Quadrature-Based Exponential-Type Approximations for the Gaussian $Q$-Function

Islam M. Tanash and Taneli Riihonen
Faculty of Information Technology and Communication Sciences, Tampere University, Finland

Abstract—In this paper, we present a comprehensive overview of (perhaps) all possible approximations resulting from applying the most common numerical integration techniques on the Gaussian $Q$-function. We also present a unified method to optimize the coefficients of the resulting exponential approximation for any number of exponentials and using any numerical quadrature rule to produce tighter approximations. Two new tight approximations are provided as examples by implementing the Legendre numerical rule with Quasi-Newton method for two and three exponential terms. The performance of the different numerical integration techniques is evaluated and compared, and the accuracy of the optimized ones is verified for the whole argument-range of interest and in terms of the chosen optimization criterion.

I. INTRODUCTION

The Gaussian $Q$-function $Q(x)$ and the related complementary error function $\text{erfc}(x)$ represent significant value in the performance analysis of different communication systems, where noise or interference is typically modeled as a Gaussian random variable, such as evaluating error probabilities. As the Gaussian $Q$-function does not have an exact closed-form expression, several approximations and bounds have been proposed in the literature to facilitate its applications, especially when complicated integrals involving it are encountered.

The authors in [1] propose tight approximations for the $Q$-function using rational Chebyshev functions. In [2], the authors use integration by parts to derive new bounds and take their geometric mean to yield an approximation; another complex approximation with two controllable parameters that define the level of accuracy is also derived. A relatively accurate approximation for the $Q$-function and integer powers thereof are provided in [3], [4]. Based on [3], an upper bound is later developed in [5] and the approximation of the first power is modified in [6] using Taylor series expansion to get a simpler form. An infinite series expression for $\text{erfc}(x)$ that is more accurate for large values of $x$ is derived in [7].

The authors in [8] present an accurate polynomial approximation. In [9], a relatively complicated new family of bounds is proposed using Jensen’s inequality with the Cotes trapezoidal integration rule and convex–concave partitions of the integrand of the Craig’s formula:

$$Q(x) \triangleq \frac{1}{\pi} \int_0^\infty \exp \left( -\frac{1}{2\sin^2 \theta} x^2 \right) \, d\theta \quad [\text{for } x \geq 0].$$

Moreover, tight bounds are presented in [10] as a sum of two exponentials with respective constant and rational factors. The authors in [11] propose a simple and accurate mathematical expression as an exponential function with a polynomial argument of the second degree. In addition, [12] develops an efficient approximation based on a semi-infinite Gauss–Hermite quadrature rule that results in a finite sum of exponential functions.

Chiani et al. in [13] propose a simple family in the form of exponential sums, in which they apply the trapezoidal integration rule with optimizing the center point to minimize the integral of relative error in the range of interest. This family was later generalized in [14] to be applied to polynomials and integer powers of the $Q$-function, or even to generic functions thereof using an original minimax methodology. Other approximations and bounds of this form are presented in [15]–[19]. In [15], the composite trapezoidal rule is used with an optimized number of sub-intervals, whereas in [16], a coarse single-term approximation from the classic Chernoff bound is presented. Since the $Q$-function can be well approximated as an infinite sum of exponentials, [17] presents the Prony approximation. An invertible exponential approximation is presented in [18] and a single-term exponential lower bound is introduced in [19] by upper-bounding the logarithmic function.

The approximations and bounds in [1]–[12] have relatively complex mathematical forms, which makes them inconvenient for algebraic manipulations in statistical performance analysis despite being accurate. On the other hand, those with the exponential form [13], [15]–[19] are more suitable to be used due to their analytical tractability. However, they still provide an inadequate accuracy for some of the argument range of interest, i.e., some of them are suitable only for some certain range. This leads us toward the various numerical integration methods [20], i.e., quadrature rules, which can be implemented to approximate the Craig’s form of the $Q$-function (1) to obtain a flexible approximations and bounds of the same form as in [13, Eq. (8)] with numerical coefficients instead. Based on the accuracy level required in a given range, the suitable numerical quadrature rule is selected to approximate the $Q$-function.

