Quantile Approximation of the Erlang Distribution using Differential Evolution Algorithm

Hilary I. Okagbue[1] *, **Muminu O. Adamu[2]**, **Timothy A. Anake[1]**, **Ashiribo S. Wusu[3]**
[1]Department of Mathematics, Covenant University, Ota, Nigeria, *hilary.okagbue@covenantuniversity.edu.ng
[2]Department of Mathematics, University of Lagos, Akoka, Nigeria
[3]Department of Mathematics, Lagos State University, Ojo, Nigeria

ABSTRACT

Erlang distribution is a particular case of the gamma distribution and is often used in modeling queues, traffic congestion in wireless sensor networks, cell residence duration and finding the optimal queueing model to reduce the probability of blocking. The application is limited because of the unavailability of closed-form expression for the quantile (inverse cumulative distribution) function of the distribution. The problem is primarily tackled using approximation since the inversion method cannot be applied. This paper extended a six parameter quantile model earlier proposed to the Nakagami distribution to the Erlang distributions. Consequently, the established relationship between the two distributions is now extended to their quantile functions. The quantile model was used to fit the machine (R software) values with their corresponding quartiles in two ways. Firstly, artificial neural network (ANN) was used to establish that a curve fitting can be achieved. Lastly, differential evolution (DE) algorithm was used to minimize the errors obtained from the curve fitting and hence estimate the values of the six parameters of the quantile model that will ensure the best possible fit, for different values of the parameters that characterize Erlang distribution. Hence, the problem is constrained optimization in nature and the DE algorithm was able to find the different values of the parameters of the quantile model. The simulation result corroborates theoretical findings. The work is a welcome result for the quest for a universal quantile model that can be applied to different distributions.

Key words: Artificial Neural Networks, Differential Evolution, curve fitting, Quantile function, Erlang; Nakagami; queue, statistics.

1. INTRODUCTION

Erlang distribution is often used in modeling queues. Out of all the probability functions that characterize a probability distribution, only the quantile function can be used in simulation, which connotes the quantile function can be used in lieu of the probability density function. The utilization of the distribution is limited because of the unavailability of the closed-form expressions of cumulative distribution (CDF) and quantile function (QF) of the distribution.

The unavailability of the closed-form expression for the CDF and QF of the distribution primarily implies the following: the inversion method cannot be used to transform the CDF into QF [1] and the differentiation of the CDF cannot yield the PDF analytically. In this case, the QF cannot be used instead of the CDF [2] in recreating the distribution. In addition, the QF neither be recovered from the statistical moments [3-4] nor numerical inversion of the CDF [5-7].

Approximation remains the only viable option to use in obtaining a function that closely resembles the closed-form expression for the QF. Approximation in this context is the use of numerical optimization, although other methods such as functional approximation, the use of series expansions are available [8-11].

Numerical optimization is the selection of the optimum solution over several available candidate solutions in a given optimization problem. The CDF of Erlang distribution is intractable and its inversion to QF is computationally expensive and can be seen as a numerical optimization problem in general and as a constrained optimization problem in particular.

To reduce the complexity of inverting the CDF, QF available in standard software can be fitted with given curves and the error between the fitted and the machine values can be minimized using some numerical optimization methods. The curve fitting is not simple as it seems because the QF has uneven orientation at the tail areas making it difficult for curve fitting. This becomes a multi-objective optimization problem, which can be handled using evolutionary computation methods. Differential evolution (DE) is one of such methods. Evolutionary computation methods are nature-inspired numerical optimization methods that are very capable of handling multi-objective and multi-modal problems amongst other NP-hard problems [12].

This work used DE as a two-edged sword of firstly, fitting the machine values of the QF of Erlang distribution with a proposed curve and minimizing the error between the two functions. The result will be important in wireless communication because of the wide applicability of Erlang distribution in the area. Furthermore, the work becomes one of

interesting areas where differential evolution is applied in wireless networks and communications.

2. LITERATURE REVIEW

2.1 Erlang Distribution in Wireless Communication

Erlang distribution has been applied extensively in wireless communications. Traditionally, the distribution was developed to model queues. In wireless communication, some systems are modeling with Queueing systems [13]. The recurring application of Erlang distribution in this context is in the modeling of traffic congestion in wireless sensor networks [14] which is sometimes referred to as teletraffic analysis [15]. Blocking is usually encountered problem in teletraffic analysis where telephone calls are either queued or lost. Erlang distribution is often applied in finding the optimal queueing model to reduce the probability of blocking [16], calculating the blocking probability [17], thereby, leading to effective resource and call management [18]. Aside from blocking, Erlang distribution, on the other hand, has been applied in outage detection [19]. Other similar applications are in modeling waiting times in public cloud [20], managing selfish nodes in ad hoc networks [21], optimal power allocation in channel [22] and others [23]. Erlang distribution is applied in modeling the vehicular traffic [24], which are often modeled and studied as networks [25]. Headway can be modelled using the Erlang distribution [26]. The distribution was used to model cell residence duration in an embedded Markov chain model [27] and the residual time of spectrum holes [28]. In addition, the distribution was used in the analysis of rain fades in radio networks [29].

2.2 Differential Evolution in Wireless Communication

Evolutionary computation methods have been applied extensively in wireless communications. Prominent among the methods used in this context are genetic algorithm, differential evolution, ant bee colony and particle swarm optimization. The use of differential evolution is motivated by the nature of the problems arising in this area which are often modeled as multi-objective, multimodal or non-convex optimization problems, of which DE can handle effectively [30]. An example of a multi-objective problem is the case where DE was employed to optimize resource utilization, energy consumption and data transmission with security controls as constraints [31]. The multi-objective problem is known as deployment issues which include variables such as connectivity, coverage, lifetime, clustering and reliability [32], construction cost and total head loss in the network [33].

