AppliedMath

Article

New Handy and Accurate Approximation for the Inverse Error Function and Cumulative Distribution Integrals with Applications

Mario Alberto Sandoval-Hernandez [1], Arturo Sarmiento-Reyes [2], Fernando Ivan Molina-Herrera [3], Hugo Jimenez-Islas [4], Uriel Antonio Filobello-Nino [5], Gerardo Ulises Diaz-Arango [5], Francisco Marroquin-Gutierrez [1], Rogelio Alejandro Callejas-Molina [5], Sandra Ysabel Campos-Dominguez [6], Cristian Dumay Hernandez-Garcia [7] and Hector Vazquez-Leal [5,*]

1 Facultad de Ingeniería en Aeronáutica, Universidad Politécnica Metropolitana de Hidalgo, Tolcayuca 43860, Hidalgo, Mexico; marioalberto.sandoval.cb190@dgeti.sems.gob.mx (M.A.S.-H.); fmarroquin@upmh.edu.mx (F.M.-G.)
2 Instituto Nacional de Astrofísica, Óptica y Electrónica, Luis Enrique Erro #1, Sta María Tonanzintla, San Andrés Cholula 72840, Puebla, Mexico; jarocho@inaoep.mx
3 Departamento de Ingeniería Agroindustrial, Universidad de Guanajuato, Mutualismo S/N, Celaya 38060, Guanajuato, Mexico; fi.molina@ugto.mx
4 Departamento de Ingeniería Bioquímica y Ambiental, Tecnológico Nacional de México en Celaya, Antonio García Cubas Pte #600, Esq. Ave. Tecnológico, Celaya 38010, Guanajuato, Mexico; hugo.jimenez@itcelaya.edu.mx
5 Facultad de Instrumentación Electrónica, Universidad Veracruzana, Cto. Gonzalo Aguirre Beltrán S/N, Xalapa 91000, Veracruz, Mexico; ufilobello@uv.mx (U.A.F.-N.); gerardiaz@uv.mx (G.U.D.-A.); rcallejas@uv.mx (R.A.C.-M.)
6 Facultad de Arquitectura, Universidad Veracruzana, Cto. Gonzalo Aguirre Beltrán S/N, Xalapa 91090, Veracruz, Mexico; sacampos@uv.mx
7 Facultad de Ingeniería, Universidad Veracruzana, Cto. Gonzalo Aguirre Beltrán S/N, Xalapa 91090, Veracruz, Mexico; crishernandez@uv.mx
* Correspondence: hvazquez@uv.mx

Abstract

This paper presents analytical approximations for the inverse error function, its complementary inverse, and the cumulative distribution function using the Power Series Extender Method (PSEM). The proposed expressions exhibit high accuracy over a wide portion of the domain, particularly in the central region, while maintaining a compact structure based on elementary functions. This formulation ensures practical implementation and computational efficiency without the need for specialized numerical algorithms. The use of strategically selected cancellation points further enhances the accuracy of the approximations, especially in regions of interest. As expected for this class of elementary approximations, a gradual loss of accuracy is observed near the boundaries of the domain due to the asymptotic behavior of the inverse functions. To demonstrate the effectiveness and practical relevance of the proposed expressions, two case studies are presented, involving applications in statistical analysis and engineering contexts.

Keywords: approximative methods; power series extender method; analytic approximation; cancellation points; significant digits

Check for updates

Received: 12 March 2026
Revised: 6 May 2026
Accepted: 7 May 2026
Published: 14 May 2026

1. Introduction

Statistics is the branch of mathematics concerned with the collection, organization, analysis, and interpretation of data to support inference and decision-making. Its theoretical foundation relies on probability theory, which underpins applications across a wide range of scientific disciplines.

For continuous random variables, probabilities are computed through probability density functions (PDFs), which describe how probability is distributed over possible values. The corresponding cumulative distribution function (CDF) represents the accumulated probability up to a given value.

Among probability distributions, the normal distribution is one of the most widely used due to both its empirical relevance and its theoretical importance through the central limit theorem. Many natural phenomena are well approximated by a Gaussian distribution, which is symmetric and bell-shaped [1]. Historically, it was introduced by de Moivre and later generalized by Laplace, while Gauss established its connection with the method of least squares [2].

In this context, the error function (erf) plays a fundamental role in quantifying probabilities associated with normally distributed variables. It is closely related to the normal cumulative distribution function and allows the evaluation of probabilities over specified intervals. The error function and its relation to the normal CDF are given by

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt, \tag{1}$$

and

$$CDFN(x) = \frac{1}{\sqrt{2\pi\sigma}} \int_{-\infty}^{x} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \tag{2}$$

which are related through

$$\mathrm{CDFN}(x,\mu,\sigma) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{x-\mu}{\sqrt{2}\sigma}\right)\right), \tag{3}$$

where μ and σ denote the mean and standard deviation, respectively.

For many distributions, particularly the normal distribution, the evaluation of probabilities involves integrals that do not admit closed-form expressions in terms of elementary functions. Consequently, numerical methods or tabulated values are commonly used. However, such approaches do not provide explicit analytical expressions that reveal the dependence on model parameters.

To overcome this limitation, several semi-analytical approximations have been proposed for the error function [3–7], the cumulative distribution function [3,4], and other special functions such as the Fresnel integrals [8], elliptic integrals [9,10], the Gamma function [9], and the Lambert W function [11].

In particular, for the inverse functions $\operatorname{erf}^{-1}(z)$ and $\mathrm{CDFN}^{-1}(z)$, several numerical and semi-analytical methods have been developed. For example, the Geometric Mean approach has been used to approximate the inverse error function [6], while polynomial-based algorithms have been proposed for the normal CDF and its inverse [12,13]. Notably, Acklam [14] and Wichura [15] introduced highly accurate methods based on rational Chebyshev approximations.

In addition, closed-form and semi-analytical approximations have been proposed for inverse functions. For instance, ref. [16] presents logarithmic and polynomial expressions for approximating $\mathrm{CDFN}^{-1}(z)$, while [17] employs neural-network-based approaches. Other works focus on compact approximations suitable for practical implementation [18–21].

In this paper, we propose two analytical approximations expressed in terms of elementary functions for $\operatorname{erf}^{-1}(x)$ and $\mathrm{CDFN}^{-1}(x)$ using the Power Series Extender Method (PSEM). The proposed approach avoids the use of lookup tables and complex numerical

https://doi.org/10.3390/appliedmath6050078

algorithms. To illustrate its applicability, two case studies are presented: one involving transient diffusion in a semi-infinite medium, and another related to statistical analysis using the inverse cumulative distribution function.

This article is organized as follows. Section 2 presents the basic procedure of PSEM. Section 3 introduces the proposed approximations. Section 4 provides the convergence and error analysis. Section 5 presents the applications in statistics and transport phenomena. Section 6 discusses the numerical results, and Section 7 concludes the paper.

2. The Basic Procedure of PSEM

A nonlinear differential equation can be expressed as follows:

$$L(u) + N(u) - f(x) = 0, \quad x \in \Omega, \tag{4}$$

subject to the boundary condition:

$$B\left(u, \frac{\partial u}{\partial \eta}\right) = 0, \quad x \in \Gamma, \tag{5}$$

where L and N denote the linear and nonlinear operators, respectively; $f(x)$ is a known analytic function; B is the boundary operator; Γ is the boundary of the domain Ω; and $\frac{\partial u}{\partial \eta}$ represents the derivative along the outward normal from Ω [22].

Next, we express the solution of (4) as a Taylor expansion:

$$u = \sum_{k=0}^{\infty} v_k (x - x_0)^k, \tag{6}$$

where v_k ($k = 0, 1, 2, \ldots$) are the coefficients of the power series, and x_0 is the expansion point.

In [23], it is proposed that the solution to (4) can be approximated as a finite sum of functions:

$$u = u_0 + \sum_{i=0}^{n} f_i(x, u_i), \tag{7}$$

or alternatively,

$$u = \frac{u_0 + \sum_{i=0}^{n} f_i(x, u_i)}{1 + \sum_{j=n+1}^{2n} f_j(x, u_j)}, \tag{8}$$

where u_i are parameters to be determined via PSEM, $f_i(x, u_i)$ are arbitrary trial functions, and n and $2n$ represent the approximation orders for (7) and (8), respectively. Equations (7) and (8) are collectively referred to as trial functions (TF).

We then compute the Taylor expansion of (7) or (8), yielding the following power series:

$$u = u_0 + \sum_{i=0}^{n} P_{i,0} + \sum_{k=1}^{\infty} \sum_{i=0}^{n} P_{i,k} (x - x_0)^k, \tag{9}$$

$$u = u_0 + \sum_{i=0}^{n} P_{i,0} + \sum_{k=1}^{\infty} \sum_{i=0}^{2n} P_{i,k} (x - x_0)^k, \tag{10}$$

respectively, where the Taylor coefficients $P_{i,k}$ are expressed in terms of the parameters u_i.

Finally, we match the coefficients of the power series in (9) or (10) with those of (6) to determine the values of u_i, and substitute them into (7) or (8) to obtain the PSEM approximation [23].

The accuracy of the PSEM approximation can be further improved by incorporating specific evaluation points, known as cancellation points (CPs). At each CP, one can impose additional conditions by requiring that the trial function and a selected number of its derivatives satisfy prescribed values. These conditions lead to a set of auxiliary equations that are added to the existing system to solve for the unknown parameters u_i. The purpose of introducing CPs is to enhance the local precision of the approximation in targeted regions of the domain. Importantly, each new condition imposed at a CP supersedes the corresponding higher-order terms that would otherwise appear in the standard formulation of PSEM. The selection of CPs is performed over the interval by minimizing the error between the trial function and the exact function.

The trial function (TF) can be constructed by combining various types of functions, provided that their Taylor expansions exist. The reordering of terms by powers of x in Equations (9) and (10) is valid because the functions f_j are assumed to be analytic within a specified domain of the independent variable x. Consequently, the corresponding Taylor series converge for values of x within that domain [24,25]. By restricting Equation (4) to this domain, the sum of the Taylor series of the functions f_j remains convergent [24,25]. Moreover, the quotient of two analytic functions—such as the case in (8)—is also analytic at x_0 provided that the denominator is nonzero at that point. It is important to emphasize that the convergence of PSEM is strongly influenced by the appropriate choice of the trial function. Therefore, combining carefully selected trial functions can improve the approximation in specific regions of interest.

3. PSEM Applied to Approximate Special Functions

In this section, we present approximations for the inverse special functions: the inverse error function and the inverse cumulative distribution function, denoted as $\widetilde{\mathrm{erf}}^{-1}(x)$ and $\widetilde{\mathrm{CDFN}}^{-1}(x)$, respectively.

The inverse of a function, denoted by f^{-1}, reverses the effect of the original function f. That is, if $f(x) = y$, then $f^{-1}(y) = x$. For a function to have an inverse, it must be bijective; that is, it must be injective (one-to-one) and surjective (onto) [26]. In other words, the composition of a function and its inverse yields the identity: $f(f^{-1}(x)) = x$ and $f^{-1}(f(x)) = x$.

