International Journal of Mathematics and Statistics Invention (IJMSI)
E-ISSN: 2321 – 4767 P-ISSN: 2321 - 4759
www.ijmsi.org Volume 14 Issue 6 || Nov. – Dec., 2026 || PP-01-11

A Residual-Based A Posteriori Error Bound for Scalar Caputo Fractional Initial-Value Problems: Rigorous Derivation, Certified Sampling, and Manufactured Verification

Navjeet Kaur
Independent Research Manuscript | Corresponding author: Navjeet Kaur
Manuscript type: Mathematical analysis and verification study

ABSTRACT
This paper presents a transparent residual-based a posteriori error-control framework for scalar initial-value problems governed by a Caputo derivative of order $0<\alpha<1$. The differential equation is written as an equivalent weakly singular Volterra integral equation, and the integral-equation residual of a continuous candidate function is defined explicitly. Under existence of an exact solution, a state-variable Lipschitz condition on the right-hand side, and a certified uniform residual bound, a fractional Grönwall argument gives the estimate $||y-v||_{[0,T]} \le \varepsilon\rho\, E\alpha(LT^{\alpha})$, where $E\alpha$ is the one-parameter Mittag–Leffler function. A separate sampling result converts residual values on a uniform grid into a full-interval bound when an independently justified Lipschitz constant for the residual is available. The framework is demonstrated using the manufactured exact solution $y^*(t)=1+t^2$ and the perturbed candidate $v(t)=1+(1+\delta)t^2$ for four combinations of α and λ. The residual, its derivative bound, and the Mittag–Leffler stability factor are evaluated analytically. The computed a posteriori bounds exceed the exact maximum error in all cases. The study distinguishes a rigorous certificate, which requires verified constants and arithmetic error control, from a heuristic residual indicator. The contribution is a reproducible scalar verification and certification workflow; the fractional Grönwall estimate and the sampling inequality are not claimed as new general theorems.
Keywords: *Caputo fractional derivative; Volterra integral equation; a posteriori error estimate; residual certification; fractional Grönwall inequality; Mittag–Leffler function; manufactured solution.*

Date of Submission: 11-11-2026 Date of Acceptance: 23-11-2026

I. INTRODUCTION

Fractional differential equations are used to represent systems in which the present state depends on the history of the process. Caputo models are common in anomalous transport, viscoelasticity, diffusion, and other applications because their initial conditions can be stated in terms of conventional integer-order values. For orders between zero and one, the Caputo initial-value problem is naturally equivalent, under standard regularity assumptions, to a Volterra integral equation with a weakly singular kernel.

The numerical analysis of such equations has developed substantially. Diethelm, Ford, and Freed established detailed error analysis for a fractional Adams method [1]. Liu, Roberts, and Yan studied graded meshes to address reduced regularity near the initial time [2]. A posteriori error estimation and adaptive-grid strategies for nonlinear fractional equations have also been investigated [3]. These works show that error assessment for fractional problems is an established research area; a residual-based estimate must therefore be positioned carefully and compared with existing numerical-analysis literature.

The contribution is deliberately framed as a verification-oriented synthesis, not as a claim that residual estimators, fractional Grönwall bounds, graded meshes, or Adams PECE methods are new. Existing work already provides a posteriori estimates and adaptive strategies for nonlinear Caputo equations [3], rigorous a posteriori estimators for fractional initial-value problems using GJF-Petrov–Galerkin methods [14], and graded-mesh error analysis for weakly singular solutions [8]. The present manuscript isolates a scalar Volterra-residual inequality, states explicitly the extra regularity and validation information needed to turn sampled residuals into a full-interval certificate, and demonstrates both a smooth Lipschitz-residual certificate and a separate reference-assisted certificate for a fractional-power manufactured solution. The potentially useful distinction is this transparent, reproducible certification workflow for piecewise-linear approximants in a scalar benchmark

setting; priority or broad novelty is not claimed, and a systematic comparison with existing estimators remains necessary.

The paper addresses three practical questions: (i) under what assumptions does the Volterra residual control the solution error; (ii) what additional information is needed to turn sampled residuals into a uniform residual bound; and (iii) does an analytically constructed example satisfy the resulting estimate?

II. PRELIMINARIES AND PROBLEM FORMULATION

2.1 Caputo derivative and equivalent integral equation

Let $T>0$ and $0<\alpha<1$. For an absolutely continuous function $y:[0,T]\to\mathbb{R}$, the Caputo derivative is

$$ {}^{c}D_{0}^{+\alpha}\, y(t) = (1/\Gamma(1-\alpha)) \int_0^t (t-s)^{(-\alpha)}\, y'(s)\, ds, \quad 0<t\le T. \qquad (1)$$

We consider the initial-value problem

$$ {}^{c}D_{0}^{+\alpha}\, y(t) = f(t,y(t)), \quad y(0)=y_0. \qquad (2)$$

Under the standard hypotheses ensuring the equivalence between the Caputo problem and its integral formulation, (2) can be written as

$$ y(t) = y_0 + (1/\Gamma(\alpha)) \int_0^t (t-s)^{(\alpha-1)}\, f(s,y(s))\, ds. \qquad (3)$$

Here Γ denotes the gamma function. The kernel $(t-s)^{(\alpha-1)}$ is integrable but weakly singular at $s=t$.

