APPENDICES

Analysis of Grade of Service (GoS) in Mobile Communication using GoS Calculator

MOHD DANIAL BIN ROZAINI
Faculty of Electrical Engineering,
Universiti Teknologi MARA
40450 Shah Alam Selangor
Email: sukusyi26@ymail.com

***Abstract* - This paper proposes a Grade of Service (GoS) calculator created using Matlab software in order to calculate the probability of a call being blocked or delayed. The aim of this project is to analyze the data that we get from the calculator by using three different formulas. The calculator used Erlang B formula to obtain the value of GoS and to get more accurate value; extended Erlang B formula can be applied. From the calculated values, comparison between Erlang B and extended Erlang B can be made. The probability that a call is queued and the average call waiting time, a.k.a. the Average Speed of Answer (ASA) can be measured using Erlang C formula. The various values obtained from Erlang C calculator can tell how long subscribers need to wait before they can make a call during peak hours.**

***Keywords:* Grade of Service, Erlang B, extended Erlang B, Erlang C and Average Speed of Answer.**

[1] INTRODUCTION

In telecommunication engineering, the quality of voice service is specified by two measures: the GoS and the quality of service (QoS). GoS is the probability of a call in a circuit group being blocked or delayed for more than a specified interval, expressed as a vulgar fraction or decimal fraction. This is always with reference to the busy hour when the traffic intensity is the greatest [3]. Grade of service may be viewed independently from the perspective of incoming versus outgoing calls, and is not necessarily equal in each direction or between different source-destination pairs. On the other hand, the quality of service that a single circuit is designed or conditioned to provide, e.g. voice grade or program grade is called the QoS [6]. Quality criteria for such circuits may include equalization for amplitude over a specified band of frequencies, or in the case of digital data transported via analogue circuits, may include equalization for phase. Criteria for mobile quality of service in cellular telephone circuits include the probability of abnormal termination of the call [4].

When a user attempts to make a telephone call, the routing equipment handling the call has to determine whether to accept the call, reroute the call to alternative equipment, or reject the call entirely [2]. Rejected calls occur because of heavy traffic loads (congestion) on the system and can result in the call either being delayed or lost. If a call is delayed, the user simply has to wait for the traffic to decrease, however if a call is lost then it is removed from the system [2]. The GoS is one aspect of the quality a customer can expect to experience when making a telephone call. In a Loss System, the GoS is described as that proportion of calls that are lost due to congestion in the busy hour. For a Lost Call system, the GoS can be measured using equation 1.

$$\boldsymbol{GoS = \frac{number\ of\ lost\ calls}{number\ of\ offered\ calls}} \quad (\mathbf{1})$$

The GoS can be measured using different sections of a network. When a call is routed from one end to another, it will pass through several exchanges. If the GoS is calculated based on the number of calls rejected by the final circuit group, then the final circuit group blocking criteria determines the Grade of Service. If the GoS is calculated based on the number of rejected calls between exchanges, then the GoS is determined by the exchange-to-exchange blocking criteria. The Gos should be calculated using both the access networks and the core networks as it is these networks that allow a user to complete an end-to-end connection [7]. Furthermore, the GoS should be calculated from the average of the busy hour traffic intensities of the 30 busiest traffic days of the year. This will cater for most scenarios, as the traffic intensity will seldom exceed the reference level. The grade of service is a measure of the ability of a user to access a trunk system during the busiest hour [1]. The busy is based upon customer demand at the busiest hour during a week, month or year.

The telecommunications provider is usually aware of the required GoS for a particular product. To achieve and maintain a given GoS, the operator must ensure that sufficient telecommunications circuits or routes are available to meet a specific level of demand [1]. It should also be kept in mind that too many circuits will create a situation where the operator is providing excess capacity which may never be used, or at the very least may be severely underutilized. This adds costs that must be borne by other parts of the network. To determine the correct number of circuits that are required, telecommunications service providers make use of traffic tables known as Erlang B table.