It is important to highlight that all power series expansions presented in this work were obtained using the built-in `MultiSeries` package of Maple 2021.

3.1. Approximation Procedure for $\widetilde{\mathrm{erf}}^{-1}(x)$

We begin by expanding the equation

$$\mathrm{erf}(y(x)) = x \tag{11}$$

as a Taylor series about the point $x_0 = 0$. Evaluating the function at this point yields $y_0 = \mathrm{erf}^{-1}(x_0) = 0$.

To determine $y(x)$, we differentiate Equation (11) with respect to x and evaluate the first derivative at $x_0 = 0$:

$$y'(x)\big|_{y_0=0} = \frac{\sqrt{\pi}}{2\exp(-y(x)^2)}\bigg|_{y_0=0} = 0.886226925455. \tag{12}$$

Next, we compute the second derivative:

$$y''(x)\big|_{y(x)=y_0,\, y'(x)=0.8862} = \frac{\sqrt{\pi}\, y(x)\, y'(x)}{\exp(-y(x)^2)}\bigg|_{y(x)=0,\, y'(x)=0.8862} = 0. \tag{13}$$

We then evaluate the third derivative:

$$y'''(x)\big|_{y(x)=0,\ y'(x)=0.8862,\ y''(x)=0} = \sqrt{\pi}\exp(y(x)^2)\left(2y(x)^2y'(x)^2 + y(x)y''(x)^2 + y'(x)^2\right)\Big|_{y(x)=0,\ y'(x)=0.8862,\ y''(x)=0} = 1.39208199922. \tag{14}$$

By continuing the process of successive differentiation, we obtain the following higher-order derivatives evaluated at $x = 0$:

$$\begin{aligned} y^{(4)}(x)\big|_{y(x)=0, y'(x)=0.8862, y''(x)=0, y'''(x)=1.392} &= 0,\\ y^{(5)}(x)\big|_{y(x)=0, y'(x)=0.8862, y''(x)=0, y'''(x)=1.392, y^{(4)}=0} &= 6.56003187294,\\ &\vdots\\ y^{(13)}(x)\big|_{y(x)=0, y'(x)=0.8862, y''(x)=0, y'''(x)=1.392, y^{(4)}=0,\ldots, y(x)^{(12)}=0} &= 138384.897545, \end{aligned} \tag{15}$$

These derivatives correspond to the coefficients of the Taylor series expansion for the inverse error function. Thus, the resulting truncated Taylor expansion is given by:

$$\begin{aligned} T_y(\cdot) = 0.886226925455y + 0.232013666537y^3 + 0.127556175308y^5 +\\ 0.0701958213544y^7 + 0.0924674964374y^9 + 0.0669600918581y^{11}\\ +0.0478037341165y^{13} + \cdots. \end{aligned} \tag{16}$$

As a candidate for the trial function, we propose the following form:

$$\begin{aligned} T_{\mathrm{F}} = a_0x + a_1x\ln(-x^2+1) + a_2x\ln(a_3x^2+1) +\\ a_4x\ln(a_5x^2+1) + a_6x\ln(x^2+1). \end{aligned} \tag{17}$$

where $a_1, \cdots, a_6$ are constants to be determined by the PSEM. Next, the Taylor series for Trial Function (17), results in

$$\begin{aligned} T_{\mathrm{F}}(\cdot) = a_0x + (a_2a_3 + a_4a_5 - a_1 + a_6)x^3 + \left(-\frac{a_1}{2} - \frac{a_2a_3^2}{2} - \frac{a_4a_5^2}{2} - \frac{a_6}{2}\right)x^5 +\\ \left(-\frac{a_1}{3} + \frac{a_2a_3^3}{3} + \frac{a_4a_5^3}{3} + \frac{a_6}{3}\right)x^7 + \left(-\frac{a_1}{4} - \frac{a_2a_3^4}{4} - \frac{a_4a_5^4}{4} - \frac{a_6}{4}\right)x^9 +\\ \left(-\frac{a_1}{5} + \frac{a_2a_3^5}{5} + \frac{a_4a_5^5}{5} + \frac{a_6}{5}\right)x^{11} + \cdots. \end{aligned} \tag{18}$$

Cancellation points are proposed at the specific values $x = [0.7,\ 0.875,\ 0.4]$. By substituting these into Equation (11) and using Maple's `fsolve` command, the corresponding values of y are computed numerically, yielding the results [0.732869077959, 1.08478704007, 0.370807158594]. These x and y values are then substituted into the trial function (17), leading to a system of seven equations, which are used to determine the unknown parameters of the approximation.

$$
\begin{aligned}
x: &\quad a_0 = 0.886226925455 \\
x^3: &\quad a_2 a_3 + a_4 a_5 - a_1 + a_6 = 0.232013666537 \\
x^5: &\quad -\tfrac{a_1}{2} - \tfrac{a_2 a_3^2}{2} - \tfrac{a_4 a_5^2}{2} - \tfrac{a_6}{2} = 0.127556175308 \\
x^7: &\quad -\tfrac{a_1}{3} + \tfrac{a_2 a_3^3}{3} + \tfrac{a_4 a_5^3}{3} + \tfrac{a_6}{3} = 0.0701958213548 \\
\mathrm{CP}_1: &\quad 0.7a_0 - 0.471341187285a_1 + 0.7a_2\ln(0.49a_3+1) + \\
&\quad 0.7a_4\ln(0.49a_5+1) + 0.279143283970a_6 = 0.732869077959 \\
\mathrm{CP}_2: &\quad 0.875a_0 - 1.26947877198a_1 + 0.875a_2\ln(0.765625a_3+1) + \\
&\quad 0.875a_4\ln(0.765625a_5+1) + 0.4974416434344a_6 = 1.08478704007 \\
\mathrm{CP}_3: &\quad 0.4a_0 - 0.0697413548588a_1 + 0.4a_2\ln(0.16a_3+1) + \\
&\quad 0.4a_4\ln(0.16a_5+1) + 0.0593680020472a_6 = 0.370807158594.
\end{aligned}
\tag{19}
$$

Solving Equation (19) and substituting the obtained values for $a_0, a_1, \ldots, a_7$ into the trial function, we obtain

$$
\begin{aligned}
\widetilde{\mathrm{erf}}(x) = {} & (0.886226925455)x - 0.256094870456x\ln(-x^2+1) - \\
& 7.375785360 \times 10^{-8} x\ln(91.3025173122x^2+1) - \\
& 0.112567359034x\ln(0.351870454478x^2+1) + \\
& 0.0155346581412x\ln(x^2+1).
\end{aligned}
\tag{20}
$$

Using Maple's *convert(Aprox, rational, 10)* command and the properties of logarithms, we obtain:

$$
\widetilde{\mathrm{erf}}^{-1}(x) = x\left(\frac{51418}{58019} + \ln\left(\frac{(x^2+1)^{\frac{3517}{226397}}}{(1-x^2)^{\frac{4643}{18130}}\left(\frac{210361}{2304}x^2+1\right)^{\frac{4}{54231513}}\left(\frac{8757}{24887}x^2+1\right)^{\frac{4157}{36929}}}\right)\right). \tag{21}
$$

Figure 1 shows a comparison between the approximation in (21) and Maple's built-in `erf(x)` command (see Appendix B for the implementation details).

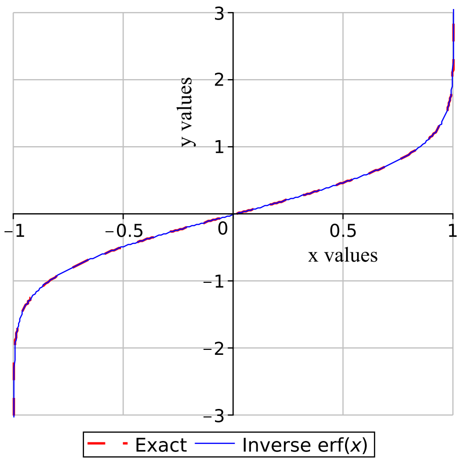


Figure 1. Graphical comparison between Equation (21) and the numerical evaluation of $\mathrm{erf}^{-1}(x)$.

3.2. Approximation of $\widetilde{CDFN}^{-1}(x)$

To obtain the Taylor series of $\widetilde{\text{CDFN}}^{-1}(x)$, we propose

$$\frac{1}{2}\,\text{erf}\left(\frac{y}{\sqrt{2}}\right) = x, \tag{22}$$

at the expansion point $x_0 = 0$. Repeating the same procedure used in the previous section to construct the Taylor series for $\widetilde{\text{CDFN}}^{-1}(x)$, we obtain

$$\begin{aligned} T_y(\cdot) = 2.506628274631x + 2.62493499095377x^3 + 5.77253353861182x^5 + \\ 12.7068009702237x^7 + 37.243456865786x^9 + 92.1147115311144x^{11} + \\ 229.011712685174x^{13}. \end{aligned} \tag{23}$$

The proposed trial function is

$$T_F(x) = a_0x + a_1x\ln(-x^2+1) + a_2x\ln(a_3x^2+1) + a_4x\ln(a_5x^2+1). \tag{24}$$

where $a_1, \cdots, a_5$ are constants to be determined by the PSEM. The Taylor series for Trial Function (17), results in

$$\begin{aligned} T_{TF}(\cdot) = a_0x + (a_2a_3 + a_4a_5 - a_1)x^3 + \left(-\frac{a_1}{2} - \frac{a_2a_3^2}{2} - \frac{a_4a_5^2}{2}\right)x^5 + \\ \left(-\frac{a_1}{3} + \frac{a_2a_3^3}{3} + \frac{a_4a_5^3}{3}\right)x^7 + \left(-\frac{a_1}{4} - \frac{a_2a_3^4}{4} - \frac{a_4a_5^4}{4}\right)x^9 + \\ \left(-\frac{a_1}{5} + \frac{a_2a_3^5}{5} + \frac{a_4a_5^5}{5}\right)x^{11} + \cdots. \end{aligned} \tag{25}$$

Now, we will use two cancellation points in (22) for $x = [0.499999999, 0.375]$. The corresponding values found for y are $y = [5.99780701500769, 1.15034938037601]$. By matching the coefficients of (23) with those of (25) and substituting the values of x and y into the trial function (23), we obtain a system of six equations, resulting in:

$$\begin{aligned}
x: &\quad a_0 = 2.50662827463101 \\
x^3: &\quad a_2a_3 + a_4a_5 - a_1 = 2.62493499095377 \\
x^5: &\quad -\tfrac{a_1}{2} - \tfrac{1}{2}a_2a_3^2 - \tfrac{1}{2}a_4a_5^2 = 5.77253353861182 \\
x^7: &\quad -\tfrac{a_1}{3} + \tfrac{1}{3}a_2a_3^3 + \tfrac{1}{3}a_4a_5^3 = 12.7068009702237 \\
\text{CP}_1: &\quad 0.499999999a_0 - 0.143841035271542a_1 + \\
&\quad 0.499999999a_2\ln(0.249999999a_3+1) + \\
&\quad 0.499999999a_4\ln(0.249999999a_5+1) = 5.99780701500769 \\
\text{CP}_2: &\quad 0.375a_0 - 0.0568312117977004a_1 + 0.375a_2\ln(0.140625a_3+1) + \\
&\quad 0.375a_4\ln(0.140625a_5+1) = 1.15034938037601
\end{aligned} \tag{26}$$

Solving the equation system (26) using the Newton–Raphson method, we obtain the coefficients: $a_0 = 2.50662827463101$, $a_1 = 0.334554690537131$, $a_2 = -8.66213146478425\times10^{-7}$, $a_3 = 217.395052089578$, $a_4 = -0.739921245524523$, and $a_5 = -3.99999055283906$.

