e-Journal of Analysis and Applied Mathematics **2025** (2025), 44–56
DOI 10.62780/ejaam/2025-005 EJAAM 2025

A polynomially accelerated fixed-point iteration for vector problems

Francesco Alemanno

Abstract. Fixed-point solvers are ubiquitous in nonlinear PDEs, yet their progress collapses whenever the Jacobian at the solution carries an eigenvalue arbitrarily close to one. We ask whether such stagnation can be removed without storing long histories or solving dense least squares. Under two assumptions—(A1) the linearised error e_n is dominated by a multiplier m with $|m| < 1$ and (A2) residuals shrink monotonically—we construct a quadratic blend of three iterates whose error polynomial has a double root at m. This three-point polynomial accelerator (TPA) cancels the stubborn mode up to $o(\|e_n\|)$, reduces to Aitken's Δ^2 process in one dimension, and matches a doubly blended Anderson step with depth $m = 2$ when the regularisation vanishes, yet it keeps the Picard memory footprint. The only extra ingredient is a residual-based estimate of $w = (1 - m)^{-1}$ obtained from a closed-form regularised least-squares fit that remains stable even when two residuals nearly coincide. Numerical experiments on linear systems with clustered spectra, a 320-dimensional nonlinear tanh fixed point, and a 50×50 Poisson discretisation show that TPA reaches the 10^{-8} residual tolerance in 32, 36, and 244 map evaluations (respectively). In the same settings SOR requires 663 steps and Anderson acceleration with depth $m = 5$ consumes 52, 38, and 955 evaluations. TPA therefore supplies a parameter-free, constant-memory drop-in accelerator whenever a single contraction factor throttles convergence.

Keywords. Fixed-point iteration, acceleration methods, three-point polynomial accelerator, Anderson acceleration, polynomial extrapolation.

2020 Mathematics Subject Classification. 65H10, 65F10, 65B05.

1 Introduction

Fixed-point iterations of the form

$$x_{k+1} = T(x_k), \tag{1}$$

constitute a standard tool for solving nonlinear systems and discretised partial differential equations. It is widely understood that the spectral structure of the Jacobian of T at the fixed point $x_\star$ dictates their effectiveness, stagnation is common when a single mode contracts markedly more slowly than the others. Classical remedies—such as relaxation strategies or extrapolation methods—attenuate this difficulty, yet they typically demand problem-dependent parameter tuning or introduce dense linear-algebra kernels that compromise the economy of the underlying iteration.

In this work we develop a *three-point polynomial accelerator* (TPA) that preserves the austere structure of a Picard loop (i.e. the plain iteration of (1)) while restoring rapid convergence. The mechanism interrogates successive residuals, forms an on-the-fly estimate of the dominant error contribution, and produces a regularised three-point update of the most recent iterates so that such error contribution is annihilated. Moreover, since the routine only manipulates three vectors at any time and eschews expanding history windows, it integrates seamlessly within existing implementations and adds a negligible computational burden.

From a formal standpoint, the analysis that leads to TPA builds upon the classical literature on sequence transformations. Aitken's Δ^2 process removes the leading error term in one dimension [1]; minimal-polynomial extrapolation transfers the annihilating-polynomial perspective to vector-valued settings [4]; and Anderson acceleration constructs optimal mixing coefficients via constrained least squares [2]. Successive over-relaxation [5] remains the canonical spectral shaper but hinges on careful damping-parameter selection. A comprehensive synthesis of these acceleration paradigms and their extensions is presented in the survey [3]. In order to make this bridge between our algorithm and pre-existing methods, in Sections 4.1 and 4.2 we summarise its precise relation to Aitken's Δ^2 process and Anderson acceleration of depth two.

In a nutshell: The main contributions of this manuscript are summarised as follows:

- the derivation of a constant-memory polynomial accelerator that elevates Aitken's principle to vector settings without dense auxiliary solves;
- a suite of linear and nonlinear experiments demonstrating consistent reductions in map evaluations relative to Picard, SOR, and Anderson acceleration, with the advantage over deeper Anderson variants widening as the test problems grow more challenging, all while maintaining minimal overhead.

We now move on to the formal derivation of the accelerator, establishing the assumptions and algebraic structure that underpin the subsequent analysis.