To determine the Grade of Service of a network when the traffic load and number of circuits are known, telecommunications network operators make use of equation 2, which is the Erlang B equation.

$$\boldsymbol{GoS} = \frac{\boldsymbol{E^M}/\boldsymbol{M}!}{\sum_{\boldsymbol{n}=\boldsymbol{0}}^{\boldsymbol{M}} \boldsymbol{E^n}/\boldsymbol{n}!} \qquad (\boldsymbol{2})$$

Where E is the number of Erlangs (Computing Erlangs requires that call frequency and call duration be in the same units of time) and M is the number of phone lines.

Extended Erlang B is used to calculate the probability that a call is dropped and is known as the GoS for the call center. The Extended Erlang B formula differs from the regular Erlang B formula in that some callers retry after their call is denied. By factoring in the rate of repeat callers, one can more accurately calculate the GoS. The retry or recall rate is a number between 0 and 1, for example, if 30% of the blocked callers hang up and call again, the retry rate is 0.3. If we are unsure how many callers try again, use a retry value of 0.5.

When the GoS is low, i.e., less than 0.05, then the call center is efficient at handling traffic. GoS values higher than 0.1 indicate that a call center does not have enough trunks to adequately handle its traffic.

To calculate GoS that is adjusted to account for a recall factor of R, the formula in equation 3 can be used.

There steps need to be followed in order to calculate GoS using extended Erlang B formula:

1. Calculate GoS using regular Erlang B formula and the known values of E and M. Call the initial Erlang value E_0.
2. Calculate successive values of the Erlang by using equation 3.
3. Repeat the recursion until the values of E_n and P_n stabilize. The stable value of P_n is the true GoS for the system with a retry rate of R.

$$\boldsymbol{E_{n+1}} = \boldsymbol{E_0} + (\boldsymbol{R} \times \boldsymbol{E_n} \times \boldsymbol{P_n}) \qquad (\boldsymbol{3})$$

Where P_n is the GoS for a traffic level of E_n Erlangs.

If a system in which calls are queued rather than dropped, then the Erlang C can be used to calculate the probability that a call is queued and the average call waiting time, also known as the Average Speed of Answer (ASA). To use the Erlang C formula, you need to know the traffic level and the number of trunks or lines that are available to take calls.

If the Erlangs of traffic is E and the number of trunks is M, then the Erlang C formula for the probability that a call is queued is given by the equation (4).

$$\boldsymbol{P_{queued}} = \frac{\boldsymbol{E^M}/\boldsymbol{M}!}{(\boldsymbol{E^M}/\boldsymbol{M}!) + (\boldsymbol{1} - \boldsymbol{E}/\boldsymbol{M}) \sum_{\boldsymbol{n}=\boldsymbol{0}}^{\boldsymbol{M}-\boldsymbol{1}} \boldsymbol{E^n}/\boldsymbol{n}!} \qquad (\boldsymbol{4})$$

The average call waiting or Average Speed of Answer is given by the equation (5).

$$ASA = \frac{T \times P_{queued}}{M - E} \quad (5)$$

Where M is the number of trunks, E is the Erlangs of traffic and T is the average duration of the call.

