ON SOME NUMERICAL METHODS FOR THE COMPUTATION OF ERLANG AND ENGSET FUNCTIONS

Jerzy KUBASIK

Institute of Electronics, Technical University of Poznań
Poznań, Poland

ABSTRACT

In the paper simple, efficient and accurate numerical methods for the evaluation of Erlang and Engset loss formulas are presented. The paper contains a review of practical formulas enabled to construction of convenient programs that rapidly evaluate some most frequently used traffic problems. The following problems are discussed: determination of probability of loss, determination of permissible offered traffic, determination of number of servers using both Erlang and Engset functions. Furthermore, approximations of Erlang B function for noninteger number of trunks are mentioned. Several of these methods are based on new improved procedures that are first presented in this paper. The described methods were tested on a computer over a wide range of arguments.

1. INTRODUCTION

The Erlang and Engset functions for lost call systems are the basic formulas in telecommunications traffic theory. They are the relation between four quantities: loss probability of call congestion - B, offered traffic intensity - A, number of traffic sources - S, and number of servers - N. The Erlang B formula is valid for the infinite value of S. The importance of these functions in current traffic engineering is well known. Many tables and charts have been published, e.g. [2, 4, 5, 9, 13, 17], that allow one more readily to apply these formulas in practice. However, many current traffic analysis and design problems can be solved only with the aid of computers. From the analysis point of view, "table or chart look ups", in a computer can be time and space consuming, inaccurate, and inconvenient. It is therefore advantageous to have methods or subroutines available, which can easily be incorporated into the main program for routine maximum likelihood estimation.

The methods presented below have these necessary capabilities and thereby eliminate the need for "table look ups" or interpolation in existing tables. This paper does not fully develop the theories of all the formulas used. Instead it, gives enough description and explanation to make the development of the computational formulas understandable. For theoretical accuracy one should consult the references.

2. ERLANG LOSS FUNCTION

2.1 Determination of Loss Probability

The Erlang B function arises in the study of the M/M/1/n queue. The probability of loss, i.e. probability that an arriving call is rejected because no trunk is available, is expressed by Erlang loss formula, when a Poisson stream of calls offering A Erl to a fully available group of N trunks is mentioned. Several of these methods are based on new improved procedures that are first presented in this paper. The described methods were tested on a computer over a wide range of arguments.
2.2 Case of Noninteger Number of Servers

The Erlang loss formula can only have any physical significance when \( N \) is a positive integer. The case of \( N \) not being an integer has, however, practical significance in many problems concerning the telecommunication network dimensioning or determination of blocking probability in the multistage switching networks.

For this reason it is also desirable to extend the definition for continuous number of devices. There are known numerous equivalent forms of the expansion for Erlang loss formula. The most frequently used expressions are the following

\[
E(X,A)^{-1} = \int_0^\infty (X+t)^{-1}e^{-A}dt
\]

(5)

\[
E(X,A)^{-1} = A \int_0^\infty (1+t)X\cdot e^{-At}dt
\]

(6)

\[
E(X,A)^{-1} = e^{AX}X \int_0^\infty tX\cdot e^{-At}dt
\]

(7)

\[
E(X,A)^{-1} = e^{AX}X \int_0^\infty (X+1)A\cdot e^{-At}dt
\]

(8)

where

\[
\Gamma(X,A) = \int_0^\infty tX\cdot e^{-At}dt
\]

(9)

is the Incomplete Gamma function of the second kind. In these formulas \( X \) denotes the noninteger number of trunks.

Because all of these expressions contain the integrals or special functions, they are not convenient for numerical evaluation and for practical use. Therefore, many authors have developed the approximations of Erlang function for \( X \).

