normal Random Variables," Journal of the American Statistical Association, 71, 893-896.

Lewis, T. G. (1975), Distribution Sampling for Computer Simulation, Lexington, Massachusetts: D. C. Heath and Company.

Lurie, D., and Hartley, H. O. (1972), "Machine Generation of Order Statistics for Monte Carlo Computations," The American Statistician, 26, 26-27.

McGrath, E. J., and Irving, D. C. (1973), Techniques for Efficient Monte Carlo Simulation. Vol. II. Random Number Generation for Selected Probability Distributions, Springfield, Virginia: National Technical Information Service.

Patel, J. K., Kapadia, C. H., and Owen, D. B. (1976), Handbook of Statistical Distributions, New York: Marcel Dekker.

Ramberg, J. S., and Tadikamalla, P. R. (1977), "On the Generation of Subsets of Order Statistics," Journal of Statistical Computation and Simulation, forthcoming.

Roberts, A. W., and Varberg, D. E. (1973), Convex Functions, New York: Academic Press, p. 10.

Schmeiser, B. W., and Deutsch, S. J. (1977), "A Versatile Four Parameter Family of Probability Distributions Suitable for Simulation," AIIE Transactions, 9, forthcoming.

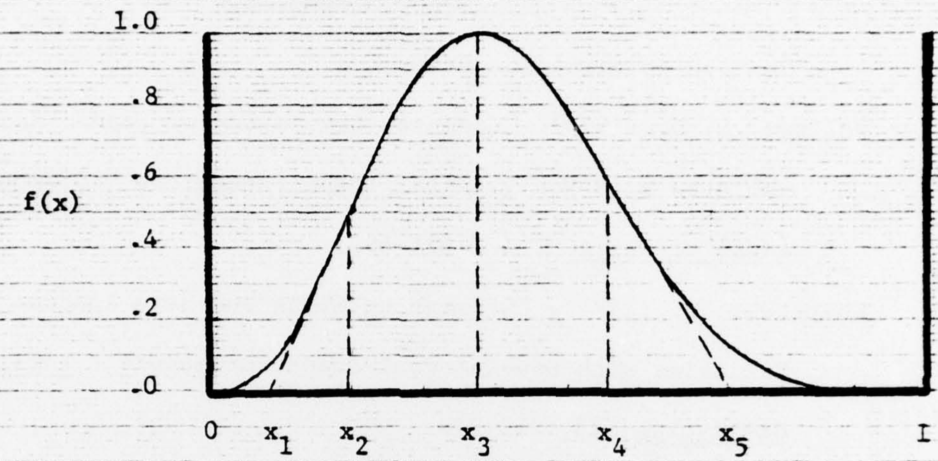
Schucany, R. W. (1972), "Order Statistics in Simulation," Journal of Statistical Computation and Simulation, 1, 281-286.

Tocher, K. D. (1963), The Art of Simulation, London: The English Universities Press Ltd.

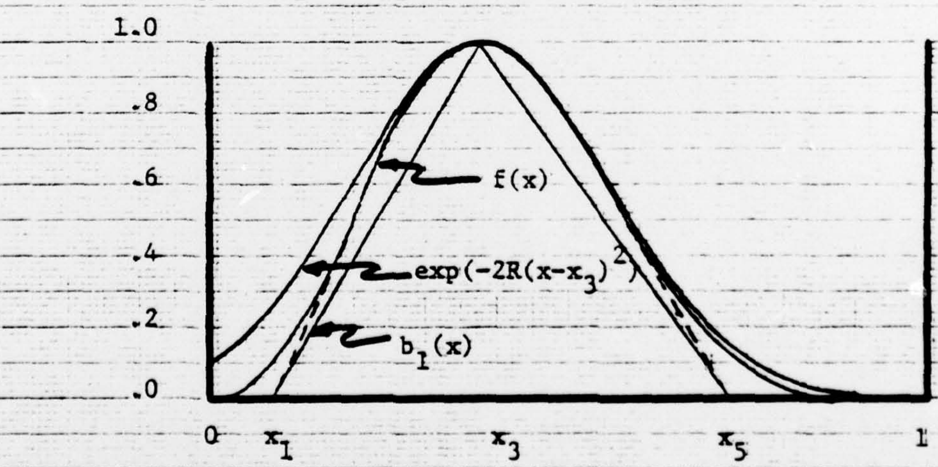
Wallace, N. D. (1974), "Computer Generation of Gamma Random Variates with Non-integral Shape Parameters," Communications of the ACM, 17, 691-695.