2.2 Integral-equation residual

For a candidate approximation $v\in C[0,T]$ satisfying $v(0)=y_0$, define its Volterra residual by

$$ \rho_v(t) = v(t)-y_0-(1/\Gamma(\alpha))\int_0^t (t-s)^{(\alpha-1)} f(s,v(s))\, ds. \qquad (4)$$

This is an integral-equation residual. It is not automatically identical to the differential residual ${}^{c}D_{0}^{+\alpha}v-f(t,v)$. A conversion between the two requires sufficient regularity and a justified fractional-operator calculation. The distinction matters: a small differential residual or a small set of sampled values does not, by itself, establish a uniform bound on (4).

For a continuous function q, write $\|q\|_\infty = \max_{0\le t\le T}|q(t)|$. A certified residual estimate is a rigorously justified number ε_ρ such that $\|\rho_v\|_\infty\le\varepsilon_\rho$.

III. RESIDUAL-TO-SOLUTION ERROR ESTIMATE

3.1 Assumptions

Assume that an exact solution $y\in C[0,T]$ of (3) exists. Assume that f is continuous in t and Lipschitz in its second argument on the ranges traversed by y and v: there is $L\ge 0$ such that $|f(t,u)-f(t,w)|\le L|u-w|$. Finally, assume that a certified $\varepsilon_\rho\ge 0$ is available with $|\rho_v(t)|\le\varepsilon_\rho$ for every $t\in[0,T]$.

3.2 Theorem 1

Under these assumptions, the solution error satisfies

$$ |y(t)-v(t)| \le \varepsilon_\rho\, E_\alpha(Lt^{\alpha}), \quad 0\le t\le T, \qquad (5)$$

where $E_\alpha(z)=\sum_{k=0}^{\infty} z^k/\Gamma(\alpha k+1)$ is the one-parameter Mittag–Leffler function.

Proof. From (3) and (4), subtract the two integral relations and use the triangle inequality. With $e(t)=|y(t)-v(t)|$,

$$ e(t) \le \varepsilon_\rho + (L/\Gamma(\alpha))\int_0^t (t-s)^{(\alpha-1)} e(s)\, ds. \qquad (6)$$

The fractional Grönwall inequality applied to (6) yields $e(t)\le\varepsilon_\rho E_\alpha(Lt^{\alpha})$, which proves (5). □

Taking the maximum over the interval gives the convenient global form

$$ \|y-v\|_\infty \le \varepsilon_\rho\, E_\alpha(LT^{\alpha}), \text{ since } E_\alpha(Lt^{\alpha}) \text{ is nondecreasing for } L\ge 0. \qquad (7)$$

The estimate is conditional. It is a rigorous certificate only when both L and ε_ρ are rigorously bounded. If either is obtained from unvalidated numerical fitting, the result should be described as an error indicator rather than a certified bound.

IV. CERTIFYING THE RESIDUAL FROM GRID VALUES

4.1 Uniform-grid sampling inequality

Let $\rho:[0,T]\to\mathbb{R}$ be K-Lipschitz, meaning $|\rho(t)-\rho(s)|\le K|t-s|$ for all $s,t\in[0,T]$. Let $t_j=jh$, $j=0,\ldots,N$, with $h=T/N$, and set $R_N=\max_{0\le j\le N}|\rho(t_j)|$.

$$\|\rho\|_\infty \le R_N + Kh/2. \tag{8}$$

Proof. For any $t\in[0,T]$, at least one grid node t_j is within distance $h/2$. Hence $|\rho(t)|\le|\rho(t_j)|+K|t-t_j|\le R_N+Kh/2$. Taking the supremum over t proves (8). □

The inequality is elementary; its practical value lies in how K is obtained. A maximum slope computed from the same grid samples is not, in general, a proof of a Lipschitz bound between nodes. K must be established analytically or by a validated derivative bound.

4.2 Incorporating quadrature and arithmetic uncertainty

In a computational implementation, the residual at a node is rarely evaluated exactly. Suppose a validated computation provides numbers r_j and certified nodewise error allowances η_j such that $|\rho(t_j)-r_j|\le\eta_j$. Then

$$\|\rho\|_\infty \le \max_j(|r_j|+\eta_j) + Kh/2. \tag{9}$$

The η_j terms must account for all relevant numerical effects, including quadrature truncation, approximation of special functions, and floating-point rounding. If the residual integral is evaluated by a numerical quadrature rule without a rigorous error estimate, the resulting value is an empirical residual estimate, not a mathematical certificate.

Combining (7) and (9) gives the practical certification pipeline: establish L; derive an independent K; compute node residuals with validated η_j; form $\varepsilon\rho=\max_j(|r_j|+\eta_j)+Kh/2$; and finally report $\varepsilon\rho E\alpha(LT^\alpha)$ as the global solution-error bound.

V. MANUFACTURED VERIFICATION EXAMPLE

5.1 Exact solution and model problem

On [0,1], choose $y^*(t)=1+t^2$. For $0<\alpha<1$, its Caputo derivative is

$${}^cD_{0^+}^{\alpha}\, y^*(t) = 2t^{(2-\alpha)}/\Gamma(3-\alpha). \tag{10}$$

For $\lambda\ge0$, define

$${}^cD_{0^+}^{\alpha}\, y(t)=\lambda y(t)+g(t), \quad y(0)=1, \tag{11}$$

$$g(t)=2t^{(2-\alpha)}/\Gamma(3-\alpha)-\lambda(1+t^2). \tag{12}$$

By construction, y^* is an exact solution. The function $f(t,y)=\lambda y+g(t)$ is Lipschitz in y with constant $L=\lambda$.