It should be noted that the definition (1) as well as the continuous expansions (5), (6), (7), and (8) satisfy the recurrence formula (2), which is valid for all nonnegative \( X \). Furthermore, Rapp [15] has shown that this relation will not propagate errors in the initial value \( E(X_0,A) \). Therefore, \( X_0 \) can be chosen to be the fractional part of \( X \), i.e. \( 0 < X_0 < 1 \), and if \( E(X_0,A) \) can be easily computed, then applying (2) \( X-X_0 \) times will yield \( E(X,A) \).

Rapp [15] has proposed the approximation of \( E(X_0,A) \) based on quadratic interpolation, namely

\[
E(X_0,A) \approx C_0 + C_1X_0 + C_2X_0^2
\]

(10)

where

\[
C_0 = 1
\]

(11)

\[
C_1 = -(2+A)/(1+5A^2)
\]

(12)

\[
C_2 = 1/(1+4A+4A^2+A^3)
\]

(13)

Another expression for computing \( E(X_0,A) \) has been given by Szybicki [18]

\[
E(X_0,A) = \left[ (2-X_0)^{A+2}/(X_0+2A+A^2) \right].
\]

(14)

Next approximation developed from Newton's interpolation formula has been described by Jagerman [7] as follows

\[
E(X,A) \approx \sum_{i=1}^{n} \frac{1}{B_i} B_i(1+x_i/A)X_0(1-X_0)/2
\]

(15)

In this formula \( B_1, B_2 \), and \( B_3 \) denote \( E(N,A) \), \( E(N+1,A) \), and \( E(N+2,A) \), respectively, where \( N \) is the integral part of \( X \). Using this formula one obtain value of loss probability directly for \( X \) and not for \( X_0 \).

Farmer and Kaufman [6] have proposed to calculate the \( E(X_0,A) \) by means of the Laguerre quadrature; i.e. the Erlang function given by definition (5) is approximated by the sum

\[
E(X_0,A)^{-1} = \frac{\sum_{i=1}^{n} w_i (1+t_i/A)}{X_0},
\]

(16)

where \( w_i \) and \( t_i \) are the weight factors and abscissas respectively and they are given in [1]. Naturally, the better accuracy is achieved by increasing \( n \).

Another method, in which the series expansion of Gamma function is applied, is described by Urmonei [19]. He expresses the Erlang function in the following way

\[
E(X_0,A)^{-1} = \left[ (X_0+1) \sum_{n=0}^{\infty} (0_n - D_n) \right]
\]

(17)

where

\[
C_n = A^{n-X_0}/n!
\]

(18)

\[
D_n = A^{n+1}/(X_0+n+2)
\]

(19)

The above coefficients can be computed recurrently

\[
C_n = C_{n-1}/A/n!
\]

(20)

\[
D_n = D_{n-1}/(X_0+n+2)/A
\]

(21)

with the starting values

\[
C_0 = 0
\]

(22)

\[
D_0 = 0
\]

(23)

\[
D_0 = A/(X_0+2)
\]

(24)

The process of calculation of the \( C_n \) and \( D_n \) is stopped when given accuracy indicated by \( |C_n - D_n| \) is obtained. The Gamma function in the formulas (15) and (16) can be computed by the series expansion

\[
\Gamma(Y) = \sum_{k=1}^{\infty} c_k Y^k
\]

(25)

The factors \( c_k \) are given in [1, 19].

One should remember that for Gamma function the following recursive relation is valid

\[
\Gamma(Y+1) = Y\Gamma(Y)
\]

(26)
Therefore, to save the computation time, the procedure (19) can be used only one times. The \( \gamma(X_0+1) \) in (15) must be calculated from (19), while for \( X_0+2 \) in (18) it may be obtained from the recurrence (20).

