

Benchmarking results for the Newton–Anderson method

Sara Pollock^{*}, Hunter Schwartz

Department of Mathematics, University of Florida, Gainesville, FL, 32611, United States of America



ARTICLE INFO

Article history:

Received 12 November 2019

Received in revised form 9 January 2020

Accepted 13 January 2020

Available online 30 January 2020

MSC:

65B05

65H10

Keywords:

Anderson acceleration

Newton iteration

Degenerate problems

Superlinear convergence

ABSTRACT

This paper primarily presents numerical results for the Anderson accelerated Newton method on a set of benchmark problems. The results demonstrate superlinear convergence to solutions of both degenerate and nondegenerate problems. The convergence for nondegenerate problems is also justified theoretically. For degenerate problems, those whose Jacobians are singular at a solution, the domain of convergence is studied. It is observed in that setting that Newton–Anderson has a domain of convergence similar to Newton, but it may be attracted to a different solution than Newton if the problems are slightly perturbed.

© 2020 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

The purpose of this paper is to present some standard benchmark tests, including both degenerate and nondegenerate problems, for Anderson acceleration applied to Newton iterations, referred to as the Newton–Anderson method. The motivation behind studying this method stems from the results of [1], where it was shown how Anderson acceleration can locally improve the convergence rate of linearly converging fixed-point iterations for nondegenerate problems, and from the numerical results of [2], where Anderson acceleration applied to Newton's method was demonstrated to allow convergence for a finite element discretization of the steady Navier–Stokes equations with a Reynold's number high enough to cause standard Newton iterations to diverge. In that setting, the convergence history for Newton–Anderson was similar to that of a damped Newton method.

A further investigation in [3] proved that in one dimension, Anderson(1), Anderson acceleration with an algorithmic depth of $m = 1$, applied to Newton provides higher order (superlinear) convergence to nonsimple roots of scalar equations. Newton alone is known to only provide linear convergence for such problems [4, Chapter 6], at the rate $1 - 1/p$, where p is the multiplicity of the root. Generalizing to higher dimensions, it is reasonable to ask whether Newton–Anderson also provides superlinear convergence for degenerate systems: those whose Jacobians are singular at a solution. The convergence of Newton's method for such systems has been shown to be locally linear, in the intersection of a ball and a star-shaped domain about the solution [5,6]. The numerical results in Section 2.2 indicate that Newton–Anderson can provide such locally superlinear convergence, and that an algorithmic depth of $m = 1$ can be sufficient to accomplish this.

Results for the first accelerated Newton method of [7], which has been shown both theoretically and numerically to provide superlinear convergence for degenerate problems [7–9], are shown alongside Newton and Newton–Anderson. This

^{*} Corresponding author.

E-mail address: s.pollock@ufl.edu (S. Pollock).

method, which can be viewed either as a predictor–corrector or an extrapolation method applied to every other iteration, is similar in form to Anderson(1) (particularly when considered from the extrapolation viewpoint), but has convergence properties that are sensitive to the tuning of two parameters. This method will be referred to as accelerated-Newton (or KS-acc. N., in the tables of results). The advantage of Newton–Anderson(1), is that, to the present authors' knowledge, there is not a robust method of determining *a priori* the two parameters, $C > 0$ ($C \in \mathbb{R}$ as originally presented in [7]) and $\alpha \in (0, 1)$, of the accelerated-Newton method. A brief discussion of convergence theory for Newton–Anderson in the nondegenerate setting is included in Section 1.1. The analysis of the convergence properties of Newton–Anderson however, particularly in the degenerate setting, remains pending.

In Section 2, the Newton–Anderson method is first tested on problems from the standard benchmark problem set of [10] to identify further problem classes on which Newton–Anderson is either advantageous or disadvantageous; and, to identify problems or problem classes where Newton–Anderson(m), with $m > 1$, is preferable to Newton–Anderson(1). The only problem identified from this set that falls in the last category is the Powell badly scaled function. Another such example designed to demonstrate that there exist problems where depth $m > 1$ is preferred is (19) shown in Section 2.2. That degenerate problem was purposely designed to demonstrate Newton–Anderson(m) converging at iterate x_{m+1} , for a problem $f(x) = 0$, of the form $f_i(x) = [(Ax - b)_i]^{p_i}$, where $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, and there are $m \leq n$ distinct exponents p_i .

Next, the iterative schemes tested on the problems of Sections 2 and 3 are specified. Each one is given below in terms of a uniform damping parameter $\beta \in (0, 1]$, as is standard practice for all but Algorithm 4. Damping ($\beta \neq 1$) was only used here for one problem, the Brown almost-linear function on \mathbb{R}^n , with $n = 20$. In that case the damping factor $\beta = 0.8$ improved the convergence of all methods tested.

Algorithm 1 (Newton's Method). Set $\beta \in (0, 1]$. Choose x_0 .

For $k = 0, 1, \dots$

$$\text{Set } x_{k+1} = x_k - \beta[f'(x_k)]^{-1}f(x_k)$$

Algorithm 2 (Newton–Anderson(1) from [11]). Set $\beta \in (0, 1]$. Choose x_0 . Compute $w_1 = -[f'(x_0)]^{-1}f(x_0)$. Set $x_1 = x_0 + \beta w_1$

For $k = 1, 2, \dots$

$$\text{Compute } w_{k+1} = -[f'(x_k)]^{-1}f(x_k)$$

$$\text{Compute } \gamma^{k+1} = (w_{k+1}, w_{k+1} - w_k) / \|w_{k+1} - w_k\|^2$$

$$\text{Set } x_{k+1} = x_k + \beta w_{k+1} - \gamma^{k+1}((x_k - x_{k-1}) + \beta(w_{k+1} - w_k))$$

Algorithm 3 (Newton–Anderson(m) from [11]). Set $\beta \in (0, 1]$ and $m \geq 0$. Choose x_0 . Compute $w_1 = -[f'(x_0)]^{-1}f(x_0)$. Set

$$x_1 = x_0 + \beta w_1$$

For $k = 1, 2, \dots$, set $m_k = \min\{k, m\}$

$$\text{Compute } w_{k+1} = -[f'(x_k)]^{-1}f(x_k)$$

$$\text{Set } F_k = ((w_{k+1} - w_k) \dots (w_{k-m_k+2} - w_{k-m_k+1})), \text{ and } E_k = ((x_k - x_{k-1}) \dots (x_{k-m_k+1} - x_{k-m_k}))$$

$$\text{Compute } \gamma^{k+1} = \operatorname{argmin}_{\gamma \in \mathbb{R}^m} \|w_{k+1} - F_k \gamma\|$$

$$\text{Set } x_{k+1} = x_k + \beta w_{k+1} - (E_k + \beta F_k) \gamma^{k+1}$$

Algorithm 4 (Accelerated Newton from Theorem 1.3 of [7]). Set $\beta \in (0, 1]$ and parameters $C > 0$ and $\alpha \in (0, 1)$. Choose x_0 .