Substituting these values into (24), and applying the *convert(TF, rational, 10)* Maple command along with logarithmic properties, we obtain:

$$\mathrm{CDFN}^{-1}(x) = \frac{41788x}{16671} - \frac{20894}{16671} + (x - \tfrac{1}{2}) \ln\left(\frac{\left(-\left(x-\frac{1}{2}\right)^2+1\right)^{\frac{8035}{24017}}}{\left(\frac{272396}{1253}(x-\frac{1}{2})^2+1\right)^{\frac{26}{30015707}} \left(-\frac{423403}{105851}(x-\frac{1}{2})^2+1\right)^{\frac{3946}{5333}}} \right). \tag{27}$$

Figure 2 shows a comparison between the approximation given in (27) and the result obtained using Maple's built-in $\mathrm{erf}(x)$ command (see Appendix B for the implementation details).

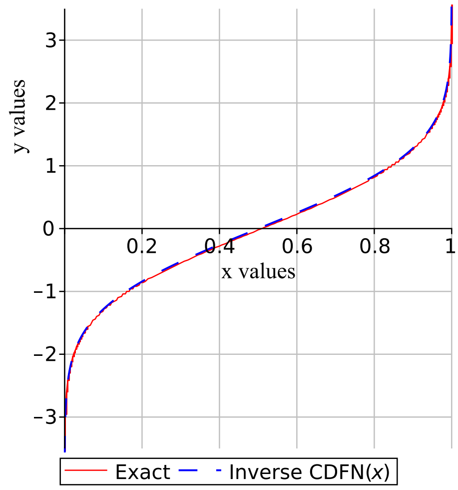


Figure 2. Graphical comparison between Equation (27) and the numerical evaluation of $\mathrm{CDFN}^{-1}(x)$.

4. Computing Convergence

First, we analyze the error differences between successive orders of the PSEM approximation to visualize the convergence behavior of the method, as follows:

$$\text{Convergence} = C_i = \left|\widetilde{E}_{i+1}(x) - \widetilde{E}_i(x)\right|, \qquad i = 1, 2, \ldots, n \tag{28}$$

where i denotes the successive orders of the $\widetilde{E}_i(x)$ PSEM approximations.

Second, the analysis of significant digits (S.D.) [27] are calculated using

$$\text{Significant Digits} \sim -\log_{10}\left|\frac{H(x) - \widetilde{H}(x)}{H(x)}\right|, \tag{29}$$

where $H(x)$ denotes the 'exact' solution (numerical solution obtained from Maple), and $\widetilde{H}(x)$ represents the approximate solution.

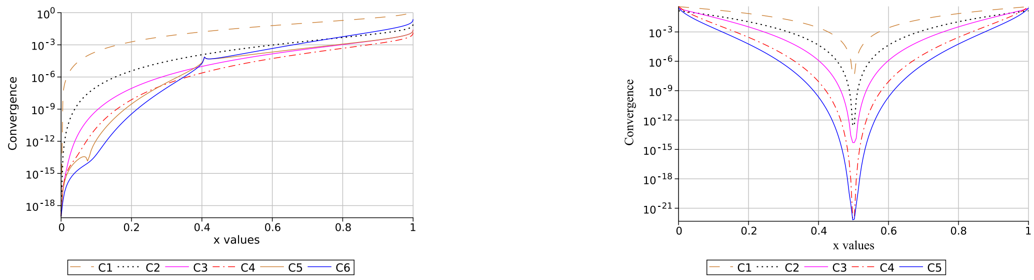
In this work, two trial functions combining x and logarithmic terms were employed. Table 1 presents the convergence analysis, illustrating the successive orders of approximation discussed in this paper. For instance, in the case of $\widetilde{\mathrm{erf}}^{-1}(x)$ with $i = 3$, three constants were used: a_0, a_1, and a_2.

Figure 3a,b depict the convergence behavior of the approximations for $\widetilde{\mathrm{erf}}^{-1}(x)$ and $\widetilde{\mathrm{CDFN}}^{-1}(x)$, respectively, across different approximation orders i as presented in Table 1, using Equation (28). It is important to note that in both case studies, the error tends to decrease steadily, reaching values below 1×10^{-2}.

For the case of $\widetilde{\mathrm{erf}}^{-1}(x)$, Figure 4a and Figure 4b show the relative error and the number of significant digits, respectively. Note that for approximation E_7 (i.e., $i = 7$, see Table 1), the relative error is approximately 1×10^{-3}, while the number of significant digits approaches 3 when $x = 1$.

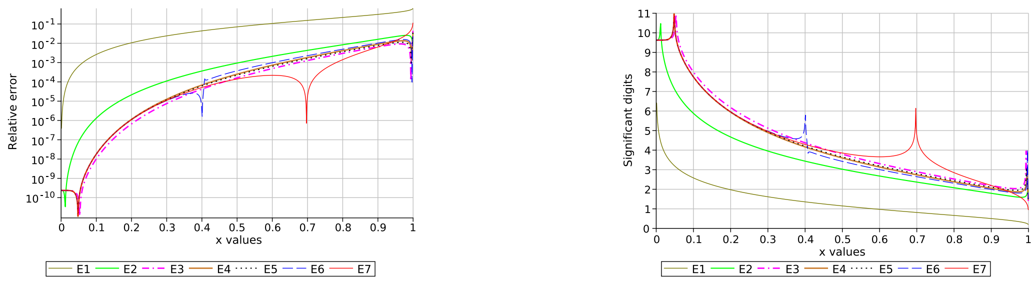
Table 1. i-th trial functions (TF_i) used in the PSEM for $\widetilde{\mathrm{erf}}^{-1}(x)$ and $\widetilde{\mathrm{CDFN}}^{-1}(x)$.

$\mathbf{TF}_i$	**Approximation**
i	$\widetilde{\mathrm{erf}}^{-1}(x)$
1	$a_0 x$
2	$a_0 x + a_1 x \ln(-x^2+1)$
3	$a_0 x + a_1 x \ln(-x^2+1) + a_2 x \ln(x^2+1)$
4	$a_0 x + a_1 x \ln(-x^2+1) + a_2 x \ln(a_3 x^2+1)$
5	$a_0 x + a_1 x \ln(-x^2+1) + a_2 x \ln(a_3 x^2+1) + a_4 x \ln(x^2+1)$
6	$a_0 x + a_1 x \ln(-x^2+1) + a_2 x \ln(a_3 x^2+1) + a_4 x \ln(a_5 x^2+1)$
7	$a_0 x + a_1 x \ln(-x^2+1) + a_2 x \ln(a_3 x^2+1) + a_4 x \ln(a_5 x^2+1) + a_6 x \ln(x^2+1)$
i	$\widetilde{\mathrm{CDFN}}^{-1}(x)$
1	$a_0 x$
2	$a_0 x + a_1 x \ln(-x^2+1)$
3	$a_0 x + a_1 x \ln(-x^2+1) + a_2 x \ln(x^2+1)$
4	$a_0 x + a_1 x \ln(-x^2+1) + a_2 x \ln(a_3 x^2+1)$
5	$a_0 x + a_1 x \ln(-x^2+1) + a_2 x \ln(a_3 x^2+1) + a_4 x \ln(x^2+1)$
6	$a_0 x + a_1 x \ln(-x^2+1) + a_2 x \ln(a_3 x^2+1) + a_4 x \ln(a_5 x^2+1)$



(**a**) Convergence for $\widetilde{\mathrm{erf}}^{-1}(x)$ (**b**) Convergence $CDFN^{-1}(x)$

Figure 3. Convergence analysis of the PSEM approximations for $\mathrm{erf}^{-1}(x)$ and $\mathrm{CDFN}^{-1}(x)$ as the approximation order i increases.

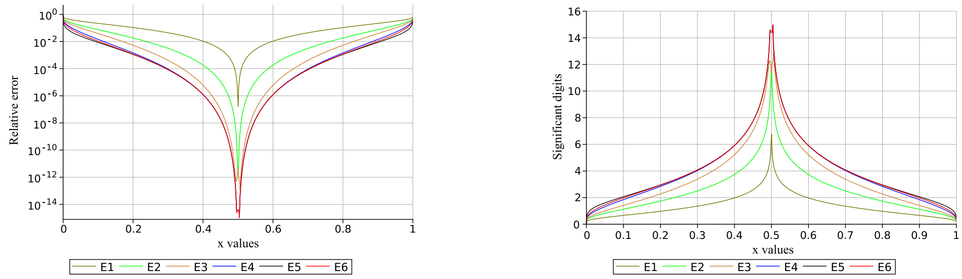


(**a**) Relative error for $\widetilde{\mathrm{erf}}^{-1}(x)$. (**b**) Significant digits for $\widetilde{\mathrm{erf}}^{-1}(x)$

Figure 4. Relative error distribution and associated significant digits for the PSEM approximation of $\mathrm{erf}^{-1}(x)$.

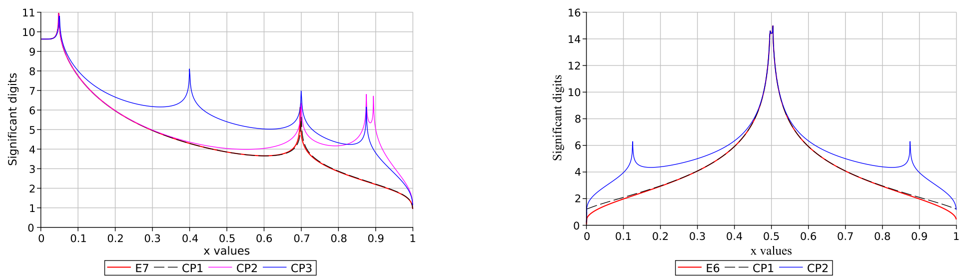
For $\widetilde{\mathrm{CDFN}}^{-1}(x)$, the relative error is shown in Figure 5a, while the number of significant digits is depicted in Figure 5b. The relative error for approximation E_6 (i.e., $i = 6$, see Table 1) approaches 1×10^{-1} near the boundaries and decreases as x approaches 0.5. Similarly, the number of significant digits tends to decrease near the boundaries and increases as x approaches 0.5, indicating improved accuracy in the central region.