The next method described here is the Lévy-Soussan procedure [12], which uses the continued-fraction expansion of the Incomplete Gamma function. The Erlang B function can be calculated as the continued-fraction expansion of some function \( F(A,-X) = E(X,A)^{-1} \). The value of \( F(A,-X) \) lays in the interval between \( P \) and \( In' \) which are the even and odd approximants of \( F \) respectively, defined as

\[
P_n(A,-X) = \frac{A}{A-X+2} + \frac{X}{A-X+4} + \cdots + \frac{n(X-n+1)}{A+2n-1} \tag{21}
\]

\[
In(A,-X) = \frac{A}{A-X} + \frac{X}{A-X+2} + \cdots + \frac{(n-1)(X-n+1)}{A+2n-1} \tag{22}
\]

The notation

\[
P(X) = b_0 + \frac{a_0}{b_1 + \frac{a_1}{b_2 + \frac{a_2}{b_3 + \cdots}}} \tag{23}
\]

can be replaced by a more legible form of continued-fraction

\[
P(X) = \frac{a_0}{b_1 + \frac{a_1}{b_2 + \frac{a_2}{b_3 + \cdots}}} \tag{24}
\]

Simultaneous computation of \( P_n \) and \( In \) is particularly convenient. The computation sequence is stopped when the first of two conditions is met: limit of \( n \) was exceeded or when \( |I_n - P_n|/I_n | < \varepsilon \) is obtained, where \( \varepsilon \) is the given accuracy. The first condition means that required accuracy is unobtainable by using this method.

The last method proposed in this section uses the basic method of numerical evaluation of Incomplete Gamma function \( \gamma(X,A) \). The \( \gamma \) is defined as

\[
\gamma(X,A) = \int_0^A t^{X-1} e^{-t} dt \tag{25}
\]

Since

\[
\gamma(X,A) = \Gamma(X) - \gamma(X,A) \tag{26}
\]

the function \( \gamma(X,A) \) can be used for determination of blocking probability from Erlang formula. Substituting \( \gamma(X+1,A) \) by (26) into (8) yields the following expression for \( E(X,A) \)

\[
E(X,A)^{-1} = A - X E[A \left( \gamma(X+1) - \gamma(X+1,A) \right) \tag{27}
\]

Furthermore, it can be shown that

\[
\lim_{A \to \infty} \gamma(X,A) = \Gamma(X) \tag{28}
\]

Hence, the function \( \gamma \) may be used for computation of the Gamma function by choosing relatively large value of \( A \).

The Incomplete Gamma function \( \Gamma \) may be written as

\[
\gamma(X,A) = X^{-1} A X e^{-A} M(1,1+X,A) \tag{29}
\]

where

\[
M(1,1+X,A) = 1 + \sum_{j=1}^{\infty} A^j / \sum_{k=1}^{\infty} (X+k) \tag{30}
\]

is the Confluent Hypergeometric function or Kummer's function, which can be represented by

\[
M(1,1+X,A) = \sum_{j=0}^{\infty} T_j \tag{31}
\]

and the terms \( T_j \) are obtained using the following recurrence formula

\[
T_j = T_{j-1} - A/(X+j) \tag{32}
\]

with \( T_0=1 \). One should mention the asymptotic expansion of Erlang loss function proposed by Akimaru and Takahashi [3]. They have found the general representation of \( E(X,A) \) in the following form

\[
E(X,A)^{-1} = \sum_{j=0}^{\infty} a_j x^{1-j}/2 \tag{33}
\]

Because the method for determination of the factors \( a_j \) is complicated, formula (33) cannot be easily applied for practical computations of \( E(X,A) \).

There exist some other methods, e.g. the described in [6], but they are very complicated and can be used only in special situations.

2.3 Determination of Traffic Intensity

Determination of the permissible offered traffic for a given number of servers and given value of loss probability is frequently required in the practical design of communication systems. Erlang formula (1) can be solved for \( A \) directly, only for

\[
N=1 \quad A = B/(1-B) \tag{34}
\]

\[
N=2 \quad A = [B + \sqrt{B(2-B)}]/(1-B) \tag{35}
\]

Although this equation for larger values of \( N \) cannot be easily inverted and calculation of \( A \) by an explicit formula is unobtainable, numerical methods can be used. This problem has been previously dealt with by several authors.