The goal of this paper is to present an overview of all the known numerical integration techniques that are commonly used to approximate the Gaussian $Q$-function and compare their performance. In particular, we consider Newton–Cotes formulas, Gaussian quadrature formulas and the composite integration rules. In addition, we generalize the work of Chiani et al. [13] from only $N = 2$ exponential terms to any $N$ and using any numerical integration technique by minimizing optimization criteria for composite intervals thereof. In terms of explicit expressions, we provide tight exponential approximations using the Legendre rule for quick reference.
II. APPROXIMATIONS FROM NUMERICAL INTEGRATION

The quadrature integration techniques that can be used to approximate the Q-function (viz. \( Q(x) \approx Q(x) \)) are categorized herein as Newton–Cotes formulas and Gaussian quadrature formulas. Due to the instability of higher-order numerical methods (especially with the Newton–Cotes rules, which have negative weights that can result in subtractive cancellation), the composite integration rules are also considered in this paper.

In general, any integral of the form \( \int_{a}^{b} W(\theta) f(\theta) d\theta \), where \( W(\theta) \) is some weighting function and \([a, b]\) is the domain of integration, can be rewritten as a finite sum of the form [20]

\[
\int_{a}^{b} W(\theta) f(\theta) d\theta = \sum_{g=1}^{G} w_g f(\theta_g) + D(\xi), \quad a < \xi < b,
\]

where \( D(\xi) \) is the resulting error term, \( \{\theta_g\}_{g=1}^{G} \) are the nodes and \( \{w_g\}_{g=1}^{G} \) are the quadrature weights. Thus, the \( Q \)-function that is defined by (1) over the interval \([0, \pi/2]\) can be numerically approximated after applying (2) as

\[
Q(x) \approx \tilde{Q}(x) \triangleq \sum_{g=1}^{G} a_g \exp(-b_g x^2)
\]

such that \( Q(x) = \tilde{Q}(x) + D(\xi, x) \) for some \( a < \xi < b \), where \( x \geq 0 \) and \( \{\{a_g, b_g\}_{g=1}^{G}\} \) is the set of numerical coefficients, which depends on the specific applied numerical integration technique as will be explained below.

A. Newton–Cotes Numerical Integration

A Newton–Cotes formula can be either closed or open, depending on whether it uses the function values at the endpoints or not. The weights for Newton–Cotes rules are derived from Lagrange basis polynomials as

\[
w_g = \int_{u}^{v} \prod_{\substack{\theta \neq \theta_g \neq \xi \neq \theta_t}}^{G} \frac{\theta - \theta_g}{\theta - \theta_t} \, d\theta = c_g \Delta \theta,
\]

where \( c_g, g = 1, 2, \ldots, G \), are constants that depend on the type of the applied Newton–Cotes rule and can be found in many mathematical books, e.g., [21], whereas \( \Delta \theta \) is the step size. For the Newton–Cotes rule, the nodes are always chosen uniformly in the integration interval. Therefore, when applied to the Gaussian Q-function, the numerical coefficients of the exponential summation in (3) are calculated as

\[
\{\{a_g, b_g\}_{g=1}^{G}\} = \left\{ \left( \frac{c_g \Delta \theta}{\pi} \right)^G \prod_{g=1}^{G} \frac{1}{2 \sin^2((g-1)\Delta \theta)} \right\} \quad \text{for closed types,}
\]

\[
\{\{a_g, b_g\}_{g=1}^{G}\} = \left\{ \left( \frac{c_g \Delta \theta}{\pi} \right)^G \prod_{g=1}^{G} \frac{1}{2 \sin^2(g \Delta \theta)} \right\} \quad \text{for open types,}
\]

where \( \Delta \theta = \frac{\pi}{2(G-1)} \) and \( \Delta \theta = \frac{\pi}{2(G+1)} \) for the closed and open types, respectively.

The error term for the Newton–Cotes rules is

\[
D(\xi, x) = \frac{f^{(G)}(\xi, x)}{G!} \int_{0}^{\pi/2} \prod_{g=1}^{G} (\theta - \theta_g) \, d\theta,
\]

where \( 0 < \xi < \pi/2, x \geq 0 \), which shows that there exists some (unknown) point \( \xi \in (0, \pi/2) \) for each \( x \), for which the respective error has exactly the displayed form.