Whittaker, J. (1974), "Generating Gamma and Beta Random Variables with Non-integral Shape Parameters," Applied Statistics, 23, 210-214.

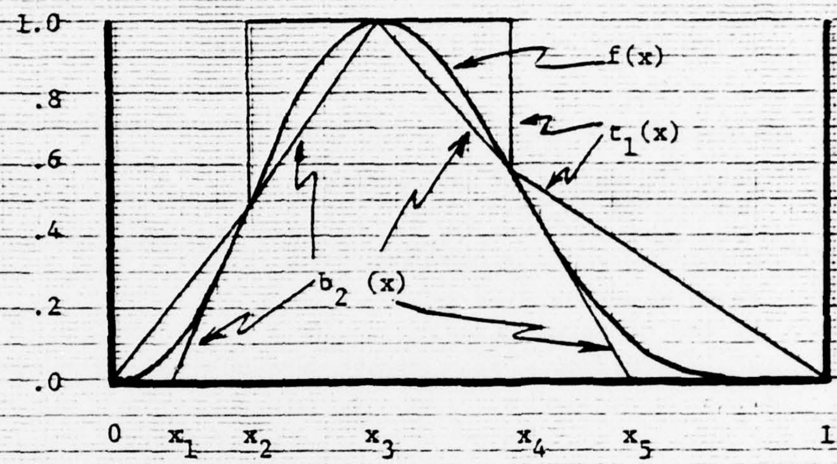
A. Example f(x) with Corresponding x_1, x_2, x_3, x_4 and x_5.



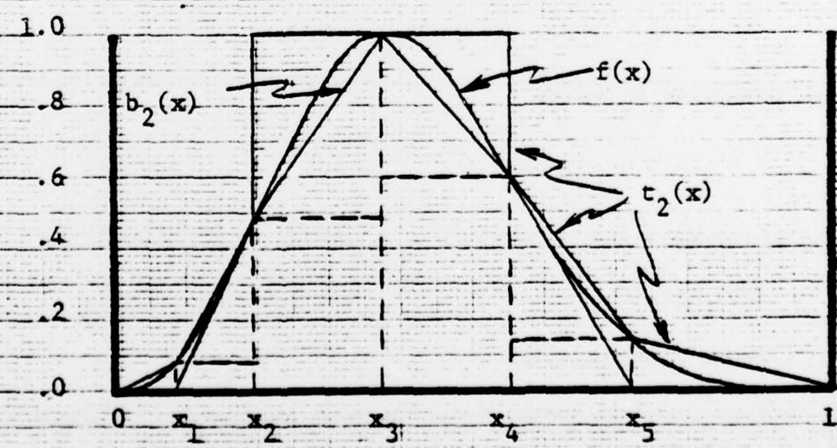
B. The Modified Ahrens and Dieter Method (BNM).



C. The Two Point Method (2P).



D. The Four Point Method (4P).



Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: IEOR 77014
2. GOVT ACCESSION NO.:
3. RECIPIENT'S CATALOG NUMBER:
4. TITLE (and Subtitle): Rejection Methods for Beta Variate Generation
5. TYPE OF REPORT & PERIOD COVERED: Technical Report
6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): Bruce Schmeiser and Mohamed A. Shalaby
8. CONTRACT OR GRANT NUMBER(s): N00014-77-C-0425
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Southern Methodist University, Dallas, TX 75275
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
11. CONTROLLING OFFICE NAME AND ADDRESS: Naval Analysis Program, Office of Naval Research, Arlington, VA 22217
12. REPORT DATE: July 1977
13. NUMBER OF PAGES: 30
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):
15. SECURITY CLASS. (of this report): Unclassified
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:
16. DISTRIBUTION STATEMENT (of this Report): Distribution of this document is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report):
18. SUPPLEMENTARY NOTES:
19. KEY WORDS (Continue on reverse side if necessary and identify by block number):