For $k = 1, 2, \dots$

$$\text{Compute } \hat{w}_{k+1} = -[f'(x_k)]^{-1}f(x_k).$$

$$\text{Set } y_{k+1} = x_k + \beta \hat{w}_{k+1}$$

$$\text{Compute } w_{k+1} = -[f'(y_{k+1})]^{-1}f(y_{k+1})$$

$$\text{Set } x_{k+1} = y_{k+1} - (2 - C \|w_{k+1}\|^\alpha) w_{k+1}$$

In accordance with [12], results are shown for Algorithm 4 with parameter choices $C = 1$ and $\alpha = 0.9$. In the cases where the iteration with those parameters failed to converge, results are additionally shown with parameters $C = 0.35$ and $\alpha = 0.1$. In [7], where Algorithm 4 is introduced, the given parameter range for C is \mathbb{R} ; however, $C > 0$ appears to be standard in presented demonstrations of the method [7–9,12], and $C \leq 0$ is not considered here.

If the norm $\|\cdot\|$ used in the optimization step of Algorithm 3 is induced by an inner product (\cdot, \cdot) , Algorithm 3 reduces to Algorithm 2, for $m = 1$ (and reduces to Algorithm 1, for $m = 0$). See [13,14] for the equivalence of this form of the Anderson algorithm to that originally stated in [11], and [15] for results on optimization in norms not induced by inner products. Throughout the remainder, the norm $\|\cdot\|$ will denote the l_2 norm over \mathbb{R}^n .

The remainder of the paper is structured as follows. In Section 1.1, superlinear convergence is shown theoretically for the accelerated algorithm. In Section 2, the first set of benchmark problems is presented in 2.1, then results are shown and discussed in 2.1.1–2.1.2. Additional degenerate problems are then presented in Section 2.2, for which results are shown and discussed in 2.3. Finally in Section 3, the domain of convergence for certain degenerate problems and small perturbations thereof are compared for the standard and accelerated algorithms.

1.1. Convergence theory

Convergence and acceleration theory for Anderson acceleration applied to fixed-point iterations has been recently studied under assumptions that the underlying fixed-point operator is contractive (linearly converging) [1,2,15–17], nonexpansive [18], or nondegenerate [1]. Here, a small contribution to the theoretical understanding of the method is introduced for the particular case of applying Anderson acceleration to Newton iterations. The first lemma shows for nondegenerate problems that Newton–Anderson(m) displays superlinear convergence under the assumption that the optimization coefficients are bounded. The second lemma shows for Newton–Anderson(1) that if the direction cosine between consecutive update steps is bounded below unity (the update steps do not become parallel), then the resulting optimization coefficient is bounded. This result is applied to the first lemma in Theorem 3 to show superlinear convergence of the error for Newton–Anderson(1). The results that follow in this section do not show that Newton–Anderson is an improvement over Newton (and where Newton converges quadratically, it generally is not), but they do show that the technique is reasonable.

While the underlying theory behind the superlinear convergence in degenerate settings is of interest to the authors, it is only demonstrated numerically here. The first part of the next assumption is a nondegeneracy condition which requires that the singular values of f' are uniformly bounded away from zero.

Assumption 1. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, be continuously Frechét differentiable, and suppose there exist positive constants σ and L such that

$$\sigma \|x\| \leq \|f'(y)x\|, \quad \text{for all } x, y \in \mathbb{R}^n, \quad (1)$$

$$\|f'(x) - f'(y)\| \leq L \|x - y\|, \quad \text{for all } x, y \in \mathbb{R}^n. \quad (2)$$

Based on Assumption 1, f' is invertible on \mathbb{R}^n , and since σ provides a lower bound on the smallest singular value of f' , its inverse satisfies

$$\|f'(x)^{-1}y\| \leq \frac{1}{\sigma} \|y\|, \quad \text{for all } x, y \in \mathbb{R}^n.$$

For the remainder of the section, suppose $f(x) = 0$ has solution x^* . The error in iteration k will be denoted by $e_k^* = x_k - x^*$.

Lemma 1 (Convergence Assuming Bounded Coefficients). Let Assumption 1 hold, and let x^* be a solution to the problem $f(x) = 0$. Then the errors $\{e_{k+1}^*\}_{k \geq 1}$ generated by Algorithm 3 with $\beta = 1$, satisfy

$$\|e_{k+1}^*\| \leq \frac{L\bar{\gamma}_{k+1}}{2\sigma} \sum_{j=k-m}^k \|e_j^*\|^2, \quad \text{where } \bar{\gamma}_{k+1} = \max \{2 \|\gamma^{k+1}\|_\infty, 1 + \|\gamma^{k+1}\|_\infty\}. \quad (3)$$

For the simple but useful case of depth $m = 1$, γ^{k+1} is a scalar coefficient and

$$\|e_{k+1}^*\| \leq \frac{L}{2\sigma} \left(|1 - \gamma^{k+1}| \|e_k^*\|^2 + |\gamma^{k+1}| \|e_{k-1}^*\|^2 \right). \quad (4)$$

The results of Lemma 1 show superlinear convergence if $\bar{\gamma}_{k+1}$ (or γ^{k+1} for $m = 1$) can be shown to be bounded. In Lemma 2, a condition is found on the direction cosines between w_{k+1} and w_k to guarantee this bound for the particular case of depth $m = 1$. While the more general result for depths $m > 1$ is not presented here, it can likely be obtained through an application of the machinery developed in [1] in which a QR decomposition is used to analyze the defect introduced by the matrix F_k of Algorithm 3 having columns that are not orthogonal.

Proof. For an arbitrary index $l \geq 1$, the usual estimates for Newton's method hold under Assumption 1. In particular expanding $f(x_l)$ about $0 = f(x^*)$ for an arbitrary index $l \geq 0$ allows $f(x_l) = \int_0^1 f'(x^* + te_l^*)e_l^* dt$. The error in $x_l + w_{l+1}$ then satisfies

$$x_l + w_{l+1} - x^* = e_l^* - f'(x_l)^{-1}f(x_l) = f'(x_l)^{-1}(f'(x_l)e_l^* - f(x_l)) = f'(x_l)^{-1} \int_0^1 (f'(x_l) - f'(x^* + te_l^*))e_l^* dt.$$

Under the two conditions of Assumption 1, the usual bound holds

$$\|x_l + w_{l+1} - x^*\| \leq \frac{L}{2\sigma} \|e_l^*\|^2. \quad (5)$$

Expanding the update step of [Algorithm 3](#), with m_k denoted by m , and the entries of γ^{k+1} denoted $\gamma_1, \dots, \gamma_m$, for simplicity

$$\begin{aligned} x_{k+1} &= x_k + w_{k+1} - \sum_{j=k-m+1}^k ((x_j - x_{j-1}) + (w_{j+1} - w_j))\gamma_{k-j+1} \\ &= (x_k + w_{k+1})(1 - \gamma_1) + (x_{k-1} + w_k)(\gamma_1 - \gamma_2) + \dots + (x_{k-m+1} + w_{k-m+2})(\gamma_{m-1} - \gamma_m) + (x_{k-m} + w_{k-m+1})\gamma_m \\ &= (x_k + w_{k+1})(1 - \gamma_1) + \sum_{j=k-m+1}^{k-1} (x_j + w_{j+1})(\gamma_{k-j} - \gamma_{k-j+1}) + (x_{k-m} + w_{k-m+1})\gamma_m. \end{aligned}$$

