IMA Journal of Numerical Analysis (2021) **41**, 2841–2872
https://doi.org/10.1093/imanum/draa095
Advance Access publication on 6 January 2021

Anderson acceleration for contractive and noncontractive operators

Sara Pollock*
Department of Mathematics, University of Florida, Gainesville, FL 32611, USA
*Corresponding author: s.pollock@ufl.edu

and

Leo G. Rebholz
School of Mathematical and Statistical Sciences, Clemson University, Clemson, SC 29634, USA

[Received on 18 June 2020; revised on 15 October 2020]

A one-step analysis of Anderson acceleration with general algorithmic depths is presented. The resulting residual bounds within both contractive and noncontractive settings reveal the balance between the contributions from the higher and lower order terms, which are both dependent on the success of the optimization problem solved at each step of the algorithm. The new residual bounds show the additional terms introduced by the extrapolation produce terms that are of a higher order than was previously understood. In the contractive setting these bounds sharpen previous convergence and acceleration results. The bounds rely on sufficient linear independence of the differences between consecutive residuals, rather than assumptions on the boundedness of the optimization coefficients, allowing the introduction of a theoretically sound safeguarding strategy. Several numerical tests illustrate the analysis primarily in the noncontractive setting, and demonstrate the use of the method, the safeguarding strategy and theory-based guidance on dynamic selection of the algorithmic depth, on a p-Laplace equation, a nonlinear Helmholtz equation and the steady Navier–Stokes equations with high Reynolds number in three spatial dimensions.

Keywords: Anderson acceleration; extrapolation; noncontractive operators.

1. Introduction

Anderson acceleration (AA) is an extrapolation technique that recombines a given number of the most recent iterates and update steps in a fixed-point iteration to improve the convergence properties of the sequence. The coefficients of the linear combination used in the update are recomputed at each iteration by the solution to an optimization problem, which determines a least-length update step. The technique was originally introduced in the context of integral equations in Anderson (1965). It has since been used in many applications over the past decade for various types of flow problems, for instance in Lott *et al.* (2012); Both *et al.* (2019); Pollock *et al.* (2019); Evans *et al.* (2020); geometry optimization in Peng *et al.* (2018); electronic structure computations in Fang & Saad (2009); radiation diffusion and nuclear physics in Toth *et al.* (2015); An *et al.* (2017); computing nearest correlation matrices in Higham & Strabic (2016); molecular interaction in Stasiak & Matsen (2011); and on a wide range of nonlinear problems in Walker & Ni (2011), among others.

In terms of its analysis AA was shown to be in the class of generalized quasi-Newton methods in Eyert (1996) and Fang & Saad (2009). In Walker & Ni (2011) it was shown that in the linear case, the variant of the method related to Type II Broyden methods is 'essentially equivalent' to Generalized Minimal RESidual method (GMRES), while the Type I variant is essentially equivalent to Arnoldi. In the remainder we restrict our attention to the (standard) Type II variant and consider its use on the

solution of nonlinear problems. Recently, in Brezinski *et al.* (2018) a nontrivial (cf. Walker & Ni, 2011) mathematical connection between AA and classical extrapolation algorithms used to accelerate vector sequences, including the (Modified) Minimal Polynomial Extrapolation, Topological and Vector Epsilon Algorithms and Reduced Rank Extrapolation algorithms, was established (see the review paper Smith *et al.*, 1987, and the references therein for further discussion on the relation between these more classical methods). Meanwhile, the first mathematical results showing local convergence of AA for contractive nonlinear operators were developed in Toth & Kelley (2015) and sharpened in Kelley (2018). The first results to prove how AA improves the convergence rate in fixed point iterations were written by the authors in Pollock *et al.* (2019) and Evans *et al.* (2020). The present work improves upon the results by Evans *et al.* (2020), by further exploiting the relationship between the optimization coefficients and optimization gain, made possible by analyzing the least-squares problem as it is discussed in Fang & Saad (2009) using a QR factorization.

This paper presents a novel one-step analysis that both sharpens and generalizes the AA convergence theory developed for contractive operators in Evans *et al.* (2020). The new one-step estimates hold for fixed-point iterations of contractive operators, or for zero-finding fixed-point iterations based on operators whose Jacobians do not degenerate. The latter are of particular importance in the numerical approximation of nonlinear partial differential equations (PDEs). The presented theory does not guarantee convergence of the sequence of iterations for noncontractive operators, unless the optimization problem is assumed to be sufficiently successful at each iteration. However, it succeeds at explaining the mechanism by which AA applied to this broad class of noncontractive fixed-point operators often does converge, and it provides insight into the design of more robust and efficient algorithms, as demonstrated in the practical guidance and in the numerical results.

One of the fundamental aspects of the theory that (to the knowledge of the authors) has not been exploited in previous investigations for general algorithmic depths, is the relation between the optimization coefficients and the gain from the optimization problem, which, as shown here, can be understood through a QR factorization. For this reason the analysis is restricted to $\mathbb{R}^n$ (trivially extendable to $\mathbb{C}^n$), with the norm from the optimization problem induced by an inner product. While the analysis and theory extend to more general Hilbert space settings, this allows for a clean presentation of the central ideas, and it is the most interesting for the solution of systems assembled from the discretization of nonlinear PDEs.

