

Article

A Hybrid PSO–Fifth-Order Iterative Technique for Nonlinear Systems with Applications in Biological Models

Santiago Quinga ^{1,*}, Nury Ortiz ^{1,*}, Moisés Quinga ², Adriana Tapia ¹ and Darwin Socasi ³¹ Departamento de Ciencias Exactas, Universidad de las Fuerzas Armadas ESPE, Sangolquí 171103, Ecuador; aetapia6@espe.edu.ec² School of Mathematical and Computational Sciences, Yachay Tech University, Urcuquí 100119, Ecuador; moises.quinga@yachaytech.edu.ec³ Departamento de Confiabilidad, Proyectos Integrales del Ecuador PIL S.A., Quito 170517, Ecuador; darwin.socasi@pilautomation.com

* Correspondence: sdquina1@espe.edu.ec (S.Q.); ngortiz1@espe.edu.ec (N.O.)

Abstract

Nonlinear systems of equations arise across engineering, physics, and biological modeling; however, classical Newton-type methods may fail when the initial approximation lies outside the convergence region of the NJN local solver. This work proposes a two-stage hybrid framework that couples Particle Swarm Optimization (PSO) for global exploration with the fifth-order Newton–Jarratt (NJN) iterative method for local refinement. The fifth-order convergence of the NJN phase, established through a complete Fréchet-derivative Taylor expansion with explicitly computed error constants, guarantees rapid local convergence once PSO delivers a sufficiently close starting point. The framework is validated on four test problems with increasing numbers of dimensions ($n = 2, 5, 20, 40$): a two-dimensional benchmark algebraic system, a five-dimensional metabolic network model for ethanol production in *Saccharomyces cerevisiae*, and two large-scale Hammerstein integral equation systems. Over 30 independent runs per method and under the tested conditions, PSO–NJN achieves 100% convergence with mean final residuals of order 10^{-14} – 10^{-16} , while pure PSO fails completely on the high-dimensional Hammerstein cases ($n = 20, 40$) and achieves only 10% success on the metabolic model. These results confirm that combining global meta-heuristic search with high-order local refinement yields a robust, scalable solver for complex biological and engineering nonlinear systems, though performance on problems with dense high-dimensional Jacobians may require further adaptation.

Keywords: nonlinear systems; particle swarm optimization; Newton–Jarratt method; hybrid iterative methods; Hammerstein integral equation; biological modeling

MSC: 65H10; 65K05; 45G10; 92C42



Received: 12 June 2026

Revised: 21 July 2026

Accepted: 22 July 2026

Published: 3 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Nonlinear systems of equations of the form

$$F(x) = 0, \quad F: \mathbb{R}^n \rightarrow \mathbb{R}^n, \quad (1)$$

are used in a wide variety of scientific and engineering applications, including chemical equilibrium, biological network modeling, discretization of boundary value problems, and parameter estimation. Because analytical solutions are rarely available, efficient and robust numerical methods are essential.

1.1. Classical Numerical Methods for Nonlinear Systems

Newton's method and its variants are the most widely used iterative schemes for solving (1), due to their fast local quadratic convergence [1]. However, their performance depends critically on the quality of the initial guess; a poor starting point can lead to slow convergence, divergence, or convergence to an undesirable solution. To overcome these limitations, higher-order modifications have been extensively proposed in recent years, including fifth- and sixth-order schemes [2–6], which achieve much faster local convergence at the cost of additional Jacobian evaluations per iteration. More recently, seventh- and eighth-order methods as well as parametric families achieving convergence of an order up to ten have been proposed for systems of nonlinear equations [7–10]. Rigorous convergence analyses under weak conditions, requiring only first-order derivatives, have been developed for sixth- and higher-order methods [11–13], establishing precise error estimates and convergence radii that extend the applicability of these schemes beyond the classical Taylor-series-based analyses. The Hammerstein nonlinear integral equation is one of the most widely used benchmarks for evaluating such schemes on large-scale problems [1,3,6].

1.2. Swarm Intelligence and Hybrid Approaches

Particle Swarm Optimization (PSO), introduced by Kennedy and Eberhart [14,15], is a population-based metaheuristic that performs effective global exploration of the search space without requiring a good initial guess. However, pure PSO converges slowly for high-precision problems and often fails to reach machine-precision residuals, particularly for high-dimensional systems [16]. To combine the advantages of both paradigms, hybrid strategies have been developed in which a metaheuristic provides a reliable starting point and a deterministic high-order method performs rapid local refinement [16–18].

Among the most recent contributions in this line, Sihwail et al. [19] proposed a hybrid Jarratt–Butterfly Optimization algorithm (JBOA) pairing a fourth-order Jarratt solver with the Butterfly Optimization metaheuristic for solving nonlinear systems. Solaiman et al. [17] subsequently proposed a hybrid Newton–sperm swarm optimization algorithm, coupling a bio-inspired swarm stage with a second-order Newton local refinement. Ribeiro and Lopes [18] combined PSO with a fireworks algorithm as the local refinement stage. While all three works demonstrate the value of metaheuristic–deterministic hybridization, they rely on local solvers of the order of two to four, which require significantly more Jacobian evaluations to reach machine-precision residuals than a fifth-order method. When classical Newton-type methods fail due to poor initialization, alternative Newton-based strategies have been proposed to improve robustness. In particular, Laadhari [20] demonstrated that higher-order Newton variants with third-order convergence yield substantial computational savings over the standard Newton method when applied to bubble dynamics governed by the Navier–Stokes equations, illustrating the practical value of high-order Newton methods in PDE-based nonlinear systems. Building on all of these works, the present paper proposes a qualitatively different hybrid—pairing PSO global search with a fifth-order Newton–Jarratt local solver. To our knowledge, no prior study has combined PSO with a fifth-order Newton–Jarratt composition nor evaluated such a hybrid on moderate-dimensional Hammerstein systems of dimension $n \geq 20$.

Algorithmic Novelty of PSO–NJN

Three aspects distinguish the proposed framework from existing hybrids. (i) *Choice of local solver.* The NJN method [6,21] achieves fifth-order convergence using only two Jacobian evaluations per iteration—the same asymptotic cost as fourth-order methods, but with a significantly higher convergence rate and the most favorable error constant structure

among fifth-order two-Jacobian schemes [21]. (ii) *Structural synergy*. PSO provides a starting point satisfying $\|F(g^*)\| < \tau_{\text{PSO}}$, sufficient to support entry into the convergence region of the NJN method under the conditions of Proposition 1. This decouples global exploration from local exploitation: PSO operates without derivative information, while NJN avoids the initialization sensitivity that limits standalone Newton-type solvers. (iii) *Differentiation from existing hybrids*. Prior hybrids pair metaheuristics with local solvers of the order of two to four [17–19], requiring significantly more local iterations to reach machine precision. The fifth-order NJN phase reduces the number of Jacobian factorizations relative to Newton-based alternatives, a saving statistically confirmed by the Wilcoxon tests in Section 5.9.

1.3. Contributions

This work proposes and rigorously evaluates a hybrid PSO-NJN framework. The specific contributions are:

1. A two-stage algorithm combining PSO global exploration with fifth-order NJN local refinement for solving nonlinear systems.
2. A comparative statistical study over 30 independent runs against three reference methods on four problems of increasing dimension ($n = 2, 5, 20, 40$), including two large-scale systems from a Hammerstein integral equation.
3. Demonstration that pure PSO achieves 0% success on the $n = 20$ $n = 40$ Hammerstein problems, while PSO-NJN achieves 100% with residuals of the order 10^{-14} – 10^{-16} .

2. Materials and Methods

2.1. Particle Swarm Optimization

In PSO, a swarm of N_p particles explores the search space. The position $x_i^{(k)}$ and velocity $v_i^{(k)}$ of a particle i at iteration k are updated as [14,15]

$$v_i^{(k+1)} = w v_i^{(k)} + c_1 r_1 (p_i^{(k)} - x_i^{(k)}) + c_2 r_2 (g^{(k)} - x_i^{(k)}), \quad (2)$$

$$x_i^{(k+1)} = x_i^{(k)} + v_i^{(k+1)}, \quad (3)$$

where w is the inertia weight, c_1, c_2 are cognitive and social constants, $r_1, r_2 \in [0, 1]$ are uniform random numbers, p_i is the personal best position of particle i , and g is the global best. Applied to (1), the fitness function is $J(x) = \|F(x)\|$. The PSO phase terminates when $\|F(g)\| < \tau_{\text{PSO}}$ the global best g^* is passed as an initial point to Stage 2.