[2] METHODOLOGY

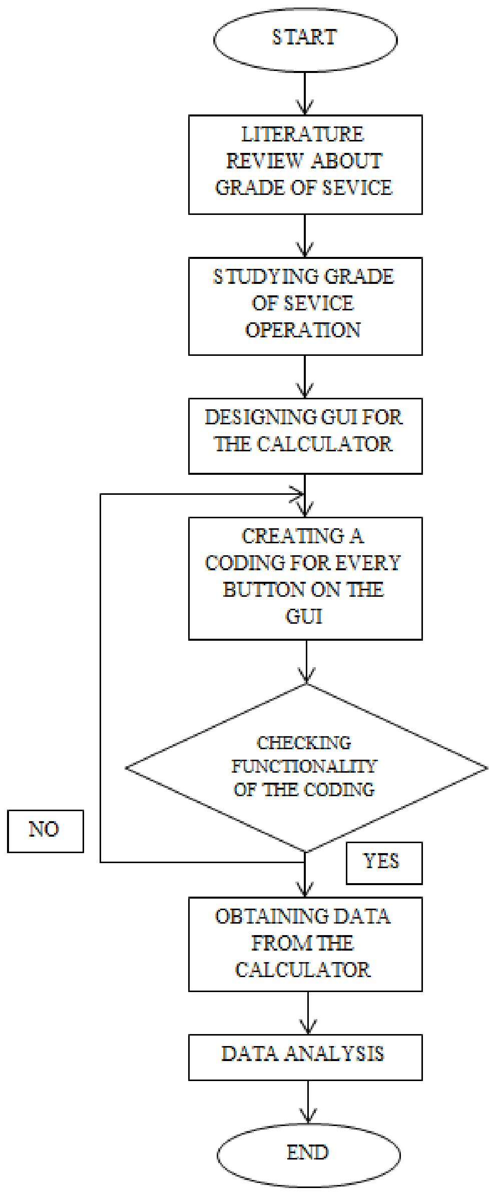

Figure 1: Flowchart of the system design

Figure 1 shows the flow of the project. The project starts from the literature review about GoS. The designing process will be done using Matlab because the software was capable to create graphical user interface (GUI) for the calculator. The next step is to enter the coding to ensure that the calculator will function perfectly. If there is an error occurred during the coding process, the problem must be detected and corrected. If there is no error, the data collecting process can be started by entering the desired values for each parameter. The last step is to analyze various values calculated using Erlang B and extended Erlang B. Various values are also calculated using Erlang C to get the probability queued and average speed of answer.

Graphical User Interface (GUI) is a program interface that takes advantage of the computer graphics capabilities to make the program easier to use. Well-designed GUI can free user from the learning complex command languages. On the other hand, many users find that they work more effectively with a command-driven interface, especially if they already know the command language.

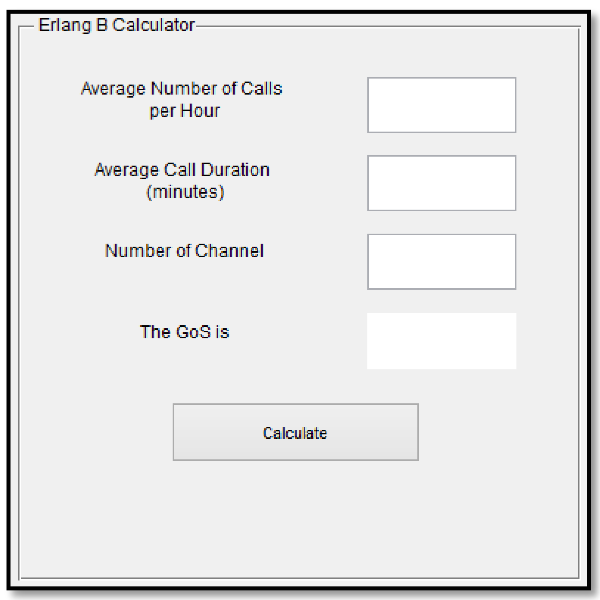

Figure 2: GUI for Erlang B calculator

Figure 2 shows the Erlang B calculator used to calculate the GoS. User can enter the desired value for average number of call per hour, average call duration and number of channel. The calculator will automatically calculate the value of GoS.