2 Derivation of the Algorithm

We study the fixed-point iteration

$$x_{n+1} = T(x_n), \quad x_n \in \mathbb{R}^d, \tag{2}$$

around a fixed point $x_\star = T(x_\star)$. Denote the error by $e_n = x_n - x_\star$.

Key assumptions. We work under two asymptotic conditions on the error sequence. There exist $|m| < 1$ and an index n_0 such that for all $n \geq n_0$ the errors obey

(A1) $e_{n+1} = me_n + o(\|e_n\|)$;

(A2) $\|e_{n+1}\| < \|e_n\|$.

Assumption A1 states that the linearisation of T at $x_\star$ is governed by a single dominant multiplier m; Assumption A2 ensures that the error magnitude shrinks, so the higher-order terms in Assumption A1 remain subordinate.

Remark 2.1 (Jacobian interpretation of Assumption A1). When T is differentiable at $x_\star$, let $J = T'(x_\star)$ denote its Jacobian. The linearisation gives $e_{n+1} = Je_n + o(\|e_n\|)$. If J admits a dominant mode with right and left vectors u and v, then

$$J = m\frac{u\,v^\top}{\langle u, v\rangle} + \text{higher-order terms}, \tag{3}$$

so the scalar multiplier m in Assumption A1 coincides with the leading spectral component that governs the residual recursion.

Quadratic blend of three iterates. Let $y_3 = x_n$, $y_2 = x_{n-1}$, and $y_1 = x_{n-2}$ with errors e_3, e_2, e_1 respectively. Any quadratic blend preserving constants can be parametrised as

$$y_+(y_1, y_2, y_3; a, b) = (1 - a - b)y_1 + ay_2 + by_3. \tag{4}$$

Lemma 2.2 (Error polynomial). *Under Assumptions A1 and A2 the error committed by the blend* (4) *satisfies*

$$y_+ - x_\star = \big(1 - a - b + am + bm^2\big)e_1 + o(\|e_1\|). \tag{5}$$

Proof. Assumption A1 yields $e_2 = me_1 + o(\|e_1\|)$. Applying Assumption A1 once more and using Assumption A2 to bound the higher-order term gives $e_3 = me_2 + o(\|e_2\|) = m^2 e_1 + o(\|e_1\|)$. Substituting $y_i = x_\star + e_i$ into (4) we obtain

$$\begin{aligned} y_+ - x_\star &= (1 - a - b)e_1 + ae_2 + be_3 \\ &= \big(1 - a - b + am + bm^2\big)e_1 + o(\|e_1\|), \end{aligned}$$

which is (5). □

Optimality conditions. To remove the dominant error mode to second order we enforce that the prefactor in (5) vanishes together with its derivative with respect to m. The conditions

$$1 - a - b + am + bm^2 = 0, \quad a + 2bm = 0, \tag{6}$$

force the error polynomial to have a double root at the unknown multiplier m.

Proposition 2.3 (Optimal quadratic coefficients). *Under Assumptions A1 and A2, solving* (6) *gives*

$$a = -\frac{2m}{(1-m)^2}, \quad b = \frac{1}{(1-m)^2}. \tag{7}$$

Consequently the blend (4) *with the optimal coefficients reduces to*

$$\begin{aligned} y_+ = y_1 + 2w(y_2 - y_1) + w^2(y_1 - 2y_2 + y_3), \\ \text{with } w := \frac{1}{1-m}, \end{aligned} \tag{8}$$

eliminates the dominant mode up to $o(\|e_1\|)$ and collapses to $x_\star$ when Assumption A1 holds exactly.

Residual dynamics. Define the residuals $r_n = x_{n+1} - x_n = e_{n+1} - e_n$. The next lemma links m to the observable sequence (r_n).