5.2 Perturbed candidate and exact residual

For $\delta>0$, take $v(t)=1+(1+\delta)t^2$, so the exact error is $e(t)=v(t)-y^*(t)=\delta t^2$ and $\|e\|_\infty=\delta$. Subtracting the Volterra equations for v and y^* gives

$$\rho_v(t)=\delta t^2-(\lambda\delta/\Gamma(\alpha))\int_0^t (t-s)^{(\alpha-1)}s^2\, ds. \tag{13}$$

Using the fractional-integral identity $I^\alpha(t^m)=\Gamma(m+1)t^{(m+\alpha)}/\Gamma(m+\alpha+1)$, we obtain

$$\rho_v(t)=\delta t^2-\lambda\delta\Gamma(3)t^{(2+\alpha)}/\Gamma(3+\alpha). \tag{14}$$

For the parameter range used below, the derivative is nonnegative on [0,1]. Indeed,

$$\rho_v'(t)=\delta t[2-\lambda\Gamma(3)(2+\alpha)t^{\alpha}/\Gamma(3+\alpha)] \ge 0, \tag{15}$$

because $\lambda\Gamma(3)(2+\alpha)/\Gamma(3+\alpha)<2$ for $\alpha\in\{0.5,0.8\}$ and $\lambda\in\{0.25,0.5\}$. Thus $\|\rho_v\|_\infty=\rho_v(1)=\delta[1-\lambda\Gamma(3)/\Gamma(3+\alpha)]$.

5.3 Independent Lipschitz constant for the residual

Differentiating (14) and using $0\le t\le1$ gives the valid bound

$$K = \delta[2+\lambda\Gamma(3)(2+\alpha)/\Gamma(3+\alpha)], \quad \text{so} \quad |\rho_v'(t)|\le K. \tag{16}$$

The mean value theorem therefore establishes that ρ_v is K-Lipschitz. This supplies the independent constant required by the grid-sampling inequality. For illustration, with N=20 and h=0.05, the grid-based residual

certificate is R_N+K(0.025). Because t=1 is a grid node and the residual is increasing, R_N equals the exact residual norm in this particular manufactured example; the additional Kh/2 term is a conservative sampling allowance.

VI. RESULTS AND DISCUSSION

Set δ=0.01, T=1, α∈{0.5,0.8}, and λ∈{0.25,0.5}. The Mittag–Leffler values are evaluated from its defining series. The residual norms follow directly from (14); no numerical time-stepping solver is used.

α	λ	True error δ	$\|\|\rho_v\|\|\infty$	Eα(λ)	Theorem bound	K (Lipschitz)
0.5	0.25	0.010000	0.00849549	1.35864237	0.01154234	0.02376126
0.5	0.50	0.010000	0.00699099	1.95236049	0.01364893	0.02752253
0.8	0.25	0.010000	0.00893485	1.31792408	0.01177545	0.02298242
0.8	0.50	0.010000	0.00786970	1.76320367	0.01387588	0.02596484

Table 1. Exact manufactured errors, residual norms, stability factors, a posteriori bounds, and analytically derived residual Lipschitz constants.

In all four cases, the theorem bound is larger than the exact maximum error of 0.01, as required by the estimate. The bounds range from 0.01154234 to 0.01387588. Increasing λ decreases the residual norm in this constructed family but increases the Mittag–Leffler stability factor; consequently, the final error bound does not decrease. This illustrates why a residual magnitude should not be interpreted independently of the stability factor.

6.1 Grid-certificate illustration

For N=20 (h=0.05), the sampling allowance is Kh/2=0.025K. The resulting residual upper bounds and propagated solution-error bounds are:

α	λ	R_N+Kh/2	Eα(λ)	Grid-based error bound
0.5	0.25	0.00908953	1.35864237	0.01234942
0.5	0.50	0.00767905	1.95236049	0.01499228
0.8	0.25	0.00950941	1.31792408	0.01253268
0.8	0.50	0.00851882	1.76320367	0.01502042

Table 2. Conservative bounds obtained from the grid-sampling inequality with N=20. The grid includes t=1, so R_N is the exact residual maximum for this monotone example; the Kh/2 term remains an additional conservative allowance.

This example verifies the internal consistency of the derived inequalities for an analytically prescribed candidate. It does not establish convergence order, computational efficiency, robustness under noisy data, or performance of a particular numerical integrator. Those claims would require an implemented scheme, mesh-refinement studies, independent benchmark solutions, and a reproducible computational protocol.

VII. COMPUTATIONAL METHODS AND REPRODUCIBLE EXPERIMENTS

To assess the behavior of the analytical estimate against computed approximations, two independently coded time-discretisation methods were applied to the same manufactured problem: the classical L1 finite-difference formula and the fractional Adams–Bashforth–Moulton predictor–corrector (PECE) method. The implementations use separate discretisation formulas and do not call a common fractional-equation solver library. Both methods were tested on uniform meshes with N=50, 100, 200, 400, and 800 subintervals, T=1, α∈{0.5,0.8}, and λ∈{0.25,0.5}.

7.1 L1 finite-difference scheme

For t_n=nh and h=T/N, the Caputo derivative is approximated by the L1 convolution formula

$$D_C^{\alpha} y(t_n) \approx [h^{(-\alpha)}/\Gamma(2-\alpha)] \Sigma_{k=0}^{n-1} a_k (y_{n-k}-y_{n-k-1}), \quad a_k=(k+1)^{(1-\alpha)}-k^{(1-\alpha)}.$$

For the linear test equation $D_C^{\alpha}y=\lambda y+g(t)$, the unknown y_n appears linearly and is solved at each time level by rearranging the current-increment term. The remaining history increments are known from previous time levels. The implementation uses the full history sum, so its computational work is $O(N^2)$ and its storage is $O(N)$.