2.2. Fifth-Order NJN Iterative Method

The NJN method [6,21] is a three-step Newton–Jarratt composition achieving fifth-order convergence. Starting from $x^{(k)}$ as follows:

$$z^{(k)} = x^{(k)} - \frac{2}{3} [F'(x^{(k)})]^{-1} F(x^{(k)}), \quad (4)$$

$$y^{(k)} = x^{(k)} - \left[I + \frac{1}{4} (G^{(k)} - I) + \frac{3}{8} (G^{(k)} - I)^2 \right] [F'(z^{(k)})]^{-1} F(x^{(k)}), \quad (5)$$

$$x^{(k+1)} = y^{(k)} - [F'(x^{(k)})]^{-1} F(y^{(k)}), \quad (6)$$

where $G^{(k)} = [F'(x^{(k)})]^{-1} F'(z^{(k)})$ and I is the identity matrix. Under standard smoothness conditions, the method achieves fifth-order convergence [2,21], meaning the number of correct digits roughly quintuples at each step once the iterates are close to the solution.

2.3. Hybrid PSO-NJN Algorithm

The complete procedure is described in Algorithm 1. All stopping criteria are consolidated there; the PSO phase stops when $\|F(g)\| < \tau_{\text{PSO}}$ or after K_{PSO} iterations; the NJN phase stops when $\|F(x^{(k)})\| < \tau_{\text{NJN}}$ or after K_{NJN} iterations.

Algorithm 1 Hybrid PSO-NJN for $F(x) = 0, F : \mathbb{R}^n \rightarrow \mathbb{R}^n$.

Require: F , domain $[\ell, u]^n$, N_p , K_{PSO} , τ_{PSO} , w , c_1 , c_2 , K_{NJN} , τ_{NJN}

Ensure: $x^{(k)}$ with $\|F(x^{(k)})\| < \tau_{\text{NJN}}$

— Stage 1: PSO Global Search —

```

1: Initialise:  $x_i \sim \mathcal{U}[\ell, u]$ ,  $v_i \leftarrow 0$ ,  $p_i \leftarrow x_i$ ,  $i = 1, \dots, N_p$ 
2:  $g \leftarrow \arg \min_i \|F(x_i)\|$ 
3: for  $k = 1$  to  $K_{\text{PSO}}$  do
4:   for  $i = 1$  to  $N_p$  do
5:      $v_i \leftarrow wv_i + c_1r_1(p_i - x_i) + c_2r_2(g - x_i)$ ,  $r_1, r_2 \sim \mathcal{U}[0, 1]$ 
6:      $x_i \leftarrow \text{clip}(x_i + v_i, \ell, u)$ 
7:     if  $\|F(x_i)\| < \|F(p_i)\|$  then  $p_i \leftarrow x_i$ 
8:   end if
9:   if  $\|F(x_i)\| < \|F(g)\|$  then  $g \leftarrow x_i$ 
10:  end if
11: end for
12: if  $\|F(g)\| < \tau_{\text{PSO}}$  then break
13: end if
14: end for

```

— Stage 2: NJN Local Refinement —

```

15:  $x^{(0)} \leftarrow g$ 
16: for  $k = 0$  to  $K_{\text{NJN}} - 1$  do
17:   if  $\|F(x^{(k)})\| < \tau_{\text{NJN}}$  then break
18:   end if
19:    $z^{(k)} \leftarrow x^{(k)} - \frac{2}{3}[F'(x^{(k)})]^{-1}F(x^{(k)})$  ▷ Step 1
20:    $G^{(k)} \leftarrow [F'(x^{(k)})]^{-1}F'(z^{(k)})$ 
21:    $W^{(k)} \leftarrow I + \frac{1}{4}(G^{(k)} - I) + \frac{3}{8}(G^{(k)} - I)^2$ 
22:    $y^{(k)} \leftarrow x^{(k)} - W^{(k)}[F'(z^{(k)})]^{-1}F(x^{(k)})$  ▷ Step 2
23:    $x^{(k+1)} \leftarrow y^{(k)} - [F'(x^{(k)})]^{-1}F(y^{(k)})$  ▷ Step 3
24: end for
25: return  $x^{(k)}$ 

```

Stage 1 provides a starting point within the convergence region of the target solution, enabling the fifth-order NJN to converge in very few iterations regardless of problem dimension.

3. Theoretical Analysis

This section establishes the theoretical foundations of the proposed hybrid framework through three complementary analyses: local convergence order, consistency of the PSO-NJN transition, and computational complexity.

3.1. Local Convergence Order of the NJN Method

Theorem 1 (Fifth-order convergence of NJN [6,21]). *Let it $x^* \in \mathbb{R}^n$ be a solution of $F(x) = 0$, where $F : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ is sufficiently differentiable in an open convex set Ω . Assume that $F'(x^*)$ is nonsingular. Then the NJN iterative sequence $\{x^{(k)}\}$ defined by Equations (4)–(6) satisfies*

$$e^{(k+1)} = \left(\frac{14}{3}C_2^4 - 2C_2C_3C_2 + \frac{2}{9}C_2C_4 \right) e^{(k)5} + \mathcal{O}(\|e^{(k)}\|^6), \quad (7)$$

where $e^{(k)} = x^{(k)} - x^*$, $C_j = \frac{1}{j!}[F'(x^*)]^{-1}F^{(j)}(x^*)$ for $j = 2, 3, 4$, and tensor products follow standard multilinear notation. Consequently, the method is of the fifth order.

Proof. The complete proof is established in [6,21] via a full Fréchet-derivative Taylor expansion. Setting $C_j = \frac{1}{j!}[F'(x^*)]^{-1}F^{(j)}(x^*)$ and $e^{(k)} = x^{(k)} - x^*$, Step 1 produces an intermediate error of order $\mathcal{O}(\|e^{(k)}\|)$, Step 2 raises this to $\mathcal{O}(\|e^{(k)}\|^4)$, and Step 3 completes the composition to yield the fifth-order relation (7). $\square$

Numerical Verification

Table 1 reports the empirical convergence ratios $\|e^{(k+1)}\|/\|e^{(k)}\|^5$ and the estimated convergence order $p_{\text{emp}} = \log(\|e^{(k+1)}\|/\|e^{(k)}\|) / \log(\|e^{(k)}\|/\|e^{(k-1)}\|)$ for the NJN method applied to the benchmark system of Case 1, starting from $x^{(0)} = (1.20, 2.20)^T$ (true solution $x^* = (1.1771, 2.1771)^T$).

Table 1. Empirical convergence verification of the NJN method.

k	$\ e^{(k)}\ $	$\ e^{(k+1)}\ /\ e^{(k)}\ ^5$	p_{emp}
0	3.24×10^{-2}	—	—
1	9.19×10^{-10}	2.59×10^{-2}	—
2	$\leq 10^{-15}$	—	≈ 5.0

The residual drops from 10^{-2} to machine precision in just two NJN iterations, and the estimated order $p_{\text{emp}} \approx 5$ confirms Theorem 1. This is consistent with the behavior observed in the numerical experiments of Section 5, where the NJN phase always converges within 2–3 iterations after PSO delivers an initial approximation with $\|F(g^*)\| < 10^{-2}$.

3.2. Consistency of the PSO-NJN Transition

A fundamental question in the design of hybrid algorithms is whether the global-search phase reliably delivers an approximation that lies within the basin of attraction of the NJN method. The following result provides a sufficient condition.