Lemma 2.4 (Residual recursion). *Under Assumptions A1 and A2 we have*

$$r_{n+1} = mr_n + o(\|e_n\|). \tag{9}$$

Proof. Using Assumption A1 twice yields $e_{n+2} = me_{n+1} + o(\|e_{n+1}\|)$ and $e_{n+1} = me_n + o(\|e_n\|)$. Therefore

$$\begin{aligned} r_{n+1} = e_{n+2} - e_{n+1} &= (m-1)e_{n+1} + o(\|e_{n+1}\|) \\ &= (m-1)(me_n + o(\|e_n\|)) + o(\|e_n\|) \\ &= m(m-1)e_n + o(\|e_n\|). \end{aligned}$$

Similarly $r_n = (m-1)e_n + o(\|e_n\|)$, whence (9) follows. □

Subtracting successive residuals and applying (9) gives

$$r_n - r_{n+1} = (1-m)r_n + o(\|e_n\|). \tag{10}$$

Combining (10) with the $w = (1-m)^{-1}$ from (8) shows that

$$w(r_n - r_{n+1}) = r_n + o(\|e_n\|). \tag{11}$$

For the triple (y_1, y_2, y_3) this relation reads

$$w(r_1 - r_2) = r_1 + o(\|e_1\|), \quad r_1 = y_2 - y_1, \quad r_2 = y_3 - y_2. \tag{12}$$

Regularised estimation of w. We infer w by fitting (12) in least squares with a quadratic regulariser that favours $w \approx 1$ when the residual gap is nearly singular:

$$\widehat{w}_\theta = \operatorname*{argmin}_{w \in \mathbb{R}} \|w(r_1 - r_2) - r_1\|_2^2 + \theta^2 (w-1)^2, \tag{13}$$

with $\theta > 0$. The first-order optimality condition is

$$\left(\|r_1 - r_2\|_2^2 + \theta^2\right) w = \langle r_1 - r_2, r_1 \rangle + \theta^2, \tag{14}$$

yielding the closed-form estimate

$$\widehat{w}_\theta = \frac{\langle r_1 - r_2, r_1 \rangle + \theta^2}{\|r_1 - r_2\|_2^2 + \theta^2}. \tag{15}$$

When the scalar model of Assumption A1 is exact, $r_2 = mr_1$ and (15) recovers $w = (1-m)^{-1}$.

Inserting $\widehat{w}_\theta$ into (8) produces the practical accelerator

$$y_+ = y_1 + 2\widehat{w}_\theta (y_2 - y_1) + \widehat{w}_\theta^2 (y_1 - 2y_2 + y_3), \tag{16}$$

Algorithm 1 TPA, Three-point polynomial accelerator

Require: map T, initial iterate x_0, tolerance `tol`, maximum loop count K, regularisation $\theta > 0$

$y_1 \leftarrow x_0$
$y_2 \leftarrow T(y_1)$, $y_3 \leftarrow T(y_2)$
$\rho \leftarrow \|y_3 - y_2\|_\infty$
if $\rho <$ `tol` **then return** y_3
end if
for $i = 1, \dots, K$ **do**
  $r_1 \leftarrow y_2 - y_1, \quad r_2 \leftarrow y_3 - y_2$
  $w \leftarrow \frac{\langle r_1 - r_2, r_1 \rangle + \theta^2}{\|r_1 - r_2\|_2^2 + \theta^2}$ ▷ θ stabilises the fit
  $y_1 \leftarrow y_1 + 2w(y_2 - y_1) + w^2(y_1 - 2y_2 + y_3)$
  $y_2 \leftarrow T(y_1), \quad y_3 \leftarrow T(y_2)$
  $\rho \leftarrow \|y_3 - y_2\|_\infty$
  if $\rho <$ `tol` **then return** y_3
  end if
end for
return y_3 ▷ Maximum iterations reached without convergence

which inherits the dominant-mode cancellation while remaining stable when r_1 and r_2 nearly coincide.

The pseudocode in Algorithm 1 consolidates these steps.

Each loop costs two evaluations of T and a handful of inner products, and the method stores only three iterates and two residuals. The next section evaluates this accelerator on nonlinear vector fixed-point problems, comparing it with Picard iteration in terms of contraction per step and overall work.