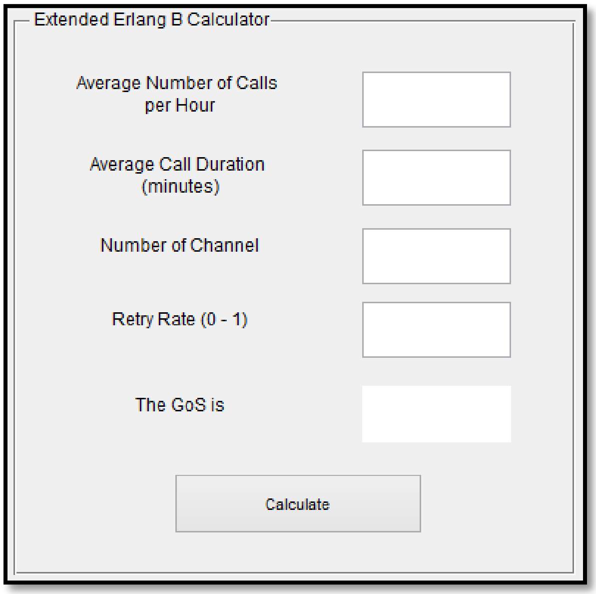

Figure 3: GUI for extended Erlang B calculator

Figure 3 shows the extended Erlang B calculator used to calculate GoS. It looks similar with Erlang B calculator but the different is at retry rate option. The user can enter the percentage of user that is making a recall after their call is denied.

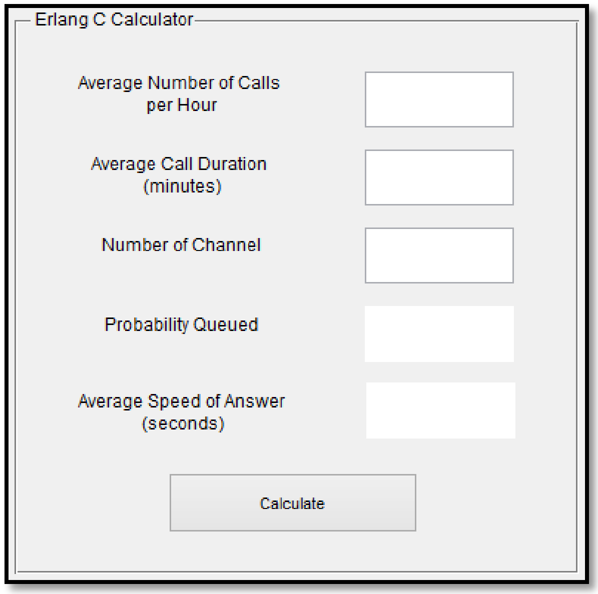

Figure 4: GUI for Erlang C calculator

Figure 4 show the Erlang C calculator used to calculate the probability queued and average speed of answer. User can enter the desired value for average number of call per hour, average call duration and number of channel.

[3] RESULT AND DISCUSSION

Table 1 shows the value of GoS, probability queued and average speed of answer when the number of channel is varied from 65 to 80. The average number of call per hour, the average call duration and retry rate is set to 120, 30 and 0.75 respectively.

Table 1: Effect of number of channel to GoS, probability queued and average speed of answer

Number of channel	Erlang B	Extended Erlang B	Erlang C	
	Blocking probability	Blocking probability	Probability Queued	Average Speed of Answer (seconds)
61	0.086498	0.122485	0.852420	1534.36
65	0.052779	0.070871	0.420072	151.226
69	0.028376	0.035240	0.182939	36.5879
71	0.019671	0.023402	0.114664	18.7632
75	0.008327	0.009161	0.040293	4.83515
79	0.002938	0.003064	0.012104	1.14665
80	0.002197	0.002273	0.008737	0.78633

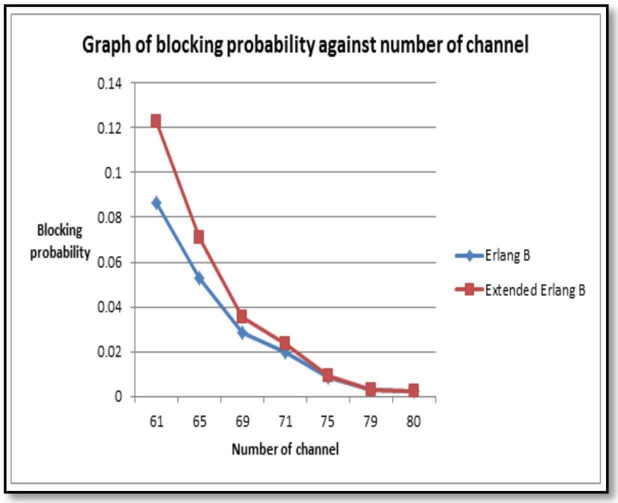

Figure 5: Experimental result of GoS correspond to the variation of number of channel

Data in table 1 shows that the GoS decrease from 0.086498 to 0.002197 for Erlang B and 0.122485 to 0.002273 for extended Erlang B as the number of channel varied from 8 to 20. The number of channel is inversely proportional to probability queued and average speed of answer. The value of probability queued decrease from 0.852420 to 0.008737 while the value of average speed of answer decrease from 1534.36 to 0.78633 seconds when the number of channel is increasing.

