GENERATING DIRICHLET RANDOM VECTORS USING A REJECTION PROPERTY

ȘTEFAN V. ȘTEFĂNESCU

In this paper a generalization of Jöhnk and Loukas results reported in [4] and [5] is given and a rejection algorithm for computer generation of Dirichlet random vectors is proposed. Comparisons with other suitable algorithms are also discussed.

1. INTRODUCTION

Let \( a_j, 1 \leq j \leq n + 1 \), be \( n + 1 \) positive real numbers.

The random vector \((X_1, X_2, \ldots, X_n)\) has a Dirichlet \( D(n; a_1, a_2, \ldots, a_n, a_{n+1})\) distribution if its probability density function (p.d.f.) is given by (cf. Wilks [14]):

\[
f(x_1, x_2, \ldots, x_n) = Kx_1^{a_1-1}x_2^{a_2-1}\cdots x_n^{a_n-1}(1 - x_1 - x_2 - \cdots - x_n)^{a_{n+1}-1}
\]

for every \((x_1, x_2, \ldots, x_n) \in S_n\), where

\[
S_n = \{(x_1, x_2, \ldots, x_n) \mid x_i > 0, 1 \leq i \leq n; x_1 + x_2 + \cdots + x_n < 1\}
\]

\[
K = \Gamma(a_1 + a_2 + \cdots + a_n + a_{n+1})/(\Gamma(a_1) \Gamma(a_2) \cdots \Gamma(a_n) \Gamma(a_{n+1}))
\]

with

\[
\Gamma(a) = \int_0^\infty t^{a-1} \exp(-t) \, dt \quad a > 0.
\]

To the best of our knowledge the computer generation of the Dirichlet distribution has not been enough studied in the literature (cf. [3], [6]–[10]).

In this context we can mention a lot of procedures:
- The gamma method based on the relation between Dirichlet and gamma distributions (Wilks [14], p. 179);
- Different rejection techniques (Ștefănescu [12]) or the use of the classical inverse method (considering the marginal distributions, Ștefănescu [12]);
- A one-to-one transform between \( S_n \) and \((0,1)^n\) (Ștefănescu [11]);
- The use of a transformation of a uniformly random vector over a bounded domain \( D \subset \mathbb{R}^{n+1} \) (Văduva [13]).

The performance of these algorithms were compared in [12]. We can conclude
that the gamma method is the fastest; a similar result (for a particular \( n, n = 2 \))
was also obtained by Loukas [5].

In what follows, using a rejection property suggested by Loukas [5], we shall
give a new algorithm for generating Dirichlet \( D(n; a_1, a_2, \ldots, a_n, a_{n+1}) \) random
vectors.

2. THE MAIN RESULT

**Theorem 1.** Let \( U_1, U_2, \ldots, U_n, U_{n+1} \) be \( n + 1 \) independent random variables
uniformly distributed over the interval \((0, 1)\) and

\[
X_i = U_i^{1/a_i}(U_1^{1/a_1} + U_2^{1/a_2} + \ldots + U_n^{1/a_n} + U_{n+1}^{1/a_{n+1}})
\]

\(1 \leq i \leq n\). Then the random vector \((X_1, X_2, \ldots, X_n)\) conditioned on

\[
U_1^{1/a_1} + U_2^{1/a_2} + \ldots + U_n^{1/a_n} + U_{n+1}^{1/a_{n+1}} < 1
\]

has a Dirichlet distribution \( D(n; a_1, a_2, \ldots, a_n, a_{n+1}) \).