3 Numerical Experiments

We investigate TPA on three representative fixed-point problems: a linear system with a clustered spectrum, a nonlinear tanh map, and a discretised Poisson equation. All solvers start from the zero vector and terminate once the residual criterion $\|T(x_k) - x_k\|_\infty <$ `tol` with `tol` $= 10^{-8}$ is satisfied or after 10^6 evaluations of the map T, and we keep the regularisation parameter fixed at $\theta = 10^{-9}$ across all trials. To anchor the comparisons we also compute a reference fixed point in closed form for each scenario, enabling us to report both the final residual $\|T(x_k) - x_k\|_\infty$ and the final error $\|x_k - x_\star\|_\infty$.

Before turning to the individual benchmarks it is useful to contrast the cost structure of the competitors. Relative to Anderson acceleration of depth m, which stores m iterates and solves a dense $m \times m$ least-squares problem each step, TPA requires only three iterates, two residuals, and no dense solves. Compared with weighted Jacobi or SOR, TPA selects the effective damping factor automatically from the residuals and removes per-problem tuning.

3.1 Linear System with Clustered Spectrum

We first test TPA on a linear system whose contraction factors cluster in $[0.9, 0.99]$, a regime where single-parameter relaxations perform poorly. The test map is built in dimension 80 by drawing a random orthogonal matrix Q from the QR factorisation of a standard-normal matrix and forming $M = Q\,\mathrm{diag}(\lambda)\,Q^\top$ with evenly spaced eigenvalues $\lambda \in [0.9, 0.99]$. We synthesise a fixed point $x_\star$ from the same random generator and apply the affine map $T(x) = Mx + c$ with $c = x_\star - Mx_\star$. Table 1 compares TPA with SOR using relaxation $\omega = 1.8$, Picard iteration ($\omega = 1$), and Anderson acceleration at depths $m \in \{2, 3, 5\}$, while Fig. 1 reports the convergence histories.

Table 1. **Linear system with clustered spectrum.** Performance of TPA compared with Picard, SOR, and Anderson($m = 2, 3, 5$).

Method	N. evals	$\|T(x) - x\|_\infty$	$\|x - x_\star\|_\infty$
TPA	**32**	0.16×10^{-8}	0.12×10^{-6}
Picard	1197	1.00×10^{-8}	0.98×10^{-6}
SOR ($\omega = 1.80$)	663	0.99×10^{-8}	0.97×10^{-6}
Anderson(m=2)	419	0.98×10^{-8}	0.86×10^{-6}
Anderson(m=3)	183	0.95×10^{-8}	0.94×10^{-6}
Anderson(m=5)	52	0.94×10^{-8}	0.72×10^{-6}

In this regime TPA converges in 32 map evaluations, whereas SOR and Picard require 663 and 1197 iterations to reach the same tolerance. To calibrate the comparison we also examine Anderson acceleration, which shortens the run to 419 evaluations at depth two, 183 at depth three, and 52 at depth five, yet TPA still uses roughly one thirteenth of the work of the shallowest variant and about 40% fewer evaluations than the deepest scheme while avoiding dense $m \times m$ solves. Fig. 1 reinforces this gap by showing the smoother decay produced by TPA relative to the plateaus observed for the competing schemes. Although this instance is nearly monotone, monotonicity is not ensured in general; consequently the regularised coefficient estimate in (15) moderates the blend when residual gaps are small, and

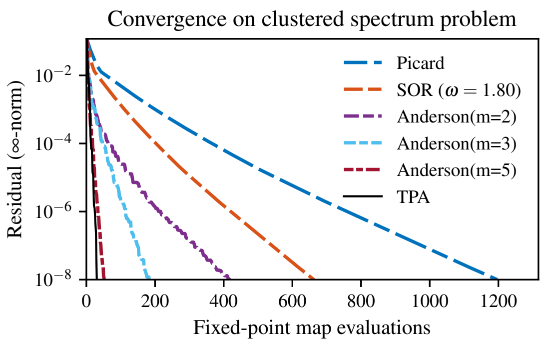

Figure 1. **Linear system with clustered spectrum.** Residual histories showing that TPA damps the dominant modes faster than the reference methods.

all convergent methods reach comparable final errors.