B. Gaussian Quadrature Numerical Integration

Another type of numerical integration techniques is the Gaussian quadrature family. For this type, the domain of integration in (2) is \([-1, 1]\), but since the \( Q \)-function is defined over \([u, v] = [0, \pi/2]\) for Craig’s formula, a change of variables is needed. This results in multiplying the weights by \( \frac{u - v}{\theta_g - \theta_t} \) and transforming the nodes as \( \frac{u - v}{\theta_g + \theta_t} \), thus, the numerical coefficients of the exponential sum in (3), when applying Gaussian quadrature numerical rules, are

\[
\{\{a_g, b_g\}_{g=1}^{G}\} = \left\{ \left( \frac{1}{4} w_g ; \frac{1}{2} \sin^2 \left( \frac{\pi}{4} \theta_g + \frac{\pi}{4} \right) \right) \right\} \quad \text{for closed types,}
\]

\[
\{\{a_g, b_g\}_{g=1}^{G}\} = \left\{ \left( \frac{1}{4} w_g ; \frac{1}{2} \sin^2 \left( \frac{\pi}{4} \theta_g + \frac{\pi}{4} \right) \right) \right\} \quad \text{for open types,}
\]

Five Gaussian rules are used herein for comparison purposes, namely Legendre, Chebyshev first and second kinds, Radau’s, and Lobatto’s rules. For the Chebyshev first and second kind rules, we should consider their weighting functions which result in \( a_g = \frac{1}{4} w_g \sqrt{(1 - \theta_g^2)} \) and \( a_g = \frac{1}{4} w_g \sqrt{(1 - \theta_g^2)} \), respectively. Table I summarizes the expressions for finding \( w_g, \theta_g \) and \( D(\xi, x) \) of these five Gaussian rules [21] while \( \phi_G(\theta) \) in the table is the Legendre polynomial of degree \( G \).

C. Composite Integration

The composite quadrature rules are preferred to approximate the \( Q \)-function for higher orders due to the oscillatory nature of high-degree polynomials in non-composite rules. The integration interval, \([u, v] = [0, \pi/2]\), can be divided into \( M \) smaller uniform or non-uniform sub-intervals, \([u_m, v_m], m = 1, 2, \ldots, M \), and simpler \( K \)-point integration rule is used for each sub-interval, where \( K \) is the number of nodes in each sub-interval. Therefore, the Gaussian \( Q \)-function is approximated by applying any composite integration rule as

\[
\tilde{Q}(x) = \sum_{m=1}^{M} \sum_{k=1}^{K} a_{m,k} \exp(-b_{m,k} x^2).
\]
where \( w_k \) and \( \theta_k \) are the same weights and nodes as illustrated in Table I. When considering equal-spaced intervals, the numerical coefficients become

\[
\{(a_{m,k}, b_{m,k})\}_{k=1}^{K} = \left\{ \left( w_k, \frac{2}{4M} 2 \sin^2 \left( \frac{\pi}{4M} \theta_k + \frac{(2m-1)\pi}{4M} \right) \right) \right\}_{k=1}^{K}.
\]

As a remark, one should note that the single-point Legendre rule is mathematically the same as the single-point open Newton–Cotes (a.k.a. rectangular) rule, and the two-point Lobatto’s rule is the same as the two-point closed Newton–Cotes (a.k.a. trapezoidal) rule. In addition, the weighting functions of Chebyshev rules should be also considered here when calculating the numerical coefficients as explained above.

### D. Implementation Aspects and Numerical Results

Accuracy comparison between the different composite and non-composite numerical integration techniques is presented herein for the same number of non-zero exponential terms which we refer to as \( N \). In general, if the left endpoint of the integration domain is used in the summation in (3) as a node, i.e., \( \theta_1 = 0 \), to evaluate the integration then the function’s value at that node is equal to zero, i.e., integrand in (1) evaluates to zero, and hence the first term in the summation is neglected. In this case, in order to establish exactly \( N \) exponential terms in the summation, \( G = N+1 \), since the first term in the \((N+1)\)-term summation is zero and hence the total number of exponential terms is \( N \). On the other hand, if the left endpoint is not included in the summation, then \( G = N \).

In plain words, \( G \) refers to the total number of terms including zero if any, and \( N \) refers to that of the non-zero terms.

The same applies to the composite rules for which we want to construct an \( N \)-term expression in (7). Hence, \( N = MK \) for the composite open Newton–Cotes, Legendre and Chebyshev first and second type rules, while \( N = M \) \((K-1)\), for the composite closed Newton–Cotes and Lobatto’s rules since the two endpoints of each sub-interval are included, which results in adding the last term of the \( n \)-th sub-interval to the first term of the \((m+1)\)th sub-interval to produce a single term (in addition to the fact that the first term of the first sub-interval is neglected since its \( b_{1,1} \to \infty \)). This is also the reason why for the composite Radu’s numerical rule \( N = MK - 1 \).