Figure 6a,b show the number of significant digits obtained for $\widetilde{\mathrm{erf}}^{-1}(x)$ and $\widetilde{\mathrm{CDFN}}^{-1}(x)$, respectively. In both cases, an improvement in the number of significant digits is observed due to the use of cancellation points. For instance, in Figure 6a, the approximation labeled CP_3, which includes three cancellation points, yields more significant digits compared to CP_1, CP_2, or even the E_7 approximation.



(**a**) (**b**)

Figure 5. Relative error distribution and associated significant digits for the PSEM approximation of $\mathrm{CDFN}^{-1}(x)$. (**a**) Relative error for $\widetilde{\mathrm{CDFN}}^{-1}(x)$. (**b**) Significant digits for $\widetilde{\mathrm{CDFN}}^{-1}(x)$.



(**a**) (**b**)

Figure 6. Comparison of significant digits obtained using CPs for $\widetilde{\mathrm{erf}}^{-1}(x)$ and $\widetilde{\mathrm{CDFN}}^{-1}(x)$. (**a**) Significant digits with CPs for $\widetilde{\mathrm{erf}}^{-1}(x)$. (**b**) Significant digits with CPs for $\widetilde{\mathrm{CDFN}}^{-1}(x)$.

5. Applications to Science and Engineering

5.1. Study Case 1: Transient Diffusion in a Semi-Infinite Medium

Chemical species A (e.g., toluene) is dropped at the soil level ($y = 0$), producing a superficial concentration of $C_{A_1} = 0.02\ \mathrm{kg/m^3}$. The objective is to predict the concentration of A at various depths and times, considering the subsurface as a semi-infinite medium ($y \geq 0$); see Figure 7.

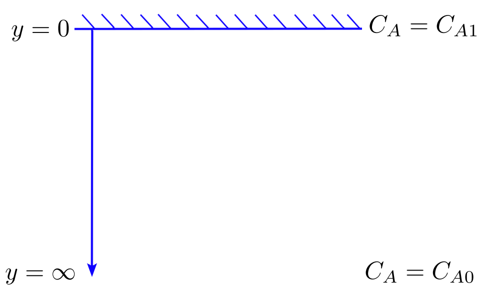


Figure 7. Concentration in the entire subsoil volume.

Fick's second law model:

$$\frac{\partial C_A}{\partial t} = D_{\mathrm{eff}} \cdot \frac{\partial^2 C_A}{\partial y^2} \tag{30}$$

Boundary (BC) and initial conditions (IC):

$$\begin{aligned} C_A(0,t) &= C_{A_1} \\ C_A(y \to \infty, t) &= C_{A_0} \\ C_A(y,0) &= C_{A_0} \end{aligned} \tag{31}$$

where C_{A_0} is the initial concentration in the entire subsoil volume (initially, the soil has no contamination, $C_{A_0} = 0$), [kg/m^3]; C_{A_1} is concentration at the soil surface $(y = 0)$, [kg/m^3]; y is depth [m]; t is the time [days]; D_{eff} is the effective diffusivity in soil [m^2/day].

Solution: The experiment on the diffusion of toluene in the subsoil was conducted. It was measured that at a depth of $y = 0.10$ m from the soil surface and at a time of 5 days, the concentration of toluene is 4.58×10^{-3} kg/m^3. The concentration of toluene at the soil surface is 0.02 kg/m^3. Estimate the value of the effective diffusivity of toluene (D_{eff}) in m^2/day.

At the beginning of the experiment, the subsoil was free of toluene: $C_{A_0} = 0.00$ kg/m^3. From the solution of Equation (30):

$$\phi = \frac{C_A - C_{A_0}}{C_{A_1} - C_{A_0}} = \operatorname{erfc}(B), \quad \text{with} \quad B = \frac{y}{\sqrt{4D_{\text{eff}}t}}. \tag{32}$$

By inverting

$$\operatorname{erfc}^{-1}(\phi) = \frac{y}{\sqrt{4D_{\text{eff}}t}} \quad \Rightarrow \quad D_{\text{eff}} = \frac{y^2}{4\left[\operatorname{erfc}^{-1}(\phi)\right]^2 t}. \tag{33}$$

By substituting the values for C_A, C_{A_0} and C_{A_1} in Equation (32) we have $\phi = 0.22905$.

To evaluate $\widetilde{\operatorname{erfc}}^{-1}(x)$, we use (21). First, we replace x by $x - 1$ to shift the domain. Then, the result is multiplied by -1 to reflect it with respect to the new origin at $x = 1$. Thus, we obtain

$$\widetilde{\operatorname{erfc}}^{-1}(x) = -(x-1)\left(\frac{51418}{58019} + \ln\left(\frac{((x-1)^2+1)^{\frac{3517}{226397}}}{(1-(x-1)^2)^{\frac{4643}{18130}}\left(\frac{210361(x-1)^2}{2304}+1\right)^{\frac{4}{54231513}}\left(\frac{8757(x-1)^2}{24887}+1\right)^{\frac{4157}{36929}}}\right)\right) \tag{34}$$

By substituting (34) $\operatorname{erfc}^{-1}(0.22905) \approx 0.8506428$ and $D_{\text{eff}} \approx 6.9099 \times 10^{-6}$ m^2/day. Figure 8 shows the use $\operatorname{erfc}^{-1}(x)$ to find the corresponding value of ϕ on domain.

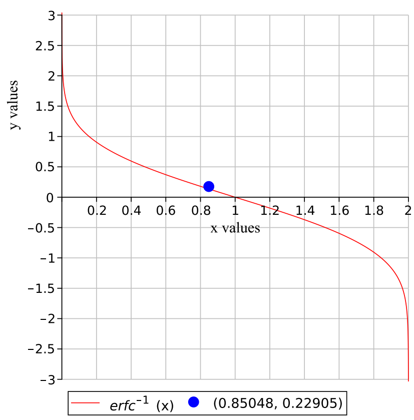


Figure 8. Evaluation of $\operatorname{erfc}^{-1}(x)$ to find value of ϕ.

5.2. Study Case 2: Application of the $\widetilde{\mathrm{CDFN}}^{-1}$ Function in Statistics

If the probability that a randomly selected individual scores 15 points on a manual dexterity test administered by a therapist is 0.0228, what should the mean be for a group of patients, assuming the standard deviation of that group is known to be 2.5?

Solution: This problem was solved in [28]; however, we will now solve the inverse. On this occasion, we aim to find the mean.

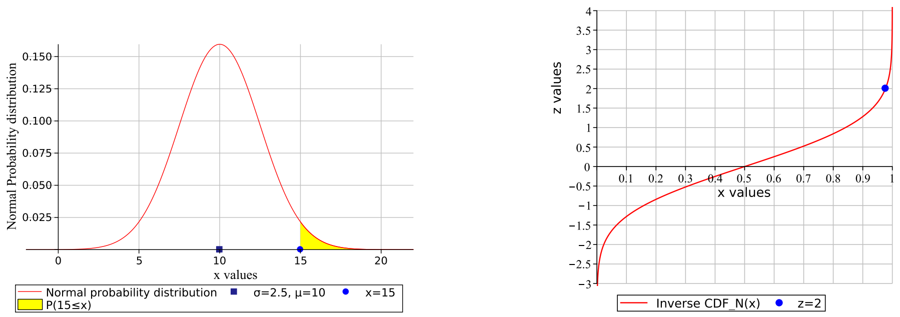
The shaded area in Figure 9a is equal to 0.0228, corresponding to randomly selected individual scores of 15 points. Thus, the area no shaded is equal to 0.9772, considering the transformation formula

$$z = \frac{x-\mu}{\sigma}, \tag{35}$$

where z is the transformed variable, x is the value seeked, μ is the mean, and σ is the standard desviation.

$$z = \widetilde{\mathrm{CDFN}}^{-1}(0.9772) = 2.0086 \tag{36}$$

Figure 9b shows the value searched. Substituting $z = 2.0086$, $x = 15$, $\sigma = 2.5$, we obtain $\mu = -9.978$.



(**a**) (**b**)

Figure 9. Evaluation of $\widetilde{\mathrm{CDFN}}^{-1}(x)$ to obtain numerical solution for value z. (**a**) Normal probability density function. (**b**) Evaluation using $\widetilde{\mathrm{CDFN}}^{-1}(x)$ to determine value z.

6. Numerical Comparison and Discussion

Mathematical, transcendental, and special functions play a fundamental role in science and its applications [4,8,11,29]. In particular, the determination of inverse functions is of great importance.

The inverse function, denoted by $f^{-1}(x)$, reverses the effect of the original function $f(x)$. Specifically, if $f(a) = b$, then $f^{-1}(b) = a$. For a function to possess an inverse, it must be bijective; that is, it must be both injective (one-to-one) and surjective (its range coincides with its codomain) [26]. Therefore, it is relevant to have an inverse function that enables the recovery of input values from known output values.

It is important to emphasize that, in the case of inverse functions $f^{-1}(\cdot)$, the exponent -1 does not represent a reciprocal operation. This distinction is particularly significant in the context of trigonometric functions. For example, $\tan^{-1}(x)$ refers to the inverse tangent (or arctangent) function, and not to the cotangent function. This is a frequent source of confusion in introductory trigonometry courses, primarily due to the conventions established in school mathematical discourse [30,31].

Table 2 reports the root mean square error (RMSE) and the maximum absolute error (MAE) [32] for $\mathrm{erfc}^{-1}(x)$ within each subinterval, along with the corresponding location x, using a step size of $\Delta x = 0.001$. This provides a localized assessment of the approximation accuracy.

It is observed that Equation (A1) exhibits a lower RMSE than Equations (21) and (A2)–(A4). However, it is important to note that Equation (A1) is defined only over the domain of positive values of x. Within this domain, Equation (A1) yields smaller errors compared to the other approximations.

The numerical algorithm proposed by Martila–Groote [6] exhibits the lowest RMSE among all methods considered. In all cases, the absolute error attains its largest values as $x \to 1$. For all the reported methods in this study, the MAE occurs at $x = 0.999$ (see Table 2), except for Equation (A2), where the maximum error is located at $x = 0.995$, due to the limited validity range of the approximation.

Table 2. RMSE and MAE for the PSEM approximation of $\widetilde{\text{erf}}^{-1}(x)$ for $x \in (-1, 1)$.

Work	RMSE	MAE	Location of MAE
This work Equation (21)	0.00661	0.12554	$x = \pm 0.999$
Winitzki Equation (A1)	0.000939	0.00793	$x = \pm 0.999$
Karlsson Equation (A2)	0.01549	0.20121	$x = 0.995$
Vedder Equation (A3)	0.09947	0.12694	$x = \pm 0.999$
Carlitz series Equation (A4)	0.06932	0.83940	$x = \pm 0.999$
Martila–Groote [6]	3.56124×10^{-7}	9.0190×10^{-6}	$x = 0.999$

Table 3 presents the maximum absolute error in each subinterval (MAEs) for inverse error function approximations over $x \in [-0.900, 0.900)$. For each method, the first row indicates the location of the maximum error and the second row reports the corresponding MAE. In general, the absolute error increases toward the right side of each subinterval for $x > 0$ and toward the left side for $x < 0$. Furthermore, the largest absolute errors are observed in the vicinity of $x = \pm 0.900$.