7.2 Fractional Adams predictor–corrector (PECE)

The second method is the product-integration predictor–corrector algorithm of Diethelm, Ford, and Freed [7]. The predictor uses piecewise-constant product integration of the Volterra equation, while the corrector uses a product-trapezoidal update. The predicted endpoint value is inserted once into the corrector (PECE: Predict–Evaluate–Correct–Evaluate). The method is implemented directly from the published convolution weights; the corrected value is then used to evaluate the right-hand side for the next step.

7.3 Mesh-refinement protocol and error metric

At each grid point, the numerical solution y_n was compared with the exact value $y^*(t_n)=1+t_n^2$. The reported maximum-norm error is $E_N=\max_{0\le n\le N}|y_n-y^*(t_n)|$. The observed order between consecutive meshes is $p_N=\log_2(E_N/E_{2N})$. This is an empirical refinement order, not a proof of an asymptotic convergence theorem. All experiments use the same manufactured forcing $g(t)=2t^{2-\alpha}/\Gamma(3-\alpha)-\lambda(1+t^2)$.

7.4 L1 mesh-refinement results

α	λ	N	Maximum error E_N	Observed p_N
0.5	0.25	50	1.80863×10^{-3}	—
0.5	0.25	100	6.49579×10^{-4}	1.4773
0.5	0.25	200	2.32145×10^{-4}	1.4845
0.5	0.25	400	8.26870×10^{-5}	1.4893
0.5	0.25	800	2.93854×10^{-5}	1.4926
0.5	0.50	50	2.39098×10^{-3}	—
0.5	0.50	100	8.59837×10^{-4}	1.4755
0.5	0.50	200	3.07571×10^{-4}	1.4831
0.5	0.50	400	1.09624×10^{-4}	1.4884
0.5	0.50	800	3.89757×10^{-5}	1.4919
0.8	0.25	50	8.75795×10^{-3}	—
0.8	0.25	100	3.83029×10^{-3}	1.1931
0.8	0.25	200	1.67194×10^{-3}	1.1959
0.8	0.25	400	7.28959×10^{-4}	1.1976
0.8	0.25	800	3.17601×10^{-4}	1.1986
0.8	0.50	50	1.05157×10^{-2}	—
0.8	0.50	100	4.59719×10^{-3}	1.1937
0.8	0.50	200	2.00653×10^{-3}	1.1960
0.8	0.50	400	8.74851×10^{-4}	1.1976
0.8	0.50	800	3.81179×10^{-4}	1.1986

Table 3. L1 scheme: maximum nodal errors and observed refinement orders.

7.5 Fractional Adams PECE mesh-refinement results

α	λ	N	Maximum error E_N	Observed p_N
0.5	0.25	50	5.94419×10^{-4}	—
0.5	0.25	100	2.16073×10^{-4}	1.4600
0.5	0.25	200	7.78682×10^{-5}	1.4724
0.5	0.25	400	2.78993×10^{-5}	1.4808
0.5	0.25	800	9.95614×10^{-6}	1.4866
0.5	0.50	50	1.56378×10^{-3}	—
0.5	0.50	100	5.62406×10^{-4}	1.4754
0.5	0.50	200	2.01140×10^{-4}	1.4834
0.5	0.50	400	7.16761×10^{-5}	1.4886
0.5	0.50	800	2.54800×10^{-5}	1.4921
0.8	0.25	50	8.67694×10^{-5}	—
0.8	0.25	100	2.66120×10^{-5}	1.7051
0.8	0.25	200	8.11816×10^{-6}	1.7129
0.8	0.25	400	2.46169×10^{-6}	1.7215
0.8	0.25	800	7.41991×10^{-7}	1.7302
0.8	0.50	50	2.72340×10^{-4}	—
0.8	0.50	100	8.05720×10^{-5}	1.7571
0.8	0.50	200	2.37829×10^{-5}	1.7604
0.8	0.50	400	7.00404×10^{-6}	1.7637
0.8	0.50	800	2.05794×10^{-6}	1.7670

Table 4. Fractional Adams PECE scheme: maximum nodal errors and observed refinement orders.

7.6 Comparative interpretation

Both schemes show decreasing maximum error under mesh refinement for all four parameter combinations. For $\alpha=0.5$, the L1 observed order approaches approximately 1.5, consistent with the smooth-solution rate $2-\alpha$. For $\alpha=0.8$, the L1 order approaches approximately 1.2, again consistent with $2-\alpha$. The PECE results show observed orders increasing toward approximately $1+\alpha$: the finest-mesh values are about 1.49 for $\alpha=0.5$ and 1.73–1.77 for $\alpha=0.8$. The latter have not fully reached the nominal asymptotic value 1.8 on the tested mesh range.

For this smooth manufactured problem, PECE produces smaller nodal errors than L1 at the same N for all tested parameter combinations. This is an empirical comparison for this problem and these mesh sizes; it is not a universal ranking of the methods. The methods have different constants, convolution weights, and computational costs. Both implementations use direct history summation and therefore have quadratic time complexity in N; a fair performance comparison would additionally measure wall-clock time, memory, and implementation environment.