Differential evolution has been used as a node localization algorithm whose solitary role is to find the optimal position of sensors with minimal cost [34]. Coverage area maximization is one major area where DE has consistently been applied in wireless networks. The coverage problem is usually modeled as a multi-objective constrained optimization problem. The highest possible coverage is determined which satisfies the variables and constraints that predispose the model [35].

Coverage area is usually optimized simultaneously with network energy leading to minimal energy consumption and maximum coverage area [36]. DE is applied as clustering algorithm in order to ensure minimal energy consumption and network resilience and sustainability [37]. Other applications can be seen in [38-40].

3. MATERIALS AND METHODS

The general steps taken to arrive at the result are given.
a). Identify the probability distribution and explore possible models to be adapted (change the model) or adopted (apply the model unchanged).
b). Obtain the exact (machine values) of the quantile function of the distribution.
c). Use artificial neural networks to show that a curve is possible.
d). Use the adapted model (for this research) to obtain the approximate values.
e). Perform the curve fitting.
f) Use Differential Evolution (DE) to reduce the error of the curve fitting.
g). Read out the values of the parameters that provides the best fit. Note that RMSE close to zero or R square close to one indicate good fit.
The steps are discussed in details.

3.1 Identification of the Probability Distribution

Erlang distribution is considered because of the intractable and non-differentiable nature of the CDF of the distribution. Intractability connotes that the inversion method cannot be used to explicitly transform the CDF to obtain the quantile functions of the respective distributions. Non-differentiability nature of the CDF means that the PDF of the distributions cannot be obtained analytically from the differentiation of the CDF. On the other hand, the CDF cannot be recovered from the integration of the PDF.

The proposed quantile model in this paper is the same proposed for Nakagami distribution by [41]. The model will be extended to the Erlang distribution because of the link between the Nakagami and Gamma distributions of which the Erlang distribution is a particular case. The relationship between Gamma and Nakagami distributions were discussed by Huang [42]. Although, the relationship has long been in existence.

Definition
Given a random variable $X \in \text{Nakagami}(m, \Omega)$ and $Y \in \text{Gamma}(k, \lambda)$. Define $k = m$ and $\lambda = \dfrac{\Omega}{m}$, then $Y = X^2$ or $X = \sqrt{Y}$. Gamma distribution of which Erlang distribution is a particular case is the square of Nakagami distribution. Alternatively, the Nakagami distribution is the square root of the Erlang distribution.
It is on this fact that the result obtained for Nakagami

distribution in [41] can always be extended to the Erlang distribution.

3.2 Obtain the exact (machine) values of the quantile function of the distribution

The machine values of the quantile function of Erlang distribution for different ordered and distinct quartiles are obtained from the R software using the same code. Any library can be declared because the quantile function of Erlang is similar to Gamma distribution.

qgamma(p, shape parameter = k,

scale parameter = λ)

The Quantile function was obtained for p = 0.01, 0.02, …0.99.

3.3 Model Adaption

The relationship between the Nakagami and Erlang distributions is extended to their quantile approximation. Following the argument of [43-44] which are appreciably improvements over [45-46]. A six-parameter quantile model for Nakagami distribution earlier developed by [41] was adapted.

The proposed six parameters Quantile model of the Erlang distributions is given by

$$Q(p) = a_1 \tanh^{-1}(p) + a_2 \exp(p) + a_3$$
$$+ a_4 p + a_5 p^2 + a_6 p^3$$

where the parameters a_1, a_2, a_3, a_4, a_5 and a_6 are chosen to minimize the error between the machine (R software) and the approximate quantiles. The cubic terms were added because of the relationship between the quantile function and cubic splines [47-48]. Graphically, the quantile function is similar to the cubic polynomial except that the domain of the quartiles is strictly non-negative and interval bounded between 0 and 1 while the domain of the support of the cubic polynomial can assume any interval and not necessarily non-negative. Finally, quantile approximations that are polynomial are usually effective in capturing the extreme quantiles of a given distribution [48-49].

3.4 Curve Fitting

The steps outlined in the curve fitting does not guarantee a good fit until some methods are used to reduce the error inherent in the process.

The inversion method is a process used to transform the CDF of probability distributions to its inverse CDF called the Quantile function. If that is possible, then, curve fitting is not desirable. Unfortunately, there are probability distributions whose QF cannot be easily obtained and that implies that the closed-form expressions are also available. Approximations become the only alternative and several methods have been discussed in the literature. One such method is the use of numerical algorithms.

It could be noted that the early researchers in this area often apply one numerical method or algorithm. Currently, there seems to be a combination of different methods to improve the use of such methods.

Manku et al. [50] submitted that regardless of the method, a good quantile approximation method or algorithm should be standalone; the approximation should be flexible and adjustable and compute in a single pass. Also, it should provide a platform of which all the quantile functions can be obtained for different quartile and at ease, utilize little computer memory and be simple to understand and code using different programming platforms.

The nature of the quantile function is the reason why most curve estimation models failed, especially at the extreme tails of the distributions. Nonlinear model fit permits the user to define nonlinear equations that will guarantee a better quantile fit.

Nonlinear curve fit can be defined as $Q(p) = f(p)$ where $f(p)$ can be linear addition or subtraction of sum or product of mathematical functions of the quartiles; such as trigonometric, hyperbolic, exponential, and logarithmic and so on.

The initial results are often refined by using numerical or numerical optimization methods. This can be in the form of experimental nonlinear model fit or Probability distribution nonlinear model fit.