Rappaport [16] presented scheme of determination of carried traffic \( C \) and offered traffic \( A \) using method of false position ('regula falsi'). He proposed to solve for \( C \) the equation

\[
P(C) = E(C) - B = 0 \tag{36}
\]

where \( B = E[N,C/(1-B)] \). The solution must lie in the interval from 0 to \( N \). The described method con-
tain three parts. To reduce the number of iterations a simple method of finding roots, e.g. method of bisections, was used first. Initial search was applied to isolate the solution within an interval \( N/100 \). In the first part of iterations, beginning with \( C_{i-1} \) and \( C_i \) as the leftmost and rightmost points of interval, a new value \( C_{i+1} \) is calculated using formula

\[
C_{i+1} = \frac{C_{i-1}E(C_i) - C_iE(C_{i-1}) + B(C_i-C_{i-1})}{E(C_i) - E(C_{i-1})}
\] (37)

This new value replaces that value of \( C_{i-1} \) or \( C_i \), for which the function (36) has the same sign. The process is repeated until the denominator of (37) becomes smaller than 0.01B. Second part of iterations is described by the formula

\[
C_{i+1} = C_i - P \cdot F(C_i)
\] (36)

where the constant \( P \) is calculated after first part of iterations by

\[
P = \frac{(C_i-C_{i-1})}{E(C_i) - E(C_{i-1})}
\] (39)

After each step a new value replaces the former value \( C \). Iterative process is stopped when given accuracy is obtained. Then offered traffic can be determined as \( A = \frac{E(N)}{1-B} \).

Szybicki [18] described special numerical method for offered traffic calculation from Erlang function. His method is given by the formula

\[ A_0 = N/(1-B) \] (41)

Both presented methods are the first order convergence procedures. Because the first derivative of Erlang loss formula with respect to \( A \) is well known

\[ E'(A) = E(A) - B + E'(A)N/A \] (42)

some faster numerical methods can be applied to determine \( A \). There is Newton's iterative method for calculation of a simple zero of a nonlinear equation

\[ F(A) = 0 \] (43)

In this method iterations are defined by

\[ A_{i+1} = A_i - \frac{F(A_i)}{F'(A_i)} \] (44)

This is the second order convergence method. It can be used to solve the discussed problem, if an appropriate function, which have characteristic needed in this procedure is chosen. For example it may be the function

\[ F(A) = A - E(A) - B \] (45)

First derivative of this function is given by

\[ F'(A) = E(A) - B + A \cdot E'(A) \] (46)

whereas, authors of [6] suggest to assume

\[ A_0 = N, \] and

\[ A_1 = N - \frac{E(N)-B}{E(N)^2} \] (49)

as the left side of equation (43). The derivative of (47) is identical as (42). The proposed starting values \( A_0 \) are also different. In [7] we have

\[ A_0 = N/(1-B) - 1 \] (43)

where \( E(N) \) can be approximated by

\[ E(N) = E(N,A) \approx 2/(N+1) \] (50)

In [4] Newton's method is applied with the function

\[ F(A) = 1/E(A) - 1/B \] (51)

There is proposed a new, third order convergence method, based on Halley's formula [14]

\[ A_{i+1} = A_i - \frac{F(A_i)}{F'(A_i) - \frac{2F''(A_i)}{F'(A_i)^2}} \] (52)

As function \( F \) the expression (45) can be used. Second derivative of this function with respect to \( A \) is given by

\[ F''(A) = E'(A) \cdot (N-A+2) + E(A)N/A \] (53)

This procedure is more complicated than Newton's method, but it needs less iterations to obtain the result with the same accuracy. Number of iterations needed to obtain one element with the accuracy to six significant figures and processing time per element are compared in [10]. The proposed method makes it possible to calculate Erlang loss function significantly faster than known procedures. This method is especially attractive for small values of loss probability.

One should be confirm that all presented in this section methods can be applied for computation both with integer and with noninteger numbers of servers in Erlang formula.