Simulation
Process Generation
Beta Distribution

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The computer generation of pseudo-random variates from the beta distribution is discussed. Presented are three exact methods applicable for parameter values $p > 1$ and $q > 1$. All three methods use rejection from regions defined by the location of the points of inflexion. The methods differ in the amounts of set-up required, with higher set-up times generally resulting in lower marginal generation times. Generation times are compared to three methods in current use: those of Johnk,

next page

DD FORM 1473, 1 JAN 73 — EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Fishman, and Ahrens and Dieter. Over a wide range of parameter values marginal generation times are reduced about 50%. Substantial set-up time savings and moderate marginal time savings are obtained, compared to Forsythe's approach as recently developed by Atkinson and Pearce.

Computer Codes

For

REJECTION METHODS FOR BETA

VARIATE GENERATION

Bruce Schmeiser

and

Mohamed Shalaby

July 1977

APPENDIX

```
      PROGRAM MAIN (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C
C         THIS IS THE MAIN ROUTINE TO TEST VARIOUS METHODS OF
C         GENERATING BETA RANDOM VARIABLES
C
C         SCHMEISER AND SHALABY       JUNE 1977
C
      DIMENSION NAME(6),METHOD(6)
      DATA NAME/[illegible]/
   10 READ(5,1000) P,Q,N,METHOD
 1000 FORMAT (2F10.0,I10,6I1)
      IF (EOF(5)) 999,20
   20 TMEAN=P/(P+Q)
      TVAR=P*Q/((P+Q)*(P+Q)*(P+Q+1))
      STE=SQRT(TVAR/N)
      WRITE (6,3000) P,Q,N,TMEAN,TVAR,STE
 3000 FORMAT (1H0,.......................................................*/,
     * * P=*,F8.3,*    Q=*,F8.3,*    SAMPLE SIZE=*,I5/,
     **11X,*TIME        MEAN    VARIANCE   STD ERROR*,/,
     * * TRUE             *, 3F10.3)
C
      DO 100 I=1,6
      IF (METHOD(I) .NE. 0)  GO TO 100
      SUMT=0.
      DO 150 J=1,6
      SUM=0.
      SUM2=0.
      TIME=SECOND(A)
C
      DO 300 K=1,N
      GO TO (1,2,3,4,5,6), I
    1 CALL BJK(P,Q,ISEED,X)
      GO TO 200
    2 CALL BRG(P,Q,ISEED,X)
      GO TO 200
    3 CALL BN (P,Q,ISEED,X)
      GO TO 200
    4 CALL HNM (P,Q,ISEED,X)
      GO TO 200
    5 CALL H2P(P,Q,ISEED,X)
      GO TO 200
    6 CALL R0P(P,Q,ISEED,X)
  200 SUM=SUM+X
  300 SUM2=SUM2+X*X
C
      TIME=1000.*(SECOND(A)-TIME)/N
      SUMT=SUMT+TIME
      AVGT=SUMT/J
      XMEAN=SUM/N
      VAR=SUM2/N-XMEAN*XMEAN