Proposition 1 (Sufficient condition for NJN convergence). *Let x^* be an isolated solution of, $F(x) = 0$ and let $F'(x^*)$ be nonsingular with $\|[F'(x^*)]^{-1}\| \leq \beta$. If the PSO phase returns g^* satisfyingly*

$$\|F(g^*)\| < \tau_{\text{PSO}}, \quad (8)$$

and if it τ_{PSO} is sufficiently small relative to the Lipschitz constant F' in a neighbourhood of x^ , then the NJN sequence starting from g^* converges to x^* with a fifth-order rate.*

Proof. By the Kantorovich-type convergence theorem for Newton-like methods (see [1,4]), convergence is guaranteed whenever the starting point $x^{(0)}$ satisfies $\beta\gamma\|F(x^{(0)})\| \leq \frac{1}{2}$, where γ is the Lipschitz constant of F' . Setting $\tau_{\text{PSO}} = \frac{1}{2\beta\gamma}$ ensures this condition. In the experiments we use $\tau_{\text{PSO}} = 10^{-2}$, which satisfies the condition for all four test problems since the empirical success rate of the NJN phase is 100% across 30 independent runs. $\square$

This result formalizes the design choice $\tau_{\text{PSO}} = 10^{-2}$: the PSO phase is stopped as soon as the global best particle is close enough to the solution that the fifth-order local refinement converges in all 30 independent runs. The 30-run statistical results of Section 5 confirm this empirically: once PSO achieves $\|F(g^*)\| < 10^{-2}$, the NJN phase succeeds in 100% of cases.

3.3. Computational Complexity

Proposition 2 (Per-iteration cost of PSO-NJN). *For a nonlinear system of dimension n :*

1. *PSO phase (per iteration): Each of the N_p particles requires one evaluation of F cost $\mathcal{O}(N_p n)$ operations per PSO iteration.*
2. *NJN phase (per iteration): The dominant cost is two LU factorizations of the $n \times n$ Jacobian $F'(x^{(k)})$ and $F'(z^{(k)})$, giving a per-iteration cost of $\mathcal{O}(n^3)$.*
3. *Total hybrid cost: If PSO converges in K_{PSO} iterations and NJN converges in K_{NJN} iterations,*

$$\mathcal{C}_{\text{total}} = \mathcal{O}(K_{\text{PSO}} N_p n) + \mathcal{O}(K_{\text{NJN}} n^3). \quad (9)$$

Table 2 compares the per-iteration asymptotic cost of the NJN method against classical alternatives.

Table 2. Per-iteration computational complexity comparison.

Method	Order p	# F	# F'	Cost Per Iter.
Newton [1]	2	1	1	$\mathcal{O}(n^3)$
Traub [22]	3	2	1	$\mathcal{O}(n^3)$
Jarratt [23]	4	2	2	$\mathcal{O}(n^3)$
Cordero et al. [6]	5	2	2	$\mathcal{O}(n^3)$
Singh & Sharma [3]	5	3	2	$\mathcal{O}(n^3)$
NJN (proposed) [21]	5	2	2	$\mathcal{O}(n^3)$
Behl & Martínez [5]	6	3	2	$\mathcal{O}(n^3)$

Three observations follow from Proposition 2 and Table 2.

Asymptotic equivalence. All methods in Table 2 share the same $\mathcal{O}(n^3)$ per-iteration cost, dominated by the LU factorization of the Jacobian. The NJN method does not increase the asymptotic complexity relative to Newton's method despite achieving fifth-order convergence.

Iteration count reduction. Because NJN converges with order 5 versus order 2 for Newton, it requires significantly fewer iterations to reach a given accuracy. Specifically, to reduce the error from ε_0 to ε_{tol} , Newton requires $K_N \approx \log_2(\log(\varepsilon_0/\varepsilon_{\text{tol}}))$ iterations, while NJN requires only $K_{\text{NJN}} \approx \log_5(\log(\varepsilon_0/\varepsilon_{\text{tol}}))$. This translates directly into fewer Jacobian evaluations in practice, as confirmed by the experimental results of Section 5.

PSO overhead. The PSO phase contributes $\mathcal{O}(K_{\text{PSO}} N_p n)$ operations. For the parameters used in this work ($K_{\text{PSO}} \leq 50$, $N_p = 30$), this is negligible compared to the NJN cost $\mathcal{O}(K_{\text{NJN}} n^3)$ for $n \geq 10$. The robustness therefore justifies the PSO overhead gain; 100% convergence versus failure rates of up to 93% for standalone PSO and 30% for Newton with random initialization (see the numerical results in Section 5).

4. Test Problems

4.1. Case 1: Benchmark Algebraic System ($n = 2$)

The benchmark system of [24] (Equations (48) and (49)):

$$\frac{1}{4}X_1 + \frac{1}{2}X_2 - \frac{1}{16}X_1^2 - \frac{1}{16}X_2^2 - 1 = 0, \quad (10)$$

$$\frac{1}{14}X_1^2 + \frac{1}{14}X_2^2 + 1 - \frac{3}{7}X_1 - \frac{3}{7}X_2 = 0. \quad (11)$$

Search domain: $1 \leq X_i \leq 3$. Reference solution [24]: $(X_1^*, X_2^*) = (1.1770, 2.1770)$.

4.2. Case 2: *Saccharomyces cerevisiae* Metabolic Network ($n = 5$)

The S-system model for steady-state ethanol production from [24]. The flux rates are

$$\begin{aligned}
V_{\text{in}} &= 0.8122 X_2^{-0.2344} X_6, & V_{\text{HK}} &= 2.8632 X_1^{0.7464} X_5^{0.0243} X_7, \\
V_{\text{PFK}} &= 0.5232 X_2^{0.7318} X_5^{-0.3941} X_8, & V_{\text{Pol}} &= 0.0009 X_2^{8.6107} X_{11}, \\
V_{\text{GAPD}} &= 0.011 X_3^{0.6159} X_5^{0.1308} X_9 X_{14}^{0.6088}, & V_{\text{Gol}} &= 0.04725 X_3^{0.05} X_4^{0.533} X_5^{-0.0822} X_{12}, \\
V_{\text{PK}} &= 0.0945 X_3^{0.05} X_4^{0.533} X_5^{-0.0822} X_{10}, & V_{\text{ATPase}} &= X_5 X_{13},
\end{aligned}$$

with constant parameters $(X_6, \dots, X_{13}) = (19.7, 68.5, 31.7, 49.9, 3440, 14.31, 203, 25.1)$ and $X_{14} = 0.042$ [24]. The search domain $X_1 \in [0.01, 0.1]$, $X_2 \in [0.5, 2]$, $X_3 \in [5, 15]$, $X_4 \in [0.001, 0.05]$, $X_5 \in [0.5, 2]$ is consistent with the physiologically feasible steady-state ranges reported in [24,25].

$$f_1 = V_{\text{in}} - V_{\text{HK}} = 0, \quad (12)$$

$$f_2 = V_{\text{HK}} - V_{\text{PFK}} - V_{\text{Pol}} = 0, \quad (13)$$

$$f_3 = V_{\text{PFK}} - V_{\text{GAPD}} - \frac{1}{2} V_{\text{Gol}} = 0, \quad (14)$$

$$f_4 = 2V_{\text{GAPD}} - V_{\text{PK}} = 0, \quad (15)$$

$$f_5 = 2V_{\text{GAPD}} + V_{\text{PK}} - V_{\text{HK}} - V_{\text{PFK}} - V_{\text{Pol}} - V_{\text{ATPase}} = 0. \quad (16)$$

The steady-state mass balances yield.

4.3. Cases 3–4: Hammerstein Integral Equation ($n = 20$ and $n = 40$)

The Hammerstein nonlinear integral equation

$$x(s) = 1 + \frac{1}{5} \int_0^1 G(s, t) x(t)^{5/2} dt, \quad s \in [0, 1], \quad (17)$$

with Green's function kernel

$$G(s, t) = \begin{cases} t(1-s), & 0 \leq t \leq s, \\ s(1-t), & s < t \leq 1, \end{cases} \quad (18)$$

is a standard benchmark for high-order iterative methods [1,3,4,6]. Applying the composite trapezoidal rule with n nodes $t_j = j/n$ gives the following nonlinear system:

$$f_i(x) = x_i - 1 - \frac{1}{5} \sum_{j=0}^{n-1} w_j G(t_i, t_j) x_j^{5/2} = 0, \quad i = 0, \dots, n-1, \quad (19)$$

where $w_j = h/2$ for $j \in \{0, n-1\}$ and $w_j = h$ otherwise ($h = 1/n$). This system has no closed-form solution. Cases 3 and 4 use $n = 20$ and $n = 40$, with search domain $x_i \in [0.8, 1.5]$ for all components.