2.4 Determination of Number of Trunks

The problem of determination of minimum number of devices or trunks, which are able to serve the offered traffic with given grade of service, is also frequently studied. Since the expression for the derivative of Erlang loss function with respect to \( X \) is rather complicated, three approaches are possible for computation of the required value.

First, which yields the solution in the integer numbers, is based on the repetitions of calculations of \( E(N,A) \) for a given \( A \) and for increasing \( N=1,2,3,\ldots \), using the recurrence formula (2). The process is continued until the
obtained, in each step, \( E(N,A) \) becomes smaller than the assumed probability \( B \). The value \( N \), for which the last computation is performed, is the required number of servers [19].

The second possible approach is the application of Newton’s or other iterative methods with approximate values of the derivative. Jagerman [7] has proposed the following approximation of the derivative of Erlang function with respect to \( X \)

\[
E'(X) = E(X) \cdot \frac{1/(2a) - \ln a}{1 - E(X)/(2a)}, \tag{54}
\]

where

\[
\alpha = E(X) + (1+X)/A,
\tag{55}
\]

and \( E(X) = E(X,A) \).

It is possible to apply the another approximation of \( E'(X) \) given in [18]

\[
E'(X) = \left[ E(X+y) - E(X-y) \right] / (2y), \tag{56}
\]

where \( y \ll 1 \), is the free chosen small real number. Knowing \( E'(X) \) we can solve for \( X \) the nonlinear equation

\[
P(X) = 0, \tag{57}
\]

where

\[
P(X) = E(X) - B, \tag{58}
\]

using the Newton’s method, in which each iteration is given by

\[
X_{i+1} = X_i - \frac{P(X_i)}{P'(X_i)}. \tag{59}
\]

The derivative \( P'(X_i) \) is simply equal to \( E'(X_i) \), and the suitable starting value is \( X_0 = (A+1)(1-B) \) [7].

Third possible method employs the iterative method for finding a simple zero of the equation (57) other than Newton’s method, e.g. regula falsi [6]. In this method, beginning from \( X_{i-1} \) and \( X_i \) as leftmost and rightmost points of interval, in which the solution must lie, a new value \( X_{i+1} \) is calculated using the formula

\[
X_{i+1} = \frac{X_{i-1}E(X_{i-1}) - X_iE(X_i)}{E(X_{i-1}) - E(X_i)}. \tag{60}
\]

This new value replaces that value of \( X_{i-1} \) or \( X_i \) for which the function (58) has the same sign. As the starting values \( X_0 \) and \( X_1 \) two numbers must be chosen, for which \( P(X_0) \) and \( P(X_1) \) have the opposite signs, e.g. \( X_0 = 0 \) and \( X_1 = A \). Hence, \( E(0)=1 \) and from (50)

\[
E(A) = E(A,A) = \sqrt{2/(\pi A)}, \tag{51}
\]

the next \( X_2 \), i.e. \( X_2 \) is obtained immediately

\[
X_2 = A - B \left[ 1 - \sqrt{2/(\pi A)} \right], \tag{61}
\]

whereafter, if it is necessary, one continues the iteration using (60).

For the calculation of \( E(X) \) in all cases any method described in section 2.2 may be used.

Naturally, the other approaches are possible for solving this problem more accurate. The search for a simple, fast, and accurate method for evaluate the first and the next derivatives of the Erlang loss function with respect to the number of devices seems to be especially attractive. Such a method should be applied in higher order convergence methods for fast determination of \( X \).

3. ENGSET FUNCTION

3.1 Determination of Loss Probability

The Erlang loss formula is valid with the assumption that the Poisson distribution is used as the traffic source model, which implies infinite number of traffic generating sources. There are many situations, however, where the traffic is originated by a small number of subscribers, e.g. in PBXs, small rural exchanges, line concentrators. In such cases the Engset/ truncated Binomial distribution is a better traffic model than Poisson distribution, but it is also more difficult to manipulate numerically.