The MAEs in the subintervals $(-1, -0.900)$ and $(0.900, 1)$ are not included, since in these regions the global MAE over the entire interval occurs near $x = \pm 1$, as shown in Table 2.

Table 3. The MAEs for $\widetilde{\text{erf}}^{-1}(x)$ over $x \in [-0.900, 0.900]$.

Interval MAEs	This Work	Vedder	Carlitz	Winitzki	Karlsson	Martila–Groote
$[-0.900, -0.600)$ MAEs	$x = -0.900$ 1.808×10^{-4}	$x = -0.900$ 0.15538	$x = -0.900$ 0.02733	–	–	–
$[-0.600, -0.300)$ MAEs	$x = -0.600$ 5.520×10^{-6}	$x = -0.600$ 0.10461	$x = -0.600$ 2.503×10^{-5}	–	–	–
$[-0.300, 0.000)$ MAEs	$x = -0.300$ 1.814×10^{-7}	$x = -0.300$ 0.05271	$x = -0.300$ 5.677×10^{-10}	–	–	–
$[0.000, 0.300)$ MAEs	$x = 0.299$ 1.800×10^{-7}	$x = 0.299$ 0.05254	$x = 0.299$ 5.397×10^{-10}	$x = 0.299$ 1.135×10^{-6}	$x = 0.299$ 1.814×10^{-7}	$x = 0.299$ 1.076×10^{-11}
$[0.300, 0.600)$ MAEs	$x = 0.599$ 5.494×10^{-6}	$x = 0.599$ 0.10444	$x = 0.599$ 2.437×10^{-5}	$x = 0.599$ 5.058×10^{-5}	$x = 0.599$ 5.272×10^{-5}	$x = 0.582$ 1.603×10^{-10}
$[0.600, 0.900]$ MAEs	$x = 0.899$ 1.690×10^{-4}	$x = 0.899$ 0.15522	$x = 0.899$ 0.02671	$x = 0.899$ 1.002×10^{-3}	$x = 0.899$ 0.02162	$x = 0.899$ 3.010×10^{-8}

In general, according to Tables 2 and 3, and for all reported approximations, the absolute error increases as $x \to \pm 1$. This behavior is attributed to the asymptotic nature of the function $\widetilde{\text{erf}}^{-1}(x)$ near the boundaries of the domain.

Figure 10 illustrates the behavior of the absolute error for all approximations presented in Tables 2 and 3, where a clear increase in error is observed as $x \to \pm 1$. As shown in these tables, the largest deviations are concentrated near the boundaries of the interval, while smaller errors are observed in the central region. It is worth noting that Equation (21) generally exhibits a lower absolute error than Equations (A1)–(A3). However, its RMSE is higher than that of Equation (A1), since the worst-case MAE of Equation (21) is larger, occurring at $x = 0.999$.

Table 4 shows RMSE and MAE for inverse normal cumulative distribution function approximations, computed using $\Delta x = 0.001$. Equation (27) exhibits a lower RMSE compared to the other analytical approximations, with the exception of the numerical algorithms proposed by Acklam [14] and Wichura [15]. Similarly, the MAE obtained from Equation (27) is smaller than that of the remaining methods, except for those same numerical algorithms. It is also observed that, for most methods, the MAE occurs near the left extreme of the interval, except (A7) in $x = 0.999$, whereas for the numerical algorithms, the maximum error may appear at both locations within the domain.

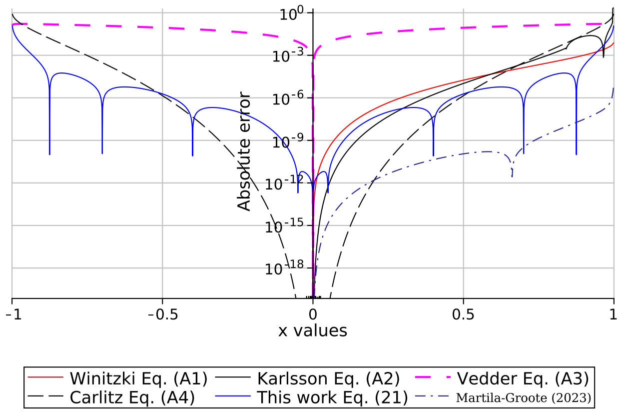


Figure 10. Comparison of Absolute error for Equation (21) versus [6,20,21,33,34].

Table 4. The RMSE and MAE $\widetilde{\text{CDFN}}^{-1}(x)$ approximations over $x \in (0, 1)$.

Work	RMSE	MAE	Location of MAE
This work Equation (27)	0.01068	0.15155	$x = 0.001$
Schmeisser Equation (A5)	0.11139	0.59272	$x = 0.001$
Nakagami Equation (A6)	1.26147	3.68553	$x = 0.001$
Dyer Equation (A7)	0.18601	1.43630	$x = 0.999$
Acklam [14]	7.8831×10^{-10}	2.895×10^{-9}	$x = 0.005, 0.995$
Wichura [15]	3.6340×10^{-17}	1.069×10^{-16}	$x = 0.075$

The Table 5 shows the MAEs for inverse normal cumulative distribution function approximations over $x \in (0, 1)$. For each method, the first row indicates the location of the maximum error and the second row reports the corresponding MAE. In general, note that MAE occurs in subinterval $(0, 0.250]$ for all works. For each method, the first row indicates the location of the maximum error, and the second row reports the corresponding MAE. In general, the MAE occurs in the subinterval $(0, 0.250]$ for all methods. Likewise, that erf, the MAEs occurs in the extremes of domain. In general the error increments from $x = 0.5$ towards extremes for (27), (A5), Acklam [14] and Wichura [15], except to (A7) and (A6).

Figure 11 shows the absolute error of Equation (27) in comparison with Equations (A5)–(A7), as well as the numerical methods reported in [14,15]. It can be observed that Equation (27) yields a lower absolute error than the other analytical methods. Furthermore, the absolute error increases in the vicinity of $x = 0$ and $x = 1$, which is associated with the asymptotic behavior of the inverse function.

Figure 12a presents a comparison of the number of significant digits achieved by our approximation (21) versus other reported formulas, namely (A1)–(A4) and Martila–Groote [6], which correspond to the approximations proposed in [20,21,33,34], respectively.

By the other hand, it is notable that our proposed expressions are the only ones that maintain high accuracy throughout the interval $0.01 < x < 0.942$. In general, all evaluated approximations exhibit a decline in significant digits, particularly near $x = 1$, due to the inherent asymptotic behavior of the inverse error function in that region.

Table 5. MAEs for $\widetilde{\mathrm{CDFN}}^{-1}(x)$ approximations over $x \in (0, 1)$.

Work **MAEs**	$(\mathbf{0}, \mathbf{0.250}]$	$(\mathbf{0.250}, \mathbf{0.500}]$	$(\mathbf{0.500}, \mathbf{0.750}]$	$(\mathbf{0.750}, \mathbf{1})$
This work Equation (27) MAE s	$x = 0.001$ 0.15155	$x = 0.251$ 1.550×10^{-4}	$x = 0.750$ 1.580×10^{-4}	$x = 0.999$ 0.15155
Schmeisser Equation (A5) MAEs	$x = 0.001$ 0.59272	$x = 0.251$ 0.07021	$x = 704$ 0.01385	$x = 0.704$ 0.01385
Nakagami Equation (A6) MAEs	$x = 0.001$ 3.68553	$x = 0.251$ 1.50001	$x = 501$ 0.98127	$x = 0.999$ 1.41092
Dyer Equation (A7) MAEs	$x = 0.025$ 0.02653	$x = 0.500$ 0.02029	$x = 750$ 0.09805	$x = 0.999$ 1.43630
Acklam [14] MAEs	$x = 0.005$ 2.895×10^{-9}	$x = 0.252$ 8.458×10^{-11}	$x = 749$ 5.617×10^{-11}	$x = 0.995$ 2.895×10^{-9}
Wichura [15] MAEs	$x = 0.075$ 1.069×10^{-16}	$x = 0.280$ 4.311×10^{-17}	$x = 750$ 3.169×10^{-17}	$x = 0.925$ 1.069×10^{-16}

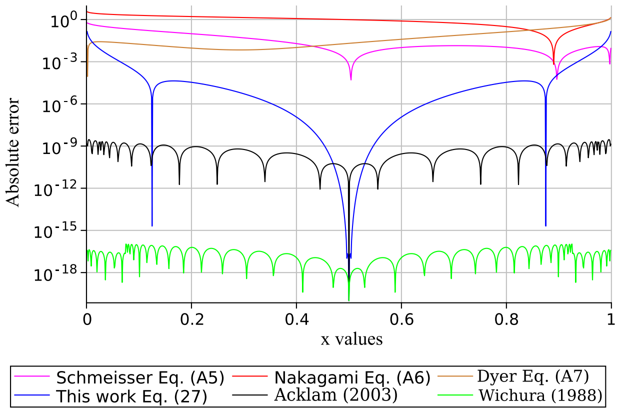
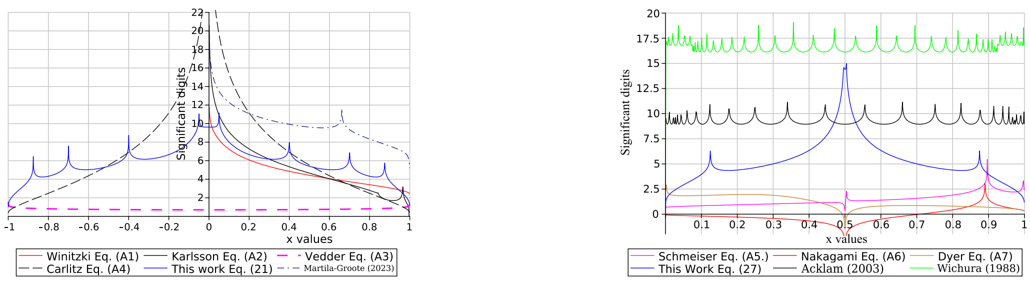


Figure 11. Absolute error for Equation (27) vs. others methods [14–16,18,19].