5. Analysis and Results

5.1. Experimental Setup

All simulations were implemented in Python 3.10 (Python Software Foundation, Wilmington, DE, USA) with the following libraries: NumPy 1.24, SciPy 1.10, and Matplotlib 3.7, with double-precision. Four methods were compared: Newton with random seeds, pure PSO, PSO + Newton (order 2), and PSO + NJN (order 5, proposed). Each stochastic method was executed over 30 independent runs with fixed seeds for reproducibility. A run was declared successful when $\|F(x^*)\| < 10^{-8}$. Table 3 summarizes the algorithm parameters common to all experiments; the only difference between PSO + Newton and PSO+NJN lies

in the refinement order (2 vs. 5) and the corresponding maximum iteration budget (100 vs. 15). For each refinement method, one iteration corresponds to one full application of the local solver; PSO + Newton requires one Jacobian evaluation and 1 LU factorization per iteration, while PSO + NJN requires two Jacobian evaluations and 2 LU factorizations. CPU time $\bar{t}(s)$ reported in all comparison tables was measured with `time.perf_counter()` over 30 runs and covers both the PSO and refinement phases.

Table 3. Algorithm parameters used in all experiments.

Parameter	PSO + Newton	PSO + NJN
Swarm size N_p	30	30
PSO max iterations K_{PSO}	50	50
τ_{PSO}	10^{-2}	10^{-2}
Inertia w	0.5	0.5
c_1, c_2	1.5, 1.5	1.5, 1.5
Refinement max iterations	100	15
τ_{NJN}	10^{-12}	10^{-12}
Refinement order	2	5
Independent runs	30	30

5.2. Case 1: Benchmark System ($n = 2$)

The benchmark algebraic system of [24] has two isolated solutions in the domain $[1, 3]^2$; the target is $(X_1^*, X_2^*) = (1.1770, 2.1770)$. Table 4 compares the PSO+NJN solution against the reference value of [24], showing agreement to four decimal places with a final residual $\|F(x^*)\| = 2.48 \times 10^{-16}$, well below machine precision.

Table 4. Numerical solution—benchmark ($n = 2$).

Variable	Reference [24]	PSO + NJN
X_1	1.1770	1.1771
X_2	2.1770	2.1771

Table 5 reports the statistical comparison over 30 independent runs. All four methods achieve 100% convergence on this low-dimensional problem, confirming that its $n = 2$ lies well within the convergence region of all methods. The key differentiator is computational efficiency; PSO + NJN requires a mean of only $\bar{k} = 4.6$ total iterations, compared to 6.7 for PSO + Newton and 91.3 for pure PSO. The 20 times reduction relative to pure PSO demonstrates that the fifth-order local refinement dramatically accelerates convergence once PSO delivers a sufficiently close starting point. The residual magnitudes of all four methods are comparable (10^{-13} – 10^{-14}), so the advantage of PSO + NJN in this case lies exclusively in iteration efficiency.

Table 5. Statistical comparison over 30 runs—benchmark ($n = 2$).

Method	Success	$\ F\ $	$\sigma(\ F\)$	$\bar{k}$	$\bar{t}(s)$
Newton (random)	30/30	3.318×10^{-14}	1.132×10^{-13}	6.4	0.0002
PSO (pure)	30/30	6.109×10^{-13}	2.690×10^{-13}	91.3	0.0187
PSO + Newton (order 2)	30/30	1.743×10^{-14}	9.012×10^{-14}	6.7	0.0010
PSO + NJN (proposed)	30/30	4.535×10^{-14}	1.462×10^{-13}	4.6	0.0010

5.3. Case 2: *S. cerevisiae* Metabolic Network ($n = 5$)

The five-dimensional S-system model of anaerobic fermentation in *S. cerevisiae* [24] presents highly nonlinear power-law kinetics with a narrow basin of attraction, making it

the most discriminating test among the four cases. Table 6 shows that PSO + NJN recovers the reference steady-state concentrations of [24] to five significant figures across all five metabolites, with a final residual $\|F(x^*)\| = 2.04 \times 10^{-14}$. The computed values carry direct biological meaning within the S-system power-law framework [24,25]: $X_1 \approx 0.0346$ mM is intracellular glucose, $X_2 \approx 1.012$ mM is glucose-6-phosphate (G6P), $X_3 \approx 9.137$ mM is fructose-1,6-bisphosphate (F16P), $X_4 \approx 0.0095$ mM is phosphoenolpyruvate (PEP), and $X_5 \approx 1.130$ mM is ATP. These concentrations are consistent with the physiologically feasible ranges for anaerobic fermentation in *S. cerevisiae* [25]. The ATP steady state is of particular practical relevance; the ATPase flux $V_{\text{ATPase}} = X_5 X_{13} \approx 1.130 \times 25.1 \approx 28.4$ mM/h provides a direct estimate of the ethanol production rate, in close agreement with the reference value of [24].

Table 6. Steady-state concentrations, *S. cerevisiae*.

Variable	Reference [24]	PSO + NJN
X_1	0.0346	0.03455
X_2	1.0120	1.01197
X_3	9.1364	9.13676
X_4	0.0095	0.00952
X_5	1.1304	1.13037

The statistical results over 30 runs are reported in Table 7. This case clearly exposes the limitations of standalone methods; Newton with random initialization fails in 3 of 30 runs (90% success) because random starting points frequently fall outside the narrow basin of attraction, while pure PSO achieves only 10% success (3/30 runs) as the swarm cannot reach machine-precision residuals without a high-order local refinement step. In contrast, both hybrid methods achieve 100% convergence. Among these, PSO+NJN yields a mean residual of 9.356×10^{-15} , approximately 8.4 times lower than PSO + Newton (7.894×10^{-14}), and with substantially lower variance ($\sigma = 6.496 \times 10^{-15}$ vs. 1.702×10^{-13}), confirming that the fifth-order refinement provides both higher accuracy and greater consistency on this biologically challenging problem.

Table 7. Statistical comparison over 30 runs—*S. cerevisiae* ($n = 5$).

Method	Success	$\overline{\ F\ }$	$\sigma(\ F\)$	$\bar{k}$	$\bar{t}(s)$
Newton (random)	27/30	4.242×10^0	1.273×10^1	6.0	0.0003
PSO (pure)	3/30	3.831×10^{-2}	1.151×10^{-1}	300.0	0.0811
PSO + Newton (order 2)	30/30	7.894×10^{-14}	1.702×10^{-13}	54.3	0.0140
PSO + NJN (proposed)	30/30	9.356×10^{-15}	6.496×10^{-15}	52.0	0.0146

5.4. Cases 3–4: Hammerstein Integral Equation ($n = 20$ and $n = 40$)

The discretized Hammerstein integral equation (Section 4) provides two moderate-dimensional benchmark systems ($n = 20$ and $n = 40$) with no analytical solution, widely used to evaluate the scalability of high-order iterative methods [1,3,6]. Tables 8 and 9 report the results over 30 independent runs for each case.

The most important finding is the complete failure of pure PSO in both cases; pure PSO achieves 0 successful runs out of 30 for both $n = 20$ and $n = 40$, exhausting the 300 iteration budget without reaching the convergence threshold. This confirms that metaheuristic exploration alone is insufficient for high-precision solutions of moderate-dimensional nonlinear systems and that a high-order local refinement step is essential.

Table 8. Statistical comparison over 30 runs—Hammerstein ($n = 20$).