The well-known Engset function is an implicit function. When traffic \( A \) Erl generated by \( S \) independent sources with equal calling intensities is offered to a full availability group of \( N \) trunks, the probability of call congestion \( B \) is a function of \( S, N, A, \) and also \( B \) itself. It is usually stated in the form

\[
B = B(S,N,A) = \frac{(S-1)A/S}{1-(1-B)A/S} \tag{52}
\]

For convenience in writing, the following abbreviations are used

\[
\alpha = a/[1-a(1-B)], \tag{63}
\]

and

\[
a = A/S. \tag{64}
\]

The \( \alpha \) is the average traffic generated per source in Erl, whereas, \( \alpha \) means the average traffic originated by one source when it is free. Thus, from (62) we obtain

\[
B(S,N,\alpha) = \frac{(S-1)\alpha^N}{N} \sum_{i=0}^{N-1} \frac{(S-1)^i}{1-(1-B)A/S} \tag{65}
\]

The blocking probability /time congestion/ in the Engset systems is defined by

\[
B(S,N,\alpha) = \frac{(S-1)\alpha^N}{N} \sum_{i=0}^{N-1} \frac{(S-1)^i}{1-(1-B)A/S} \tag{66}
\]

with \( \alpha \) given by (63). There exist the interrelation formulas between \( B \) and \( E \).
In this section problem of determination \( B \) with respect to \( S, N, \) and \( A \) will be discussed. The case of integer number of sources and servers will only be developed.

Beforehand, the methods for computation of \( B \) when \( S, N, \) and \( \alpha \) are given. First of them is a well-known recurrence relation

\[
B(N) = \frac{(S-N)\cdot \alpha B(N-1)}{S-N+1} \quad \text{for} \quad N > 2
\]

where \( B(N) \) is the abbreviation from \( B(S,N,\alpha) \). The initial value \( B(0) \) is evidently equal to unity. The other form of this formula has been given by Joys \([8]\) for the inverse of \( B \) when \( N = 2 \).

\[
B(N) = \frac{(S-N)\cdot \alpha B(N-1)}{S-N+1} \quad \text{for} \quad N = 2
\]

\[
B(N) = \frac{S-N\cdot \alpha B(N-1)}{S-N+1} \quad \text{for} \quad N = 1
\]

The terms \( T_j \) are successively computed using the recurrence formula

\[
T_j = \frac{T_{j-1} \cdot (N-j+1)}{\alpha (S-N-j+1)} \quad \text{for} \quad j = 1, 2, \ldots, N
\]

with \( T_0 = 1 \). The calculations can be terminated before \( j = N \) because the individual terms for increasing indexes \( j \) become very small and have no significant effect on the series sum.

This new method may be used for fast calculation of loss probability from Engset function especially for large trunk groups.

Now we return to the problem of computation of \( B \) when the value \( A, \) not \( \alpha, \) is given. However, \((65)\) is not an explicit equation, since \( \alpha \) is not a constant, but a function of \( B \) \((63)\). Therefore, an iterative procedure is necessary. For this purpose an auxiliary function

\[
D = B - F(B) \quad \text{for} \quad N > 2
\]

is used, where \( F(B) \) is the new estimate of \( B \) computed from \((70), (71), \) or \((73)\) with \( \alpha \) obtained from \((65)\) using original \( B \). For computational efficiency equation \((71)\) has been simplified to the following form

\[
1/\alpha = C + B
\]

where \( C = S/A - 1 \), and hence, \( C \) is a constant for a given combination of traffic and the number of sources.

The above described method, accompanied with the Joys recurrence \((71)\), was employed in \([2, 17]\).

3.2 Determination of Traffic Intensity

The determination of the permissible offered traffic for a given other quantities from Engset distribution is also the frequently required problem. Since from \((65)\) and \((64)\) we have

\[
A = S/(1/\alpha + 1 - B)
\]

the equation

\[
B(S,N,\alpha) = B
\]

where \( B \) is the given probability of loss, may be solved for \( \alpha \) and the result used for the calculation of \( A \).