Writing $e^* = e^*((1 - \gamma_1) + (\gamma_1 - \gamma_2) + \dots + (\gamma_{m-1} - \gamma_m) + \gamma_m)$, the error e_{k+1}^* is represented by

$$e_{k+1}^* = (x_k + w_{k+1} - x^*)(1 - \gamma_1) + \sum_{j=k-m+1}^{k-1} (x_j + w_{j+1} - x^*)(\gamma_{k-j} - \gamma_{k-j+1}) + (x_{k-m} + w_{k-m+1} - x^*)\gamma_m. \quad (6)$$

Taking norms and applying (5) to each term of (6), yields the result (3).

For the simple case of $m = 1$, letting $\gamma = \gamma^{k+1}$, the estimate (5) is applied to the expansion

$$x_{k+1} - x^* = x_k + w_{k+1} - x^* - \gamma((x_k - x_{k-1}) + (w_{k+1} - w_k)) = (1 - \gamma)(x_k + w_{k+1} - x^*) + \gamma(x_{k-1} + w_k - x^*),$$

to produce the result (4).

The next lemma is specific to the algorithmic depth of $m = 1$, and produces a simple verifiable condition under which the optimization coefficient γ^{k+1} for depth $m = 1$ is bounded in magnitude by 1.

Lemma 2 (Boundedness Conditions for γ^{k+1} for Newton–Anderson(1)). *The coefficient γ^{k+1} produced by [Algorithm 2](#) (or [Algorithm 3](#) with $m = 1$) is bounded, so long as $\cos(w_{k+1}, w_k) \leq \bar{\alpha}$, for some $\bar{\alpha} < 1$. Specifically, for $\bar{\alpha} = 0.942$, it holds that $|\gamma| < 2$.*

Proof. For depth $m = 1$, $\gamma = \gamma^{k+1}$ is given by

$$\gamma = \frac{(w_{k+1}, w_{k+1} - w_k)}{\|w_{k+1} - w_k\|^2} = \frac{\|w_{k+1}^2\| - (w_{k+1}, w_k)}{\|w_{k+1}\|^2 + \|w_k\|^2 - 2(w_{k+1}, w_k)} = \frac{\|w_{k+1}\| / \|w_k\| - \cos(w_{k+1}, w_k)}{\|w_{k+1}\| / \|w_k\| + \|w_k\| / \|w_{k+1}\| - 2\cos(w_{k+1}, w_k)}, \quad (7)$$

where $\cos(u, v)$ is the direction cosine between vectors $u, v \in \mathbb{R}^n$. For $w_{k+1}, w_k \neq 0$, the right-hand side of (7) is of the form

$$\gamma = \psi_\alpha(r) = \frac{r - \alpha}{r + 1/r - 2\alpha}, \quad r > 0, \quad -1 \leq \alpha \leq 1. \quad (8)$$

Clearly $\psi_1(r)$ has a singularity as $r \rightarrow 1$; however for $\alpha < 1$, there are no singularities. Hence, so long as $\cos(w_{k+1}, w_k) \leq \bar{\alpha} < 1$, the coefficient γ remains bounded.

To be more precise and determine a value of $\bar{\alpha}$ for which $|\gamma| < 2$, for instance, it is first noted that $|\psi_\alpha(r)| < 1$ for $\alpha \in [-1, 0]$ and $r > 0$, and $\psi_\alpha(1) = 1/2$. So, it remains to investigate $\alpha \in (0, 1)$ for $r > 0$, $r \neq 1$. The extreme values of $\psi_\alpha(r)$ can first be found for each fixed value of $\alpha \in (0, 1)$, by setting $0 = \psi'_\alpha(r) = (-\alpha r^2 + 2r - \alpha)/(r^2 - 2\alpha r + 1)^2$. The numerator yields a quadratic equation in r to which the solution for $r < 1$ is $r_\alpha = (1 - \sqrt{1 - \alpha^2})/\alpha$. Plugging this into (8) yields

$$\psi_\alpha(r_\alpha) = \frac{(2/\alpha^2 - 1)(1 - \sqrt{1 - \alpha^2}) - 1}{2(1/\alpha^2 - 1)(1 - \sqrt{1 - \alpha^2})},$$

which is a decreasing function of α , for $\alpha > 0$. Investigating this function numerically, it is seen, for example, that $\psi_\alpha(r) > -0.990$ for $\alpha < 0.942$.

For $r > 1$, the extremum of $\psi_\alpha(r)$ occurs at $r_\alpha = (1 + \sqrt{1 - \alpha^2})/\alpha$, for which

$$\psi_\alpha(r_\alpha) = \frac{(2/\alpha^2 - 1)(1 + \sqrt{1 - \alpha^2}) - 1}{2(1/\alpha^2 - 1)(1 + \sqrt{1 - \alpha^2})},$$

which is an increasing function of α , for $\alpha > 0$; and, for which $\psi_\alpha(r) < 1.99$ for $\alpha < 0.942$.

Theorem 3 (Superlinear Convergence for Newton–Anderson(1)). *Let [Assumption 1](#) hold, and let x^* be a solution to the problem $f(x) = 0$. Then the errors $\{e_{k+1}^*\}_{k \geq 1}$ generated by [Algorithm 2](#) (or [Algorithm 3](#) with $m = 1$) and $\beta = 1$ converge superlinearly to zero if the direction cosines between update steps satisfy $\cos(w_{k+1}, w_k) < \bar{\alpha}$, for some constant $\bar{\alpha} < 1$.*

Proof. The result follows immediately from applying the results of Lemma 2 to the result (4) of Lemma 1.

This shows that if consecutive update steps are not parallel (anti-parallel is harmless), then convergence is locally better than linear. The superlinear convergence for nondegenerate problems agrees with the numerical results shown below. Consistent with the theory, the convergence is also observed to be subquadratic. While it is not demonstrated here, superlinear convergence for degenerate problems (which violate Assumption 1) is also observed.

2. Benchmark results

The first set of problems considered are the systems of nonlinear equations that map $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, from the standard test set of Moré et al. [10]. The problems and results are stated in Section 2.1. Section 2.2 that follows specifically considers additional degenerate problems.

Throughout this section, each method was terminated on the residual reduced below the tolerance $\|f(x_k)\| < 10^{-8}$. In Section 3 discussing the domain of convergence to particular solutions, iterations were terminated based on error dropping below tolerance $\|x_k - x^*\| < 10^{-8}$, where x^* is a given (known) solution. All computations were done in Matlab on an 8 core Intel Xeon W-2145 CPU @ 3.70 GHz, with 64 GB memory. Timing was performed using Matlab's tic/toc commands. Each algorithm/problem pair was run 100 times, and the results were averaged to produce the time displayed in the tables of results.