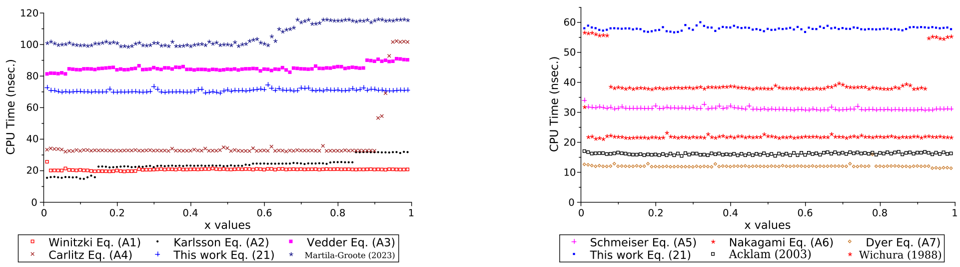
(**a**) (**b**)

Figure 12. Comparison of significant digits obtained by our proposal versus other approximations for $\widetilde{\mathrm{erf}}^{-1}(x)$ and $\widetilde{\mathrm{CDFN}}^{-1}(x)$. (**a**) Comparison of significant digits for $\widetilde{\mathrm{erf}}^{-1}(x)$ versus other methods [6,20,21,33,34]. (**b**) Comparison of significant digits for $\widetilde{\mathrm{CDFN}}^{-1}(x)$ versus other methods [14–16,18,19].

Figure 12b illustrates the comparison of significant digits for $\widetilde{\mathrm{CDFN}}^{-1}(x)$ against the algebraic approximations given in Equations (A5)–(A7), reported in [18], [19], and [16], respectively, as well as the numerical algorithms proposed by Acklam [14] and Wichura [15]. It can be observed that the proposed approximation in Equation (27) achieves at least three significant digits over almost the entire domain, outperforming the other methods. In contrast, the approximations reported by other authors do not consistently reach this level of accuracy. Only numerical methods achieve higher accuracy than the proposed approach. It is worth noting that, although the title of [16] refers to approximations of the error function, the authors actually present an expression for the *inverse* cumulative distribution function (CDF), specifically Equation (A7).

Both special functions are essential in computational simulations across a wide range of applications in physics and engineering. To evaluate their efficiency, we measured the CPU execution time in nanoseconds using a Fortran 90 implementation described in Appendix B. Figure 13a illustrates that the approximation given in Equation (21) requires the most computational time. This is primarily due to the greater algebraic complexity of the PSEM-based expression, which includes a logarithmic function and polynomial terms. In contrast, the approximations expressed in Equations (A1), (A2) and (A4) have simpler structures with constant coefficients. In the case of Equation (A4), it corresponds to a series expansion. In this study, Equation (A4) is represented as a polynomial of degree 13;However, it is important to note that high-degree polynomial approximations may exhibit numerical instability, particularly near the boundaries of the interval (as observed in Tables 2 and 3). This behavior is associated with Runge's phenomenon, where oscillations increase as the polynomial degree grows. Consequently, although adding more terms may improve local accuracy, it can adversely affect the overall stability of the approximation [35,36]. In the case, Equation (A3) has a structure composed of hyperbolic functions; consequently, it requires higher computational cost.

A similar behavior is observed for Equation (27) in Figure 13b, where the increased number of algebraic terms leads to longer computation times but also results in improved accuracy. In contrast, Equation (A7) has a simpler structure with constant coefficients, as it is derived from least-squares fitting techniques. Despite the higher computational cost, Equation (21) provides superior accuracy. For reference, Appendix A includes a summary of other approximations reported in the literature.



(**a**) (**b**)

Figure 13. Comparison of CPU execution times (in nanoseconds) for the proposed and reported approximations of $\widetilde{\mathrm{erf}}^{-1}(x)$ and $\widetilde{\mathrm{CDFN}}^{-1}(x)$. (**a**) Execution time for $\widetilde{\mathrm{erf}}^{-1}(x)$ and other approximations [6,20,21,33,34]. (**b**) Execution time for $\widetilde{\mathrm{CDFN}}^{-1}(x)$ and other approximations [14–16,18,19].

In this work, two case studies were presented using Equations (21) and (27) to demonstrate the procedure for recovering domain values when the corresponding codomain values are provided. In Case Study 1, an application related to transport phenomena was addressed. Specifically, the complementary inverse error function was employed to recover a parameter involved in transient diffusion within a semi-infinite medium. In this context, it was necessary to use the function $\widetilde{\mathrm{erfc}}^{-1}(x)$, which was derived through the corresponding algebraic procedure. The resulting approximation is given in Equation (34), and its application is illustrated in Figure 8.

The solution obtained for Case Study 2 shows good agreement with the data reported in [28], specifically for the parameters $z = 2$ and $\mu = 10$. In our analysis, we found $z = 2.0086$, a result that lies very close to the reference value and demonstrates at least two significant digits of accuracy, as illustrated in Figure 12b. The methodology that followed to determine the value of z is detailed in Figure 9b.

Future research should aim to further improve the proposed approximations by enhancing their computational efficiency and increasing their accuracy.

7. Concluding Remarks

In this article, the PSEM method was employed to propose two practical approximations: the first for $\widetilde{\mathrm{erf}}^{-1}(x)$ and the second for $\widetilde{\mathrm{CDFN}}^{-1}(x)$. It is worth highlighting that the resulting expressions are both compact and computationally efficient, eliminating the need for numerical integration or interpolation from data tables. Moreover, our proposals are expressed in terms of elementary operations and logarithmic functions, making them easy to implement in embedded systems or low-resource environments. A comparative analysis with previously published approximations revealed that our methods consistently yielded the lowest relative errors across the tested intervals. Finally, two case studies demonstrated the utility of the proposed approximations in practical applications. Notably, the inverse complementary error function $\widetilde{\mathrm{erfc}}^{-1}(x)$ was algebraically derived and applied in the second case study.

Author Contributions: Conceptualization, M.A.S.-H. and A.S.-R.; methodology, M.A.S.-H., A.S.-R., F.I.M.-H., H.J.-I., U.A.F.-N., G.U.D.-A. and F.M.-G.; software, M.A.S.-H., H.J.-I., G.U.D.-A. and R.A.C.-M.; validation, F.I.M.-H., U.A.F.-N., F.M.-G., R.A.C.-M., S.Y.C.-D., C.D.H.-G. and H.V.-L.; formal analysis, A.S.-R., F.I.M.-H., H.J.-I., U.A.F.-N., G.U.D.-A., F.M.-G., R.A.C.-M., S.Y.C.-D. and C.D.H.-G.; investigation, M.A.S.-H., A.S.-R., F.I.M.-H., H.J.-I., U.A.F.-N. and G.U.D.-A.; writing—original draft preparation, M.A.S.-H., A.S.-R., F.I.M.-H. and H.J.-I.; writing—review and editing, M.A.S.-H., G.U.D.-A., F.M.-G. and H.V.-L.; visualization, U.A.F.-N., G.U.D.-A., F.M.-G., R.A.C.-M., S.Y.C.-D., C.D.H.-G. and H.V.-L.; supervision, H.J.-I., U.A.F.-N., S.Y.C.-D. and C.D.H.-G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: The authors thank Roberto Ruiz Gomez for his support during the writing of this article.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Existing Approximations for Comparison

$$\widetilde{\mathrm{erf}}_1^{-1}(x) = \left(-\frac{2}{a\pi} - \frac{1}{2}\ln(-x^2+1) + \sqrt{\left(\frac{2}{a\pi} + \frac{1}{2}\ln(-x^2+1)\right)^2 - \frac{\ln(-x^2+1)}{a}} \right)^{\frac{1}{2}}, \quad a = 14. \tag{A1}$$

$$\mathrm{erf}_2^{-1}(x) \approx \begin{cases} \arcsin\left(\arcsin\left(\frac{x\sqrt{\pi}}{2}\right)\right), & 0 \le x \le 0.84 \\ (\arcsin(x))^{1.5}, & 0.84 \le x \le 0.995 \end{cases} \tag{A2}$$

$$\mathrm{erf}_3(x) = \frac{8\sqrt{61149}}{561}\sinh\left(\frac{1}{3}\operatorname{arcsinh}\left(\frac{51\,\operatorname{arctanh}(x)\sqrt{1744}\sqrt{561}}{55808}\right)\right). \tag{A3}$$

$$\mathrm{erf}_3^{-1}(x) = \frac{\sqrt{\pi}}{2}x + \frac{\pi^{3/2}}{24}x^3 + \frac{7\pi^{5/2}}{960}x^5 + \frac{127\pi^{7/2}}{80640}x^7 + \frac{4369\pi^{9/2}}{11612160}x^9 + \frac{34807\pi^{11/2}}{364953600}x^{11} + \frac{20036983\pi^{13/2}}{797058662400}x^{13}. \tag{A4}$$

$$\widetilde{CDFN}_1^{-1}(x) = 0.2 + 6.26566416040100\, x^{0.14} - 6.26566416040100\,(1-x)^{0.09}. \tag{A5}$$

$$\widetilde{CDFN}_2^{-1}(x) = 1.2552\mathrm{arctanh}(x)\exp(-1.3854 * x) + 0.4347\exp(0.6679x). \tag{A6}$$

$$\widetilde{CDFN}_3^{-1}(x) = 1.724 - 2.22\sqrt{-\ln(x)} - 0.15\ln(x). \tag{A7}$$