3.5 Application of Differential Evolution

Nonlinear probability fit was employed using the proposed model. The errors emanating from the curve fitting is reduced using differential evolution (DE). Details on DE as one of the robust evolutionary algorithm can be seen in [51-55].

This is the most basic form of the DE algorithm, which has been modified serially to yield different variants. The algorithm works like this:

A population of candidate solutions (called vectors) is initialized and moved around in the search space by using the mathematical formulae defined for the objective function to combine the positions of existing vectors from the population using a combination of uniform distribution, amplitude factor and crossover rate.

Algorithm 1: Basic DE Algorithm

    Initialization

    Evaluation

    do **while** (*termination criteria are met*)

        Mutation

        Crossover

        Evaluation

        Selection

    *e*nd do **while**

If the new position of the vector is an improvement, then it is accepted and promoted to form part of the population. Otherwise, the new position is simply discarded and sometimes archived. The process is repeated t times until a satisfactory solution is discovered. DE algorithm does not guarantee that an optimal (exact) solution will be found just like other evolutionary computational methods.

3.6 Root Mean Square Error

The root mean square root (RMSE) is used to calculate the extent to which the model was able to fit the exact (R software

values). It is one of the most widely used performance metrics in this context. Smaller values of RMSE values are desirable as they mean that the proposed model performs well. The RMSE is given as;

$$RMSE = \sqrt{\frac{\sum_{i=1}^{k}(R\ value - Approximate)^2}{k}}$$

RMSE values range from 0 to 1. Values close to zero are desirable as it indicates an effective fit. This can be used to assess and compare the quality of different models against a reference model.

3.7 Nature of the Problem

The problem is a constrained optimization problem.
The proposed model is a non-linear optimization problem which can be defined as:
Minimize $Q(P) : P \in S \rightarrow \Re$

Subject to $P = (0.01, 0.02, ..., 0.99) \in S$

Where S is defined by;

$g_j(P) \leq 0, \ j = 1, 2, ..., l$

which is the number of inequality constraints (to be defined later)

$h_k(P) \leq 0, \ k = 1, 2, ..., m$

which is the number of equality constraints (to be defined later)

$0.01 \leq p_i \leq 0.99 \ (i = 1, 2, ..., 99)$

The idea is to obtain a feasible solution that satisfies all the constraints (inequality and equality). Any violation of one constraint yields infeasible solution.
A solution Q(P) is feasible if the two conditions are met

$g_j(P) \leq 0, \ j = 1, 2, ..., l$

$|h_k(P)| - \xi \leq 0, \ k = 1, 2, ..., m$

The general form of the solution imply that equality are converted to equality and ξ set at the value of 0.0001. A useful measure often used in this type of problem is called the average constraint violation, which is defined as;

$$\bar{v} = \frac{\sum_{j=1}^{l} G_j(P) + \sum_{k=1}^{m} H_k(P)}{l + m}$$

Where;

$$G_j(P) = \begin{cases} g_j(P) & \text{if } g_j(P) > 0 \\ 0 & \text{if } g_j(P) \leq 0 \end{cases}$$

$$H_k(P) = \begin{cases} |h_k(P)| & \text{if } |h_k(P)| - \xi > 0 \\ 0 & \text{if } |h_k(P)| - \xi \leq 0 \end{cases}$$

A near zero value of the average constraint violation is highly desirable.
The idea of using DE is to allow newly generated solutions to be selected to influence the selection directions of the offspring in the current generation and to speed up the convergence. This is based on three necessary rules:
a). Between two feasible vectors (candidate solutions), the one with the best value of the objective function is preferred and promoted to the next generation.
b). If one vector is feasible and the other one is infeasible, the feasible one is preferred and promoted to the next generation.

c). Between two infeasible vectors, the one with the lowest sum of the constraint violation is preferred.
Besides the aforementioned rules, equality constraints were converted to inequality by using the following tolerance value, $\xi = 0.0001$.

3.8 Problem Specification

The following are the detailed specifications of the problem used in DE to obtain the results for each parameter of the distribution.
a). Dimension of the problem: 99.
b). Type of function: The objective function is a nonlinear function given as:

$$Q(p) = a_1 \tanh^{-1}(p) + a_2 \exp(p) + a_3$$
$$+ a_4 p + a_5 p^2 + a_6 p^3$$

c). Number of equality constraints: None.
d). Number of inequality constraints: Seven, they are listed as follows; $0 < p < 1$ and collectively;

$$-\infty < a_1, a_2, a_3, a_4, a_5, a_6 < \infty.$$

e). Number of active constraints: one that is the quartiles p.
f). The termination criterion/criteria is/are defined: These are defined to facilitate convergence. In this case, it is defined as;

$$Q(p_i) - Q(p_i^*) < \xi$$

The termination criterion is defined and can happen when there is no appreciable difference between the parent and the offspring for some iteration X.
Consequently, it implies that the parameters that ensure a minimum error between the exact and the approximate are obtained for some ξ. That is the feasible solution exist when;

$$a_{1i} - a_{1i}^* < \xi, a_{2i} - a_{2i}^* < \xi, ..., a_{6i} - a_{6i}^* < \xi$$

Also, the process terminates when the quartiles lies outside the given range.