Problems from the *Systems of Nonlinear Equations* test set of [10] for which all methods converged in no more than four iterations, in addition to the Chebyquad function, are excluded from the test set below. In a slight (but standard) abuse of notation, subscripts are used in the algorithms above to indicate iteration counts, and in the problem descriptions below to denote components of $x \in \mathbb{R}^n$.

2.1. Problems $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ from [10]

B1. Powell badly scaled function, [10, (3)] and [19], $n = 2$.

$$f(x) = \begin{pmatrix} 10^4 x_1 x_2 - 1 \\ \exp(-x_1) + \exp(-x_2) - 1.0001 \end{pmatrix}, \quad \text{with } x_0 = (0, 1)^T. \quad (9)$$

B2. Helical valley function, [10, (7)] and [20], $n = 3$.

$$f(x) = \begin{pmatrix} 10(x_3 - 10\theta(x_1, x_2)) \\ 10(\sqrt{x_1^2 + x_2^2} - 1) \\ x_3 \end{pmatrix}, \quad \text{where } \theta(x_1, x_2) = \frac{1}{2\pi} \arctan(x_2/x_1), \quad \text{with } x_0 = (-1, 0, 0)^T, \quad (10)$$

where $\arctan(\cdot)$ is the four-quadrant arctangent function.

B3. Powell singular function, [10, (13)] and [21], $n = 4$.

$$f(x) = \begin{pmatrix} x_1 + 10x_2 \\ \sqrt{5}(x_3 - x_4) \\ (x_2 - 2x_3)^2 \\ \sqrt{10}(x_1 - x_4)^2 \end{pmatrix}, \quad \text{with } x_0 = (3, -1, 0, 1)^T. \quad (11)$$

B4. Watson function, [10, (20)] and [22], $n = 31$.

$$f_i(x) = \sum_{j=2}^n (j-1)x_j t_i^{j-2} - \left(\sum_{j=1}^n x_j t_i^{j-1} \right)^2 - 1, \quad \text{where } t_i = i/29, \quad 1 \leq i \leq 29, \\ f_{30}(x) = x_1, \quad f_{31}(x) = x_2 - x_1^2 - 1, \quad \text{with } x_0 = (0, \dots, 0)^T. \quad (12)$$

B5. Trigonometric function, [10, (26)] and [20], $n = 100, 1000, 10000$.

$$f_i(x) = n - \sum_{j=1}^n \cos x_j + i(1 - \cos x_i) - \sin x_i, \quad \text{with } x_0 = (1/n, \dots, 1/n)^T. \quad (13)$$

B6. Brown almost-linear function, [10, (27)] and [23], $n = 5, 20$.

$$f_i(x) = x_i + \sum_{j=1}^n x_j - (n+1), \quad 1 \leq i < n, \\ f_n(x) = \left(\prod_{j=1}^n x_j \right) - 1, \quad \text{with } x_0 = (1/2, \dots, 1/2)^T. \quad (14)$$

Table 1

Results for problems from [10] where Newton–Anderson(1) converges.

Problem	Method	Iterations (k)	$\ f(x_k)\ $	$\ w_k\ $	q_k	Time (s)
B2. (10) $n = 3$	Newton	10	1.325e−14	4.161e−08	2.235	9.655e−05
	N. Anderson(1)	10	5.485e−13	1.001e−08	1.794	0.0001249
	KS-acc. N. (1.0, 0.9)	F	–	–	–	–
	KS-acc. N. (0.35, 0.1)	11	1.042e−16	6.591e−18	2.391	0.0001292
B3. (11) $n = 4$	Newton	16	2.954e−09	3.743e−05	1.076	0.0002417
	N. Anderson(1)	3	4.163e−17	0.2157	40.27	0.0001412
	KS-acc. N. (1.0, 0.9)	3	5.964e−10	0.003071	2.638	0.0001279
B4. (12) $n = 31$	Newton	5	6.004e−14	13.5	1.71	0.01759
	N. Anderson(1)	7	2.863e−10	3.341	1.73	0.02455
	KS-acc. N. (1.0, 0.9)	4	2.215e−13	9.818	2.807	0.02716
B5. (13) $n = 100$	Newton	10	1.892e−11	6.137e−07	1.726	0.003895
	N. Anderson(1)	8	9.565e−13	1.159e−08	1.518	0.003095
	KS-acc. N. (1.0, 0.9)	7	3.568e−12	3.571e−12	1.741	0.003868
B5. (13) $n = 1000$	Newton	13	9.906e−11	4.454e−07	1.575	0.4408
	N. Anderson(1)	11	1.653e−11	2e−08	1.4	0.3734
	KS-acc. N. (1.0, 0.9)	6	1.302e−09	1.308e−09	1.534	0.3695
B7. (15) $n = 1000$	Newton	4	1.065e−09	4.555e−05	2.312	0.01596
	N. Anderson(1)	6	1.59e−14	6.612e−09	1.788	0.02284
	KS-acc. N. (1.0, 0.9)	4	1.904e−13	6.395e−14	2.187	0.02831
B8. (16) $n = 1000$	Newton	8	5.669e−09	2.419e−08	1.24	0.09291
	N. Anderson(1)	9	3.668e−10	1.768e−09	1.2	0.1056
	KS-acc. N. (1.0, 0.9)	9	3.204e−10	6.07e−11	1.19	0.1814
B6. (14) $n = 5$	Newton	18	3.14e−15	6.481e−08	2.058	0.0002524
	N. Anderson(1)	24	5.031e−12	2.935e−07	1.555	0.0003786
	KS-acc. N. (1.0, 0.9)	F	–	–	–	–
	KS-acc. N. (0.35, 0.1)	9	3.397e−15	2.334e−16	1.963	0.0002039
B6. (14) $n = 20$	Newton, $\beta = 0.8$	368	4.743e−09	4.854e−07	1.092	0.02743
	N. Anderson(1), $\beta = 0.8$	52	7.394e−10	3.853e−09	1.088	0.004206
	KS-acc. N. (1.0, 0.9), $\beta = 0.8$	F	–	–	–	–
	KS-acc. N. (0.35, 0.1), $\beta = 0.8$	147	1.931e−09	4.224e−08	1.602	0.02054

B7. Broyden tridiagonal function, [10, (30)] and [24], $n = 1000$.

$$f_i(x) = (3 - 2x_i)x_i - x_{i-1} - 2x_{i+1} + 1, \quad (x_0 = x_{n+1} = 0), \quad \text{with } x_0 = (-1, \dots, -1)^T. \quad (15)$$

B8. Broyden banded function, [10, (31)] and [25], $n = 1000$.

$$f_i(x) = x_i(2 + 5x_i^2) + 1 - \sum_{j \in J_i} x_j(1 + x_j), \quad J_i = \{j : j \neq i, \max(1, i - 5) \leq j \leq \min(n, i + 1)\},$$

$$\text{with } x_0 = (-1, \dots, -1)^T. \quad (16)$$

2.1.1. Results

The first set of results, those for which Anderson(1) was preferred over Anderson(m), with $m > 1$, are summarized in Table 1. Here, results are given for Newton, Newton–Anderson(1), and accelerated-Newton. An entire predictor–corrector step is counted as a single iteration for the accelerated-Newton method. For each problem and method, the number of iterations (k), terminal residual $\|f(x_k)\|$, norm of the final update step $\|w_k\|$, the terminal approximate order of convergence $q_k = \log \|f(x_k)\| / \log \|f(x_{k-1})\|$, and the (average) time in seconds, are displayed.