**Proof.** Let introduce a one-to-one transformation \( T: (0, 1)^{n+1} \rightarrow D, D \subset \mathbb{R}^{n+1}, \)

\[
T(y_1, \ldots, y_n, y_{n+1}) = (x_1, \ldots, x_n, x_{n+1})
\]

defined by the relations

\[
T^{-1}:
\begin{align*}
(x_i &= y_i/(y_1 + y_2 + \ldots + y_n + y_{n+1}); \quad 1 \leq i \leq n \\
\quad x_{n+1} &= y_1 + y_2 + \ldots + y_n
\end{align*}
\]

The inverse \( T^{-1} \) of the transform \( T, T^{-1} : D \rightarrow (0, 1)^{n+1}, \)

\[
T^{-1}(x_1, x_2, \ldots, x_n, x_{n+1}) = (y_1, y_2, \ldots, y_n, y_{n+1})
\]

is given by

\[
T^{-1}:
\begin{align*}
(y_i &= x_{n+1}x_i/(x_1 + x_2 + \ldots + x_n); \quad 1 \leq i \leq n \\
i_{n+1} &= x_{n+1}(1 - x_1 - x_2 - \ldots - x_n)/(x_1 + x_2 + \ldots + x_n)
\end{align*}
\]

Let \( J \) be the Jacobian of \( T^{-1}, \)

\[
J = D(y_1, y_2, \ldots, y_n, y_{n+1})/D(x_1, x_2, \ldots, x_n, x_{n+1}) = \det (A)
\]

where \( \det (A) \) is the determinant of the matrix \( A = (a_{ij})_{1 \leq i, j \leq n+1}, \) with \( a_{ij} = \)

\[
\partial y_i/\partial x_j, \quad 1 \leq i, j \leq n + 1.
\]

Adding to the row 1 of the matrix \( A \) all remaining rows and using afterwards
the last relation (4) we get

\[
J = (-1)^{n+2} \cdot \det (B)
\]

where \( B = (b_{ij})_{1 \leq i, j \leq n}, \) with \( b_{ij} = \partial y_{i+1}/\partial x_j, \quad 1 \leq i, j \leq n. \)

From (5) we obtain the derivatives \( \partial y_{i+1}/\partial x_j, \quad 1 \leq i, j \leq n, \) and therefore

\[
\det (B) = (-x_{n+1}^n/(x_1 + x_2 + \ldots + x_n)^2)^n \det (C)
\]

where \( C = (c_{ij})_{1 \leq i, j \leq n}, \) with

\[
c_{nj} = 1; \quad 1 \leq j \leq n \\
c_{ij} = x_{i+1}; \quad 1 \leq i \leq n - 1, \quad 1 \leq j \leq n, \quad j \neq i + 1 \\
c_{i,i+1} = x_{i+1} - (x_1 + x_2 + \ldots + x_n); \quad 1 \leq i \leq n - 1
\]
Subtracting column 1 of the matrix $C$ from the other columns of $C$ and then moving the last row on the first place we obtain a lower triangular matrix; hence

$$\det (C) = (x_1 + x_2 + \ldots + x_n)^{n-1}$$

From (6)–(9) we conclude that

$$J = x_1^n(x_1 + x_2 + \ldots + x_n)^{n-1}$$

Let $U_1, U_2, \ldots, U_n, U_{n+1}$ be $n + 1$ independent random variables uniformly distributed over the interval (0, 1). If $Y_i = U_i^{1/a_i}, 1 \leq i \leq n + 1$, then the p.d.f. $g_1$ of the random vector $(Y_1, Y_2, \ldots, Y_n, Y_{n+1})$ takes the form

$$g_1(y_1, y_2, \ldots, y_n, y_{n+1}) = \prod_{j=1}^{n+1} a_j y_j^{a_j-1}; \quad 0 < y_j < 1, \quad 1 \leq j \leq n + 1$$