4. RESULTS

4.1 Initial Model Verification using Neural Network

Neural network training was applied to precipitate the initial nature in graphic form of the machine values of the quantile function through fitting. The training done using MATLAB software revealed that a general model could be obtained. Some applications of neural network in wireless communications can be seen in [56-59].
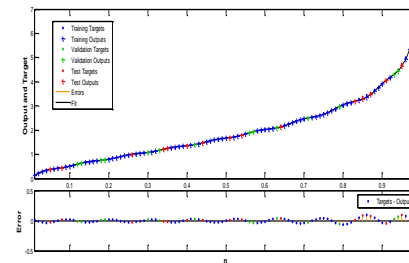

Figure 1 a: The Output versus input curve for the quantile function of both shape and scale parameter equals one.
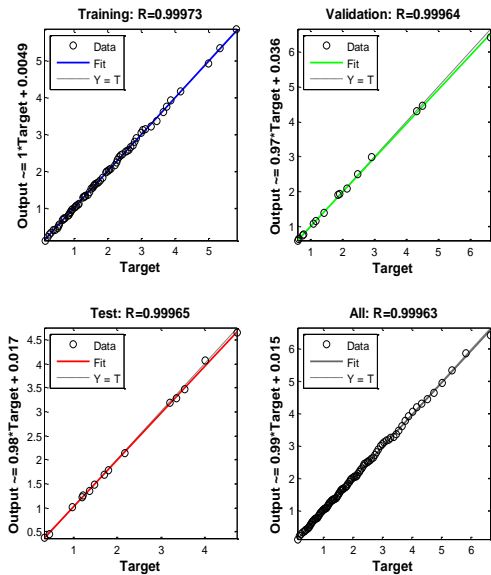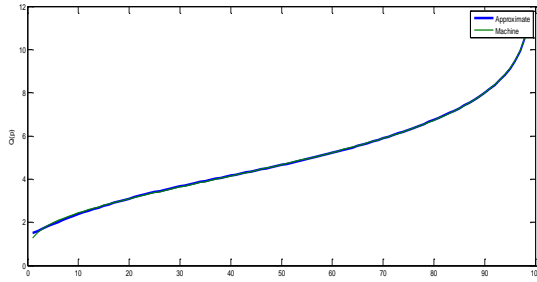
Figure 1 b: ANN Model fit summary

Figure 2: Comparison between the Machine and Approximate values for Erlang (5, 1)

The output is the machine values where the input is the quartiles. The training algorithm used is the Levenberg-Marquardt algorithm and is shown in Figures 1a and b. The nature of the curve in Figure 1 is synonymous with the quantile function of Erlang distribution. The values of R in Figure 1 b indicate an almost perfect fit.

4.2 Main Results

The machine values of the Erlang distribution were fitted with their quartile using the proposed quantile model for degrees of freedom (k), 1 to 15 and scale parameter (λ) equals one only. Other scale parameters are multiples of the scale

parameter of one. It should be noted that the distribution is supported by two parameters. The values of the six parameters and the RMSE obtained using the differential evolution to minimize the error of the quantile model are presented in Table 1. The plots that showed how the model performed against the machine values are shown in Figures 2 and 3. The quantile function of other scale parameters of the distribution for any shape parameter is multiple of one.

A close look at the figures showed that the precision reduces as the values of the shape parameter increase and hence the distance between the machine and approximate widens from Figures 2 and 3. This is corroborated with the decreasing values of RMSE in Table 1 and the result is desirable since approximations are often very difficult at low values of shape parameters of probability distributions. In addition, the near zero RMSE value obtained when the shape parameter equals to one is expected since Erlang distribution and by extension, the gamma distribution have closed form expressions there.

4.3 Simulation

Quantile function generates non-uniform variates using the uniform random variables. The quartile (p) in the Quantile model is replaced with standard uniform distribution and is used to generate non-uniform random variables with a population size of 10,000.
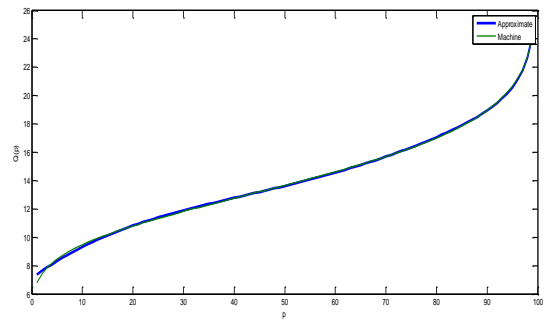

Figure 3: Comparison between the Machine and Approximate values for Erlang (14, 1)

EQI = The machine value of the lower quartile. SQI = The average of the simulated value of the lower quartile obtained using the new quantile model. EQ2 = The machine value of the median. SQ2 = The average of the simulated value of the median obtained using the new quantile model. EQ3 = The machine value of the upper quartile. SQ3 = The average of the simulated value of the upper quartile obtained using the new quantile model.

Table 1: The estimates of the parameters of the quantile model of Erlang
distribution for shape parameter equal 1 to 15 and scale parameter equals one

k	a_1	a_2	a_3	a_4	a_5	a_6	RMSE
1	1.999426	0.885764	-0.88593	-1.88007	0.016987	-0.35035	3.61E-05
2	2.359142	-34.9766	35.12954	36.67837	12.94206	10.73063	0.007409
3	2.660176	-78.6911	79.16279	82.94576	29.28917	23.89266	0.016676
4	2.921711	-120.043	120.9354	126.5696	44.87536	36.27293	0.025581
5	3.15554	-158.308	159.6875	166.8805	59.34531	47.70107	0.033883
6	3.368779	-193.829	195.7455	204.274	72.8021	58.29597	0.041621
7	3.56600	-227.032	229.5223	239.2108	85.39468	68.19114	0.048872
8	3.75032	-258.278	261.3716	272.078	97.25409	77.49773	0.055708
9	3.923967	-287.856	291.5773	303.1838	108.4867	86.30391	0.062187
10	4.088598	-315.997	320.3655	332.7724	119.1778	94.67937	0.068356
11	4.245483	-342.884	347.9175	361.0392	129.396	102.6798	0.074254
12	4.395619	-368.667	374.3807	388.1425	139.1972	110.3502	0.079913
13	4.539807	-393.469	399.8758	414.2122	148.6275	117.7274	0.085358
14	4.678703	-417.392	424.5037	439.3561	157.7253	124.8423	0.090613
15	4.812849	-440.522	448.3489	463.6649	166.5228	131.7205	0.095695