Graph in Figure 5 shows that there is significant different in the value of GoS calculated using Erlang B formula and extended Erlang B formula. The graph As the number of channel increase, the value of GoS will decrease. Therefore, we can conclude that the blocking probability is inversely proportional to the number of channel.

Table 2 shows the value of GoS, probability queued and average speed of answer when the average number of call per hour is varied from 139 to 159. The number of channel, the average call duration and retry rate is set to 80, 30 and 0.75 respectively.

Table 2: Effect of average number of call per hour to GoS, probability queued and average speed of answer

Average number of call per hour	Erlang B	Extended Erlang B	Erlang C	
	Blocking probability	Blocking probability	Probability Queued	Average Speed of Answer (seconds)
139	0.023136	0.028294	0.152865	26.2055
143	0.032006	0.040839	0.237336	50.2593
146	0.039668	0.052116	0.320685	82.4617
149	0.048108	0.064875	0.423671	138.656
153	0.060408	0.083877	0.595062	306.032
156	0.070282	0.099328	0.751478	676.331
159	0.080599	0.115540	0.933450	3360.42

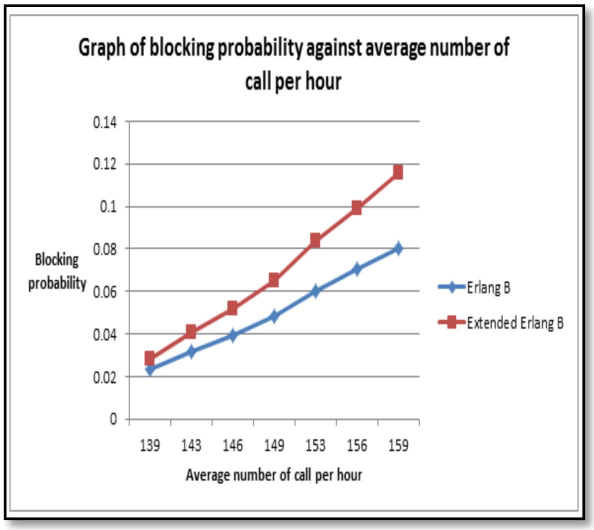

Figure 6: Experimental result of GoS correspond to the variation of average number of call per hour

Data in table 2 shows that the GoS increase from 0.023136 to 0.080599 for Erlang B and 0.028294 to 0.115540 for extended Erlang B as the average number of call per hour varied from 139 to 159. The average number of call per hour is directly proportional to probability queued and average speed of answer. The value of probability queued increase from 0.152865 to 0.933450 while the value of average speed of answer increase from 26.2055 to 3360.42 seconds when the average number of call per hour is increasing.

Graph in Figure 6 shows that there is significant different in the value of GoS calculated using Erlang B formula and extended Erlang B formula. As the average number of call per hour increase, the value of GoS also will increase. Therefore, we can conclude that the blocking probability is directly proportional to the average number of call per hour.

Table 3 shows the value of GoS, probability queued and average speed of answer when the average call duration (minutes) is varied from 27 to 39. The average number of call per hour, the number of channel and retry rate is set to 120, 80 and 0.75 respectively.