The results support numerical consistency with the expected smooth-solution rates, but they do not prove a general convergence theorem. The test solution $y^*(t)=1+t^2$ is smooth and has $y^{*\prime}(0)=0$; it does not probe the weak initial-time singularities common in fractional differential equations. A second manufactured solution with a fractional-power term, together with graded meshes, is required before drawing conclusions about nonsmooth solutions. Relevant analyses emphasize that solution regularity and mesh grading can materially alter the observed order [2,3].

7.7 Reproducibility and implementation status

A separate Python script accompanies this manuscript and implements the L1 and PECE schemes, generates the mesh-refinement tables, and computes the observed orders from the exact manufactured solution. The numerical values in Tables 3 and 4 are generated by that script. The implementation is a research prototype using double-precision arithmetic; it is not an interval-arithmetic or validated-numerics certificate. Consequently, the numerical experiment establishes reproducible empirical errors, not a rigorous computer-assisted proof.

VIII. NONSMOOTH SOLUTIONS AND GRADED-MESH EXPERIMENTS

The smooth polynomial test does not expose the weak regularity often encountered in fractional initial-value problems. To examine this issue, a second manufactured solution was introduced: $y^*(t)=1+t^q$ with $q=0.9$. Its Caputo derivative is $\Gamma(q+1)t^{q-\alpha}/\Gamma(q+1-\alpha)$, and the forcing was selected as $g(t)=\Gamma(q+1)t^{q-\alpha}/\Gamma(q+1-\alpha)-\lambda(1+t^q)$. This produces an exact solution while allowing $y'(t)=q\,t^{q-1}$ to be unbounded as t approaches zero.

The nonuniform L1 formula was implemented on graded meshes $t_n=(n/N)^r$, with grading exponents $r=1$ (uniform), 1.5, 2, and 3. The PECE method was evaluated on uniform meshes only; no nonuniform PECE implementation is claimed. The tested values were $\alpha\in\{0.5,0.8\}$, $\lambda=0.25$, $q=0.9$, and $N\in\{50,100,200,400,800\}$.

At N=800, the maximum errors and the observed orders between N=400 and N=800 are summarised below.

α	Method	Grading r	Error at N=800	Order 400→800
0.5	L1 uniform	1.0	9.675e-05	0.905
0.5	L1 graded	1.5	6.187e-06	1.356
0.5	L1 graded	2.0	5.127e-06	1.493
0.5	L1 graded	3.0	7.362e-06	1.489
0.5	PECE uniform	1.0	4.664e-04	0.921
0.8	L1 uniform	1.0	2.125e-04	0.906
0.8	L1 graded	1.5	7.207e-05	1.183
0.8	L1 graded	2.0	6.364e-05	1.198
0.8	L1 graded	3.0	7.489e-05	1.198
0.8	PECE uniform	1.0	9.700e-04	0.902

Table 5. Nonsmooth manufactured solution: uniform and graded-mesh errors at the finest tested mesh.

For both fractional orders, graded L1 meshes reduce the maximum error relative to the uniform L1 mesh at the same N. For $\alpha=0.5$, $r=2$ reduces the error from approximately 9.70×10^{-5} to 5.0×10^{-6}. For $\alpha=0.8$, $r=2$ reduces it from approximately 2.13×10^{-4} to 6.4×10^{-5}. The observed order of uniform L1 and uniform PECE is close to 0.9, reflecting the reduced regularity of this manufactured solution. Graded L1 meshes recover higher observed orders, approaching approximately $2-\alpha$ for $r=2$ in this experiment.

These results are numerical evidence for this selected fractional-power test, not a universal statement that $r=2$ is optimal. The best grading exponent depends on α, the singularity exponent q, and the numerical method. A broader parameter sweep is needed to establish robust grading recommendations.

IX. A POSTERIORI RESIDUAL VALIDATION

The numerical Volterra residual was evaluated for the piecewise-linear interpolant through each computed solution. At selected grid nodes t_i, the residual was computed as $\rho_v(t_i)=v(t_i)-1-\Gamma(\alpha)^{-1}\int_0^{t_i}(t_i-s)^{\alpha-1}[\lambda v(s)+g(s)]ds$. The weakly singular integral was evaluated by adaptive quadrature, with the interpolation knots supplied as integration breakpoints. This diagnostic pertains to the weak-regularity $q=0.9$ experiment and is retained as an empirical comparison only; it is not used as the certified result below.

For an empirical sampling estimate, residual values were sampled at every second mesh node. The largest sampled residual R_N was augmented by $K_{emp}\,h_s/2$, where K_{emp} is the maximum absolute slope between adjacent sampled residual values and h_s is the largest sampling gap. The resulting ε_{emp} was propagated through $\varepsilon_{emp}\,E_\alpha(\lambda T^\alpha)$. Because K_{emp} is inferred from sampled slopes rather than independently proved to bound the residual derivative between samples, this is an empirical indicator and not a rigorous certificate.

α	Method	N	Sampled residual	K_emp	Empirical error bound	Actual error	Bound/error
0.5	L1	100	5.606e-04	2.803e-02	1.142e-03	6.379e-04	1.79
0.5	PECE	100	1.964e-03	9.822e-02	4.003e-03	3.223e-03	1.24
0.8	L1	100	1.371e-03	6.418e-02	2.653e-03	1.556e-03	1.70
0.8	PECE	100	5.191e-03	2.595e-01	1.026e-02	6.351e-03	1.62

Table 6. Empirical residual-based error indicators for q=0.9, λ=0.25, N=100. The error bound column is not certified because K_emp is not independently validated.