Table 2: Simulation Results for Erlang Distribution

k	λ	EQ1	SQ1	EQ2	SQ2	EQ3	SQ3
1	1	0.2876	0.2855	0.6931	0.6986	1.3862	1.3826
2	1	0.9612	0.9629	1.6783	1.6735	2.6926	2.69
3	1	1.7273	1.7349	2.674	2.677	3.9204	3.92
4	1	2.5353	2.5495	3.672	3.6734	5.1094	5.1123
5	1	3.3686	3.3874	4.6709	4.6686	6.2744	6.2779
6	1	4.2192	4.2438	5.6701	5.6663	7.4227	7.429
7	1	5.0826	5.1122	6.6696	6.6637	8.5584	8.5668
8	1	5.9561	5.9904	7.6692	7.6613	9.6844	9.6946
9	1	6.8376	6.8767	8.6689	8.6594	10.8024	10.8148
10	1	7.7258	7.7685	9.6687	9.6564	11.9138	11.9268
11	1	8.6198	8.666	10.6685	10.654	13.0196	13.034
12	1	9.5186	9.569	11.6684	11.653	14.0005	14.137
13	1	10.4217	10.476	12.6682	12.651	15.2173	15.235
14	1	11.3286	11.387	13.6681	13.65	16.3102	16.33
15	1	12.2388	12.301	14.668	14.648	17.3999	17.421

The machine and the simulated values of the estimates of the first, second and third quartiles were presented for some selected values of the distribution for the sole purpose of comparison of the extent of which the proposed quantile model was able to recreate the first, second (median) and third quartiles respectively.

The result of the comparison for some selected values of both parameters that support the Erlang distribution is presented in Table 2 and graphically in Figures 4 and 5 for some selected parameters.

It can be seen from Figures 4 and 5 that the quantile model proposed in this work was able to recreate the distribution. The distribution is heavy-tailed as seen in the values of the skewness. The decreasing value of A squared statistic connotes approximation to normality. In addition, the accuracy of the Quantile model increases with a decreasing value of the shape parameters of the respective distributions.

A further investigation of the behavior of the random variables generated using the new quantile model is required to prove that the random variables are not from the normal distribution.
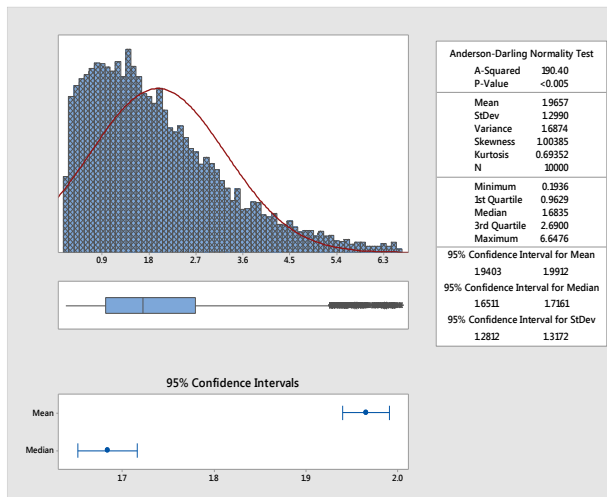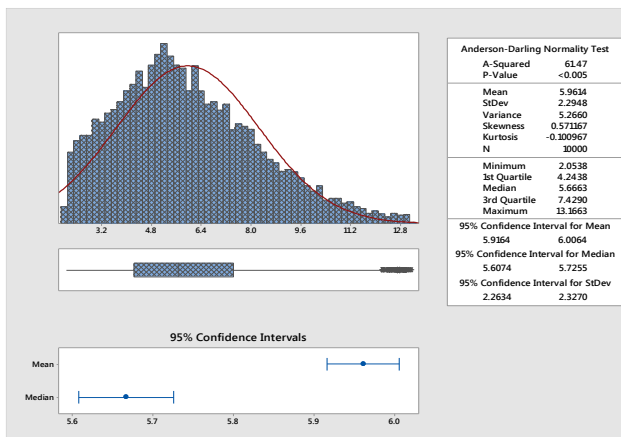
Figure 4: Simulation Result for Erlang (2, 1)

Figure 5: Simulation Result for Erlang (6, 1)

To achieve this, the values of the skewness and Kurtosis obtained from the simulation result for the two distributions are interpreted. The values are culled from Figures 4 and 5 and other selected parameters. The detailed summary is presented in Table 3.

Table 3: Summary of the Skewness and Kurtosis from the Simulated Results

k	λ	Skewness	Kurtosis
2	1	1.0000	0.6935
6	1	0.5712	-0.1010
8	1	0.4917	-0.2023
11	1	0.4162	-0.2860
15	1	0.3535	-0.3463

It can be seen that the skewness (a measure of symmetry) tends towards zero as the value of the shape parameter increases. Apart from tending to normality, the tails of the distributions point to the right. This is consistent with the properties of the distributions now recreated using the newly developed quantile model. Erlang distribution is an example of distributions with positive skewness. It can also be noticed that the Kurtosis (a measure of the nature of tails) decreases as the

value of the shape parameter increases. The increasing values of the negative Kurtosis imply that the tails of the two distributions become lighter when compared with the normal distribution as the value of the shape parameter increases. This equally corroborates theoretical findings. Although the values of distribution will continue to cluster in the center as the values of the shape parameter increases, the tails become lighter than the normal. This is an indication that the distribution is not from the normal distribution.