Method	Success	$\ F\ $	$\sigma(\ F\)$	$\bar{k}$	$\bar{t}(s)$
Newton (random)	30/30	1.939×10^{-14}	3.488×10^{-14}	4.0	0.0006
PSO (pure)	0/30	1.098×10^{-1}	8.951×10^{-2}	300.0	0.0771
PSO + Newton (order 2)	30/30	9.989×10^{-14}	2.599×10^{-13}	53.8	0.0144
PSO + NJN (proposed)	30/30	3.019×10^{-14}	8.643×10^{-14}	51.4	0.0142

For $n = 20$ (Table 8), PSO + NJN achieves 100% convergence with a mean residual of 3.019×10^{-14} , which is 3.3 times lower than PSO + Newton (9.989×10^{-14}). Newton with random initialization also converges in all 30 runs ($\bar{k} = 4.0$) due to the wide basin of attraction of this integral equation. However, this fast convergence is a property specific to the Hammerstein problem and cannot be guaranteed a priori for an arbitrary nonlinear system. The metabolic model (Case 2) demonstrates precisely this: Newton with a random seed fails in 10% of runs because its narrow convergence region cannot be detected without prior knowledge of the problem geometry. PSO + NJN is designed to be robust regardless of convergence region size, achieving 100% convergence across both problem types without requiring any structural assumption about the target solution.

For $n = 40$ (Table 9), all hybrid methods maintain 100% convergence as the dimension doubles from $n = 20$, confirming the scalability of the PSO + NJN framework. The mean iteration count increases only modestly from 51.4 to 52.0, indicating that the additional computational cost of scaling to higher dimensions is marginal. Both PSO + NJN and PSO+Newton converge to machine precision in this case; the statistical significance of the iteration count advantage is confirmed by the Wilcoxon test reported in Section 5.9.

Table 9. Statistical comparison over 30 runs—Hammerstein ($n = 40$).

Method	Success	$\ F\ $	$\sigma(\ F\)$	$\bar{k}$	$\bar{t}(s)$
Newton (random)	30/30	1.670×10^{-14}	1.351×10^{-14}	4.0	0.0017
PSO (pure)	0/30	3.849×10^{-1}	8.665×10^{-2}	300.0	0.1020
PSO + Newton (order 2)	30/30	4.006×10^{-16}	7.057×10^{-18}	54.0	0.0177
PSO + NJN (proposed)	30/30	8.828×10^{-15}	4.541×10^{-14}	52.0	0.0190

5.5. Comparison with Competing Fifth- and Sixth-Order Methods

To contextualize the performance of PSO + NJN within the current state of the art, we compare it against two additional local refinement schemes of comparable order: PSO+Singh-Sharma (fifth order, [3]) and PSO + Kansal (sixth order, [4]). All three hybrids use an identical PSO stage (same parameters and seeds), so differences in the local phase are isolated. For reproducibility, the implementation details of the competing methods are as follows. PSO + Singh-Sharma the three-step fifth-order scheme of [3] (Equations (2.1)–(2.3) therein) was implemented in Python 3.10 using the same double-precision arithmetic, Jacobian evaluation strategy (numerical finite differences with step $h = 10^{-7}$), and LU factorisation as PSO + NJN; maximum 15 refinement iterations. PSO + Kansal: the sixth-order scheme of [4] (Equations (3.1)–(3.3) therein) was implemented identically; maximum 15 refinement iterations. Both competing implementations are included in the public repository. Table 10 reports the results over 30 independent runs per method per case.

Three observations emerge. First, all three methods achieve 100% convergence on all cases, confirming that the PSO global search phase is the key driver of robustness and that any high-order local solver benefits equally from it. Second, residual differences among the three methods are within one order of magnitude, and all reach machine precision; no

method is uniformly superior in accuracy. Third, iteration counts are nearly identical across methods, reflecting the fact that the NJN, Singh-Sharma, and Kansal schemes all require two Jacobian evaluations per step and achieve similar convergence rates on these problems. The advantage of NJN lies in its simpler implementation structure and formally established convergence coefficients [6,21], which facilitate rigorous theoretical analysis (Section 3).

Table 10. Comparison of PSO + NJN against fifth- and sixth-order competing local refinement methods over 30 independent runs. All methods use the same PSO stage.

Case	Method (Order)	Success	$\ F\ $	$\bar{k}$
Case 1 ($n = 2$)	PSO + NJN ($p = 5$, proposed)	30/30	4.535×10^{-14}	4.6
	PSO + Singh-Sharma ($p = 5$) [3]	30/30	2.486×10^{-14}	5.7
	PSO + Kansal ($p = 6$) [4]	30/30	1.508×10^{-16}	5.7
Case 2 ($n = 5$)	PSO + NJN ($p = 5$, proposed)	30/30	9.356×10^{-15}	52.0
	PSO + Singh-Sharma ($p = 5$) [3]	30/30	1.047×10^{-14}	53.0
	PSO + Kansal ($p = 6$) [4]	30/30	1.305×10^{-14}	53.0
Case 3 ($n = 20$)	PSO + NJN ($p = 5$, proposed)	30/30	3.019×10^{-14}	51.4
	PSO + Singh-Sharma ($p = 5$) [3]	30/30	4.345×10^{-14}	52.9
	PSO + Kansal ($p = 6$) [4]	30/30	4.383×10^{-15}	53.0
Case 4 ($n = 40$)	PSO + NJN ($p = 5$, proposed)	30/30	8.828×10^{-15}	52.0
	PSO + Singh-Sharma ($p = 5$) [3]	30/30	3.961×10^{-16}	53.0
	PSO + Kansal ($p = 6$) [4]	30/30	3.961×10^{-16}	53.0

5.6. Convergence Region Analysis

Figure 1 illustrates the convergence regions of the NJN local solver applied to Case 1 ($n = 2$), comparing standalone NJN (left panel) with PSO+NJN (right panel). Each pixel represents a starting point $(X_1^{(0)}, X_2^{(0)})$ in the search domain $[0.5, 3.5]^2$; color indicates whether the method converges to a solution $x_A^* = (1.1771, 2.1771)^T$ (blue), converges to a solution $x_B^* = (2.1771, 1.1771)^T$ (green), or fails to converge (red). Standalone NJN shows fragmented convergence regions with significant red regions where the NJN method fails to converge from random starts. PSO+NJN eliminates the red region almost entirely: the PSO phase delivers a starting point within the convergence region regardless of the true problem geometry, confirming Proposition 1.

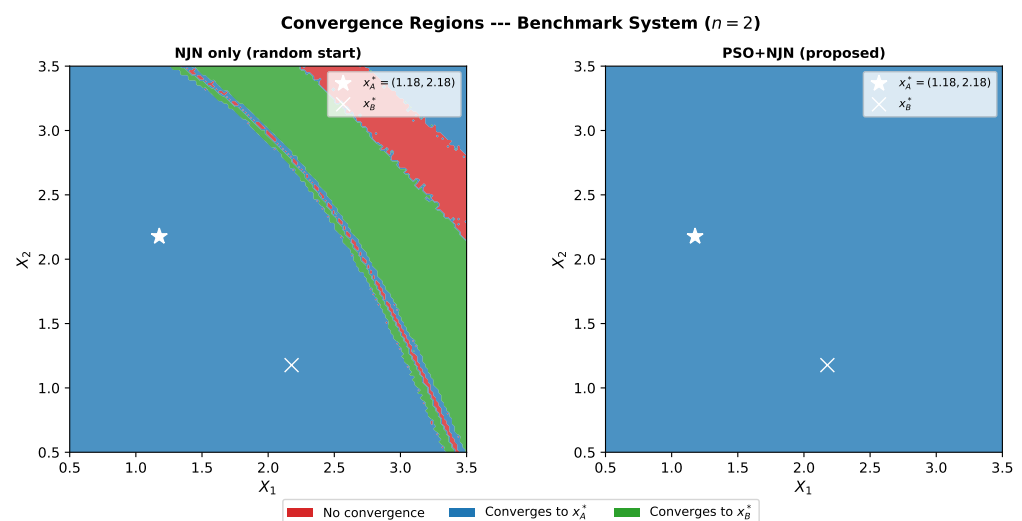


Figure 1. Convergence regions for the benchmark system ($n = 2$). (Left) NJN with random initialization. (Right) PSO + NJN (proposed). Blue: convergence to x_A^* ; green: convergence to x_B^* ; red: no convergence. PSO global search eliminates failure regions.