Denoting

$$D_1 = \{(y_1, \ldots, y_n, y_{n+1}) \mid 0 < y_j < 1, 1 \leq j \leq n + 1, y_1 + \ldots + y_{n+1} < 1\}$$

and using the following identity $(a > 0, b > 0, c > 0)$

$$\int_0^t b r^{-1}(a - t)^c \, dt = a^{b+c} \Gamma(b + 1) \Gamma(c + 1)/\Gamma(b + c + 1)$$

we obtain the value $K_1$, that is

$$K_1 = \int_{D_1} g_1(y_1, y_2, \ldots, y_n, y_{n+1}) \, dy_1 \, dy_2 \ldots \, dy_n \, dy_{n+1} =$$

$$= \int_0^1 dy_1 \int_0^{y_1} dy_2 \ldots \int_0^{y_{n-1}} \cdot \cdot \cdot \int_0^{y_{n-2}} dy_n \cdot \cdot \cdot \int_0^{y_{n-2}} g_1(y_1, y_2, \ldots, y_n, y_{n+1}) \, dy_{n+1} =$$

$$= \left( \prod_{j=1}^{n+1} \Gamma(a_j + 1) \right) / \Gamma(a_1 + a_2 + \ldots + a_n + a_{n+1} + 1)$$

Therefore the p.d.f. $g_2$ of the random vector $(Y_1, Y_2, \ldots, Y_n, Y_{n+1})$ conditioned on the inequality

$$Y_1 + Y_2 + \ldots + Y_n + Y_{n+1} < 1$$

has the form

$$g_2(y_1, y_2, \ldots, y_n, y_{n+1}) = g_1(y_1, y_2, \ldots, y_n, y_{n+1}) / K_1$$

for every $(y_1, y_2, \ldots, y_n, y_{n+1}) \in D_1$.

Let $(X_1, X_2, \ldots, X_n, X_{n+1})$ be the random vector obtained from the vector $(Y_1, Y_2, \ldots, Y_n, Y_{n+1})$ by applying the transform $T$ (formula (4)). Then the p.d.f. $f_2$ of $(X_1, X_2, \ldots, X_n)$ is given by

$$f_2(x_1, x_2, \ldots, x_n, x_{n+1}) = g_2(y_1, y_2, \ldots, y_n, y_{n+1}) \bigg| J \bigg|$$

From the relations (5), (10), (11), (13), (15), (16) and using the equality $\Gamma(a + 1) = a \Gamma(a)$ we finally obtain

$$f_2(x_1, \ldots, x_n, x_{n+1}) = f(x_1, x_2, \ldots, x_n) f_3(x_1, x_2, \ldots, x_n, x_{n+1})$$

where the function $f$ is given by (1) and $f_3$ takes the form

$$f_3(x_1, \ldots, x_{n+1}) = (a_1 + \ldots + a_{n+1}) x_1^{a_1} \ldots x_n^{a_n} x_{n+1}^{a_{n+1} - 1} (x_1 + \ldots + x_n)^{a_1 + \ldots + a_{n+1}}$$
From (5) we get
\[ y_1 + y_2 + \ldots + y_n + y_{n+1} = x_{n+1}/(x_1 + x_2 + \ldots + x_n) \]
and then the inequality (3) is equivalent to \( y_1 + \ldots + y_n + y_{n+1} < 1 \), that is
\[ 0 < x_{n+1} < x_1 + x_2 + \ldots + x_n \]

Hence the p.d.f. of \((X_1, X_2, \ldots, X_n)\) is given by
\[
f_1(x_1, x_2, \ldots, x_n) = \int_0^{x_1 + x_2 + \ldots + x_n} f_2(x_1, x_2, \ldots, x_n, x_{n+1}) \, dx_{n+1} =
= f(x_1, x_2, \ldots, x_n) \int_0^{x_1 + x_2 + \ldots + x_n} f_3(x_1, x_2, \ldots, x_n, x_{n+1}) \, dx_{n+1} =
= f(x_1, x_2, \ldots, x_n)
\]
which was required to prove.

**Remark 1.** The Beta\((a, b)\) distribution is a unidimensional Dirichlet distribution \((\text{Beta}(a, b) \equiv D(1; a, b))\).

**Remark 2.** If \( U \) is uniformly distributed over \((0, 1)\) then \( Y = U^{1/a} \) has a Beta\((a, 1)\) distribution \((Y \sim \text{Beta}(a, 1))\).