In all four runs, the empirical propagated bound exceeds the actual maximum nodal error, with ratios between approximately 1.24 and 1.79. This is a useful consistency check for the chosen experiment. It does not establish reliability for arbitrary data or meshes. A rigorous residual certificate would require a proven bound on $|\rho'_v(t)|$ (or a direct interval enclosure of ρ_v), validated quadrature errors, and outward-rounded arithmetic.

X. INTERVAL-ARITHMETIC RESIDUAL CERTIFICATION FOR THE SMOOTH TEST

This section establishes a rigorous a posteriori certificate for the smooth manufactured problem. The certificate combines interval-enclosed residual values at the mesh nodes with an analytically justified global derivative bound, a sampling inequality, and the fractional Grönwall stability factor. The result applies to the specified piecewise-linear interpolant and test parameters; it is not asserted for the nonsmooth test or for arbitrary IVPs.

Table 8. Interval-certified residual and propagated solution-error upper bounds for the smooth manufactured problem. The observed error is shown only as a posteriori consistency evidence; the certificate itself is the upper bound column.

α	N	K upper	$\|\rho\|_\infty$ upper	Error upper	Observed error
0.5	50	3.4466×10^{-2}	1.7542×10^{-3}	2.3834×10^{-3}	1.8086×10^{-3}
0.5	100	1.7192×10^{-2}	5.9355×10^{-4}	8.0642×10^{-4}	6.4958×10^{-4}
0.5	200	8.5814×10^{-3}	2.0320×10^{-4}	2.7608×10^{-4}	2.3214×10^{-4}
0.8	50	4.2453×10^{-2}	7.8097×10^{-3}	1.0293×10^{-2}	8.7579×10^{-3}
0.8	100	2.1189×10^{-2}	3.3377×10^{-3}	4.3988×10^{-3}	3.8303×10^{-3}
0.8	200	1.0585×10^{-2}	1.4376×10^{-3}	1.8946×10^{-3}	1.6719×10^{-3}

For all six runs, the interval-propagated upper bound exceeds the independently known maximum nodal error. The bound-to-observed-error ratio decreases under refinement, from 1.32 to 1.15 for α=0.5 and from 1.18 to 1.13 for α=0.8. These are genuine certificates for the stated smooth manufactured test and the represented floating-point interpolant, subject to the stated interval-arithmetic implementation. They do not yet certify the nonsmooth q=0.9 experiment or arbitrary user-supplied IVPs.

A global Lipschitz constant follows from $\rho'_v=e'-\lambda I^{\alpha}(e')$, because e(0)=0. If M=max over all cells of $\max(|m_j-2t_j|,|m_j-2t_{j+1}|)$, where m_j is the exact slope of v_N on cell j, then $\|e'\|_\infty\le M$ and $\|\rho'_v\|_\infty\le M[1+\lambda/\Gamma(\alpha+1)]$ on [0,1]. The maximum is attained at a cell endpoint because e′ is affine on each cell. The sampling inequality then gives $\|\rho_v\|_\infty\le R_N+K\,h/2$, where R_N is the largest upper endpoint of the interval-enclosed nodal residual magnitudes and K is an interval upper bound for the analytic derivative estimate. The Mittag–Leffler factor $E_\alpha(\lambda)$ was bounded above by interval summation with a geometric tail bound; monotonicity of successive term ratios follows from log-convexity of Γ.

On each mesh cell, e is a quadratic polynomial. Its fractional integral was therefore evaluated by an exact antiderivative after the substitution u=t−s. All nodal data and mesh points were interpreted as exact binary floating-point rationals; the polynomial coefficients were formed as exact rational numbers. The resulting residual values at every mesh node were enclosed using 50-decimal-digit interval arithmetic, including interval evaluation of the gamma function and fractional powers.

A rigorous certificate was constructed for the smooth manufactured solution $y^*(t)=1+t^2$ on [0,1], with $\alpha\in\{0.5,0.8\}$ and λ=0.25. Let v_N be the piecewise-linear interpolant of the computed L1 nodal values and

e=v_N−y*. Since g(t)=D_C^αy*(t)−λy*(t), the Volterra residual satisfies the exact identity ρ_v(t)=e(t)−λ(I^αe)(t). This identity avoids numerically integrating the singular forcing term separately.

10.1 REFERENCE-ASSISTED INTERVAL CERTIFICATION FOR THE NONSMOOTH TEST

The fractional-power manufactured solution y*(t)=1+t^q, q=0.9, is not Lipschitz at t=0. Consequently, the smooth-case residual derivative argument in Section 10 cannot be transferred unchanged: y*′(t)=q t^(q−1) is unbounded at the initial point. For this case we use a different, deliberately conservative route that does not assume a global residual Lipschitz constant.

Let v_N be the piecewise-linear interpolant of the computed L1 nodal values and e=v_N−y*. For the manufactured equation, the Volterra residual satisfies ρ_v=e−λ I^α e. First, an interval enclosure E_N≥||e||_∞ is obtained cell by cell. On a cell [t_j,t_(j+1)], v_N(t)=m_j t+b_j, so e′(t)=m_j−q t^(q−1). Since this derivative is strictly increasing for 0<q<1, the only possible interior extremum occurs at t*=(q/m_j)^(1/(1−q)) when m_j>0 and t* lies in the cell. The absolute error is therefore bounded by interval evaluation at the two endpoints and this possible critical point. The tabulated computations use q=9/10, so the critical-point exponent is exactly 10.