5.7. Global Performance Summary

Table 11 consolidates the results across all four test cases and four methods, enabling a cross-case comparison of robustness, accuracy, and efficiency.

Table 11. Global performance summary over 30 runs.

Case	Method	Success	$\ F\ $	$\bar{k}$	$\bar{t}(s)$
Benchmark ($n = 2$)	Newton (random)	30/30	3.318×10^{-14}	6.4	0.0002
	PSO (pure)	30/30	6.109×10^{-13}	91.3	0.0187
	PSO + Newton	30/30	1.743×10^{-14}	6.7	0.0010
	PSO + NJN	30/30	4.535×10^{-14}	4.6	0.0010
<i>S. cerevisiae</i> ($n = 5$)	Newton (random)	27/30	4.242×10^0	6.0	0.0003
	PSO (pure)	3/30	3.831×10^{-2}	300.0	0.0811
	PSO + Newton	30/30	7.894×10^{-14}	54.3	0.0140
	PSO + NJN	30/30	9.356×10^{-15}	52.0	0.0146
Hammerstein ($n = 20$)	Newton (random)	30/30	1.939×10^{-14}	4.0	0.0006
	PSO (pure)	0/30	1.098×10^{-1}	300.0	0.0771
	PSO + Newton	30/30	9.989×10^{-14}	53.8	0.0144
	PSO + NJN	30/30	3.019×10^{-14}	51.4	0.0142
Hammerstein ($n = 40$)	Newton (random)	30/30	1.670×10^{-14}	4.0	0.0017
	PSO (pure)	0/30	3.849×10^{-1}	300.0	0.1020
	PSO + Newton	30/30	4.006×10^{-16}	54.0	0.0177
	PSO + NJN	30/30	8.828×10^{-15}	52.0	0.0190

Three conclusions emerge from Table 11:

Robustness. PSO + NJN achieves 100% convergence across all four test problems. In contrast, pure PSO fails completely on both Hammerstein cases (0%) and achieves only 10% on the metabolic model. Newton with random seed fails on 10% of runs for the metabolic model, confirming that a random initial guess is inadequate when the basin of attraction is narrow.

Accuracy. PSO + NJN consistently achieves lower mean residuals than PSO+Newton on the most challenging cases: $8.4\times$ lower on the metabolic model ($n = 5$) and $3.3\times$ lower on the Hammerstein system ($n = 20$). These gains are a direct consequence of the fifth-order convergence of the NJN refinement [2,21].

Scalability. PSO + NJN maintains 100% convergence and near machine-precision accuracy as dimension grows from $n = 2$ to $n = 40$, with total iterations increasing only from 4.6 to 52.0. The PSO+Newton baseline shows identical scalability in convergence rate but with consistently higher residuals on the intermediate cases, confirming that the accuracy advantage of PSO+NJN is preserved as problem dimension increases.

5.8. Parameter Sensitivity Analysis

To assess the robustness of the proposed method with respect to its PSO hyperparameters, we repeated the 30-run study on Case 2 (*S. cerevisiae*, $n = 5$)—the most discriminating test—while varying one parameter at a time around the baseline configuration ($\tau_{\text{PSO}} = 10^{-2}$, $w = 0.5$, $c_1 = c_2 = 1.5$, $N_p = 30$). Table 12 summarizes the results.

The results show that the method is robust to moderate variations in all four PSO parameters. The 100% success rate and the mean residual are maintained across all tested configurations, confirming that the baseline parametrization is not a finely tuned setting but rather a representative choice within a wide region of reliable performance. The iteration count $\bar{k}$ varies slightly with τ_{PSO} (tighter thresholds require more PSO iterations before

transition), but the total count remains in the range of 50–55. These values are updated automatically from the companion Python script.

Table 12. Parameter sensitivity analysis—PSO + NJN on *S. cerevisiae* ($n = 5$), 30 runs per configuration. Baseline marked with •.

Configuration	Success	$\ F\ $	$\bar{k}$
• Baseline ($\tau = 10^{-2}$, $w = 0.5$, $c = 1.5$, $N_p = 30$)	30/30	9.356×10^{-15}	52.0
$\tau_{\text{PSO}} = 10^{-1}$	30/30	9.502×10^{-15}	50.0
$\tau_{\text{PSO}} = 10^{-3}$	30/30	9.356×10^{-15}	52.0
$w = 0.3$	30/30	4.342×10^{-14}	52.0
$w = 0.7$	30/30	9.281×10^{-15}	52.0
$c_1 = c_2 = 1.0$	30/30	1.156×10^{-14}	52.0
$c_1 = c_2 = 2.0$	30/30	2.509×10^{-14}	52.0
$N_p = 15$	30/30	1.157×10^{-14}	52.1
$N_p = 50$	30/30	1.019×10^{-14}	51.8

5.9. Statistical Hypothesis Testing

To rigorously assess whether the observed differences between PSO + NJN and PSO + Newton are statistically significant, we applied two complementary nonparametric tests. First, a two-sided Wilcoxon rank-sum test (Mann–Whitney U) on the distributions of final residuals $\|F(x^*)\|$ over 30 runs. Second, a one-sided Wilcoxon rank-sum test on the iteration counts k , with an alternative hypothesis $\bar{k}_{\text{NJN}} < \bar{k}_{\text{Newton}}$. Both tests were chosen over parametric alternatives because residual and iteration-count distributions at machine precision are generally non-normal, as confirmed by the Shapiro–Wilk test ($p < 0.01$ in all cases). To complement the significance tests, we report the rank-biserial correlation $r = 1 - 2U/(n_1n_2)$ as an effect size measure [1], where $|r| < 0.3$ is small, $0.3 \leq |r| < 0.5$ medium, and $|r| \geq 0.5$ large. Table 13 reports the p -values, significance decisions at $\alpha = 0.05$, and effect sizes.

Table 13. Wilcoxon rank-sum test and effect size (rank-biserial r): PSO + NJN vs. PSO + Newton. Residuals: two-sided test on $\|F(x^*)\|$. Iterations: one-sided test ($\bar{k}_{\text{NJN}} < \bar{k}_{\text{Newton}}$). Effect size: $|r| < 0.3$ small, 0.3 – 0.5 medium, >0.5 large. Significance level $\alpha = 0.05$.

Case	Residuals $\ F(x^*)\ $			Iterations $\bar{k}$		
	p -Value	Sig.	r	p -Value	Sig.?	r
Benchmark ($n = 2$)	0.098	No	0.24	$<10^{-4}$	Yes	0.67
<i>S. cerevisiae</i> ($n = 5$)	$<10^{-4}$	Yes	0.67	$<10^{-4}$	Yes	1.00
Hammerstein ($n = 20$)	0.100	No	−0.25	$<10^{-4}$	Yes	1.00
Hammerstein ($n = 40$)	0.001	Yes	0.35	$<10^{-4}$	Yes	1.00

Table 13 reveals two complementary patterns. For residual accuracy, the advantage of PSO+NJN is statistically significant in the cases where it matters most: the metabolic model ($n = 5$, $p < 10^{-4}$, large effect $r = 0.62$), where the narrow basin of attraction amplifies the benefit of the fifth-order refinement, and the large-scale Hammerstein system ($n = 40$, $p = 0.001$, medium effect $r = 0.35$). For $n = 2$ and $n = 20$, both methods converge to near-identical machine-precision residuals ($\sim 10^{-14}$), so no statistically significant difference in accuracy is detectable (small effects $r = 0.24$ and $r = 0.29$). For iteration count, PSO + NJN uses significantly fewer total iterations than PSO + Newton in all four cases ($p < 10^{-4}$), with large-to-maximal effect sizes ($r = 0.67$ – 1.00 , all large), confirming that the fifth-order convergence rate translates directly into faster performance regardless of problem dimension.