Considering \( n = 1 \), respectively \( n = 2 \), from Theorem 1 it results

**Corollary 1** (Jöhnk [4]). If \( Y_i \sim \text{Beta}(a_i, 1), i = 1, 2, \) are independent random variables so that \( Y_1 + Y_2 < 1 \) then \( X = Y_1/(Y_1 + Y_2) \sim \text{Beta}(a, b)\).

**Corollary 2** (Loukas [5]). If \( Y_1 \sim \text{Beta}(a_1, 1), Y_2 \sim \text{Beta}(a_2, 1), Y_3 \sim \text{Beta}(a_3, 1) \) are independent random variables and \( Y_1 + Y_2 + Y_3 < 1 \) then \((X_1, X_2) =
= (Y_1/(Y_1 + Y_2 + Y_3), Y_2/(Y_1 + Y_2 + Y_3)) \sim D(2; a_1, a_2, a_3)\).

### 3. THE REJECTION PROCEDURE

From Theorem 1 we get the following algorithm for generating a random vector \((X_1, X_2, \ldots, X_n)\) having a Dirichlet \( D(n; a_1, a_2, \ldots, a_n, a_{n+1}) \) distribution.

**Algorithm AGDR** (Algorithm for Generating a Dirichlet distribution using a Rejection property).

**STEP 0.** Inputs: \( a_1, a_2, \ldots, a_n, a_{n+1}, n \) \((a_j > 0, 1 \leq j \leq n + 1)\).

**STEP 1.** Generate \( n + 1 \) independent random variables \( U_1, U_2, \ldots, U_{n+1} \) uniformly distributed over the interval \((0, 1)\)
\[ X_j \leftarrow U_j^{1/a_j}, \quad 1 \leq j \leq n. \]

**STEP 2.** \( S \leftarrow X_1 + X_2 + \ldots + X_{n-1} + X_n + U_{n+1}^{1/a_{n+1}} \).

**STEP 3.** If \( S \geq 1 \) then go to Step 1.

**STEP 4.** \( X_j \leftarrow X_j/S, \quad 1 \leq j \leq n. \)

**STEP 5.** Print \((X_1, X_2, \ldots, X_{n-1}, X_n)\). STOP.
The "acceptance probability" \( P_{ac}(n; a_1, a_2, \ldots, a_n, a_{n+1}) \) (Devroye [3]) in Step 3 of the algorithm AGDR is given by

\[
P_{ac}(n; a_1, \ldots, a_n, a_{n+1}) = P(U_1^{1/a_1} + U_2^{1/a_2} + \ldots + U_n^{1/a_n} + U_{n+1}^{1/a_{n+1}} < 1) = \int_{D_2} dU_1 dU_2 \ldots dU_n dU_{n+1}
\]

where

\[
D_2 = \{(u_1, \ldots, u_{n+1}) | 0 < u_j < 1, 1 \leq j \leq n + 1; u_1^{1/a_1} + \ldots + u_{n+1}^{1/a_{n+1}} < 1\}.
\]

Considering the new variables \( y_j = u_j^{1/a_j}, 1 \leq j \leq n + 1, \) we have

\[
P_{ac}(n; a_1, \ldots, a_{n+1}) = \int_{D_1} \int_{D_1} a_1 a_2 \ldots a_{n+1} y_1 y_1^{-1} \ldots y_{n+1} y_{n+1}^{-1} dy_1 \ldots dy_{n+1}
\]

where the domain \( D_1 \) is defined by (12). Using (11) and (13) we find

\[
P_{ac}(n; a_1, \ldots, a_{n+1}) = K_1 = \Gamma(1 + a_1) \ldots \Gamma(1 + a_n) \Gamma(1 + a_{n+1})
\]

\[
/ \Gamma(1 + a_1 + \ldots + a_n + a_{n+1}).
\]