The presented bounds significantly sharpen those previously developed for contractive operators in two important ways. First, the dependence on the higher order terms is shown to be $\mathcal{O}\big(\|w_k\|(\|w_k\| + \|w_{k-1}\| + \ldots + \|w_{k-m}\|)\big)$, improving on the $\mathcal{O}(\|w_k\|^2) + \mathcal{O}(\|w_{k-1}\|^2) + \ldots + \mathcal{O}(\|w_{k-m}\|^2)$ bound proven in Evans *et al.* (2020), where w_k is the stage-k residual. This analysis produces the first residual bound for AA applied to nonlinear problems, where the most recent residual $\|w_k\|$ can be factored out of the entire bound; previously, the best bounds for the higher order terms involved only older (often larger) residuals from the history. Secondly, the new estimates show that if the solution to the optimization problem does not produce a linear combination of residuals that is strictly lesser in norm than the most recent residual, then there is no contribution from the higher order terms. The results of the analysis further motivate strategies for choosing the AA depth adaptively or dynamically, which is shown to provide a significant advantage over constant depths in the numerical tests.

The remainder of the paper is structured as follows. Section 2 states the algorithm and presents notation that will be used throughout, and Section 3 summarizes the residual expansion that is similar to that of Evans *et al.* (2020). In Section 4 the new one-step analysis is presented for algorithmic depth $m = 1$, and in Section 5 the one-step analysis is developed for $m > 1$. In section 5.1 practical guidance is presented on dynamic algorithmic depth selection and safeguarding strategies, as motivated by the

developed theory. In Section 6 numerical results are presented that both illustrate the theory and practical guidance, and demonstrate how AA can be effectively used to solve a nonlinear Helmholtz equation and the three-dimensional steady Navier–Stokes equations (NSEs) with Reynolds numbers past the first Hopf bifurcation. An appendix contains a proof of a technical lemma providing particular bounds on the entries of the inverse of the upper triangular matrix found in the QR decomposition.

2. Problem setting and preliminaries

Consider seeking a fixed point of Fréchet differentiable operator $g : X \to X$ for Hilbert space $X \subseteq \mathbb{R}^n$ equipped with inner product $(\cdot, \cdot)$ and induced norm $\|\cdot\|$, under the following conditions.

ASSUMPTION 2.1 Assume $g \in C^1(X)$ has a fixed point x^* in X, and there are positive constants κ_g and $\hat{\kappa}_g$ with

1. $\|g'(x)z\| \leqslant \kappa_g \|z\|$ for all $x, z \in X$.
2. $\|g'(x)z - g'(y)z\| \leqslant \hat{\kappa}_g \|x - y\| \|z\|$ for all $x, y, z \in X$.

A particular case of interest is finding a zero of a function $f : X \to X$, where the system of nonlinear equations $f(x) = 0$, comes from the discretization of a nonlinear PDE. Then $f(x) = g(x) - x$ converts between the fixed-point and zero-finding problems. Under Assumption 2.1 it holds that f has a zero $x^* \in X, f \in C^1(X)$, and

$$\|f'(x)z - f'(y)z\| = \|(g'(x) - I)z - (g'(y) - I)z\| \leqslant \hat{\kappa}_g \|x - y\| \|z\|, \text{ for all } x, y, z \in X. \tag{2.1}$$

The AA algorithm with depth m applied to the fixed-point problem $g(x) = x$ reads as follows.

ALGORITHM 2.2 (Anderson iteration). The AA algorithm with depth $m \geq 0$ and damping factors $0 < \beta_k \leqslant 1$ reads as follows:

Step 0: Choose $x_0 \in X$.
Step 1: Find $w_1 \in X$ such that $w_1 = g(x_0) - x_0$. Set $x_1 = x_0 + w_1$.
Step $k+1$: For $k = 1, 2, 3, \ldots$ Set $m_k = \min\{k, m\}$.
[a.] Find $w_{k+1} = g(x_k) - x_k$.
[b.] Solve the minimization problem for $\{\alpha_j^{k+1}\}_{k-m_k}^{k}$

$$\min_{\sum_{j=k-m_k}^{k} \alpha_j^{k+1} = 1} \left\| \sum_{j=k-m_k}^{k} \alpha_j^{k+1} w_{j+1} \right\|. \tag{2.2}$$

[c.] For damping factor $0 < \beta_k \leqslant 1$, set

$$x_{k+1} = \sum_{j=k-m_k}^{k} \alpha_j^{k+1} x_j + \beta_k \sum_{j=k-m_k}^{k} \alpha_j^{k+1} w_{j+1}. \tag{2.3}$$

Throughout the remainder the stage-k differences between iterates and terms are defined as

$$e_k \coloneqq x_k - x_{k-1}, \quad w_k \coloneqq g(x_{k-1}) - x_{k-1}. \tag{2.4}$$

The next assumption allows a key generalization from the previous convergence analysis frameworks by Toth & Kelley (2015); Kelley (2018); Pollock *et al.* (2019); Evans *et al.* (2020), which are specific to contractive fixed-point operators. As discussed below in Remark 2.1, it is automatically satisfied at each iteration for contractive fixed-point operators, and may be locally satisfied for finding zeros of nondegenerate functions.