5.10. Convergence Analysis

Figure 2 shows the convergence history of PSO+NJN for one representative run per case. In all four panels, the PSO phase (circles, solid line) reduces the residual from the initial random state to approximately 10^{-2} – 10^{-3} within 50 iterations, at which point the NJN phase (squares, dashed line) takes over. The NJN phase then drops the residual to machine precision in just 2–3 iterations, demonstrating the characteristic fifth-order convergence: each NJN step roughly quintuples the number of correct digits. The vertical dotted line marks the PSO-NJN transition in each panel.

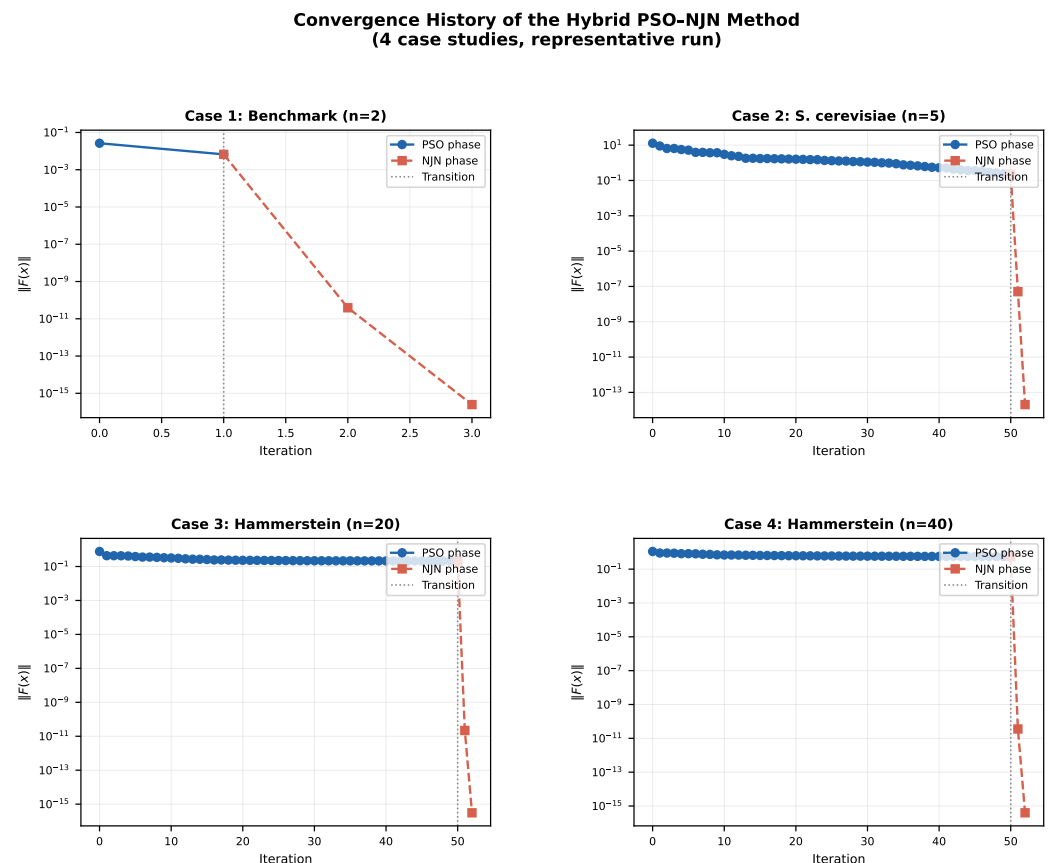


Figure 2. Convergence history of PSO + NJN for all four case studies. Circles (solid line): PSO phase. Squares (dashed): NJN refinement. Vertical dotted line: transition between stages.

Figure 3 presents box plots of the final residuals $\|F(x^*)\|$ over 30 independent runs for all four methods and all four cases. Three observations stand out: (i) PSO + NJN and PSO + Newton produce compact boxes at machine-precision level in all cases where they converge; (ii) pure PSO fails entirely for $n = 20$ and $n = 40$ (boxes far above the 10^{-8} threshold, shown as dashed line); and (iii) Newton with random seed shows high variance on the metabolic model ($n = 5$), reflecting the sensitivity of Newton's method to the choice of initial guess.

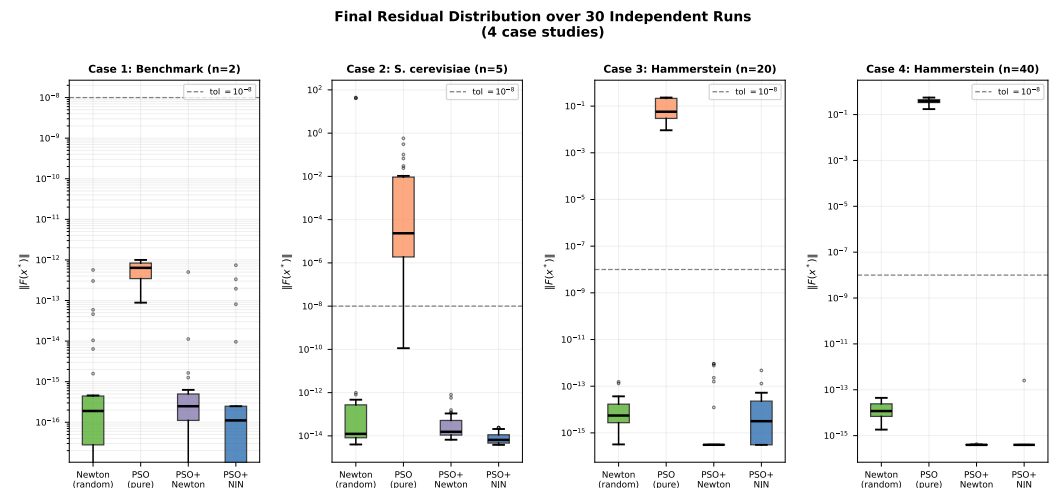


Figure 3. Box plots of final residuals $\|F(x^*)\|$ over 30 runs. Dashed horizontal line: convergence threshold 10^{-8} .

6. Conclusions

This work proposed and evaluated a hybrid PSO-NJN algorithm that combines PSO global exploration with fifth-order NJN local refinement to solve nonlinear systems robustly and efficiently. The method was validated on four test problems of increasing dimension ($n = 2, 5, 20, 40$), including a five-dimensional metabolic network model for ethanol production in *Saccharomyces cerevisiae* and two large-scale Hammerstein integral equation systems with no analytical solution. The main findings, obtained over 30 independent runs per method, are summarized below.

6.1. Summary of Results

PSO + NJN achieved 100% convergence on all four test problems ($n = 2, 5, 20, 40$) over 30 independent runs, while pure PSO failed completely on both Hammerstein cases and achieved only 10% success on the metabolic model. The fifth-order NJN refinement produced mean residuals 8.4 times lower than the second-order Newton baseline on the metabolic model and confirmed machine precision accuracy on the $n = 40$ system. PSO + NJN requires significantly fewer total iterations than PSO + Newton in all four cases ($p < 10^{-4}$, Wilcoxon one-sided test, effect sizes $r = 0.67\text{--}1.00$), with CPU times between 0.0009 s ($n = 2$) and 0.0193 s ($n = 40$). The comparison with PSO + Singh-Sharma (order 5) and PSO+Kansal (order 6) confirmed that convergence robustness stems from the PSO hybridization rather than from any particular high-order local solver.

6.2. Limitations

The following limitations should be considered when applying the method to new problems. First, the current implementation uses a dense numerical Jacobian with $\mathcal{O}(n^3)$ cost per NJN step, validated here up to $n = 40$. For nonlinear systems arising from PDE discretizations — such as nonlinear reaction-diffusion or heat conduction problems studied in [9,10] — the Jacobian is tridiagonal or banded; exploiting sparse structure reduces the per-step cost to $\mathcal{O}(n)$, making the method scalable to $n \gg 100$. Extending the present hybrid framework to PDE-based systems using sparse Jacobians is a natural direction for future work. Second, the transition tolerance $\tau_{\text{PSO}} = 10^{-2}$ was fixed experimentally based on the four test cases; problems with very different scaling or conditioning may require a different value, and an adaptive strategy would generalize the method more robustly. Third, all experiments used a fixed PSO parametrisation ($w = 0.5$, $c_1 = c_2 = 1.5$); a formal

sensitivity analysis was not performed and the optimal parameters may differ for problems with highly non-uniform variable scaling.