Table 3: Effect of average call duration (minutes) to GoS, probability queued and average speed of answer

Average call duration (minutes)	Erlang B	Extended Erlang B	Erlang C	
	Blocking probability	Blocking probability	Probability Queued	Average Speed of Answer (seconds)
27	0.000193	0.000193	0.000593	0.036941
29	0.001075	0.001095	0.003899	0.308372
31	0.004126	0.004364	0.018079	1.868140
33	0.011569	0.013117	0.062692	8.866470
35	0.025203	0.031158	0.171386	35.99110
37	0.045214	0.060468	0.387029	143.2010
39	0.070282	0.099328	0.751478	879.2300

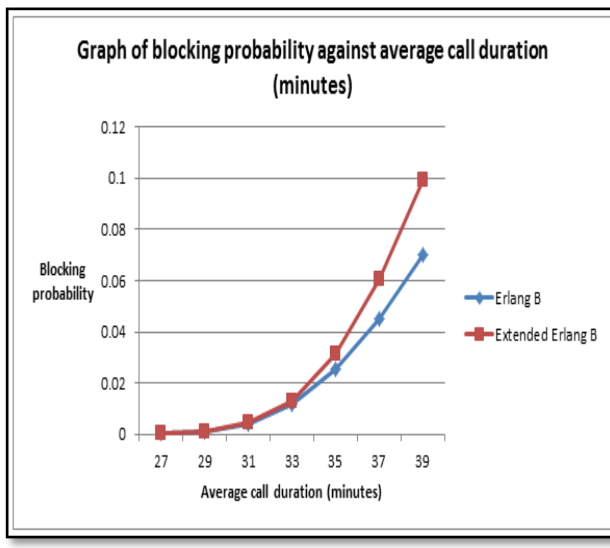

Figure 7: Experimental result of GoS correspond to the variation of average call duration (minutes)

Data in table 3 shows that the GoS increase from 0.000193 to 0.070282 for Erlang B and 0.000193 to 0.099328 for extended Erlang B as the average call duration varied from 27 to 39 minutes. The average call duration is directly proportional to probability queued and average speed of answer. The value of probability queued increase from 0.000593 to 0.751478 while the value of average speed of answer increase from 0.036941 to 879.2300 seconds when the average call duration is increasing.

Graph in Figure 7 show that there is significant different in the value of GoS calculated using Erlang B formula and extended Erlang B formula. As the average call duration increase, the value of GoS also will increase. Therefore, we can conclude that the blocking probability is directly proportional to the average call duration.

[4] CONCLUSION

The novelty of this work is that we are able to both accurately measure GoS and get to know what factor affecting the blocking probability in mobile communication system. As can be seen, the formula for Erlang B, extended Erlang B and Erlang C is very complex and by using this calculator it will reduce the probability to make a mistake during manually calculation and also time saving. This project enables us to analyze the various calculated values of GoS that we get through two different formulas. Through Erlang C calculator we can see the pattern of queued probability and average speed of answer when one of the parameter is varied and the other two is fixed.

[5] REFERENCES

[1] T L Singal "*Wireless Communications*", Tata McGraw Hill, 2011, New Delhi.

[2] T. Viswanathan "*Telecommunication switching system and networks*", Prentice-Hall, 2009, New Delhi.

[3] Roger L. Freeman "*Fundamentals of Telecommunications*", Wiley-Interscience, 2008, New Jersey.

[4] Roger L. Freeman "*Telecommunication System Engineering"*, Wiley-Interscience, 2007, New Jersey.

[5] V. Jeyasri Arokiamary "*Mobile Communications*", Technical Publications Pune, 2009, India.

[6] Piet Van Mieghem "*Performance Analysis of Communications Networks and Systems*", Cambridge University Press, 2010, New York.

[7] P. Muthu Chidambara Nathan "Wireless Communications", Prentice-Hall, 2008, New Delhi.

[8] O.Thomas Holland, Patrick Marchand *"Graphics and GUIs with MATLAB",* Dog Ear Publishing, 2002, New York.

[9] Scott T. Smith *"MATLAB: Advanced GUI Development"*, Dog Ear Publishing, 2006, Indianapolis.

[10] Antonio Siciliano *"MATLAB: Building GUIs with MATLAB"*, World Scientific Publishing, 2008, Danvers.