6. CONCLUSION

A six-parameter Quantile model developed for the Nakagami distribution has been extended to the Erlang distribution. Differential Evolution algorithm was very effective in optimization of coefficients that provide the best available fit on the machine values. The quantile model can be used as the closed form expression for the inverse cumulative function of Erlang, Gamma and Nakagami distributions. Simulation done using the model yielded estimates closed to the machine values. Different areas of wireless communications will benefit immensely from this work especially the areas of queuing and modeling fading channels. Furthermore, different evolutionary computational methods can be applied in lieu of the differential evolution [60-64].

ACKNOWLEDGEMENT

REFERENCES

1. Acklam, P. J. (2009). **An algorithm for computing the inverse normal cumulative distribution function**. http://home.online.no/~pjacklam/notes/invnorm/.
2. Nair, N.U. & Sankaran, P.G. (2009). **Quantile based reliability analysis**. *Comm. Stat. Theo. Meth*., 38, 222-232. https://doi.org/10.1080/03610920802187430
3. Karvanen, J. (2006). **Estimation of quantile mixtures via L-moments and trimmed L-moments**. *Comput. Stat. Data Analy*., 51 (2), 947-959. https://doi.org/10.1016/j.csda.2005.09.014
4. Sborshchikovi, A. (2017). **On nonparametric quantile function estimation using transformed moments**. *Bull. Georgian Nat. Acad. Sci*., 11 (3), 22-27.
5. Derflinger, G., Hörmann, W. & Leydold, J. (2010). **Random variate generation by numerical inversion when only the density is known**. *ACM Trans. Model. Comp. Simul*., 20 (4): Article 18. https://doi.org/10.1145/1842722.1842723
6. Rhein, B., Clees, T. & Ruschitzka, M. (2014). **Robustness measures and numerical approximation of the cumulative density function of response surfaces**. *Comm. Stat. Simul. Comput*., 43 (1), 5209-5220.https://doi.org/10.1080/03610918.2012.6906 37

7. Witkovský V. (2016). **Numerical inversion of a characteristic function: An alternative tool to form the probability distribution of output quantity in linear Measurement models.** *Acta IMEKO*, 5 (3), 32-44. https://doi.org/10.21014/acta_imeko.v5i3.382

8. Jiménez, F. & Jodrá, P. (2009). **On the computer generation of the Erlang and negative binomial distributions with shape parameter equal to two**. *Math. Comp. Simul*., 79 (5): 1636-1640. https://doi.org/10.1016/j.matcom.2008.07.010

9. Jodrá, P. (2012). **Computing the asymptotic expansion of the median of the Erlang distribution**. *Math. Model. Analy.*, 17 (2): 281-292. https://doi.org/10.3846/13926292.2012.664571

10. Okagbue, H.I., Adamu, M.O. & Anake, T.A. (2019). **Closed Form Expressions for the Quantile Function of the Erlang Distribution Used in Engineering Models**. *Wireless Pers. Commun*., 104(4), 1393-1408. https://doi.org/10.1007/s11277-018-6090-x

11. Okagbue, H.I., Adamu, M.O. & Anake, T.A. (2019). **Quantile mechanics: Issues arising from critical review**. *Int. J. Adv. Appl. Sci.*, 6(1), 9-23.

12. Salmon, H.M., De Farias, C.M., Loureiro, P., Pirmez, L., Rossetto, S., De A. Rodrigues, P.H., Pirmez, R., Delicato, F.C. & Da Costa Carmo, L.F.R. (2013). **Intrusion detection system for wireless sensor networks using danger theory immune-inspired techniques**. *Int. J. Wirel. Info. Netw*., 20(1), 39-66.

13. Lazov, I. (2016). **A methodology for information and capacity analysis of broadband wireless access systems**. *Telecom. Syst.*, 63(2), 127-139. https://doi.org/10.1007/s11235-015-0104-8

14. Gundalaswathi & Saravanan, R. (2014). **Improve the routing performance in wireless networks to avoid congestion using queuing model**. *Int. J. Appl. Engine. Res.*, 9(7), 825-833.

15. Kadri, T. & Smaili, K. (2015). **Convolutions of hyper-erlang and of erlang distributions**. *Int. J. Pure Appl. Math.*, 98(1), 81-98.

16. Jonckheere, M. & Prabhu, B.J. (2018). **Asymptotics of insensitive load balancing and blocking phases**. *Queueing Syst.*, 88(3-4), 243-278.

17. Samouylov, K., Gudkova, I., Markova, E. & Dzantiev, I. (2016). **On analyzing the blocking probability of M2M transmissions for a CQI-based RRM scheme model in 3GPP LTE**. *Comm. Computer Info. Sci.*, 638, 327-340. https://doi.org/10.1007/978-3-319-44615-8_29

18. Głąbowski, M. & Sobieraj, M. (2017). **Analytical modelling of multiservice switching networks with multiservice sources and resource management mechanisms**. *Telecom. Syst.*, 66(3), 559-578.

19. Bilen, T., Ayvaz, K. & Canberk, B. (2018). **QoS-based distributed flow management in Software Defined Ultra-Dense Networks**. *Ad Hoc Netw*., 79, 105-111. https://doi.org/10.1016/j.adhoc.2018.06.002

20. Santhi, K. & Saravanan, R. (2019). **Performance analysis of cloud computing using series of queues with Erlang service**. *Int. J. Internet Tech. Secured Trans.*, 9(1-2), 147-162.