  150 WRITE (6,2000)  NAME(I),TIME,AVGT,XMEAN,VAR
 2000 FORMAT(1H ,A6,2F6.2,F8.3,F10.3)
  100 CONTINUE
      GO TO 10
C
  999 STOP
      END
```

```
      SUBROUTINE BJK(P,Q,ISEED,X)
C
C         GENERATION OF ONE PSEUDO-RANDOM VARIATE USING
C         JOHNK'S METHOD AS DISCUSSED IN SECTION 1.2
C         FROM THE BETA DENSITY FUNCTION PROPORTIONAL TO
C            F(X) = (X**(P-1)) * ((1-X)**(Q-1))
C
C         P = FIRST PARAMETER
C         Q = SECOND PARAMETER
C         ISEED = A RANDOM NUMBER SEED
C         X = THE BETA VARIATE
C
      PP=1./P
      QQ=1./Q
  100 Y=RANF(ISEED)**PP
      Z=RANF(ISEED)**QQ
      IF(Y+Z .GT. 1.)  GO TO 100
      X=Y/(Y+Z)
      RETURN
      END
```

```
      SUBROUTINE BRG (P,Q,ISEED,X)
C
C         GENERATION OF ONE PSEUDO-RANDOM VARIATE USING
C         THE RATIO OF GAMMAS METHOD AS DISCUSSED IN SECTION 1.2
C         FROM THE BETA DENSITY FUNCTION PROPORTIONAL TO
C            F(X) = (X**(P-1)) * ((1-X)**(Q-1))
C
C         P = FIRST PARAMETER
C         Q = SECOND PARAMETER
C         ISEED = A RANDOM NUMBER SEED
C         X = THE BETA VARIATE
C
      Y1=0.
      Y2=0.
      Z1=0.
      Z2=0.
      K1=P
      K2=Q
      G1=P-K1
      G2=Q-K2
      IF(G1 .EQ. 0.)  GO TO 10
      Z1=-ALOG(RANF(ISEED))
      CALL BJK (G1,1.-G1,ISEED,Y1)
   10 IF(G2 .EQ. 0.)  GO TO 20
      Z2=-ALOG(RANF(ISEED))
      CALL BJK (G2,1.-G2,ISEED,Y2)
   20 X1=1.
      X2=1.
      IF (K1 .LE. 0)  GO TO 40
      DO 30 I=1,K1
   30 X1=X1*RANF(ISEED)
   40 IF (K2 .LE. 0)  GO TO 60
      DO 50 I=1,K2
   50 X2=X2*RANF(ISEED)
   60 X1=-ALOG(X1)+Y1*Z1
      X2=-ALOG(X2)+Y2*Z2
      X=X1/(X1+X2)
      RETURN
      END
```

```
      SUBROUTINE BN (P,Q,ISEED,X)
C
C         GENERATION OF ONE PSEUDO-RANDOM VARIATE USING
C         THE AHRENS AND DIETER METHOD DISCUSSED IN SECTION 2.1
C         FROM THE BETA DENSITY FUNCTION PROPORTIONAL TO
C            F(X) = (X**(P-1)) * ((1-X)**(Q-1))
C
C         P = FIRST PARAMETER    (GREATER THAN ONE)
C         Q = SECOND PARAMETER   (GREATER THAN ONE)
C         ISEED = A RANDOM NUMBER SEED
C         X = THE BETA VARIATE
C
      DATA PSAVE, QSAVE /-1.,-1./
C
C         CHECK WHETHER SET-UP IS NECESSARY
C
      IF (P .EQ. PSAVE .AND. Q .EQ. QSAVE)  GO TO 100
C
C         PERFORM SET-UP
C
      PSAVE=P
      QSAVE=Q
      PP=P-1
      QQ=Q-1
      RR=PP+QQ
      RL=RR*ALOG(RR)
      X3=PP/RR
      SIG=.5/SQRT(RR)
C
C         REJECTION PROCEDURE BEGINS HERE