4. PERFORMANCES

**Proposition 1.** If \( 0 < a_j \leq b_j, 1 \leq j \leq n + 1, \) then

\[
P_{ac}(n; a_1, a_2, \ldots, a_n, a_{n+1}) \geq P_{ac}(n; b_1, b_2, \ldots, b_n, b_{n+1})
\]

**Proof.** If \( 0 < a \leq b, c > 0, 0 < t < 1 \) then \( t^{-1} \geq e^{-1} \) and hence

\[
\int_0^1 t^{-1}(1 - t)^{c-1} dt \geq \int_0^1 e^{-1}(1 - t)^{c-1} dt
\]

From (26) it results

\[
\Gamma(a) \Gamma(a + c) \geq \Gamma(b) \Gamma(b + c), \quad 0 < a \leq b, \quad c > 0
\]

Applying the inequality (27) it obtains successively

\[
P_{ac}(n; a_1, a_2, \ldots, a_n, a_{n+1}) \geq P_{ac}(n; b_1, a_2, \ldots, a_n, a_{n+1}) \geq \ldots
\]

\[
\ldots \geq P_{ac}(n; b_1, b_2, \ldots, b_n, a_{n+1}) \geq P_{ac}(n; b_1, b_2, \ldots, b_n, b_{n+1})
\]

the proposition being proved. \( \square \)

Tables 1 and 4 contain the values of the "acceptance probability" \( P_{ac}(n; a, a, \ldots, a) \) obtained by using formula (24) (Table 1) or applying a Monte Carlo procedure (Table 4; 10 000 simulation steps); it considers \( a \in \{1.0; 0.5; 0.25; 0.2; 0.125; 0.1; 0.05\}, \) \( n = 1, 2. \)

The values from Table 1 and Table 2 are very close.

**Table 1.** The "acceptance probability" \( P_{ac}(n; a, a, \ldots, a, a) \) given by (24).

<table>
<thead>
<tr>
<th></th>
<th>( a = 1.0 )</th>
<th>( a = 0.5 )</th>
<th>( a = 0.25 )</th>
<th>( a = 0.2 )</th>
<th>( a = 0.125 )</th>
<th>( a = 0.1 )</th>
<th>( a = 0.05 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n = 1 )</td>
<td>0.5000</td>
<td>0.7854</td>
<td>0.9270</td>
<td>0.9502</td>
<td>0.9785</td>
<td>0.9857</td>
<td>0.9962</td>
</tr>
<tr>
<td>( n = 2 )</td>
<td>0.1667</td>
<td>0.5236</td>
<td>0.8103</td>
<td>0.8663</td>
<td>0.9396</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

471
Remark 3. Table 2 and Table 3 give the “acceptance probability” $P_{ac}(n; a, ..., a)$ (formula (24)) for different values of $n$. Using Proposition 1 and Tables 2, 3 it can be found a good approximation of the “acceptance probability” $P_{ac}(n; a_1, a_2, ..., a_n, a_{n+1})$ when we haven’t $a_1 = a_2 = ... = a_n = a_{n+1}$.