3.2 Nonlinear *tanh* Fixed Point

We next apply TPA to the nonlinear map $T(x) = \tanh(Bx + c)$ with $\|x_\star\|_\infty < 1$, whose Jacobian spectrum near the solution again clusters near unity. We set the dimension to 320, draw $B = Q\,\mathrm{diag}(\lambda)\,Q^\top$ with eigenvalues evenly spaced in $[0, 0.999]$, and choose a reference fixed point $x_\star$ by sampling each component uniformly in $[-0.7, 0.7]$. The offset is $c = \operatorname{arctanh}(x_\star) - Bx_\star$, so $x_\star$ is recovered exactly when T converges. Table 2 reports the iteration counts for Picard, SOR with $\omega = 1.5$, Anderson acceleration with depths $m \in \{2, 3, 5\}$, and TPA; Fig. 2 shows the residual traces.

Table 2. **Nonlinear** tanh **fixed point.** Iteration counts for Picard, SOR, Anderson, and TPA.

Method	N. evals	$\|T(x) - x\|_\infty$	$\|x - x_\star\|_\infty$
TPA	**36**	0.53×10^{-8}	0.18×10^{-7}
Picard	128	0.97×10^{-8}	0.91×10^{-7}
SOR ($\omega = 1.50$)	84	0.85×10^{-8}	0.79×10^{-7}
Anderson(m=2)	70	0.86×10^{-8}	0.64×10^{-7}
Anderson(m=3)	49	0.89×10^{-8}	0.72×10^{-7}
Anderson(m=5)	38	0.69×10^{-8}	0.49×10^{-7}

All methods level off near 10^{-8} because they share the same stopping tolerance rather than any intrinsic instability, so iteration counts provide the relevant compari-

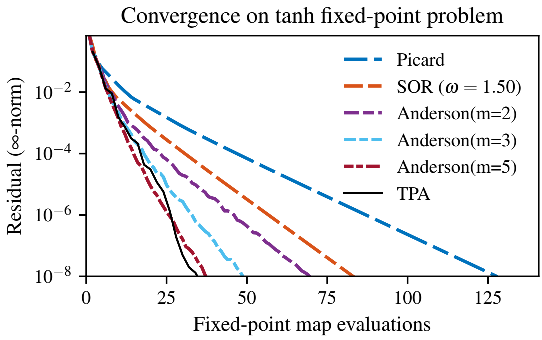

Figure 2. **Nonlinear** tanh **fixed point.** Residual histories showing the smooth convergence of TPA relative to the baselines.

son. Within that framework TPA converges in 36 evaluations, outpacing Anderson acceleration with depth two (70 evaluations) and depth three (49 evaluations) while still edging the depth-five variant (38 evaluations) and avoiding the larger least-squares solves and storage burden. The resulting three-point update therefore preserves smooth decay without depth tuning.

3.3 Discretised 2D Poisson Equation

The final experiment considers a 50×50 standard five-point finite-difference discretisation of the Poisson equation $-\Delta u = f$ on the unit square with homogeneous Dirichlet boundary conditions. The Jacobi fixed-point map arises from the usual diagonal splitting of the stiffness matrix on a grid with mesh spacing $h = 1/51$. The right-hand side is prescribed deterministically as

$$f(x, y) = \sin(\pi x^2)\,\sin(2\pi y^2) \tag{17}$$

evaluated at the interior grid points. We assemble the resulting Kronecker-structured linear system and precompute the exact grid solution $x_\star$ for error reporting. Table 3 summarises the performance of Jacobi ($\omega = 1$), Anderson acceleration at depths $m \in \{2, 3, 5\}$, and TPA; Fig. 3 plots the residual traces.

On this discretised elliptic problem TPA reaches the stopping tolerance in 244 fixed-point evaluations. Anderson acceleration requires 3722 evaluations at depth two, 1573 at depth three, and 955 at depth five, so even the strongest baseline still needs nearly four times as many iterations, whereas Jacobi demands 4317 steps, more than an order of magnitude beyond TPA. The widening gap relative to the depth-five Anderson method on this more demanding problem underscores

Table 3. **Discretised 2D Poisson equation.** Performance on the 50 × 50 grid for Jacobi (Picard), Anderson, and TPA.