C
  100 CALL NORMAL(ISEED,S)
      X=SIG*S+X3
      IF(X .LT. 0. .OR. X .GT. 1.)  GO TO 100
      U=RANF(ISEED)
      IF(ALOG(U) .LT. PP*ALOG(X/PP)+QQ*ALOG((1.-X)/QQ)+RL+S*S*.5) RETURN
      GO TO 100
      END
```

```
      SUBROUTINE BNM(P,Q,ISEED,X)
C
C        GENERATION OF ONE PSEUDO-RANDOM VARIATE USING
C        THE MODIFIED AHRENS AND DIETER METHOD DESCRIBED IN SECTION 2.1
C        FROM THE BETA DENSITY FUNCTION PROPORTIONAL TO
C           F(X) = (X**(P-1)) * ((1-X)**(Q-1))
C
C        P = FIRST PARAMETER    (GREATER THAN ONE)
C        Q = SECOND PARAMETER   (GREATER THAN ONE)
C        ISEED = A RANDOM NUMBER SEED
C        X = THE BETA VARIATE
C
      DATA PSAVE, QSAVE /-1.,-1./
C
C        CHECK WHETHER SET-UP IS NECESSARY
C
      IF (P .EQ. PSAVE .AND. Q .EQ. QSAVE)  GO TO 100
C
C        PERFORM SET-UP
C
      PSAVE=P
      QSAVE=Q
      PP=P-1
      QQ=Q-1
      RR=PP+QQ
      INEW=1
      X3=PP/RR
      SIG=.5/SQRT(RR)
      X1=0.
      X5=1.
      IF(RR .LE. 1.)  GO TO 100
      D = SQRT(PP*QQ/(RR-1.))/RR
      IF(D .GE. X3)  GO TO 60
      X2=X3-D
      X1=X2-((X2*(1.-X2))/(PP-RR*X2))
   60 IF(D .GE. (1.-X3))  GO TO 100
      X4=X3+D
      X5=X4-((X4*(1.-X4))/(PP-RR*X4))
C
C        REJECTION PROCEDURE BEGINS HERE
C
  100 CALL NORMAL(ISEED,S)
      X=SIG*S+X3
      IF(X .LT. 0. .OR. X .GT. 1.)  GO TO 100
      U=RANF(ISEED)
      IF(X .LT. X1 .OR. X .GT. X5)  GO TO 160
      IF(X .GT. X3)  GO TO 150
      IF (U*EXP(-S*S*.5) .LE. (X-X1)/(X3-X1))  RETURN
      GO TO 160
  150 IF(U*EXP(-S*S*.5) .LE. (X5-X)/(X5-X3))  RETURN
  160 IF(INEW .EQ. 0)  GO TO 170
      INEW=0
      RL=RR*ALOG(RR)
  170 IF(ALOG(U) .LT. PP*ALOG(X/PP)+QQ*ALOG((1.-X)/QQ)+RL+S*S*.5) RETURN
      GO TO 100
      END
```

```
      SUBROUTINE NORMAL(ISEED,X)
C
C        GENERATION OF ONE NORMAL(0,1) VARIATE USING
C        THE ALGORITHM GIVEN BY KINDERMAN AND RAMAGE
C        IN THE JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION 12/76
C
C        CODED BY PETER BONNER AND MODIFIED BY BRUCE SCHMEISER
C        MARCH 1976 AND JUNE 1976 RESPECTIVELY
C
      DATA TAIL/2.216035867166471/
      UU=RANF(ISEED)
      IF(UU.GE..884070402298758) GO TO 2
C
C        RETURN TRIANGULAR VARIATE 88 PERCENT OF THE TIME
C
      V=RANF(ISEED)
      X=TAIL*(1.131131635444180*UU+V-1.0)
      RETURN
    2 IF(UU.LT..973310954173898) GO TO 4
C
C        TAIL COMPUTATION
C
    3 V=RANF(ISEED)
      W=RANF(ISEED)