**Table 2.** The “acceptance probability” $P_{ac}(n; a, a, ..., a, a)$ obtained from (24).

<table>
<thead>
<tr>
<th>$a$</th>
<th>$n = 1$</th>
<th>$n = 2$</th>
<th>$n = 3$</th>
<th>$n = 4$</th>
<th>$n = 5$</th>
<th>$n = 6$</th>
<th>$n = 7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>0.5000</td>
<td>0.1667</td>
<td>0.0417</td>
<td>0.0083</td>
<td>0.0014</td>
<td>0.0002</td>
<td>0.0000</td>
</tr>
<tr>
<td>0.9</td>
<td>0.5517</td>
<td>0.2133</td>
<td>0.0639</td>
<td>0.0157</td>
<td>0.0033</td>
<td>0.0006</td>
<td>0.0001</td>
</tr>
<tr>
<td>0.8</td>
<td>0.6068</td>
<td>0.2710</td>
<td>0.0970</td>
<td>0.0292</td>
<td>0.0076</td>
<td>0.0018</td>
<td>0.0004</td>
</tr>
<tr>
<td>0.7</td>
<td>0.6647</td>
<td>0.3414</td>
<td>0.1452</td>
<td>0.0532</td>
<td>0.0173</td>
<td>0.0050</td>
<td>0.0013</td>
</tr>
<tr>
<td>0.6</td>
<td>0.7246</td>
<td>0.4255</td>
<td>0.2138</td>
<td>0.0949</td>
<td>0.0380</td>
<td>0.0140</td>
<td>0.0047</td>
</tr>
<tr>
<td>0.5</td>
<td>0.7854</td>
<td>0.5236</td>
<td>0.3084</td>
<td>0.1645</td>
<td>0.0807</td>
<td>0.0369</td>
<td>0.0159</td>
</tr>
<tr>
<td>0.4</td>
<td>0.8452</td>
<td>0.6339</td>
<td>0.4335</td>
<td>0.2749</td>
<td>0.1637</td>
<td>0.0922</td>
<td>0.0495</td>
</tr>
<tr>
<td>0.3</td>
<td>0.9014</td>
<td>0.7516</td>
<td>0.5888</td>
<td>0.4380</td>
<td>0.3117</td>
<td>0.2134</td>
<td>0.1412</td>
</tr>
<tr>
<td>0.2</td>
<td>0.9502</td>
<td>0.8663</td>
<td>0.7631</td>
<td>0.6525</td>
<td>0.5438</td>
<td>0.4429</td>
<td>0.3533</td>
</tr>
<tr>
<td>0.1</td>
<td>0.9857</td>
<td>0.9594</td>
<td>0.9232</td>
<td>0.8793</td>
<td>0.8297</td>
<td>0.7762</td>
<td>0.7204</td>
</tr>
</tbody>
</table>

**Table 3.** The “acceptance probability” $P_{ac}(n; a, a, ..., a, a)$ obtained from (24) for large $n$ and small values of $a$.

<table>
<thead>
<tr>
<th>$n$</th>
<th>$a = 0.1$</th>
<th>$a = 0.05$</th>
<th>$a = 0.01$</th>
<th>$a = 0.005$</th>
<th>$a = 0.001$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.5521</td>
<td>0.8373</td>
<td>0.9915</td>
<td>0.9978</td>
<td>0.9999</td>
</tr>
<tr>
<td>20</td>
<td>0.1597</td>
<td>0.5566</td>
<td>0.9692</td>
<td>0.9918</td>
<td>0.9997</td>
</tr>
<tr>
<td>30</td>
<td>0.0313</td>
<td>0.3157</td>
<td>0.9356</td>
<td>0.9824</td>
<td>0.9992</td>
</tr>
<tr>
<td>40</td>
<td>0.0046</td>
<td>0.1587</td>
<td>0.8931</td>
<td>0.9698</td>
<td>0.9987</td>
</tr>
<tr>
<td>50</td>
<td>0.0006</td>
<td>0.0724</td>
<td>0.8438</td>
<td>0.9543</td>
<td>0.9979</td>
</tr>
<tr>
<td>60</td>
<td>0.0000</td>
<td>0.0304</td>
<td>0.7900</td>
<td>0.9363</td>
<td>0.9971</td>
</tr>
<tr>
<td>70</td>
<td>0.0000</td>
<td>0.0119</td>
<td>0.7332</td>
<td>0.9161</td>
<td>0.9960</td>
</tr>
<tr>
<td>80</td>
<td>0.0000</td>
<td>0.0044</td>
<td>0.6753</td>
<td>0.8939</td>
<td>0.9949</td>
</tr>
<tr>
<td>90</td>
<td>0.0000</td>
<td>0.0015</td>
<td>0.6173</td>
<td>0.8700</td>
<td>0.9936</td>
</tr>
<tr>
<td>100</td>
<td>0.0000</td>
<td>0.0005</td>
<td>0.5605</td>
<td>0.8447</td>
<td>0.9921</td>
</tr>
</tbody>
</table>