Appendix B. Approximations of Inverse Functions

Appendix B.1. $\widetilde{erf}^{-1}(x)$ *Maple Code*

```
# Error function Erf^{-1}(x);
# Code Written for the authors'
# Code for computing inverse error function '

restart;
with(plots);

Digits := 12:

#Expansion point
x0 := 0;

#Inverse exact function#
F1 := erf(y) = x;

# Derivatives
# Expansion point for yo'
y0 := evalf(erf(x0));
d1 := solve(diff(erf(y(x)), x) = 1, diff(y(x), x));
define(InvG, diff(InvG(x), x) = d1);

d1n := eval(d1, y(x) = y0);
d2 := diff(d1, x);
d2n := simplify(eval(eval(d2, diff(y(x), x) = d1n), y(x) = y0), size);
d3 := diff(d2, x):
d3n := simplify(simplify(eval(eval(d3, diff(y(x), x) = d1n), y(x) = y0)), size);
d4 := diff(d3, x):
d4n := simplify(simplify(eval(eval(d4, diff(y(x), x) = d1n), y(x) = 0)), size);
d5 := diff(d4, x):
d5n := simplify(simplify(eval(eval(d5, diff(y(x), x) = d1n), y(x) = y0)), size);
d6 := diff(d5, x):
d6n := simplify(simplify(eval(eval(d6, diff(y(x), x) = d1n), y(x) = y0)), size);
d7 := diff(d6, x):
d7n := simplify(simplify(eval(eval(d7, diff(y(x), x) = d1n), y(x) = 1)), size);
d8 := diff(d7, x):
d8n := simplify(simplify(eval(eval(d8, diff(y(x), x) = d1n), y(x) = y0)), size):
d9 := diff(d8, x):
d9n := simplify(simplify(eval(eval(d9, diff(y(x), x) = d1n), y(x) = y0)), size):
d10 := diff(d9, x):
d10n := simplify(simplify(eval(eval(d10, diff(y(x), x) = d1n), y(x) = y0)), size):
d11 := diff(d10, x):
d11n := simplify(simplify(eval(eval(d11, diff(y(x), x) = d1n), y(x) = y0)), size);
d12 := diff(d11, x):
d12n := simplify(simplify(eval(eval(d12, diff(y(x), x) = d1n), y(x) = y0)), size);
d13 := diff(d12, x):
d13n := simplify(simplify(eval(eval(d13, diff(y(x), x) = d1n), y(x) = y0)), size);
d14 := diff(d13, x):
d14n := simplify(simplify(eval(eval(d14, diff(y(x), x) = d1n), y(x) = y0)), size);
d15 := diff(d14, x):
d15n := simplify(simplify(eval(eval(d15, diff(y(x), x) = d1n), y(x) = y0)), size);
d16 := diff(d15, x):
d16n := simplify(simplify(eval(eval(d16, diff(y(x), x) = d1n), y(x) = y0)), size);


# Formulation for Taylor series exact inverse
pol := convert(taylor(InvG(x), x = x0, 14), polynom):
TinvG := eval(pol, [InvG(x0) = y0, y(x0) = y0, D(y)(x0) = d1n, (D@@2)(y)(x0) = d2n,
  (D@@3)(y)(x0) = d3n, (D@@4)(y)(x0) = d4n, (D@@5)(y)(x0) = d5n, (D@@6)(y)(x0) = d6n,
  (D@@7)(y)(x0) = d7n, (D@@8)(y)(x0) = d8n, (D@@9)(y)(x0) = d9n,
  (D@@10)(y)(x0) = d10n, (D@@11)(y)(x0) = d11n, (D@@12)(y)(x0) = d12*n,
  (D@@13)(y)(x0) = d13n, (D@@14)(y)(x0) = d14n, (D@@15)(y)(x0) = d15n]);

# Taylor series for  exact function
```

```
serie1y := evalf(eval(TinvG, x = y+x0)):

# Trial function
Trial_F:=a0*x+a1*x*ln(-x^2+1)+a2*x*ln(a3*x^2+1)+a4*x*ln(a5*x^2+1)+a6*x*ln(x^2+1);

# Cancellation point 1
epx := 0.7:
Despeje := fsolve(erf(y) = epx, y = 0.6);
CP1 := eval(Trial_F, x = epx) = Despeje;

#Cancellation point 2‘
epx := 0.875:
Despeje := fsolve(erf(y) = epx, y = 0.6);
CP2 := eval(Trial_F, x = epx) = Despeje;

# Cancellation point 3
epx := 0.4:
Despeje := fsolve(erf(y) = epx, y = 0.3);
CP3 := eval(Trial_F, x = epx) = Despeje;


# Taylor series for trial function‘
serie2 := convert(taylor(Trial_F, x = x0, 14), polynom):
serie2y := eval(serie2, x = y+x0):

# System equations
sys := {CP1, CP2,CP3, seq(coeff(serie1y, y, i) = coeff(serie2y, y, i),
i = [1,3,5,7])}:

# Solving the system
solution := fsolve(sys, complex);

Aprox := subs(solution, Trial_F):
# Convert to rational
ERF_I := convert(Aprox, rational, 10);

# Plotting approxmation versus exact function
rango := -1.1 .. 1.1:
AA := implicitplot(F1, x = rango, y = -4 .. 4, grid = [800, 800],color=red, thickness=2,
linestyle = spacedash,legend="Exact"):

bb := plot(ERF_I, x = rango, numpoints = 16000, color = blue, linestyle = solid, thickness = 0,
legend="Inverse Erf(x)"):
display({AA, bb}, view = [rango, -3.5 .. 3.5],  labels = ["x values", "y"],
labelfont = ["HELVETICA", 13], axis = [gridlines = [10, color = gray]]);
```

Appendix B.2. $\widetilde{CDFN}^{1}(x)$ *Maple Code*

```
# CDFN^{-1}(x)
# Code Written for the authors
# Inverse exact function
restart;
with(plots);
rango := 0 .. 1;

# Expansion point
> x0 := 0;
> Digits := 15;
#Inverse exact function
F1 := (1/2)*erf(y/sqrt(2)) = x;


# Derivatives
y0 := evalf((1/2)*erf(x0)/sqrt(2));
d1 := solve(diff((1/2)*erf(y(x)/sqrt(2)), x) = 1, diff(y(x), x));
define(InvG, diff(InvG(x), x) = d1);

d1n := eval(d1, y(x) = y0);
d2 := diff(d1, x);
d2n := simplify(eval(eval(d2, diff(y(x), x) = d1n), y(x) = y0), size);
d3 := diff(d2, x):
d3n := simplify(simplify(eval(eval(d3, diff(y(x), x) = d1n), y(x) = y0)), size);
d4 := diff(d3, x):
d4n := simplify(simplify(eval(eval(d4, diff(y(x), x) = d1n), y(x) = 0)), size);
d5 := diff(d4, x):
```

```
d5n := simplify(simplify(eval(eval(d5, diff(y(x), x) = d1n), y(x) = y0)), size);
d6 := diff(d5, x):
d6n := simplify(simplify(eval(eval(d6, diff(y(x), x) = d1n), y(x) = y0)), size);
d7 := diff(d6, x):
d7n := simplify(simplify(eval(eval(d7, diff(y(x), x) = d1n), y(x) = 1)), size);
d8 := diff(d7, x):
d8n := simplify(simplify(eval(eval(d8, diff(y(x), x) = d1n), y(x) = y0)), size):
d9 := diff(d8, x):
d9n := simplify(simplify(eval(eval(d9, diff(y(x), x) = d1n), y(x) = y0)), size):
d10 := diff(d9, x):
d10n := simplify(simplify(eval(eval(d10, diff(y(x), x) = d1n), y(x) = y0)), size);
d11 := diff(d10, x):
d11n := simplify(simplify(eval(eval(d11, diff(y(x), x) = d1n), y(x) = y0)), size);
NULL;
d12 := diff(d11, x):
d12n := simplify(simplify(eval(eval(d12, diff(y(x), x) = d1n), y(x) = y0)), size);
d13 := diff(d12, x):
d13n := simplify(simplify(eval(eval(d13, diff(y(x), x) = d1n), y(x) = y0)), size);
d14 := diff(d13, x):
d14n := simplify(simplify(eval(eval(d14, diff(y(x), x) = d1n), y(x) = y0)), size);


# ‘Formulation for taylor series exact inverse‘
pol := convert(taylor(InvG(x), x = x0, 15), polynom):
TinvG := simplify(eval(pol, [InvG(x0) = y0, y(x0) = y0, D(y)(x0) = d1n, (D@@2)(y)(x0) = d2n,
        (D@@3)(y)(x0) = d3n, (D@@4)(y)(x0) = d4n, (D@@5)(y)(x0) = d5n, (D@@6)(y)(x0) = d6n,
        (D@@7)(y)(x0) = d7n, (D@@8)(y)(x0) = d8n, (D@@9)(y)(x0) = d9n, (D@@10)(y)(x0) = d10n,
        (D@@11)(y)(x0) = d11n, (D@@12)(y)(x0) = d12n, (D@@13)(y)(x0) = d13n,
        (D@@14)(y)(x0) = d14n])):

# Taylor series for  exact function
serie1y := evalf(eval(TinvG, x = y+x0)):

# Trial function
Trial_F := a0*x+a1*x*ln(-x^2+1)+a2*x*ln(a3*x^2+1)+a4*x*ln(a5*x^2+1):

Cancellation point 1‘
epx := 0.499999999:
Despeje := fsolve((1/2)*erf(y/sqrt(2)) = epx, y = 0.25);
CP1 := eval(Trial_F, x = epx) = Despeje:

# Cancellation point 2‘
epx := 0.375:
Despeje := fsolve((1/2)*erf(y/sqrt(2)) = epx, y = 0.25);
CP2 := eval(Trial_F, x = epx) = Despeje:

# Taylor series for trial function
# serie2 := convert(taylor(Trial_F, x = x0, 14), polynom):
# serie2y := eval(serie2, x = y+x0):

# System equations
sys := { CP1, CP2,seq(coeff(serie1y, y, i) = coeff(serie2y, y, i), i = [1, 3, 5,7])}:

#Solving the system‘
solution := fsolve(sys, complex):

Aprox := subs(solution, Trial_F):
CDFN := Aprox:
CDFN := convert(Aprox, rational, 10);


# By Simplifying with logarithm properties
CDFN := (41788*x)/16671 - 20894/16671 +
        u*ln((-u^2 + 1)^(8035/24017)/((1 + 272396/1253*u^2)^(26/30015707)*(1 -
        423403/105851*u^2)^(3946/5333)));
CDFN := subs(u = x - 1/2, CDFN);

#Plotting approxmationn versus exact~function

F1 := 0.5 + 0.5*erf(sqrt(2)*y/2) = x:
rango := -0 .. 1:
AA := implicitplot(F1, x = rango, y = -4 .. 4, grid = [800, 800],color=red, thickness=0,
      legend="Exact"):

bb := plot(CDFN, x = rango, numpoints = 1600, color = blue,
```

```
      linestyle = spacedash, thickness = 1, legendstyle = [font = ["HELVETICA", 13]], labelfont =
      ["Times,", 14],  axesfont = ["HELVETICA", "ARIAL", 13], legend="Inverse CDFN(x)"):

display({AA, bb}, view = [rango, -3.5 .. 3.5],labels = ["x values", "y values"],
axis = [gridlines = [6, color = gray]], labeldirections = ["horizontal", "vertical"],size=[600,400]);
```

Appendix C. Computing Times

All the numerical experiments were conducted on a PC with an Intel Core Pentium(R) CPU 2117U×2 running at 1.80 GHz clock under Linux Ubuntu 16.04 LTS. The computation code is written in Fortran 77/90, and compiled by the gcc 5.4.0 with level 3 optimization.