      T1=TAIL*TAIL/2.0
      T=T1-ALOG(W)
      IF(V*V*T.GT.T1) GO TO 3
      X=SQRT(2.0*T)
      IF(UU.GE..986655477086949) X=-X
      RETURN
    4 IF(UU.LT..958720824790463) GO TO 6
C
C        FIRST NEARLY LINEAR DENSITY
C
    5 V=RANF(ISEED)
      W=RANF(ISEED)
      Z=V-W
C     LET V= MAX(V,W) AND LET W=MIN(V,W)
      IF(V.GT.W) GO TO 100
      TEMP=V
      V=W
      W=TEMP
  100 T=TAIL-.630834801921960*W
      IF(V.LE..755591531667601) GO TO 9
      DIFF=EXP(-T*T*.5)/2.506628274631000-.180025191068563*
     *      (2.216035867166471-ABS(T))
      IF(ABS(Z)*.034240503750111.LE.DIFF)  GO TO 9
      GO TO 5
    6 IF(UU.LT..911312780288703) GO TO 8
C
C        SECOND NEARLY LINEAR DENSITY
C
    7 V=RANF(ISEED)
      W=RANF(ISEED)
      Z=V-W
C     LET V= MAX(V,W) AND LET W=MIN(V,W)
      IF(V.GT.W) GO TO 101
      TEMP=V
      V=W
      W=TEMP
  101 T=.479727404222441+1.105473661022070*W
      IF(V.LE..872834976671790) GO TO 9
      DIFF=EXP(-T*T*.5)/2.506628274631000-.180025191068563*
     *      (2.216035867166471-ABS(T))
      IF(ABS(Z)*.049264496373128.LE.DIFF)  GO TO 9
      GO TO 7
C
C        THIRD NEARLY LINEAR DENSITY
C
    8 V=RANF(ISEED)
      W=RANF(ISEED)
      Z=V-W
C     LET V= MAX(V,W) AND LET W=MIN(V,W)
      IF(V.GT.W) GO TO 102
      TEMP=V
      V=W
      W=TEMP
  102 T=.479727404222441-.595507138015940*W
      IF(V.LE..805577924423817) GO TO 9
      DIFF=EXP(-T*T*.5)/2.506628274631000-.180025191068563*
     *      (2.216035867166471-ABS(T))
      IF(ABS(Z)*.053377549506886.LE.DIFF)  GO TO 9
      GO TO 8
    9 X=T
      IF(Z.GE.0.0) X=-X
      RETURN
      END
```

```
      SUBROUTINE B2P (P,Q,ISEED,X)
C
C        GENERATION OF ONE PSEUDO-RANDOM VARIATE USING
C        THE TWO POINT TECHNIQUE DESCRIBED IN SECTION 2.2
C        FROM THE BETA DENSITY FUNCTION PROPORTIONAL TO
C           F(X) = (X**(P-1)) * ((1-X)**(Q-1))
C
C        P = FIRST PARAMETER    (GREATER THAN ONE)
C        Q = SECOND PARAMETER   (GREATER THAN ONE)
C        ISEED = A RANDOM NUMBER SEED
C        X = THE BETA VARIATE
C
      DATA PSAVE, QSAVE /-1.,-1./
C
C        CHECK WHETHER SET-UP IS NECESSARY
C
      IF (P .EQ. PSAVE .AND. Q .EQ. QSAVE)  GO TO 100
C
C        PERFORM SET-UP
C
      PSAVE=P
      QSAVE=Q
      PP=P-1.
      QQ=Q-1.
      RR=PP+QQ
      TRR=-RR-RR
      RL=RR*ALOG(RR)
      X1=0.
      X2=0.
      X3=PP/RR
      X4=1.
      X5=1.
      F2=0.
      F4=0.
C