Method	N. evals	$\|T(x) - x\|_\infty$	$\|x - x_\star\|_\infty$
TPA	**244**	0.97×10^{-8}	0.51×10^{-5}
Picard	4317	1.00×10^{-8}	0.53×10^{-5}
Anderson(m=2)	3722	1.00×10^{-8}	0.51×10^{-5}
Anderson(m=3)	1573	0.99×10^{-8}	0.47×10^{-5}
Anderson(m=5)	955	0.96×10^{-8}	0.24×10^{-5}

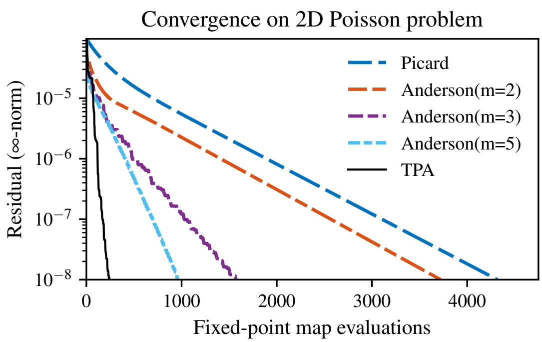

Figure 3. **Discretised 2D Poisson equation.** Residual histories highlighting the faster damping achieved by TPA.

the advantage conferred by this three-point update in two-dimensional elliptic problems with slowly decaying error components. These benchmarks motivate a closer comparison between TPA and the classical extrapolation routines it refines; the next section formalises this connection.

4 Relation to earlier work

This section situates TPA within the literature on polynomial and residual-based acceleration. We first recall how the scalar restriction and vanishing regularisation recover Aitken's Δ^2 transform, and then examine the structural link with depth-two Anderson acceleration.

4.1 Relation to Aitken's Δ^2 Process

Setting the regularisation to zero and restricting to scalar iterates recovers Aitken's classical accelerator. Let $y_1 = x_{n-2}$, $y_2 = x_{n-1}$, and $y_3 = x_n$ be three consecutive iterates of a scalar map. Their residuals are $r_1 = y_2 - y_1$ and $r_2 = y_3 - y_2$, and with $\theta = 0$ the least-squares fit (13) simplifies to

$$\widehat{w}_0 = \frac{\langle r_1 - r_2, r_1 \rangle}{\|r_1 - r_2\|_2^2} = \frac{r_1}{r_1 - r_2}. \tag{18}$$

Substituting this coefficient into the quadratic blend (8) yields

$$y_+ = y_1 + 2\widehat{w}_0(y_2 - y_1) + \widehat{w}_0^2(y_1 - 2y_2 + y_3) \tag{19}$$

$$= y_1 - \frac{(y_2 - y_1)^2}{y_3 - 2y_2 + y_1}, \tag{20}$$

which coincides with Aitken's Δ^2 transform applied to x_{n-2}.

4.2 Relation to Anderson Acceleration

Consider Anderson acceleration of depth two with iterates y_1, y_2, y_3 and residuals $r_1 = y_2 - y_1$ and $r_2 = y_3 - y_2$. The depth-two update is obtained by choosing a mixing coefficient a and forming the blend

$$y_{+2} = (1 - a)y_2 + ay_3. \tag{21}$$

The coefficient a is selected by the regularised least-squares fit

$$\widehat{a}_\theta = \underset{a \in \mathbb{R}}{\operatorname{argmin}} \|(1 - a)r_1 + ar_2\|_2^2 + \theta^2 \left[a^2 + (1 - a)^2\right], \tag{22}$$

where the residuals r_i act as a proxy for the error of y_{i+1} and the regularisation parameter θ avoids ill-conditioning.

Differentiating the objective and setting the derivative to zero yields the closed form

$$\widehat{a}_\theta = \frac{\langle r_1 - r_2, r_1 \rangle + \theta^2}{\|r_1 - r_2\|_2^2 + 2\theta^2}. \tag{23}$$

With the optimal coefficient in hand, the accelerated iterate (21) of Anderson's scheme is complete.