Appendix C.1. $\widetilde{erf}^{1}(x)$ *Fortran Code*

```
!===================================================================
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
program Erf_function_time
!
!   Computation time:
!   Written by: Mario-A. Sandoval-Hernandez and Hector~Vazquez-Leal

real :: start, finish, tt
integer iteracion,maxiter,jend,j,i
!integer*8 i
real*8 x,y,wmax,wmin,dw

maxiter=1000000
iteracion=1

jend=1E2
wmin=0.0
wmax=1.0
dw=(wmax-wmin)/dble(jend)

!This code is to measure Gamma inv Upper branch !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        write(*,"( a15, a26, a23, a8)") "--y value ","---x value","--Tiempo ns","Loop"
        write(*,"( a15, a10, a10, a10, a10, a10, a10)") "----------","----------","----------", &
        "----------","----------","----------","-------.--"
OPEN (unit=20, File ='Erf_ser-CPUtime.txt', status ='unknown')

        write(20,"(a10,a13, a26, a23, a8)") "y value  ","x value ","Tiempo ns","Loop"

        do j=1,jend
            x=wmin+dw*dble(j)
           call cpu_time(start)
           do i=1,maxiter
                y= sqrt(-0.4547284088339868D1 - log(dble(-x ** 2 + 1)) / 0.2D1 + &
                   sqrt((0.4547284088339868D1 + log(dble(-x ** 2 + 1)) / 0.2D1) ** 2 - &
                   0.7142857142857143D1 * log(dble(-x ** 2 + 1))))
           enddo

           call cpu_time(finish)
           tt=((finish-start)/maxiter)*1000000000
           write( *,"(0pf24.15, 0pf27.15, 0PF14.8, I7)" )  y, x, tt,j
           write(20,"(0pf24.15, 0pf27.15, 0PF14.8, I7)" )  y, x, tt,j

        enddo
close (20)
end program Erf_function_time
```

Appendix C.2. $\widetilde{\mathrm{erfc}}^{-1}(x)$ *Fortran Code*

```
!===================================================================
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
program Erf_function_time
!
!   Computation time: this code generates a table of CPU time
!   Written by: Mario-A. Sandoval-Hernandez and Hector~Vazquez-Leal

real :: start, finish, tt
integer iteracion,maxiter,jend,j,i
!integer*8 i
real*8 x,y,u, wmax,wmin,dw
```

```
maxiter=1000000
iteracion=1

jend=1E2
wmin=0.0
wmax=1.0
dw=(wmax-wmin)/dble(jend)

!This code is to measure Gamma inv Upper branch !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!11
        write(*,"( a15, a26, a23, a8)") "--y value ","---x value","--Tiempo ns","Loop"
        write(*,"( a15, a10, a10, a10, a10, a10, a10)") "----------","----------","----------", &
        "----------","----------","----------","-------.--"
OPEN (unit=20, File ='Erf_CDF-CPUtime.txt', status ='unknown')

        write(20,"(a10,a13, a26, a23, a8)") "y value  ","x value ","Tiempo ns","Loop"

        do j=1,jend
            x=wmin+dw*dble(j)
           call cpu_time(start)
           do i=1,maxiter
                u = x - 1.0/2.0
                y=  0.41788D5 / 0.16671D5 * x - 0.20894D5 / 0.16671D5 + &
                     dble(u) * log(dble((-u ** 2 + 1) ** (0.8035D4 / 0.24017D5)) * (0.1D1 + &
                     0.272396D6 / 0.1253D4 * dble(u ** 2)) ** (-0.26D2 / 0.30015707D8) * (0.1D1 - &
                     0.423403D6 / 0.105851D6 * dble(u ** 2)) ** (-0.3946D4 / 0.5333D4))

           enddo
           call cpu_time(finish)
           tt=((finish-start)/maxiter)*1000000000
           write( *,"(0pf24.15, 0pf27.15, 0PF14.8, I7)" )  y, x, tt,j
           write(20,"(0pf24.15, 0pf27.15, 0PF14.8, I7)" )  y, x, tt,j
        enddo
close (20)
end program Erf_function_time
```

References

1. Orrego, J.J.M. *Logística de Aprovisionamiento*; Ediciones Paraninfo, S.A.: Madrid, Spain, 2014; Consultado el 12 de Diciembre de 2017.
2. Archibald, R.C. A Rare Pamphlet of Moivre and Some of His Discoveries. *Isis* **1926**, *8*, 671–683. [CrossRef]
3. Patel, J.K.; Read, C.B. *Handbook of the Normal Distribution*, 2nd ed.; Viarcel Dekker, Inc.: New York, NY, USA; Department of Statistics, Southern Methodist University: Dallas, TX, USA, 1996.
4. Sandoval-Hernandez, M.A.; Vazquez-Leal, H.; Filobello-Nino, U.; Hernandez-Martinez, L. New handy and accurate approximation for the Gaussian integrals with applications to science and engineering. *Open Math.* **2019**, *17*, 1774–1793. [CrossRef]
5. Howard, R.M. Arbitrarily accurate analytical approximations for the error function. *Math. Comput. Appl.* **2022**, *27*, 14. [CrossRef]
6. Martila, D.; Groote, S. Analytic error function and numeric inverse obtained by geometric means. *Stats* **2023**, *6*, 431–437. [CrossRef]
7. Vazquez-Leal, H.; Castaneda-Sheissa, R.; Filobello-Nino, U.; Sarmiento-Reyes, A.; Sanchez Orea, J. High accurate simple approximation of normal distribution integral. *Math. Probl. Eng.* **2012**, *2012*, 124029. [CrossRef]
8. Sandoval-Hernandez, M.; Vazquez-Leal, H.; Hernandez-Martinez, L.; Filobello-Nino, U.; Jimenez-Fernandez, V.; Herrera-May, A.; Castaneda-Sheissa, R.; Ambrosio-Lazaro, R.; Diaz-Arango, G. Approximation of Fresnel integrals with applications to diffraction problems. *Math. Probl. Eng.* **2018**, *2018*, 4031793. [CrossRef]
9. Sandoval-Hernández, M.; Hernández-Méndez, S.; Torreblanca-Bouchan, S.E.; Díaz-Arango, G.U. Actualización de contenidos en el campo disciplinar de matemáticas del componente propedéutico del bachillerato tecnológico: El caso de las funciones especiales. *RIDE Rev. Iberoam. Investig. Desarro. Educ.* **2021**, *12*, 1–34. [CrossRef]
10. El Yazidi, Y.; Ellabib, A.; Ouakrim, Y. On the stabilization of singular identification problem of an unknown discontinuous diffusion parameter in elliptic equation. *Comput. Math. Appl.* **2021**, *97*, 267–285. [CrossRef]
11. Vazquez-Leal, H.; Sandoval-Hernandez, M.; Garcia-Gervacio, J.; Herrera-May, A.; Filobello-Nino, U. PSEM approximations for both branches of lambert function with applications. *Discret. Dyn. Nat. Soc.* **2019**, *2019*, 8267951. [CrossRef]
12. Richards, W.A.; Antoine, R.; Sahai, A.; Acharya, M.R. An efficient polynomial approximation to the normal distribution function and its inverse function. *J. Math. Res.* **2010**, *2*, 47. [CrossRef]
13. Milton, R.C.; Hotchkiss, R. Computer evaluation of the normal and inverse normal distribution functions. *Technometrics* **1969**, *11*, 817–822. [CrossRef]

14. Acklam, P.J. An Algorithm for Computing the Inverse Normal Cumulative Distribution Function. 2003. Available online: https://stackedboxes.org/2017/05/01/acklams-normal-quantile-function/ (accessed on 8 April 2026).
15. Wichura, M.J. Algorithm AS 241: The percentage points of the normal distribution. *J. R. Stat. Soc. Ser. C (Appl. Stat.)* **1988**, *37*, 477–484. [CrossRef]
16. Dyer, S.A.; Dyer, J.S. Approximations to error functions. *IEEE Instrum. Meas. Mag.* **2008**, *10*, 45–48. [CrossRef]
17. Wu, D.; Chamoin, L.; Bressan, S. Approximations of the Inverse Cumulative Distribution Function using Transport Maps and Physics-informed Neural Networks. In Proceedings of the AI4X 2025 International Conference, Singapore, 8–11 July 2025. [CrossRef]
18. Schmeiser, B.W. Approximations to the inverse cumulative normal function for use on hand calculators. *J. R. Stat. Soc. Ser. C (Appl. Stat.)* **1979**, *28*, 175–176. [CrossRef]
19. Kabalci, Y. An improved approximation for the Nakagami-m inverse CDF using artificial bee colony optimization. *Wirel. Netw.* **2018**, *24*, 663–669. [CrossRef]
20. Winitzki, S. A Handy Approximation for the Error Function and Its Inverse. 2008. Available online: https://www.academia.edu/9730974/A_handy_approximation_for_the_error_function_and_its_inverse (accessed on 6 May 2026).
21. Vedder, J.D. Simple approximations for the error function and its inverse. *Am. J. Phys.* **1987**, *55*, 762–763. [CrossRef]
22. Wang, Y.G.; Lin, W.H.; Liu, N. A homotopy perturbation-based method for large deflection of a cantilever beam under a terminal follower force. *Int. J. Comput. Methods Eng. Sci. Mech.* **2012**, *13*, 197–201. [CrossRef]
23. Vazquez-Leal, H.; Sarmiento-Reyes, A. Power Series Extender Method for the Solution of Nonlinear Differential Equations. *Math. Probl. Eng.* **2015**, *2015*, 717404. [CrossRef]
24. Zill, D.G. *A First Course in Differential Equations*; Cengage Learning: Boston, MA, USA, 2018.
25. Balser, W. *Formal Power Series and Linear Systems of Meromorphic Ordinary Differential Equations*; Springer: Berlin/Heidelberg, Germany, 2000.
26. Cárdenas, H.; Riera, E.L.; Raggi, F. *Álgebra Superior*; Trillas: Tarragona, Spain, 1990.
27. Barry, D.; Culligan-Hensley, P.; Barry, S. Real values of the W-function. *ACM Trans. Math. Softw. (TOMS)* **1995**, *21*, 161–171. [CrossRef]
28. Daniel, W.W. *Bioestadística, Base Para el Análisis de las Ciencias de la Salud*; UTEHA: Mexico City, Mexico, 2000.
29. Asiminoaei, L.; Teodorescu, R.; Blaabjerg, F.; Borup, U. Implementation and test of an online embedded grid impedance estimation technique for PV inverters. *IEEE Trans. Ind. Electron.* **2005**, *52*, 1136–1144. [CrossRef]
30. Sandoval-Hernández, M.; Vázquez-Leal, H.; Huerta-Chua, J.; Castro-González, F.J.; Filobello-Nino, U.A. Didáctica del graficado de funciones: El caso de las funciones piecewise. *RIDE Rev. Iberoam. Investig. Desarro. Educ.* **2022**, *12*. [CrossRef]
31. Sandoval Hernández, M.A.; Vázquez Leal, H.; Huerta Chua, J.; Filobello Nino, U.A.; Mayorga Cruz, D. La didáctica del cálculo integral: El caso de los procedimientos de integración. *RIDE Rev. Iberoam. Investig. Desarro. Educ.* **2022**, *13*. [CrossRef]
32. Burden, R.L.; Faires, J.D. *Numerical Analysis*; Brooks Cole: Pacific Grove, CA, USA, 2015.
33. Karlsson, H.; Bjerle, I. A simple approximation of the error function. *Comput. Chem. Eng.* **1980**, *4*, 67–68. [CrossRef]
34. Carlitz, L. The inverse of the error function. *Pac. J. Math.* **1963**, *13*, 459–470. [CrossRef]
35. Epperson, J.F. On the Runge Example. *Am. Math. Mon.* **1987**, *94*, 329–341. [CrossRef]
36. Shen, S.; Qian, W.; Zhang, J.; Pan, Y.; Yan, Y.; Shao, C. Matrix method and the suppression of Runge's phenomenon. *SciPost Phys. Core* **2024**, *7*, 034. [CrossRef]