The fractional integral operator satisfies ||I^α e||_∞≤T^α||e||_∞/Γ(α+1). Hence ||ρ_v||_∞≤E_N[1+λT^α/Γ(α+1)]. Applying Theorem 1 gives ||y−v_N||_∞≤E_N[1+λT^α/Γ(α+1)]E_α(λT^α). Interval arithmetic is used for the critical-point evaluations, gamma factors, and Mittag–Leffler series upper enclosure. The nodal approximations are treated as the exact binary floating-point numbers that define the reported piecewise-linear interpolant.

This is a rigorous manufactured-solution, reference-assisted enclosure for the specified computed interpolant, not a solution-independent a posteriori estimator: it uses the known exact function y*(t) to enclose ||v_N−y*||_∞. It therefore demonstrates how interval arithmetic can certify the benchmark error and the associated Volterra residual, but it must not be presented as an automatic certification procedure for an unknown exact solution.

Table 9. Reference-assisted interval enclosures for the nonsmooth manufactured solution (q=0.9, λ=0.25, T=1).

α	N	Enclosure E_N	Residual upper bound	Propagated error upper bound	Observed nodal error
0.5	50	0.001665	0.002134	0.0029	1.202e-03
0.5	100	0.0008888	0.00114	0.001549	6.379e-04
0.5	200	0.0004751	0.0006091	0.0008276	3.396e-04
0.5	400	0.0002542	0.0003259	0.0004427	1.811e-04
0.5	800	0.0001361	0.0001744	0.000237	9.675e-05
0.8	50	0.003083	0.00391	0.005154	3.082e-03
0.8	100	0.001557	0.001975	0.002602	1.556e-03
0.8	200	0.0007774	0.000986	0.0013	7.773e-04
0.8	400	0.0004047	0.0005133	0.0006765	3.983e-04
0.8	800	0.0002162	0.0002742	0.0003613	2.125e-04

For every reported α and N, the propagated interval upper bound exceeds the observed nodal error. The bounds decrease with refinement. They are conservative, with bound-to-observed-error ratios approximately 2.41–2.45 for α=0.5 and 1.67–1.70 for α=0.8. The observed error column is included as a consistency check, not as an ingredient in the enclosure.

XI. PERFORMANCE AND RESOURCE BENCHMARKING

Runtime and memory were measured for the two uniform-grid implementations at N=200, 400, 800, and 1600, using α∈{0.5,0.8} and λ=0.25. Each case was warmed up, timed three times, and the median wall-clock time was retained. Python memory was measured separately using tracemalloc; it reports traced Python allocations rather than total process resident memory. The following representative results are for α=0.5.

Method	N	Median runtime (s)	Peak traced memory (KiB)
L1	200	0.0055	11.2
L1	400	0.0109	20.0
L1	800	0.0136	42.4
L1	1600	0.0220	87.1
PECE	200	0.0078	18.6
PECE	400	0.0165	35.8
PECE	800	0.0383	70.2
PECE	1600	0.0940	138.9

Table 7. Representative performance measurements for α=0.5 and λ=0.25 in the current Python execution environment.

The L1 implementation was faster than PECE at all four tested mesh sizes in this particular implementation and environment. Both methods store the solution and right-hand-side history and use direct convolution sums, giving O(N) storage and $O(N^2)$ arithmetic work in the straightforward implementation. The observed wall-clock values are environment-dependent and should not be treated as portable performance claims. For a journal-grade benchmark, the machine model, processor, Python and library versions, operating system, and repeated-run variability should be reported. Fast convolution or sum-of-exponentials acceleration was not implemented.

The measured allocation values grow approximately linearly with N, consistent with storage of arrays and history values. The traced-memory measurement is not a complete measure of interpreter or native-library memory and is therefore reported only as a comparative implementation diagnostic.

XII. PECE IMPLEMENTATION CHECK AND COMPUTING ENVIRONMENT

The PECE implementation was checked against a separately written, direct-index implementation of the published fractional Adams predictor–corrector weights. The reference equations were taken from the fde12 description reproduced in the FDE-TestSet documentation: predictor weights b_k=(k+1)^α−k^α; corrector starting weight a^0_n=(n−1)^(α+1)−n^α(n−α−1); and second-difference weights a_k=(k+1)^(α+1)−2k^(α+1)+(k−1)^(α+1). The independent implementation uses explicit summation over the published index ranges rather than the vectorized weight-array construction in the primary PECE routine.

For $\alpha\in\{0.5,0.8\}$, $\lambda\in\{0.25,0.5\}$, and $N\in\{50,100,200,400\}$, the maximum pointwise difference between the two implementations was between 1.3×10^{-15} and 2.1×10^{-14} in double precision. This is a formula-level cross-check of two independently indexed implementations. It is not yet a black-box comparison against Roberto Garrappa's external fde12.m source code. That external comparison remains a reproducibility task because the reference MATLAB file was not executed in the present environment.

The external reference selected for black-box validation is Roberto Garrappa's fde12 implementation, MATLAB Central File Exchange item 32918, version 2.0.0 (updated 2 May 2025) [10]. The official listing identifies MATLAB as a requirement, and its download route requires MATLAB Central sign-in in the present environment. MATLAB and GNU Octave were not available for this execution; therefore, no external fde12 run or numerical comparison is claimed. The accompanying MATLAB driver calls fde12 directly once the official file is installed on the MATLAB path. Until that driver is executed and its output archived, the external solver validation remains pending.