In Fig. 1, we illustrate the absolute relative error resulted from applying all the different types of numerical integration rules and their composites to approximate the Gaussian

### III. Optimized Numerical Approximations

The seminal work in [13] has shown that the traditional trapezoidal rule can be optimized with respect to the integral of the relative error in the range of values of interest. This yields a tighter approximation with respect to the optimization criterion than the un-optimized one. This approach has also been implemented in [22] to propose a new sum for the trapezoidal approximation of three exponentials. In this section, we present a unified method to optimize any composite numerical rule with any \( N \) and in respect to any optimization criterion.

The composite rule is based on dividing the integration interval into smaller uniform or non-uniform sub-intervals. In the previous section, we give explicit expressions for the numerical coefficients of the uniform composite rules. On the other hand, for the non-uniform case, the integration interval can be arbitrarily partitioned and general expression

![Fig. 1. Comparison among all the different numerical integration techniques for \( N = 3 \). The abbreviation N–C refers to Newton–Cotes rule.](image-url)
are presented for their corresponding coefficients. However, the division can actually be chosen optimally in order to increase the approximation’s accuracy, e.g., like in [13].

We can conclude from the relations (8) and (9) that the optimized set of coefficients, \(\{\alpha_{m,k}, \beta_{m,k}\}_{k=1}^{K}\), are obtained by optimizing the sub-intervals boundaries, \([u_m, v_m]\), \(m = 1, 2, ..., M\), according to the chosen optimization criterion with keeping in mind that \(v_m = u_{m+1}\), \(m = 1, 2, ..., M - 1\). Thus, the whole integration range has \(M + 1\) boundary points in which for the Gaussian \(Q\)-function, \(u_m = 0\) and \(v_M = \pi/2\). This will give a total of \(M - 1\) boundary points to be optimized.

Any optimization criterion could be selected for calculating the approximation’s corresponding optimized numerical coefficients. For consistency with [13], we consider the integral of the relative error in the range of values of interest \([0, R]\)

\[
F(v) = \frac{1}{R} \int_0^R \frac{\tilde{Q}(x) - Q(x)}{Q(x)} \, dx,
\]

and the numerical approximation is optimized as

\[
v^* = \arg \min_v F(v),
\]

where \(v = [v_1, v_2, \ldots, v_{M-1}]\) is the vector of unknowns (i.e., the boundary points to optimize), \(Q(x)\) is defined in (1) and \(\tilde{Q}(x)\) with the corresponding expressions of the numerical coefficients is defined in (7), (8) and (9), respectively. It should be noted that when using two-point closed Newton–Cotes rule, i.e., the trapezoidal rule, with \(M = 2\), (7) will become [13, Eq. (12)] but in terms of the \(Q\)-function instead of \(erfc(\cdot)\).

We solve this optimization problem for all values of \(M\) and \(K\) and for an arbitrary integration rule by applying the Quasi-Newton method, which is an iterative technique for finding the roots of a given differentiable function. It can also be used in the context of optimization by applying it on the derivative of the target function which is differentiable twice. This will yield the optimized roots of the function’s derivative.

In particular, we implement the Quasi-Newton optimization method herein to minimize the target function \(F(v)\) in (10). We start with some initial guesses for the \(M - 1\) unknowns that converge eventually to the optimized values, which give the minimum possible value for the target function. The iteration process is performed as

\[
v^{(t+1)} = v^{(t)} - \gamma \left[\tilde{H}^{(t)}\right]^{-1} \tilde{J}^{(t)} \left(v^{(t)}\right),
\]

where \(t\) is the iteration counter, \(0 < \gamma \leq 1\) is the iteration step size, \(\tilde{J}(\cdot)\) is the gradient vector calculated as \(\tilde{J}(v) = \left[\frac{\partial F(v)}{\partial v_1}, \frac{\partial F(v)}{\partial v_2}, \ldots, \frac{\partial F(v)}{\partial v_{M-1}}\right]\), and \(\tilde{H}\) is an approximation to the Hessian matrix. Among the various methods developed to calculate \(\tilde{H}\), we implement the well-known Broyden–Fletcher–Goldfarb–Shanno (BFGS) method which starts from some symmetric positive-definite matrix \(\tilde{H}^{(0)}\) that is updated in the sequential iterations as

\[
\tilde{H}^{(t+1)} = \tilde{H}^{(t)} + \frac{\Delta J^{(t)} [\Delta J^{(t)}]^T}{\Delta v^{(t)} [\Delta J^{(t)}]^T} \tilde{H}^{(t)} \Delta v^{(t)} - \frac{\Delta v^{(t)} [\Delta J^{(t)}]^T}{\Delta v^{(t)} [\Delta J^{(t)}]^T} \tilde{H}^{(t)} \Delta J^{(t)},
\]

where \([\cdot]^T\) denotes the transpose, \(\Delta J^{(t)} = J^{(t+1)} - J^{(t)}\) and \(\Delta v^{(t)} = v^{(t+1)} - v^{(t)}\). The iterations of Quasi-Newton method are repeated until the difference between the values of \(v\) of two successive iterations become smaller than some predefined threshold value. It is worth mentioning that, for practical implementation, we can directly use the fminunc command in Matlab with setting its corresponding algorithm to ‘quasi-newton’ and choosing initial values in the range \((0, \pi)\) for \(v\) in order to eventually find \(v^*\) in (11).