In order to connect to TPA we may also blend the earlier pair of iterates to obtain

$$y_{+1} = (1 - \widehat{a}_\theta)y_1 + \widehat{a}_\theta y_2, \tag{24}$$

and then apply the same coefficient once more to combine y_{+1} and y_{+2}:

$$y_{++} = (1 - \widehat{a}_\theta)y_{+1} + \widehat{a}_\theta y_{+2}. \tag{25}$$

Expanding the last expression reveals the quadratic weights applied to the original iterates,

$$y_{++} = (1 - \widehat{a}_\theta)^2 y_1 + 2\widehat{a}_\theta(1 - \widehat{a}_\theta) y_2 + \widehat{a}_\theta^2 y_3. \tag{26}$$

When $\theta = 0$ the coefficient reduces to $\widehat{a}_0 = \widehat{w}_0$, so the three weights coincide with those in (8) and the doubly blended iterate y_{++} matches TPA exactly. Depth-two Anderson acceleration, however, typically outputs only y_{+2} and therefore misses the additional cancellation provided by the second blend. For $\theta > 0$ the regularisation in the denominator also deviates slightly from the safeguard in (15), yet this effect is dominated by the structural distinction between a single and a double blend.

The heuristic comparison above is intended to expose the algebraic kinship between the methods; a full proof that follows this route is feasible with additional work but falls outside the present scope. We close by distilling these implications for practice and outlining directions for refinement in the concluding section.

5 Conclusions

We have shown that a closed-form, constant-memory three-point polynomial accelerator (TPA) can markedly reduce the work required by fixed-point iterations across linear and nonlinear problems. On the clustered linear system TPA attains the convergence tolerance in only 32 map evaluations compared with 663 for SOR, 1197 for Picard, and $419/183/52$ for Anderson acceleration with depths $m = 2/3/5$, thereby trimming roughly 40% of the evaluations relative to the deepest competitor while avoiding dense solves. The nonlinear tanh benchmark in turn converges in 36 evaluations, maintaining a lead over Anderson acceleration at depths two (70), three (49), and five (38), and the Poisson experiment achieves the tolerance in 244 iterations versus $3722/1573/955$ for Anderson($m = 2/3/5$). Collectively these observations highlight a widening advantage over Anderson acceleration as the test problems grow more demanding and underscore the benefits of error-polynomial shaping informed by short-history contraction-factor estimates.

TPA is most effective when convergence is throttled by one slowly decaying mode: the residuals align, the contraction-factor estimate stabilises, and the resulting three-point update suppresses the stubborn component. Empirically it also performs well when the spectrum is only moderately clustered, provided that the dominant mode does not vary rapidly between successive steps, and this operational envelope motivates the safeguards below.

Limitations and safeguards. Strongly non-normal Jacobians can still produce alternating dominant modes that temporarily mislead the residual ratio. Users should therefore monitor the residual norm and fall back to the base iteration if acceleration induces growth. In addition, purely polynomial extrapolations such as TPA cannot stabilise iterations whose local Jacobian has eigenvalues of magnitude greater than one, so a contractive regime remains indispensable.

Looking ahead, future work includes adaptive selection of the contraction-factor regularisation, richer surrogates derived from longer iterate histories, and higher-degree extensions that exploit additional iterates without sacrificing the simplicity that renders TPA attractive in practice.

Acknowledgements

We thank Miriam Aquaro (iD) for carefully reading the manuscript and providing valuable suggestions.

Bibliography

[1] A. C. Aitken, XXV.— On Bernoulli's numerical solution of algebraic equations, *Proc. R. Soc. Edinburgh Sect. A* **46** (1927), 289–305, doi:10.1017/S0370164600022070.

[2] D. G. Anderson, Iterative Procedures for Nonlinear Integral Equations, *J. ACM* **12(4)** (1965), 547–560.

[3] Y. Saad, Acceleration methods for fixed-point iterations, *Acta Numer.* **34** (2025), 805–890.

[4] A. Sidi, Vector extrapolation methods with applications to solution of large systems of equations and to PageRank computations, *Comput. Math. Appl.* **56(1)** (2008), 1–24.

[5] D. Young, Iterative methods for solving partial difference equations of elliptic type, *Trans. Amer. Math. Soc.* **76(1)** (1954), 92–111.

Received November 18, 2025; revised December 23, 2025; accepted December 26, 2025.

Author information

Francesco Alemanno (iD), Technical office, SolarSud T.R.E., Via Tiziano Vecellio 11, San Giorgio Ionico (TA), Italy.
E-mail: `francesco.alemanno@unisalento.it`