2.1.2. Discussion

In accordance with the expected results, for the Powell singular function (11), whose Jacobian is singular at the solution (with rank 2), Newton converges linearly, and both accelerated and Anderson methods yield faster convergence, as seen in the convergence histories shown in Fig. 1, on the right. In terms of timing, Newton–Anderson(1) is advantageous over accelerated-Newton as the problem dimension n increases, as Newton–Anderson(1) solves a linear system plus computes two additional inner products to set the optimization parameter, whereas accelerated-Newton solves two linear systems. This difference is illustrated by comparing the results of problem (11) with $n = 4$ to problem (16) with $n = 1000$, where both methods converge in the same number of iterations.

Newton–Anderson(1) was found disadvantageous compared to Newton in terms of timing for the Watson function (12), with $n = 31$, and the Broyden tridiagonal function with $n = 1000$ (15), taking respectively 1.40 and 1.44 times as long as Newton to converge (but only two additional iterations). It was found nominally disadvantageous for the Broyden banded function (16), with $n = 1000$, taking 1.14 times as long to converge as Newton, with one additional iteration.

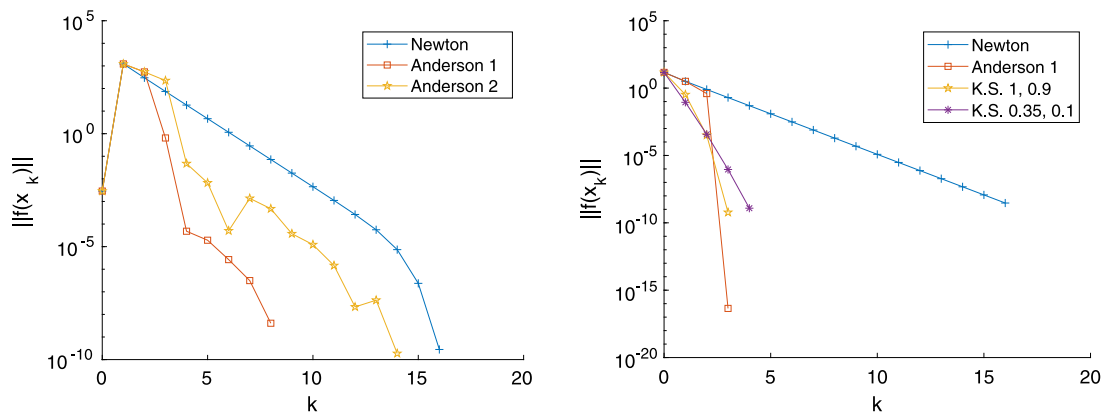


Fig. 1. Left: convergence history for the trigonometric function (13), with $n = 10,000$, for Newton and Newton–Anderson(m), $m = 1, 2$. Right: convergence history for the Powell singular function (11), illustrating the high terminal order of convergence for Newton–Anderson(1).

Table 2

Results for problems from [10] where Newton–Anderson(1) fails to converge.

Problem	Method	Iterations (k)	$\ f(x_k)\ $	$\ w_k\ $	q_k	Time (s)
B1. (9) $n = 2$	Newton	12	1.573e−11	3.987e−05	1.769	0.0001241
	N. Anderson(1)	F	–	–	–	–
	N. Anderson(2)	12	4.058e−09	0.0001451	1.518	0.0003492
	KS-acc. N. (1.0, 0.9)	8	5.773e−15	5.886e−16	1.818	7.731e−05

Newton–Anderson and Newton both converged in 10 iterations for the Helical valley function (10), with $n = 3$. In each of the above cases, Newton–Anderson converged faster than accelerated-Newton. In agreement with the results of [1,2], these results indicate that Anderson slows the convergence of quadratically-converging iterations, and should not be used on problems for which Newton is known to converge robustly and quadratically.

For the Brown almost-linear function (14), Newton–Anderson converged in 1.5 times as long as Newton (6 additional iterations) with $n = 5$ and no damping, but 0.15 times as long as Newton (316 fewer iterations), for the problem with dimension $n = 20$ with a damping factor of 0.8. In both cases, accelerated-Newton failed to converge with parameters $(C, \alpha) = (1.0, 0.9)$, and succeeded with parameters (0.35, 0.1). Taken together, these results indicate that Newton–Anderson can be useful for problems where convergence of Newton is locally quadratic but less stable as the dimensions of the problem increases.

For the trigonometric function (13), as n increases, the Newton method has an extended preasymptotic regime of linear convergence. Both accelerated methods, which display superlinear (but subquadratic) convergence, attain residual tolerance in fewer iterations. Newton–Anderson(m) for $m = 1, 2$ are illustrated for $n = 10,000$, in Fig. 1 on the left. On the right of Fig. 1, the residual history for the Powell singular function is shown. The plot illustrates the curiously high approximate order of convergence for Anderson(1). The accelerated-Newton method is shown for comparison, with both sets of tested parameters, along with Newton.

Results for the Powell badly scaled function (9) are shown in Table 2. For this problem, Newton–Anderson(m) converges for $m > 1$, but not $m = 1$. Tests (not shown) were run for $m = 1, \dots, 20$. However, as $n = 2$ for this problem, the optimization problem reduces to a solvable linear system for depth $m = 2$, and results with greater depths are not shown. Convergence can be restored for $m = 1$ by applying a safeguarding strategy based on Lemma 2. If a Newton step is taken whenever $\cos(w_{k+1}, w_k) > 0.942$, and the Anderson(1) step otherwise, then Newton–Anderson(1) converges, essentially tracking the Newton iteration.

2.2. Additional degenerate problems

Based on the results of [3], which show superlinear convergence of Newton–Anderson(1) to nonsimple roots of scalar equations, it is conjectured that Newton–Anderson also provides superlinear convergence to solutions of degenerate problems. Three such problems (in addition to the Powell singular function above) are collected in this section. The first two are standard problems from the literature and the third was chosen to demonstrate a case where Newton–Anderson(4) converges faster than Newton–Anderson(m), for $m = 1, 2, 3$. (See Table 3.)

Table 3

Results for degenerate problems.