21. Janakiraman, S. & Rajendiran, M. (2016). **An erlang factor-based conditional reliability mechanism for enforcing co-operation in MANETs**. *Serb. J. Elect. Engine.*, 13(2), 265-284. https://doi.org/10.2298/SJEE1602265J

22. Li, Z. & Liang, Q. (2016). **Capacity optimization in heterogeneous home area networks with application to smart grid**. *IEEE Trans. Vehic. Technol.*, 65(2), Art. No. 7038209, 699-706.

23. Głąbowski, M., Hanczewski, S. & Stasiak, M. (2016). **Modelling load balancing mechanisms in self-optimising 4G mobile networks with elastic and adaptive traffic.** *IEICE Trans. Comm.*, E99B (8), 1718-1726.

24. Bhadra, S. & Kundu, A. (2016). **Priority-based classification: An automated traffic approach**. *Adv. Intel. Syst. Comput.*, 379, 35-46. https://doi.org/10.1007/978-81-322-2517-1_5

25. Chen, S.-M., Zhao, H.-T., Zhu, H.-B. & Zhang, H. (2017). **Connectivity modeling for vehicular communication networks**. *J. Appl. Sci.*, 35(1), 63-70.

26. Keller, M. & Karl, H. (2017). **Response Time Optimized Service Deployment: MILP Formulations of Piece-Wise Linear Functions Approximating Bivariate Mixed-Integer Functions**. *IEEE Trans. Netw. Service Manag.*, 14(1), Art. No. 7572217, 121-135.

27. Liu, P., Liu, Y., Ge, L. & Chen, C. (2015). **An embedded Markov chain modeling method for movement-based location update scheme**. *J. Commun.*, 10(7), 512-519. https://doi.org/10.12720/jcm.10.7.512-519

28. Agarwal, P., Thomas, D.E. & Kumar, A. (2017). **Security Analysis of LTE/SAE Networks under De-synchronization Attack for Hyper-Erlang Distributed Residence Time**. *IEEE Comm. Lett.*, 21(5), Art. No. 7820073, 1055-1058.

29. Alonge, A.A. & Afullo, T.J. (2015). **Fractal analysis of rainfall event duration for microwave and millimetre networks: Rain queueing theory approach**. *IET Microwaves, Ante. and Prop.*, 9(4), 291-300.

30. Liu, H. (2019). **SINR-based multi-channel power schedule under DoS attacks: A Stackelberg game approach with incomplete information**. *Automatica*, 100, 274-280.

31. Xu, Z., Gu, R., Huang, T., Xiang, H., Zhang, X., Qi, L. & Xu, X. (2018). **An IoT-oriented offloading method with privacy preservation for cloudlet-enabled wireless metropolitan area networks**. *Sensors*, 18(9), Art. No. 3030. https://doi.org/10.3390/s18093030

32. Cao, B., Kang, X., Zhao, J., Yang, P., Lv, Z. & Liu, X. (2018). **Differential Evolution-Based 3-D Directional Wireless Sensor Network Deployment Optimization.** *IEEE Internet of Things J*., 5(5), Art. No. 8279426, 3594-3605.

33. Wisittipanich, W. & Buakum, D. (2019). **Optimal design of pipe diameter in water distribution system by multi-objective differential evolution algorithm: A**

case study of small town in Chiang Mai. *Lecture Notes in Elect. Engine.*, 513, 351-360.

34. Qin, M. & Zhu, R. (2018). **A Monte Carlo localization method based on differential evolution optimization applied into economic forecasting in mobile wireless sensor networks**. *Eurasip J. Wireless Commun. Netw.*, 2018(1), Art. No. 32.
https://doi.org/10.1186/s13638-018-1037-1

35. Xu, Y., Ye, Y., Zhang, H., Zhang, W. & Lv, Y. (2018). **A fast two-objective differential evolution for the two-objective coverage problem of WSNs**. *Memetic Computing*, DOI: 10.1007/s12293-018-0264-7.

36. Céspedes-Mota, A., Castañón, G., Martínez-Herrera, A.F., Cárdenas-Barrón, L.E. & Sarmiento, A.M. (2018). **Differential evolution algorithm applied to wireless sensor distribution on different geometric shapes with area and energy optimization**. *J. Network Comp. Appl.*, 119, 14-23.

37. Potthuri, S., Shankar, T. & Rajesh, A. (2018). **Lifetime Improvement in Wireless Sensor Networks using Hybrid Differential Evolution and Simulated Annealing (DESA)**. *Ain Shams Engine. J.*, 9(4), 655-663. https://doi.org/10.1016/j.asej.2016.03.004

38. Chen, W., Zhao, H., Li, T. & Liu, Y. (2018). **Optimal probabilistic encryption for distributed detection in wireless sensor networks based on immune differential evolution algorithm**. *Wireless Netw.*, 24(7), 2497-2507. https://doi.org/10.1007/s11276-017-1484-3

39. Baumgartner, P., Bauernfeind, T., Biro, O., Magele, C., Renhart, W. & Torchio, R. (2018). **Synthesis of NFC antenna structure under multi-card condition**. *Appl. Comput. Electromag. Soc. J.*, 33(10), 1161-1163.

40. Okagbue, H.I., Adamu, M.O. & Anake, T.A. (2019). **Differential evolution in wireless communications: A review**. *Int. J. Online Biomed. Engine.*, 15(11), 29-52.