Simple solution of \((76)\) can be found directly only for

\[
N = 1 \quad \alpha = B/(1 - B) \quad \text{and} \quad (S-1) \quad \text{and} \quad (79)
\]

\[
N = 2 \quad \alpha = \left[1 + \sqrt{1 + 2(S-2)}\right] / (S-1) \quad \text{and} \quad (80)
\]

For \( N \) greater than two \((76)\) cannot be solved directly for \( \alpha \), especially it is impossible to find the general explicit formula for all values of \( N \). The iterative procedure must be used.

Driksna and Wormald \([5]\) have used the simplest, but very slow, method of bisections. Here, the method, which employs the Newton’s iterative procedure, will be proposed. The iterations are given by

\[
\alpha_{i+1} = \alpha_i - \frac{F(\alpha_i)}{F'(\alpha_i)}
\]

The function \( F(\alpha) \) can be chosen as

\[
F(\alpha) = 1/B - 1/B(S,N,\alpha)
\]

According to \((73)\) the above equation may be written as

\[
F(\alpha) = 1/B - \sum_{j=0}^{N} T_j
\]

with the terms defined recurrrently by \((74)\) or directly performed as function of \( \alpha \)

\[
T_j = \frac{N!}{(N-j+1)! (S-N-j)!} \quad \text{for} \quad j = 1, 2, \ldots, N
\]

Hence, the first derivative of \( F(\alpha) \) with respect to \( \alpha \) has the following form

\[
F'(\alpha) = \frac{1}{\alpha} \sum_{j=0}^{N} j \cdot T_j
\]

Since the similar sum of series appears in the formulas for computation of \( F(\alpha) \) \((83)\) and its derivative \((85)\), they may be calculated simultaneously. The suitable starting value \( \alpha_0 \) is any number belonging to the interval \((0,1)\).

In the same way the second and the next derivatives of Engset function can be calculated to apply in the numerical procedures with higher degree of convergence, e.g. in the Halley’s method \([14]\).
3.3 Determination of Number of Devices and Number of Sources

Methods for determination of number of devices when A, B, and S are given or determination of number of sources generating the traffic A Erl, which is carried by N trunks with given grade of service are also frequently required in many teletraffic problems.

Since the simple formulas for the partial derivative of Engset loss function with respect to N or S are not known, the method of successive calculation of probability of loss can be applied. From (63) and (64) with given A, B, and S, \( \epsilon \) is obtained and whereafter, by repetitions of calculations of \( B(S,N,\epsilon) \) for increasing \( N = 1, 2, \ldots \) using the recurrence formulas (70) or (71). The process is stopped when the obtained in \( N \)-th step value \( B(S,N,\epsilon) \) becomes smaller than assumed grade of service B.

The recurrence formula for increasing S has been also given by Joys [8]

\[
I(S) = \frac{S + N + 1}{S + 1} \left[ (1+\epsilon)I(S-1) - \epsilon \right], \quad (66)
\]

where

\[
I(S) = \frac{1}{B(S,N,\epsilon)} \quad \text{(67)}
\]

and the starting value \( I(N+1) = \epsilon/(1+\epsilon) \). The method for determination of S is a little more complicated, because the value \( \epsilon \) is not a constant, but a function of S (63). Hence the iterative procedure should be employed, analogous to this, which is used for determination of B with respect to A.

One should be noted that the above methods yield the integer results only.

4. CONCLUSIONS

All methods described in this paper were tested on ODA 1305 computer /compatible with ICL 1900/. The programs written in FORTRAN have been used to calculate tables over a wide range of parameters.

Number of iterations needed to obtain one element with the assumed accuracy and the processing time have been compared for some algorithms in [10].

Subroutines, which illustrate all described above methods appears in [1].

REFERENCES