Remark 4. From formula (24) it follows

\[
\lim_{a_1 \to 0} \lim_{a_2 \to 0} \lim_{a_{n+1} \to 0} \cdots \lim_{a_n \to 0} P_{ac}(n; a_1, a_2, ..., a_n, a_{n+1}) = 1
\]

(29) 

\[
\lim_{a_1 \to \infty} \lim_{a_2 \to \infty} \lim_{a_{n+1} \to \infty} \cdots \lim_{a_n \to \infty} P_{ac}(n; a_1, a_2, ..., a_n, a_{n+1}) = 0
\]

which proves that the AGDR algorithm is very fast for small values of the parameters $a_1, a_2, ..., a_n, a_{n+1}$ (see also Table 3).

We will denote by AGDW (Wilks’ algorithm) the procedure for generating Di-
richlet $D(n; a_1, a_2, \ldots, a_n, a_{n+1})$ random vectors based on the following proposition

**Proposition 2** (Wilks [14], p. 179). If $Y_j$, $1 \leq j \leq n + 1$, are $n + 1$ independent random variables having a gamma distribution with parameter $a_j$, and

$$X_i = Y_i/(Y_1 + Y_2 + \ldots + Y_n + Y_{n+1}); \quad 1 \leq i \leq n$$

then the random vector $(X_1, X_2, \ldots, X_n)$ has a Dirichlet distribution.

**Table 4.** The values of “acceptance probability” $P_{ac}(n; a, \ldots, a)$ obtained using a Monte Carlo procedure (100,000 simulations).

<table>
<thead>
<tr>
<th>$a$</th>
<th>$a = 1.0$</th>
<th>$a = 0.5$</th>
<th>$a = 0.25$</th>
<th>$a = 0.2$</th>
<th>$a = 0.125$</th>
<th>$a = 0.1$</th>
<th>$a = 0.05$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 1$</td>
<td>0.4978</td>
<td>0.7824</td>
<td>0.9248</td>
<td>0.9509</td>
<td>0.9804</td>
<td>0.9879</td>
<td>0.9968</td>
</tr>
<tr>
<td>$n = 2$</td>
<td>0.1736</td>
<td>0.5211</td>
<td>0.8086</td>
<td>0.8782</td>
<td>0.9421</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

Analysing a lot of procedures for generating Dirichlet deviates we concluded that the algorithm AGDW is the fastest (cf. Štefănescu [12]; for $n = 2$ Loukas [5] obtained the same result).

**Remark 5.** The rejection procedure AGDR is based on simple operations; in addition its “acceptance probability” is very large for small values of $a_j$, $1 \leq j \leq n + 1$ (see Remark 4).

**Table 5.** The values of the threshold $q$ for which AGDR algorithm is the fastest (case $a_1 = \ldots = a_2 = \ldots = a_n = a$).

<table>
<thead>
<tr>
<th>$a$</th>
<th>$n = 1$</th>
<th>$n = 2$</th>
<th>$n = 3$</th>
<th>$n = 4$</th>
<th>$n = 5$</th>
<th>$n = 6$</th>
<th>$n = 7$</th>
<th>$n = 8$</th>
<th>$n = 9$</th>
<th>$n = 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>1.26</td>
<td>0.02</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0.9</td>
<td>1.40</td>
<td>0.11</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0.8</td>
<td>1.60</td>
<td>0.25</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0.7</td>
<td>2.05</td>
<td>0.38</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0.6</td>
<td>2.37</td>
<td>0.55</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0.5</td>
<td>2.80</td>
<td>0.80</td>
<td>0.08</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0.4</td>
<td>4.29</td>
<td>1.00</td>
<td>0.34</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0.3</td>
<td>6.75</td>
<td>1.66</td>
<td>0.66</td>
<td>0.20</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0.2</td>
<td>13.43</td>
<td>3.01</td>
<td>1.32</td>
<td>0.75</td>
<td>0.25</td>
<td>0.05</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0.1</td>
<td>100.00</td>
<td>10.2</td>
<td>4.39</td>
<td>2.07</td>
<td>1.34</td>
<td>0.96</td>
<td>0.66</td>
<td>0.44</td>
<td>0.24</td>
<td>0.11</td>
</tr>
</tbody>
</table>