Computing environment for the experiments reported here: Python 3.13.5; NumPy 2.3.5; SciPy 1.17.0; pandas 2.2.3; Linux 6.18.57 x86_64; GCC 14.2.0 runtime build. Timings were collected using Python perf_counter after one warm-up run and the median of three timed runs. Python allocation measurements used tracemalloc and do not represent total resident memory. The processor model was not exposed by the execution environment and is therefore not reported.

For the smooth manufactured case $y^*(t)=1+t^2$, Section 10 now supplies an interval-arithmetic residual certificate based on an analytic global derivative bound and exact piecewise-polynomial integration. The separate q=0.9 experiment in Section 9 remains empirical because its residual regularity at t=0 requires a

different Hölder-modulus treatment. The external fde12 black-box comparison also remains pending and is not represented as completed.

XIII. LIMITATIONS AND SCOPE

The result is a conditional a posteriori estimate, not a new fractional Grönwall theorem. Its practical certification depends on independently verified values of L, K, quadrature-error allowances, and arithmetic-error allowances. The manufactured problem is linear and deliberately constructed to have a known exact solution; it is a verification test rather than evidence of performance on a broad class of nonlinear models. No adaptive algorithm or time-stepping solver is assessed in this manuscript.

A stronger computational study would implement at least one established fractional solver, report mesh-refinement results for multiple manufactured solutions (including a solution with reduced regularity near t=0), compare the residual estimator with the observed error, and provide code and parameter files. Any claim of novelty should be made only after a systematic comparison with the existing a posteriori and fractional numerical-analysis literature.

XIV. CONCLUSION

A scalar Caputo initial-value problem can be recast as a weakly singular Volterra integral equation, and a uniform bound on its integral-equation residual controls the solution error through a fractional Grönwall inequality. This stability mechanism is established theory, not a new theorem of this paper. The manuscript contributes a transparent scalar presentation of the residual and sampling assumptions, a rigorous interval certificate for the smooth manufactured interpolants, and a separate reference-assisted interval enclosure for a fractional-power test where the residual is not globally Lipschitz at the initial point. The nonsmooth enclosure is benchmark-specific and uses the known exact manufactured solution; it is not a general-purpose estimator for unknown solutions. The numerical PECE formula cross-check remains internal, while an execution-based comparison with the official external fde12 release is still pending. These limitations define the present work as a verification study and delimit the claims that can responsibly be made.

REFERENCES

[1] K. Diethelm, N. J. Ford, and A. D. Freed, “Detailed error analysis for a fractional Adams method,” Numerical Algorithms, vol. 36, pp. 31–52, 2004. https://doi.org/10.1023/B:NUMA.0000027736.85078.be

[2] Y. Liu, J. Roberts, and Y. Yan, “Detailed error analysis for a fractional Adams method with graded meshes,” Numerical Algorithms, vol. 78, pp. 1195–1216, 2018. https://doi.org/10.1007/s11075-017-0419-5

[3] L.-B. Liu and Y. Chen, “A posteriori error estimation and adaptive strategy for a nonlinear fractional differential equation,” International Journal of Computer Mathematics, vol. 99, no. 2, pp. 240–246, 2022. https://doi.org/10.1080/00207160.2021.1906420

[4] J. R. L. Webb, “A fractional Gronwall inequality and the asymptotic behaviour of global solutions of Caputo fractional problems,” Electronic Journal of Differential Equations, vol. 2021, no. 80, pp. 1–22, 2021. https://doi.org/10.58997/ejde.2021.80

[5] M. Stynes, “A survey of the L1 scheme in the discretisation of time-fractional problems,” Numerical Mathematics: Theory, Methods and Applications, vol. 15, no. 4, pp. 1173–1192, 2022. https://doi.org/10.4208/nmtma.OA-2022-0009s

[6] K. Diethelm, The Analysis of Fractional Differential Equations. Berlin: Springer, 2010.

[7] K. Diethelm, N. J. Ford, and A. D. Freed, “A predictor-corrector approach for the numerical solution of fractional differential equations,” Nonlinear Dynamics, vol. 29, pp. 3–22, 2002. https://doi.org/10.1023/A:1016582219341

[8] M. Stynes, E. O’Riordan, and J. L. Gracia, “Error analysis of a finite difference method on graded meshes for a time-fractional diffusion equation,” SIAM Journal on Numerical Analysis, vol. 55, no. 2, pp. 1057–1079, 2017. https://doi.org/10.1137/16M1082329

[10] R. Garrappa, “Predictor-corrector for FDEs,” MATLAB Central File Exchange, Version 2.0.0, 2025. https://www.mathworks.com/matlabcentral/fileexchange/32918-predictor-corrector-for-fdes

[11] FDE-TestSet, “FDE-testset: comparing Matlab codes for solving fractional differential equations of Caputo type,” Fractal and Fractional, 2025. https://www.mdpi.com/2504-3110/9/5/312

[12] Z. Cen, A. Le, and A. Xu, “A posteriori error analysis for a fractional differential equation,” International Journal of Computer Mathematics, vol. 94, no. 6, pp. 1185–1195, 2017. https://doi.org/10.1080/00207160.2016.1184263

[14] W. Mao, Y. Chen, and H. Wang, “A-posteriori error estimations of the GJF-Petrov–Galerkin methods for fractional differential equations,” Computers & Mathematics with Applications, vol. 90, pp. 159–170, 2021. https://doi.org/10.1016/j.camwa.2021.03.021