Problem	Method	Iterations (k)	$\ f(x_k)\ $	$\ w_k\ $	q_k	Time (s)
D1. $n = 3$	Newton	14	3.991e-09	6.903e-05	1.077	0.000129
	N. Anderson(1)	5	1.656e-10	1.349e-05	1.493	7.152e-05
	KS-acc. N. (1.0, 0.9)	3	2.396e-09	0.0001877	1.993	4.476e-05
D2. $n = 1000$	Newton	16	2.628e-09	0.000382	1.075	0.9698
	N. Anderson(1)	6	1.236e-11	0.001947	1.663	0.3671
	N. Anderson(2)	6	1.912e-11	2.595e-06	1.399	0.3658
	KS-acc. N. (1.0, 0.9)	F	–	–	–	–
	KS-acc. N. (0.35, 0.1)	4	3.986e-09	0.002449	1.461	0.4772
D3. $n = 10$	Newton	46	4.339e-09	0.07587	1.057	0.001038
	N. Anderson(1)	17	7.899e-09	0.01347	1.45	0.0008288
	N. Anderson(2)	26	6.781e-11	0.0003507	1.404	0.002012
	N. Anderson(3)	6	3.964e-10	0.09431	5.87	0.000219
	N. Anderson(4)	5	7.289e-25	0.1056	23.05	0.0001861
	KS-acc. N. (1.0, 0.9)	F	–	–	–	–
	KS-acc. N. (0.35, 0.1)	F	–	–	–	–
	KS-acc. N. (0.7, 0.3)	20	1.758e-09	0.07474	1.165	0.0007143

D1. From [6], $f : \mathbb{R}^3 \rightarrow \mathbb{R}^3$

$$f(x_1, x_2, x_3) = \begin{pmatrix} x_1 + x_1x_2 + x_2^2 \\ x_1^2 - 2x_1 + x_2^2 \\ x_1 + x_2^2 \end{pmatrix} \quad \text{with } x_0 = (0.1, 0.5, 1.0)^T. \quad (17)$$

D2. Chandrasekhar H-equation from [16]. The Chandrasekhar H-equation from radiative transfer (see [16,26,27] and the references therein), is given by

$$F(H)(\mu) := H(\mu) - \left(1 - \frac{\omega}{2} \int_0^1 \frac{\mu H(v) dv}{\mu + v}\right)^{-1} = 0.$$

Discretizing the equation (as described in [16, Example 2.10]) by the composite midpoint rule with n subintervals, yields the discrete system in $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ given componentwise for $h \in \mathbb{R}^n = (h_1, \dots, h_n)^T$, by

$$f_i(h) = h_i - \left(1 - \frac{\omega}{2n} \sum_{j=1}^n \frac{ih_j}{i+j-1}\right), \quad 1 \leq i \leq n, \quad \text{with } x_0 = (1, \dots, 1)^T. \quad (18)$$

D3. Problem designed to demonstrate Newton–Anderson(4).

$$f_i(x) = [(Ax - b)_i]^{p_i}, \quad A = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}, \quad b = (-11, -7, -5, -3, -2, 2, 3, 5, 7, 11)^T, \\ p = (2, 4, 4, 2, 2, 8, 8, 2, 12, 12), \quad \text{with } x_0 = (0, \dots, 0)^T. \quad (19)$$

2.3. Discussion

In problems (17), (18), and (19), Newton–Anderson(1) converges faster than Newton by respective factors of 0.55, 0.38, and 0.8. Newton–Anderson(4) outperforms Newton–Anderson(1) by a factor of 0.22, and accelerated-Newton by a factor of 0.26, in the last problem. The approximate convergence order q_k is close to one for Newton in each case, and greater than one (generally above 1.4) for each of the accelerated methods, when they converge. The accelerated-Newton method was run with an additional parameter pair for (19) as the method did not converge under either of the other two parameter-pairs used in the rest of the tests. The varied nature of these problems in terms of both problem dimension, and the dimension of the nullspaces at their respective solutions, indicates consistent behavior of Newton–Anderson on degenerate problems, which is worth a theoretical investigation.

The first problem (17) has a Jacobian featuring a two-dimensional nullspace at the solution. For the Chandrasekhar H-equation (18), the rank of the Jacobian at the solution is $n - 1$. For the degenerate polynomial problem (19), the Jacobian has rank zero at the solution. The problem was designed to solve exactly for $m = 4$ after the first full optimization. It is shown here to illustrate that there exists a class of problems for which $m > 1$ is preferable. It is not yet clear what this problem class is.

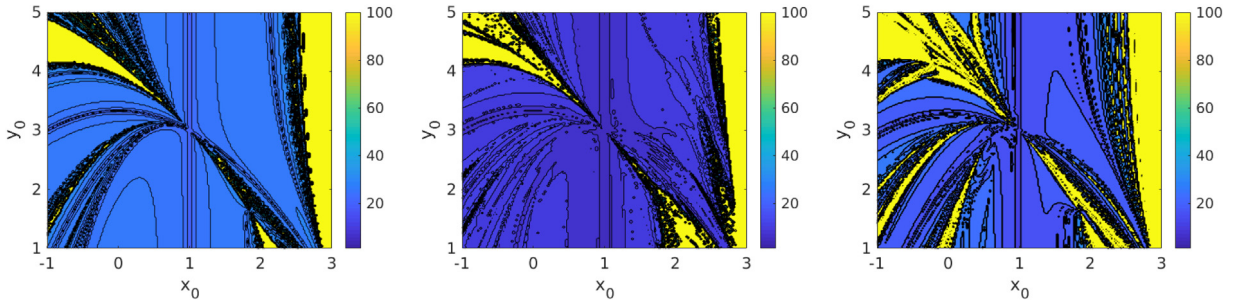


Fig. 2. Domain of convergence to $x_+ = x_-$ for (20) with $\varepsilon = 0$. Left: Newton iterations; center: Newton-Anderson(1); right: accelerated-Newton with $C = 0.5$ and $\alpha = 0.35$.

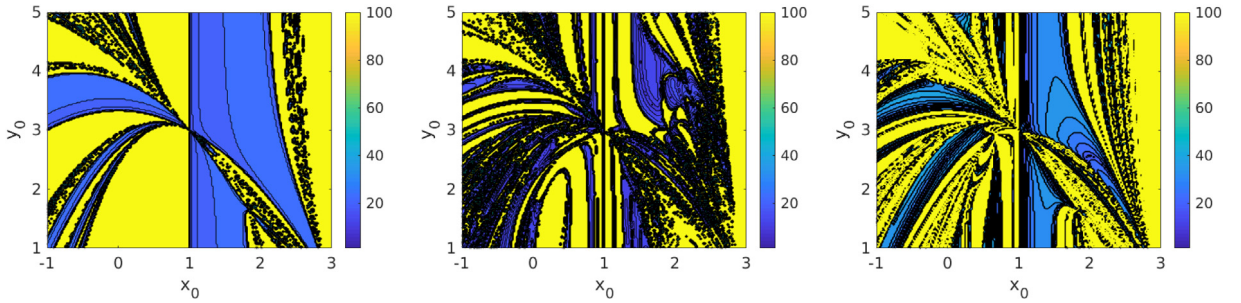


Fig. 3. Domain of convergence to x_+ for (20) with $\varepsilon = 10^{-6}$. Left: Newton iterations; center: Newton-Anderson(1); right: accelerated-Newton.