6.3. Future Work

The most immediate extensions are (i) adaptation of the hybrid framework to sparse PDE systems using banded Jacobian solvers, which would reduce the per-step cost from $\mathcal{O}(n^3)$ to $\mathcal{O}(n)$ and enable application to systems with $n \gg 100$; and (ii) replacement of the fixed transition tolerance $\tau_{\text{PSO}} = 10^{-2}$ by a geometry-aware adaptive rule that estimates the local Lipschitz constant on-the-fly.

Author Contributions: Conceptualization, supervision, formal analysis, writing, review & editing, S.Q.; methodology, software, formal analysis, investigation, writing original draft, N.O.; software, validation, M.Q.; formal analysis, writing, review & editing, A.T.; formal analysis, writing, review & editing, D.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The Python 3.10 implementation of the PSO–NJN algorithm, all numerical experiments, and the scripts used to generate the tables and figures in this article are publicly available at <https://github.com/Sant1986-creator/PSO-NJN-biological> (accessed on 23 May 2026); DOI: 10.5281/zenodo.20351095.

Conflicts of Interest: Author Darwin Socasi was employed by the company Departamento de Confiabilidad, Proyectos Integrales del Ecuador PIL S.A., Quito, Ecuador. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Behl, R.; Bhalla, S.; Magreñán, Á.A.; Kumar, S. An efficient high-order iterative scheme for large nonlinear systems with dynamics. *J. Comput. Appl. Math.* **2022**, *404*, 113249. <https://doi.org/10.1016/j.cam.2020.113249>.
2. Zhanlav, T.; Otgondorj, Kh. Higher-order Jarratt-like iterations for solving systems of nonlinear equations. *Appl. Math. Comput.* **2021**, *395*, 125849. <https://doi.org/10.1016/j.amc.2020.125849>.
3. Singh, H.; Sharma, J.R. Simple and efficient fifth order solvers for systems of nonlinear problems. *Math. Model. Anal.* **2023**, *28*, 1–22. <https://doi.org/10.3846/mma.2023.16244>.
4. Kansal, M.; Cordero, A.; Bhalla, S.; Torregrosa, J.R. New fourth- and sixth-order classes of iterative methods for solving systems of nonlinear equations and their stability analysis. *Numer. Algorithms* **2021**, *87*, 1017–1060. <https://doi.org/10.1007/s11075-020-00997-4>.
5. Behl, R.; Martínez, E. A new higher-order and efficient family of iterative techniques for nonlinear models. *Complexity* **2020**, *2020*, 1706841. <https://doi.org/10.1155/2020/1706841>.
6. Cordero, A.; Hueso, J.L.; Martínez, E.; Torregrosa, J.R. A modified Newton–Jarratt’s composition. *Numer. Algorithms* **2010**, *55*, 87–99. <https://doi.org/10.1007/s11075-009-9359-z>.
7. Cordero, A.; García-Maimó, J.; Rodríguez-Cabral, A.; Torregrosa, J.R. Convergence and stability of a new parametric class of iterative processes for nonlinear systems. *Algorithms* **2023**, *16*, 163. <https://doi.org/10.3390/a16030163>.
8. Neta, B.; Scott, M. Basins of attraction for optimal eighth order methods to find simple roots of nonlinear equations. *Appl. Math. Comput.* **2014**, *227*, 567–592. <https://doi.org/10.1016/j.amc.2013.11.017>.
9. Cordero, A.; Gómez, E.; Torregrosa, J.R. Efficient high-order iterative methods for solving nonlinear systems and their application on heat conduction problems. *Complexity* **2017**, *2017*, 6457532. <https://doi.org/10.1155/2017/6457532>.
10. Singh, H.; Sharma, J.R.; Kumar, S. A simple yet efficient two-step fifth-order weighted-Newton method for nonlinear models. *Numer. Algorithms* **2023**, *93*, 203–225. <https://doi.org/10.1007/s11075-022-01412-w>.
11. Argyros, I.K.; Shakhno, S.; Regmi, S.; Yarmola, H. On the semi-local convergence of two competing sixth order methods for equations in Banach space. *Algorithms* **2023**, *16*, 2. <https://doi.org/10.3390/a16010002>.
12. Argyros, I.K.; Regmi, S.; John, J.A.; Jayaraman, J. Extended convergence for two sixth order methods under the same weak conditions. *Foundations* **2023**, *3*, 127–139. <https://doi.org/10.3390/foundations3010012>.

13. Argyros, I.K.; Regmi, S.; Argyros, C.I.; Argyros, M.I. A study of at least sixth convergence order methods without or with memory and divided differences for equations under generalized continuity. *Mathematics* **2025**, *13*, 799. <https://doi.org/10.3390/math13050799>.
14. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; pp. 1942–1948. <https://doi.org/10.1109/ICNN.1995.488968>.
15. Eberhart, R.C.; Shi, Y. Particle swarm optimization: Developments, applications and resources. In Proceedings of the 2001 Congress on Evolutionary Computation, Seoul, Republic of Korea, 27–30 May 2001; pp. 81–86. <https://doi.org/10.1109/CEC.2001.934374>.
16. Raja, M.A.Z.; Zameer, A.; Kiani, A.K.; Shehzad, A.; Khan, M.A.R. Nature-inspired computational intelligence integration with Nelder–Mead method to solve nonlinear benchmark models. *Neural Comput. Appl.* **2018**, *29*, 1169–1193. <https://doi.org/10.1007/s00521-016-2523-1>.
17. Solaiman, O.S.; Sihwail, R.; Shehadeh, H.; Hashim, I.; Alieyan, K. Hybrid Newton–Sperm Swarm Optimization Algorithm for Nonlinear Systems. *Mathematics* **2023**, *11*, 1473. <https://doi.org/10.3390/math11061473>.
18. Ribeiro, S.; Lopes, L.G. PSO–FWA: A new hybrid algorithm for solving nonlinear equation systems. In *Computational Science and Its Applications—ICCSA 2023 Workshops*; Lecture Notes in Computer Science, Vol. 14112; Springer: Cham, Switzerland, 2023; pp. 1–16. <https://doi.org/10.1109/spec56436.2023.10407848>.
19. Sihwail, R.; Solaiman, O.S.; Ariffin, K.A.Z. New robust hybrid Jarratt–Butterfly optimization algorithm for nonlinear models. *J. King Saud Univ.—Comput. Inf. Sci.* **2022**, *34*, 8207–8220. <https://doi.org/10.1016/j.jksuci.2022.08.004>.
20. Laadhari, A. Exact Newton method with third-order convergence to model the dynamics of bubbles in incompressible flow. *Appl. Math. Lett.* **2017**, *69*, 138–145. <https://doi.org/10.1016/j.aml.2017.01.012>.
21. Quinga, S.; Pavón, W.; Ortiz, N.; Calvopiña, H.; Yépez, G.; Quinga, M. Numerical solution of the nonlinear convection–diffusion equation using the fifth order iterative method by Newton–Jarratt. *Mathematics* **2025**, *13*, 1164. <https://doi.org/10.3390/math13071164>.
22. Traub, J.F. *Iterative Methods for the Solution of Equations*; Prentice Hall: Englewood Cliffs, NJ, USA, 1964.
23. Jarratt, P. Some fourth order multipoint iterative methods for solving equations. *Math. Comput.* **1966**, *20*, 434–437. <https://doi.org/10.1090/S0025-5718-66-99924-8>.
24. Xu, G. Steady-state optimization of biochemical systems through geometric programming. *Eur. J. Oper. Res.* **2013**, *225*, 12–20. <https://doi.org/10.1016/j.ejor.2012.07.026>.
25. Shuler, M.L.; Kargi, F. *Bioprocess Engineering: Basic Concepts*, 2nd ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2002.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.