From Proposition 1 it follows that for $0 < q < a_{n+1}$ the run time of the algorithm AGDR $(n; a_1, a_2, \ldots, a_{n+1})$ is less than for the AGDW $(n; a_1, a_2, \ldots, a_n, q)$ algorithm. Therefore, for fixed values of $n, a_1, a_2, \ldots, a_n$ it can be established a threshold $q$, $q > 0$, so that for any $0 < a_{n+1} < q$, the algorithm AGDR $(n; a_1, a_2, \ldots, a_n, a_{n+1})$ is faster then the AGDW $(n; a_1, a_2, \ldots, a_n, a_{n+1})$ algorithm. The value of $q$ depends
on the computer, on the specific implementations of these algorithms or on the particular procedure for generating gamma random variates (see Proposition 2).

From the literature [3], [6]–[10] it can be selected the following algorithms for generating random values having a gamma distribution with parameter $a$, $a < 1$:

- the GT and GBH Cheng-Feast’s procedures [2] (in the case of $a > 0.5$, respectively $a > 0.25$);
- the GS Ahrens-Dieter’s procedure [1].

The GS algorithm is the fastest when it generates only one random value.

<table>
<thead>
<tr>
<th>$a_2$</th>
<th>1.0</th>
<th>0.9</th>
<th>0.8</th>
<th>0.7</th>
<th>0.6</th>
<th>0.5</th>
<th>0.4</th>
<th>0.3</th>
<th>0.2</th>
<th>0.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>1.0</td>
<td>0.02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.9</td>
<td>0.06</td>
<td>0.11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.8</td>
<td>0.12</td>
<td>0.18</td>
<td>0.25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.7</td>
<td>0.17</td>
<td>0.24</td>
<td>0.32</td>
<td>0.38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.6</td>
<td>0.23</td>
<td>0.30</td>
<td>0.39</td>
<td>0.47</td>
<td>0.55</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.5</td>
<td>0.34</td>
<td>0.42</td>
<td>0.49</td>
<td>0.59</td>
<td>0.65</td>
<td>0.80</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.4</td>
<td>0.41</td>
<td>0.50</td>
<td>0.59</td>
<td>0.68</td>
<td>0.78</td>
<td>0.90</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.3</td>
<td>0.53</td>
<td>0.62</td>
<td>0.72</td>
<td>0.79</td>
<td>0.94</td>
<td>1.06</td>
<td>1.22</td>
<td>1.66</td>
<td></td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.2</td>
<td>0.65</td>
<td>0.78</td>
<td>0.84</td>
<td>0.99</td>
<td>1.11</td>
<td>1.29</td>
<td>1.64</td>
<td>2.29</td>
<td>3.01</td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.1</td>
<td>0.77</td>
<td>0.96</td>
<td>1.12</td>
<td>1.18</td>
<td>1.46</td>
<td>1.73</td>
<td>2.14</td>
<td>2.96</td>
<td>5.08</td>
</tr>
</tbody>
</table>

Tables 5 and 6 give the $q$ threshold values for different $n$, $a_1$, $a_2$, ..., $a_n$ considering a FORTRAN implementation of the AGDR and AGDW algorithms run on a Romanian FELIX C-256 computer and using the GS Ahrens-Dieter’s procedure [1] for gamma random variates.

ACKNOWLEDGEMENT

The author is grateful to the referees for their suggestions. (Received March 21, 1989.)

REFERENCES


Dr. Ştefan Ştefănescu, University of Bucharest, Faculty of Mathematics, Computing Centre, 14 Academiei Street, Bucharest 70109, Romania.