Let us next develop new tight exponential approximations by finding new optimized numerical coefficients using the approach explained above. Based on the observations concluded from Fig. 1, we use Legendre rule to formulate the new approximations as it is one of the approximations that has the least relative error among the various quadrature rules.

For \(N = 2\), using the single-point Gauss Legendre rule \((K = 1)\), the integration interval is divided into \(M = 2\) sub-intervals which yields one unknown as \(v = v_1\). Thus, the approximated \(Q\)-function in (7) after using (9) and calculating \(w_k\) and \(\theta_k\) from Table I, will result in

\[
\tilde{Q}(x) = \frac{v_1^T}{\pi} \exp\left(-\frac{x^2}{2 \sin^2(\frac{\pi}{4})}\right) + \left(1 - \frac{v_1^T}{\pi}\right) \exp\left(-\frac{x^2}{2 \sin^2(\frac{\pi}{4} + \frac{\pi}{4})}\right).
\]

The optimum value of the parameter \(v_1\) is calculated using Quasi-Newton optimization method with respect to (11) for the case of \(R = 13\) dB and is found to be \(v_1^* = 0.967\).

For \(N = 3\), the integration interval is divided into \(M = 3\) sub-intervals which yields two unknowns as \(v = [v_1, v_2]\). Thus, the \(Q\)-function’s approximation is written as

\[
\tilde{Q}(x) = \frac{v_1^T}{\pi} \exp\left(-\frac{x^2}{2 \sin^2(\frac{\pi}{4})}\right) + \frac{v_2^* - v_1^*}{\pi} \exp\left(-\frac{x^2}{2 \sin^2(\frac{\pi}{4} + \frac{\pi}{4})}\right) + \left(1 - \frac{v_2^*}{\pi}\right) \exp\left(-\frac{x^2}{2 \sin^2(\frac{\pi}{4} + \frac{\pi}{4})}\right).
\]

The optimum parameters are \(v_1^* = 0.5571\) and \(v_2^* = 1.0702\).

In Fig. 2, we compare the accuracy of the optimized single-point Legendre rule with that of the trapezoidal rule, i.e., composite two-point closed Newton–Cotes rule. In particular, our optimized approximation outperforms that of Chiani et al. for \(N = 2\) over most of the \(x\)-range, and over the whole \(x\)-range for \(N = 3\) where they have the least relative error. The numerical approximations with \(N = 3\) are tighter than those with \(N = 2\). The integral of the absolute relative error is also calculated and plotted in Fig. 3 for the optimized and un-optimized trapezoidal and single-point Legendre rules up to \(N = 10\). As expected, the optimized approximation has a better total accuracy. In addition, it can be concluded from the decaying curves that, as the number of exponential terms increases, the accuracy of the numerical method increases.
reported based on the optimized composite Legendre rule. The coefficients for the Gaussian quadrature rules. We also contributed the optimizations of composite and non-composite Newton–Cotes and for the corresponding coefficients of all the possible approximations for it. We presented explicit expressions under the grant 326448 “Generalized Fading Distributions and Matrix

Fig. 2. Comparison between the optimized trapezoidal rule and the optimized Legendre rule.

IV. CONCLUSION
This paper provided a mathematical study on the different numerical integration techniques that can be applied on the Craig’s form of the Gaussian Q-function to produce exponential approximations for it. We presented explicit expressions for the corresponding coefficients of all the possible approximations of composite and non-composite Newton–Cotes and Gaussian quadrature rules. We also contributed the optimization of the numerical rules to increase their tightness by adopting Quasi-Newton optimization method. The coefficients of the exponential approximation with two and three terms are reported based on the optimized composite Legendre rule.

ACKNOWLEDGMENT
This research work was funded in part by the Academy of Finland under the grant 326448 “Generalized Fading Distributions and Matrix Functions for the Analysis of Wireless Communication Systems.”

REFERENCES