J. King Saud Univ., Vol. 2, *Eng. Sci.* (2), pp. 347–353 (1410H/1990)

SHORT COMMUNICATION

A New Method for Computing the Last Trunk Occupancy for High-Usage Telephone Routes

Saad Haj Bakry
College of Engineering, King Saud University,
P.O.Box 800, Riyadh-11421, Saudi Arabia

Abstract. In this paper, a new formula is derived for the last trunk occupancy of a full-availability group of trunks, assuming sequential hunting, which is an important performance measure for high-usage telephone routes. The formula can be easily programmed on the computer, and it avoids the occurrence of arithmetic overflow during computation, allowing the evaluation of the last-trunk occupancy for large groups of trunks.

Introduction

An important measure in the evaluation of the performance of high-usage telephone routes, which consist of full-availability groups of trunks, is the last-trunk occupancy, assuming sequential hunting. This measure, P (n,a), is given in terms of the number of trunks (n) of the high-usage route concerned, and the traffic offered in Erlang a, as follows [1].

$$P(n,a) = B(n,a)\,F(n,a) \qquad (1)$$

where B (n,a) is the lost-call congestion given by the Erlang-B formula as follows,

$$B(n,a) = \frac{a^n/n!}{\sum_{i=0}^{n} (a^i / i!)} \qquad (2)$$

and where F (n,a) is the last trunk occupancy factor, given as follows,

$$F(n,a) = \frac{\sum_{i=0}^{n-1} [(n-i)\, a^i / i!]}{\sum_{i=0}^{n-1} (a^i / i!)} \qquad (3)$$

B (n,a) has been programmed on the computer using a recursive algorithm similar to the following, for the denominator element h (i,a) = (a^i / i!) of equation (2) [2,3]

$$h(0,a) = 1$$
$$h(i+1,a) = [a / (i+1)]\, h(i,a) \qquad (4)$$

The same principle can be used for programming F (n,a). Although, exact results for B (n,a), F (n,a), and consequently P (n,a), can be obtained using the principle, arithmetic overflow problems exist for routes with large number of trunks.

In ref. [2], the arithmetic overflow problem has been solved for B (n,a), using some approximations. However, the approximations used are only valid for small B (n,a), which is not suitable for high-usage routes. Recent work has derived a new expression for B (n,a) that enables its evaluation on the computer with no restriction on the size or on the traffic level of the system considered [4].

The purpose of this paper is to develop a new method for computing P (n,a) with no restriction. This is done by using the B (n,a) expression given in [4], deriving a new and suitable form for F (n,a), and making use of the range of real numbers available on any computer system.

New Formulas

B (n,a) and F (n,a) can be expressed in simpler mathematical forms than those given in equations (2) and (3) using the following Stirling approximation and logarithmic principle [5]. This approximation has proved to be reasonable as discussed in the application section of this paper.

$$i! \approx (2\pi i)^{\frac{1}{2}}\, i^i \exp(-i) \qquad (5)$$
$$g^i = \exp[i \ln(g)]$$

In a recent work [4], B (n,a) has been expressed as follows:

$$B(n,a) \approx 1 / \sum_{i=0}^{n} \exp[x(i) - x(n)] \tag{6}$$

In the same way, F (n,a) can be expressed as follows:

$$F(n,a) \approx \sum_{i=0}^{n-1} \{(n-i) \exp[x(i)]\} / \sum_{i=0}^{n-1} \exp[x(i)] \tag{7}$$

where

$$x(0) = 0$$
$$x(i) = i[1 + \ln(a/i)] - (1/2)\ln(2\pi i)$$
$$x(n) = n[1 + \ln(a/n)] - (1/2)\ln(2\pi n)$$

Using the above expressions, B (n,a) and F (n,a) can be easily programmed on the computer using the exponential function, the natural logarithm function and some looping for computing summations.

Range of Computations

Computations on the computer, using real numbers, are safe in the range (10^{-d}, 10^{+d}), where d depends on the computer system used. As we are dealing with exponential expressions, this range can be expressed as [exp (–k), exp (+k)], where k is given in terms of d as follows.

$$k = d \ln(10) \tag{8}$$

To avoid the occurrence of arithmetic overflow, computations should be done within the specified range.

For computing B (n,a), while making use of the range of real number, the following expression has been derived assuming $q<k$ [4].

$$B(n,a) \approx \exp(q) / \sum_{i=0}^{i=n} \exp[x(i) - x(n) + q] \tag{9}$$

Non-significant exponential elements in the summation above, which are less than exp (–k) are disregarded [4].