41. Okagbue, H.I., Adamu, M.O., Anake, T.A. & Wusu, A.S. (2019). **Nature Inspired Quantile estimates of the Nakagami distribution**. *Telecom. Syst.*, 72(4), 517-541. https://doi.org/10.1007/s11235-019-00584-6

42. Huang, L. F. (2016). **The Nakagami and its related distributions**. *WSEAS Trans. Math.*, 15, 477-485.

43. Kabalci, Y. (2016). **On the Nakagami-m Inverse Cumulative Distribution Function: Closed-Form Expression and Its Optimization by Backtracking Search Optimization Algorithm**. *Wireless Pers. Commun.*, 91(1), 1-8.

44. Kabalci, Y. (2018). **An improved approximation for the Nakagami-m inverse CDF using artificial bee colony optimization**. *Wireless Netw.*, 24(2), 663-669.

45. Beaulieu, N.C. & Cheng, C. (2005). **Efficient Nakagami-m fading channel simulation**. *IEEE Trans. Vehic. Technol.*, 54(2), 413-424.

46. Bilim, M. & Develi, I. (2015). **A new Nakagami-m inverse CDF approximation based on the use of genetic algorithm**. *Wireless Pers. Commun.*, 83(3), 2279-2287.

47. He, X. & Ng, P. (1999). **Quantile splines with several covariates**. *J. Stat. Plan. Infer.*, 75(2), 343-352. https://doi.org/10.1016/S0378-3758(98)00153-0

48. Leydold, J. & Hormann, W. (2011). **Generating generalized inverse Gaussian random variates by fast inversion**. *Comput. Stat. Data Analy.*, 55 (1), 213-217.

49. Tsai, A.C., Liou, M., Simak, M. & Cheng, P.E. (2017). **On hyperbolic transformations to normality**. *Comput. Stat. Data Analy.*, 115, 250-266.

50. Manku, G.S., Rajagopalan, S. & Lindsay, B.G. (1998). **Approximate medians and other quantiles in one pass and with limited memory**. In *ACM SIGMOD Record*, 27 (2), 426-435. https://doi.org/10.1145/276305.276342

51. Storn, R. & Price, K. (1997). **Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces**. *J. Global Optimization* 11, 341-359.

52. Storn, R. (1996). **On the usage of differential evolution for function optimization**. Biennial Conference of the North American Fuzzy Information Processing Society (NAFIPS). pp. 519–523.

53. Das, S. & Suganthan, P.N. (2011). **Differential evolution: A survey of the state-of-the-art**. *IEEE Trans. Evol. Comput.*, 15 (1), 4-31.

54. Das, S., Abraham, A., Chakraborty, U.K. & Konar, A. (2009). **Differential evolution using a neighborhood based mutation operator**. *IEEE Trans. Evol. Comput.*, 13(3): 526–553.
https://doi.org/10.1109/TEVC.2008.2009457

55. Das, S., Konar, A. & Chakraborty, U. (2005). **Improved differential evolution algorithms for handling noisy optimization problems**. In the Proceedings of IEEE Congress on Evol. Comput., 2: 1691-1698.

56. Wan, Q., Weng, M.-J. & Liu, S. (2019). **Optimization of wireless sensor networks based on differential evolution algorithm**. *Int. J. Online Biomed. Engine.*, 15(1), 183-195.

57. Hamzaoui, Y., Choukri, A., Amnai, M. & Fakhri, Y. (2019). **Classification algorithm based on nodes similarity for MANETs**. *Int. J. Online Biomed. Engine.*, 15(5), 86-100. https://doi.org/10.3991/ijoe.v15i05.9742

58. Africa, A.D.M., Asuncion, F.X., Tiberio, J.L. & Munchua, R.M.F.A. (2019). **Sensor-based traffic control network with neural network based control system**. *International Journal of Advanced Trends in Computer Science and Engineering*, 8(4), 983-989. https://doi.org/10.30534/ijatcse/2019/01842019

59. Kollu, P.K. & Prasad, R.S. (2019). **Effective intrusion detection using deeper recurrent neural networks**. *International Journal of Advanced Trends in Computer Science and Engineering*, 8(4), 1134-1138. https://doi.org/10.30534/ijatcse/2019/21842019.

60. Halim, S.A., Rosli, H.M. & Hasri, H.F. (2019). **Moth-flame optimization algorithm with different course for optimal photovoltaic location and sizing**. *International Journal of Advanced Trends in Computer Science and Engineering*, 8(1), 145-152. https://doi.org/10.30534/ijatcse/2019/2381.62019

61. Ayoppan, A.H.A.H., Ibrahim, M.F. & Mohd Zaman, M.H. (2019). **A genetic algorithm with online learning approach for improving loop closure detection of a visual SLAM**. *International Journal of Advanced Trends in Computer Science and Engineering*, 8(1), 159-166. https://doi.org/10.30534/ijatcse/2019/2581.62019

62. Kamari, N.A.M., Rahmat, N.A. & Musirin, I. (2019). **Optimal power scheduling strategy in power systems using swarm optimization technique**. *International Journal of Advanced Trends in Computer Science and Engineering*, 8(1.6), 246-251. https://doi.org/10.30534/ijatcse/2019/3781.62019

63. Mohd Din, N.A., Salimin, R.H., Musirin, I., Honnoon, N.M.S. & Aminuddin, N. (2019). **Solving economic load dispatch for power generation using genetic algorithm techniques**. *International Journal of Advanced Trends in Computer Science and Engineering*, 8(1), 337-344. https://doi.org/10.30534/ijatcse/2019/6181.32019

64. Soni, A. & Chande, S.V. (2019). **Impact of varying parameters in improved multi colony ant optimization algorithm for join order problem in distributed databases**. *International Journal of Advanced Trends in Computer Science and Engineering*, 8(5), 2442-2448. https://doi.org/10.30534/ijatcse/2019/87852019