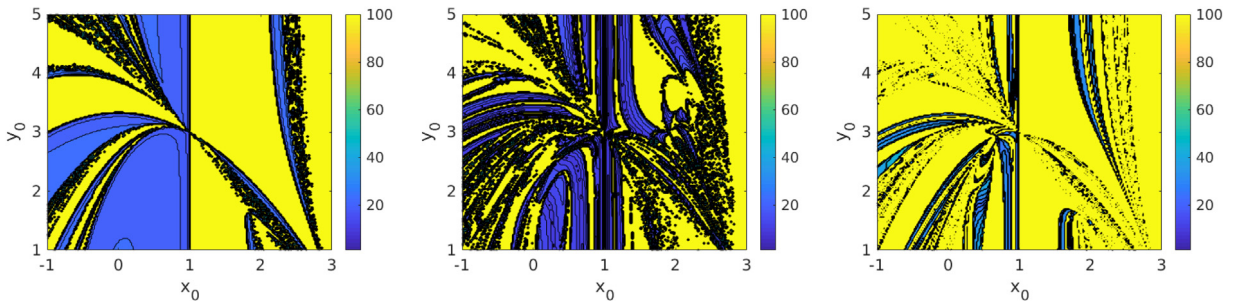


Fig. 4. Domain of convergence to x_- for (20) with $\varepsilon = 10^{-6}$. Left: Newton iterations; center: Newton-Anderson(1); right: accelerated-Newton.

3. Domain of convergence

In this section, the domains of convergence for Newton, Newton-Anderson(1), and accelerated-Newton are compared on standard benchmark degenerate and nearly degenerate polynomial systems.

The first investigated problem is the polynomial system $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ from [8, section 5].

$$f(x_1, x_2) = \begin{pmatrix} (x_1 - 1) + (x_2 - 3)^2 \\ \varepsilon(x_2 - 3) + (3/2)(x_1 - 1)(x_2 - 3) + (x_2 - 3)^2 + (x_2 - 3)^3 \end{pmatrix}. \quad (20)$$

As presented in [8], for $\varepsilon > 0$ there are two distinct roots in the vicinity of $x_+ = (1, 3)^T$, the second being $x_- = (1 - \eta^2, 3 + \eta)^T$, for $\eta = 1 - \sqrt{1 + 2\varepsilon}$. For $\varepsilon = 0$, the degenerate case, the two roots coincide; and, for $\varepsilon \ll 1$ but $\varepsilon > 0$, the near-degenerate case, the two roots are distinct but close. For Figs. 2–4, each algorithm is run from the initial iterate (x_0, y_0) for $x_0 \in [-1, 3]$ and $y_0 \in [1, 5]$ over a grid of 40,000 equally spaced points, and the number of iterations to convergence to a particular solution is displayed. The maximum of 100 iterations indicates not converging to the specified solution. As demonstrated in Fig. 2, Algorithm 2 has a similar domain of convergence to Newton (Algorithm 1) and accelerated-Newton, (Algorithm 4); but, as illustrated in Figs. 3–4, as ε is varied away from zero, the Anderson method has substantially different domains of attraction to the roots x_- and x_+ . For this example, Algorithm 4 was run with parameters $C = 0.5$ and $\alpha = 0.35$, presented as optimal for $\varepsilon = 10^{-6}$ in [8].

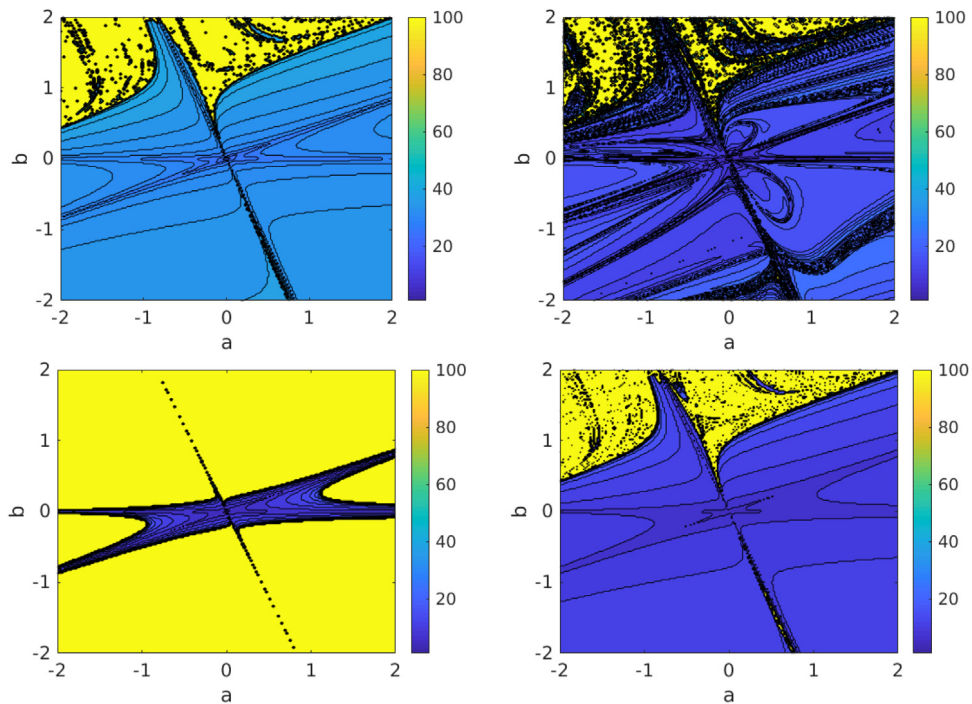


Fig. 5. Domain of convergence to $x^* = (0, 0, 0)^T$ for (17), with $x_0 = (a, b, 10^{-3})^T$. Top left: Newton iterations; top right: Newton-Anderson(1); bottom left: accelerated-Newton with $C = 1.0$, $\alpha = 0.9$; bottom right: accelerated-Newton with $C = 0.1$, $\alpha = 0.35$.

The second system investigated for domain of convergence is (17), from Section 2.2. In this system, the Jacobian degenerates for coordinates $x_3 = 0$, and also for $x_1 = x_2 = 0$. Here the domain of convergence is investigated for $x_0 = (a, b, 10^{-3})^T$, as the first two components are varied, again over a grid of 40,000 equally spaced initial iterates. This example points out that the domains of convergence are similar for the three algorithms, but the accelerated-Newton method is sensitive to parameter choice. With α and C chosen small enough, the convergence is similar to Newton and Newton-Anderson, but with the parameters chosen too large ($C = 1$, $\alpha = 0.9$), the star-shaped domain of convergence is considerably smaller. (See Fig. 5.)