For computing F (n,a) using equation (7), it can be seen that the largest element which would cause an arithmetic overflow when n is large is exp [x (i)] for i = (n–1). To avoid such an overflow, it is necessary to perform the computation within the range of real values accepted by the computer system used. This can be achieved by multiplying the summation of the nominator and that of the denominator of equation (7) by the element exp (–y), where,

$$y = x(n-1) - q$$

$$x(n-1) = (n-1)\{1 + \ln[a/(n-1)]\} - (1/2)\ln[2\pi(n-1)] \qquad (10)$$

Equation (6), therefore becomes,

$$F(n,a) \approx \sum_{i=0}^{n-1} \{(n-i)\exp[x(i)-y]\} \Big/ \sum_{i=0}^{n-1} \exp[x(i)-y] \qquad (11)$$

The largest exponential element in this equation will be exp (q). Exponential elements less than exp (–k) will be insignificant and can be disregarded during computations, avoiding arithmetic underflow.

Applications and Verifications

For computing P (n,a), equation (1), and the new equations (9) and (11) have been programmed on an IBM-PC computer using TURBO PASCAL-3 where d = 38, and consequently k = 88; with q given the value of 80. For comparison, equations (2) and (3) have also been programmed using the recursive algorithm given in (4), so that P (n,a) can be computed in the usual conventional method. Table 1 presents results obtained for P (n,a), using both the conventional method and the new method. The numerical results given for P (n,a) consider 4 significant digits. The relative percentage difference (error), of the new method relative to the conventional exact method is given. Although, the numerical values given in the table only consider two significant digits, for this relative difference, it has been computed using the 11 significant digits of the usual real values that occupy 6 bytes each. The reason for using less significant digits is to reduce table space, while maintaining reasonable accuracy.

The Table demonstrates the limitation of the conventional method, as an arithmetic overflow occurred during the computation of P (n,a) for a group of 100 trunks, while the new method was free of such limitation. In addition, the Table shows how the relative difference between the two methods decreases, as the number of trunks

Table 1. The conventional method versus the new method in computing the last trunk occupancy (Real value limits: 10^{-38} to 10^{+38})

No. of Trunks	Traffic Offered (erlang)	Last trunk occupancy		Relative Error %
		Conventional method	New Method	
10	7	0.3035	0.3025	0.330
20	15	0.2715	0.2712	0.120
30	24	0.2910	0.2909	0.057
40	33	0.2944	0.2943	0.036
50	42	0.2914	0.2913	0.026
60	51	0.2854	0.2853	0.020
70	61	0.3152	0.3152	0.014
80	70	0.3041	0.3041	0.012
90	79	0.2931	0.2931	0.010
100	89	Overflow	0.3131	–
200	183	Overflow	0.2903	–
300	279	Overflow	0.2910	–
400	376	Overflow	0.2970	–
500	474	Overflow	0.3097	–

increases. At 10 trunks the relative difference is 0.33% (0.0033), while at 90 trunks, the relative difference is 0.01% (0.0001). The decrease of this relative difference is related to the Stirling factorial approximation used, which provides better accuracy at higher values.

Another source of error in the new method is related to ignoring the values under exp (–k). However, such errors are insignificant, as equations (9) and (11) are designed to include large values reaching exp (q), for any number of trunks at any traffic level. The ignored values will therefore only represent exp (–k–q) of the large values, and this would insure their insignificance.

Conclusions

The new method presented in this paper enables the computations of the last-trunk occupancy for full-availability groups of trunks, assuming sequential hunting, with no restriction on the size of the group considered, or on the level of traffic offered.

References

[1] Bear, D. *Principles of Telecommunication-traffic Engineering.* London: Peter Peregrinus Ltd., IEE, 1978.

[2] Hebuterne, G. *Traffic Flow in Switching Systems.* Norwood, MA, USA: Artech House, 1987.

[3] Bakry, S.H. "Algorithm for the investigations of basic teletraffic engineering problems." King Saud Univ., *J. Eng. Sc.*, 10, Nos. 1 and 2 (1984), 1-7.

[4] Bakry, S.H. "A New Method for Computing Erland-B Formula." *International Journal of Computers and Mathematics with Applications* (Pergaman Press, U.K.) 19, No.2 (1990), 73-77.

[5] Hughes, W.F. and Gaylord, E.W. *Basic Equations of Engineering Science.* New York: McGraw-Hill, 1964.

طريقة جديدة لحساب شُغْل دارة الربط الأخيرة لطريق هاتفي كثير الاستخدام

سعد الحاج بكري

كلية الهندسة، جامعة الملك سعود، ص.ب ٨٠٠،
الرياض ١١٤٢١، المملكة العربية السعودية

ملخص البحث. يستخرج هذا البحث علاقة جديدة لحساب شُغْل دارة الربط الأخيرة لطريق هاتفي مكون من مجموعة من دارات الربط المتيسرة بشكل كامل، وذلك بفرض أن البحث عن دارة ربط حرة يتم تسلسليا. ويعتبر حساب هذا الشُغل مهما في تقويم أداء الطرق الهاتفية الكثيرة الاستخدام. وتتميز العلاقة الجديدة بسهولة برمجتها على الحاسوب، وتجنبها لحدوث الفيض الحسابي أثناء التنفيذ، وهي بذلك تسمح بتقويم شُغْل دارة الربط الأخيرة لطرق هاتفية مكونة من مجموعات كبيرة من دارات الربط.