      IF (RR .LE. 1.)  GO TO 80
      D=SQRT(PP*QQ/(RR-1.))/RR
      IF(D .GE. X3)  GO TO 60
      X2=X3-D
      X1=X2-((X2*(1.-X2))/(PP-RR*X2))
      F2=EXP(PP*ALOG(X2/PP)+QQ*ALOG((1.-X2)/QQ)+RL)
   60 IF (D .GE. (1.-X3))  GO TO 80
      X4=X3+D
      X5=X4-((X4*(1.-X4))/(PP-RR*X4))
      F4=EXP(PP*ALOG(X4/PP)+QQ*ALOG((1.-X4)/QQ)+RL)
C
C        CALCULATE AREAS OF THE FOUR REGIONS
C
   80 P1=X3-X2
      P2=(X4-X3)+P1
      P3=P2
      IF(F2 .GT. 0.)  P3=F2*X2*.5+P2
      P4=P3
      IF(F4 .GT. 0.)  P4=F4*(1.-X4)*.5+P3
C
C        REJECTION PROCEDURE BEGINS HERE
C
  100 U=RANF(ISEED) * P4
      W=RANF(ISEED)
C
C        LEFT RECTANGULAR REGION
C
      IF(U.GT.P1)GO TO 200
      X=X2+W*(X3-X2)
      V=U/P1
      IF(V .LE. F2+(1.-F2)*W)  RETURN
      GO TO 500
C
C        RIGHT RECTANGULAR REGION
C
  200 IF(U.GT.P2)GO TO 300
      X=X3+W*(X4-X3)
      V=(U-P1)/(P2-P1)
      IF (V .LE. 1.-(1.-F4)*W)  RETURN
      GO TO 500
C
C        LEFT TRIANGULAR REGION
C
  300 W2=RANF(ISEED)
      IF(W2.GT.W)W=W2
      IF(U.GT.P3) GO TO 400
      X=W*X2
      V=((U-P2)/(P3-P2))*W*F2
      IF (X .LE. X1)  GO TO 500
      IF(V.LE.F2*(X-X1)/(X2-X1))RETURN
      GO TO 500
C
C        RIGHT TRIANGULAR REGION
C
  400 X=1.-W*(1.-X4)
      V=((U-P3)/(P4-P3))*(1.-X)*F4/(1.-X4)
      IF (X .GT. X5)  GO TO 500
      IF(V.LE.F4*(X5-X)/(X5-X4))RETURN
C
C        CHECK EASY REJECTION VIA COMPARISON WITH THE NORMAL DENSITY
C
  500 ALV=ALOG(V)
      IF (ALV .GT. (X-X3)*(X-X3)*TRR)  GO TO 100
C
C        PERFORM REJECTION BASED ON F(X)
C
      IF(ALV.LE.PP*ALOG(X/PP)+QQ*ALOG((1.-X)/QQ)+RL)  RETURN
      GO TO 100
      END
```

```
      SUBROUTINE B4P (P,Q,ISEED,X)
C
C        GENERATION OF ONE PSEUDO-RANDOM VARIATE USING
C        THE FOUR POINT METHOD DESCRIBED IN SECTION 2.3
C        FROM THE BETA DENSITY FUNCTION PROPORTIONAL TO
C           F(X) = (X**(P-1)) * ((1-X)**(Q-1))
C
C        P = FIRST PARAMETER    (GREATER THAN ONE)
C        Q = SECOND PARAMETER   (GREATER THAN ONE)
C        ISEED = A RANDOM NUMBER SEED
C        X = THE BETA VARIATE
C
      DATA PSAVE, QSAVE /-1.,-1./
C
C        CHECK WHETHER SET-UP IS NECESSARY
C
      IF (P .EQ. PSAVE .AND. Q .EQ. QSAVE)  GO TO 100
C
C        PERFORM SET-UP
C
      PSAVE=P
      QSAVE=Q
      PP=P-1.
      QQ=Q-1.
      RR=PP+QQ
      TRR=-RR-RR
      RL=RR*ALOG(RR)
      X1=0.
      X2=0.
      X3=PP/RR
      X4=1.
      X5=1.
      F1=0.
      F2=0.
      F4=0.
      F5=0.
C