4. Conclusion

This results of this numerical benchmarking investigation of the Newton-Anderson method indicate a superlinear but (usually) subquadratic order of convergence on both degenerate problems and nondegenerate problems. The superlinear convergence for nondegenerate problems is also shown theoretically. Collectively, the numerical results presented here show the main advantage of the Newton-Anderson method, and Newton-Anderson(1) in particular, is robust superlinear convergence for degenerate problems. The increase in performance for these problems is obtained with minimal additional computational effort per iteration, plus the requirement of storing $2m$ additional vectors. The secondary advantage of the method is improved performance for nondegenerate problems featuring an extended preasymptotic regime, such as the trigonometric function (13) considered here, if a good initial iterate for Newton's method is not available. Further investigations show the method can show a similar domain of convergence to the standard Newton iteration for degenerate problems; however, under small perturbation of a problem parameter, the Newton-Anderson method may be attracted to a different solution than the Newton iteration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Sara Pollock: Conceptualization, Investigation, Funding acquisition. **Hunter Schwartz:** Investigation.

Acknowledgments

SP was partially supported by NSF, United States of America DMS 1852876. The authors would like to thank David Burrell (University of Florida) for preliminary numerical results on degenerate systems.

References

- [1] Pollock S, Rebholz LG. Anderson acceleration for contractive and noncontractive iterations. 2019, Available as: [arXiv:1810.08455](https://arxiv.org/abs/1810.08455) [math.NA].
- [2] Evans C, Pollock S, Rebholz L, Xiao M. A proof that Anderson acceleration improves the convergence rate in linearly converging fixed point methods (but not in those converging quadratically). *SIAM J Numer Anal* 2020 [in press].
- [3] Pollock S. Fast convergence to higher-multiplicity zeros. 2019, Available as: [arXiv:1911.10647](https://arxiv.org/abs/1911.10647) [math.NA].
- [4] Quarteroni A, Sacco R, Saleri F. Numerical mathematics. Texts in applied mathematics, Berlin, Heidelberg: Springer-Verlag; 2006.
- [5] Reddien GW. On Newton's method for singular problems. *SIAM J Numer Anal* 1978;15(5):993–6.
- [6] Reddien G. Newton's method and high order singularities. *Comput Math Appl* 1979;5(2):79–86. [http://dx.doi.org/10.1016/0898-1221\(79\)90061-0](https://doi.org/10.1016/0898-1221(79)90061-0).
- [7] Kelley CT, Suresh R. A new acceleration method for Newton's method at singular points. *SIAM J Numer Anal* 1983;20(5):1001–9. [http://dx.doi.org/10.1137/0720070](https://doi.org/10.1137/0720070).
- [8] Decker D, Kelley CT. Expanded convergence domains for Newton's method at nearly singular roots. *SIAM J Sci Stat Comput* 1985;6(4):951–66. [http://dx.doi.org/10.1137/0906064](https://doi.org/10.1137/0906064).
- [9] Griewank A. On solving nonlinear equations with simple singularities or nearly singular solutions. *SIAM Rev* 1985;27(4):537–63.
- [10] Moré JJ, Garbow BS, Hillstom KE. Testing unconstrained optimization software. *ACM Trans Math Software* 1981;7(1):17–41. [http://dx.doi.org/10.1145/355934.355936](https://doi.org/10.1145/355934.355936).
- [11] Anderson DG. Iterative procedures for nonlinear integral equations. *J Assoc Comput Mach* 1965;12(4):547–60. [http://dx.doi.org/10.1145/321296.321305](https://doi.org/10.1145/321296.321305).
- [12] Kelley CT. A Shamanskii-like acceleration scheme for nonlinear equations at singular roots. *Math Comp* 1986;47(176):609–23.
- [13] Fang H, Saad Y. Two classes of multisection methods for nonlinear acceleration. *Numer Linear Algebra Appl* 2009;16(3):197–221. [http://dx.doi.org/10.1002/nla.617](https://doi.org/10.1002/nla.617).
- [14] Walker HF, Ni P. Anderson acceleration for fixed-point iterations. *SIAM J Numer Anal* 2011;49(4):1715–35. [http://dx.doi.org/10.1137/10078356X](https://doi.org/10.1137/10078356X).
- [15] Toth A, Kelley CT. Convergence analysis for Anderson acceleration. *SIAM J Numer Anal* 2015;53(2):805–19. [http://dx.doi.org/10.1137/130919398](https://doi.org/10.1137/130919398).
- [16] Kelley CT. Numerical methods for nonlinear equations. *Acta Numer* 2018;27:207–87. [http://dx.doi.org/10.1017/S0962492917000113](https://doi.org/10.1017/S0962492917000113).
- [17] Pollock S, Rebholz L, Xiao M. Anderson-accelerated convergence of Picard iterations for incompressible Navier–Stokes equations. *SIAM J Numer Anal* 2019;57(2):615–37. [http://dx.doi.org/10.1137/18M1206151](https://doi.org/10.1137/18M1206151).
- [18] Zhang J, O'Donoghue B, Boyd S. Globally convergent type-I Anderson Acceleration for non-smooth fixed-point iterations. 2018, Available as: [arXiv:1808.03971](https://arxiv.org/abs/1808.03971) [math.OC].
- [19] Powell MJD. A hybrid method for nonlinear equations. In: Rabinowitz P, editor. Numerical methods for nonlinear algebraic equations. New York: Gordon and Breach; 1970, p. 87–114.
- [20] Fletcher R, Powell MJD. A rapidly convergent descent method for minimization. *Comput J* 1963;6(2):163–8. [http://dx.doi.org/10.1093/comjnl/6.2.163](https://doi.org/10.1093/comjnl/6.2.163).
- [21] Powell MJD. An iterative method for finding stationary values of a function of several variables. *Comput J* 1962;5(2):147–51. [http://dx.doi.org/10.1093/comjnl/5.2.147](https://doi.org/10.1093/comjnl/5.2.147).
- [22] Kowalik JS, Osborne MR. Methods for unconstrained optimization problems. Modern analytic and computational methods in science and mathematics, vol. 13, New York: Elsevier; 1968.
- [23] Brown KM. A quadratically convergent Newton-like method based upon Gaussian elimination. *SIAM J Numer Anal* 1969;6(4):560–9. [http://dx.doi.org/10.1137/0706051](https://doi.org/10.1137/0706051).
- [24] Broyden CG. A class of methods for solving nonlinear simultaneous equations. *Math Comp* 1965;19(92):577–93. [http://dx.doi.org/10.1090/S0025-5718-1965-0198670-6](https://doi.org/10.1090/S0025-5718-1965-0198670-6).
- [25] Broyden CG. The convergence of an algorithm for solving sparse linear systems. *Math Comp* 1971;25(114):285–94. [http://dx.doi.org/10.1090/S0025-5718-1971-0297122-5](https://doi.org/10.1090/S0025-5718-1971-0297122-5).
- [26] Busbridge IW. The mathematics of radiative transfer. Cambridge tracts in mathematics and mathematical physics, no. 50, University Press; 1960.
- [27] Chandrasekhar S. Radiative transfer. Engineering special collection, Dover Publications; 1960.