      IF (RR .LE. 1.)  GO TO 80
      D=SQRT(PP*QQ/(RR-1.))/RR
      IF(D .GE. X3)  GO TO 60
      X2=X3-D
      X1=X2-((X2*(1.-X2))/(PP-RR*X2))
      F1=EXP(PP*ALOG(X1/PP)+QQ*ALOG((1.-X1)/QQ)+RL)
      F2=EXP(PP*ALOG(X2/PP)+QQ*ALOG((1.-X2)/QQ)+RL)
   60 IF(D .GE. (1.-X3))  GO TO 80
      X4=X3+D
      X5=X4-((X4*(1.-X4))/(PP-RR*X4))
      F4=EXP(PP*ALOG(X4/PP)+QQ*ALOG((1.-X4)/QQ)+RL)
      F5=EXP(PP*ALOG(X5/PP)+QQ*ALOG((1.-X5)/QQ)+RL)
C
C        CALCULATE AREAS OF THE TEN REGIONS
C
   80 P1=0.
      IF(F2 .GT. 0.)  P1=F2*(X3-X2)
      P2=P1
      IF(F4 .GT. 0.)  P2=F4*(X4-X3)+P1
      P3=P2
      IF(F1 .GT. 0.)  P3=F1*(X2-X1)+P2
      P4=P3
      IF(F5 .GT. 0.)  P4=F5*(X5-X4)+P3
      P5=(1.-F2)*(X3-X2)+P4
      P6=(1.-F4)*(X4-X3)+P5
      P7=P6
      IF(F2 .GT. F1)  P7=(F2-F1)*(X2-X1)*.5+P6
      P8=P7
      IF(F4 .GT. F5)  P8=(F4-F5)*(X5-X4)*.5+P7
      P9=P8
      IF(F1 .GT. 0.)  P9=F1*X1*.5+P8
      P10=P9
      IF(F5 .GT. 0.)  P10=F5*(1.-X5)*.5+P9
C
C        REJECTION PROCEDURE BEGINS HERE
C
  100 U=RANF(ISEED) * P10
C
C        THE FOUR REGIONS WITH ZERO PROBABILITY OF REJECTION
C
      IF(U.GT.P4)GO TO 500
      IF(U.GT.P1)GO TO 200
      X=X2+U/F2
      RETURN
  200 IF(U.GT.P2)GO TO 300
      X=X3+(U-P1)/F4
      RETURN
  300 IF(U.GT.P3)GO TO 400
      X=X1+(U-P2)/F1
      RETURN
  400 X=X4+(U-P3)/F5
      RETURN
C
C        THE TWO REGIONS USING RECTANGULAR REJECTION
C
  500 W=RANF(ISEED)
      IF(U.GT.P5)GO TO 600
      X=X2+(X3-X2)*W
      IF((U-P4)/(P5-P4) .LE. W)  RETURN
      V=F2+(U-P4)/(X3-X2)
      GO TO 1300
  600 IF(U.GT.P6)GO TO 700
      X=X3+(X4-X3)*W
      IF((P6-U)/(P6-P5) .GE. W)  RETURN
      V=F4+(U-P5)/(X4-X3)
      GO TO 1300
C
C        THE FOUR TRIANGULAR REGIONS
C
  700 W2=RANF(ISEED)
      IF(W2.GT.W)W=W2
      IF(U.GT.P7)GO TO 800
      X=X1+(X2-X1)*W
      V=F1+2.*W*(U-P6)/(X2-X1)
      IF (V .LE. F2*W)  RETURN
      GO TO 1300
  800 IF(U.GT.P8)GO TO 900
      X=X5-W*(X5-X4)
      V=F5+2.*W*(U-P7)/(X5-X4)
      IF (V .LE. F4*W)  RETURN
      GO TO 1300
  900 IF(U.GT.P9)GO TO 1000
      X=W*X1
      V=2.*W*(U-P8)/X1
      GO TO 1300
 1000 X=1.-W*(1.-X5)
      V=2.*W*(U-P9)/(1.-X5)
C
C        CHECK EASY REJECTION VIA COMPARISON WITH NORMAL DENSITY
C
 1300 ALV=ALOG(V)
      IF (ALV .GT. (X-X3)*(X-X3)*TRR)  GO TO 100
C
C        PERFORM THE STANDARD REJECTION
C
      IF(ALV.LE.PP*ALOG(X/PP)+QQ*ALOG((1.-X)/QQ)+RL)RETURN
      GO TO 100
      END
```