ASSUMPTION 2.3 The stage-j iterates and residuals satisfy the relationship

$$\left\|w_{j+1} - w_j\right\| \geqslant \sigma \left\|e_j\right\|. \tag{2.5}$$

REMARK 2.1 Assumption 2.3 is reasonable to require as it is satisfied (not necessarily exhaustively) under the two following important settings.

1. If g is a contractive operator, then its Lipschitz constant given by Assumption 2.1 satisfies $\kappa_g < 1$, and by the triangle inequality $\left\|w_{j+1} - w_j\right\| \geqslant \left\|x_j - x_{j-1}\right\| - \left\|g(x_j) - g(x_{j-1})\right\| \geqslant (1 - \kappa_g)\left\|e_j\right\|$. Then (2.5) is always satisfied with $\sigma = (1 - \kappa_g)$.
2. In terms of seeking a zero of a function f as the fixed point of $g(x) = f(x) + x$, the nonlinear residual is $w_{j+1} = g(x_j) - x_j = f(x_j)$. Assumption 2.3 is then satisfied locally if the smallest singular value of the Jacobian f' is uniformly bounded away from zero on X, and $\left\|x_j - x_{j-1}\right\|$ is small enough. Specifically, if for each $x, y \in X$ it holds that $\left\|f'(x)y\right\| \geqslant \sigma_f \left\|y\right\|$, for some $\sigma_f > 0$. This is similar to the usual assumption for Newton iterations that the Jacobian is nondegenerate at a solution, and could be localized to the vicinity of a solution without undue complication. Then, under Assumption 2.1, and in accordance with (2.1), it holds that

$$\begin{aligned}
\|f(x) - f(y)\| &= \left\| f'(y)(x-y) + \int_0^1 (f'(y + t(x-y)) - f'(y))(x-y)\,\mathrm{d}t \right\| \\
&\geqslant \sigma_f \|x - y\| - \frac{\kappa_g}{2}\|x - y\|^2 \\
&\geqslant \frac{\sigma_f}{2}\|x - y\|, \text{ for } \|x - y\| \leqslant \frac{\sigma_f}{\hat{\kappa}_g}.
\end{aligned}$$

Then for $\left\|e_j\right\| \leqslant \sigma_f/\hat{\kappa}_g$ it holds that $\left\|w_{j+1} - w_j\right\| \geqslant \frac{\sigma_f}{2}\left\|e_j\right\|$, which satisfies (2.5) with $\sigma = \sigma_f/2$.

Define the following averages given by the solution $\alpha^{k+1} = \{\alpha_j^{k+1}\}_{j=k-m_k}^{k}$ to the optimization problem (2.2) by

$$x_k^{\alpha} = \sum_{j=k-m_k}^{k} \alpha_j^{k+1} x_j, \quad w_{k+1}^{\alpha} = \sum_{j=k-m_k}^{k} \alpha_j^{k+1} w_{j+1}. \tag{2.6}$$

Then the update (2.3) can be written in terms of the averages x_k^α and w_{k+1}^α, by

$$x_{k+1} = x_k^\alpha + \beta_k w_{k+1}^\alpha. \tag{2.7}$$

The stage-k gain θ_k that quantifies the success of the optimization problem is defined by

$$\left\| w_k^\alpha \right\| = \theta_k \left\| w_k \right\|. \tag{2.8}$$

This important quantity is shown in Evans *et al.* (2020) to scale the first-order term in the residual expansion (also shown below). Up to that scaling this term is the residual in the standard fixed-point iteration. The higher-order terms on the other hand are shown below to be scaled by a factor of $\sqrt{1-\theta_k^2}$, meaning a successful optimization increases the relative weight of the higher-order terms, and an unsuccessful optimization increases the relative weight of the first-order term in the residual expansion.

The constrained optimization problem (2.2) is often useful for analysis of the method (see, e.g., Toth & Kelley, 2015; Kelley, 2018; Pollock *et al.*, 2019; Evans *et al.*, 2020). In the current view, however, the following unconstrained form of the optimization problem (2.2) that is more easily implemented in practice is also more convenient for the analysis.

Define the matrices E_k and F_k formed by the respective differences between consecutive iterates and residuals by

$$\begin{aligned} E_k &\coloneqq \begin{pmatrix} e_k & e_{k-1} & \cdots & e_{k-m_k+1} \end{pmatrix}, \\ F_k &\coloneqq \begin{pmatrix} (w_{k+1} - w_k) & (w_k - w_{k-1}) & \ldots & (w_{k-m_k+2} - w_{k-m_k+1}) \end{pmatrix}. \end{aligned} \tag{2.9}$$

Then (2.2) is equivalent to the unconstrained minimization problem

$$\gamma^{k+1} = \operatorname{argmin}_{\gamma \in \mathbb{R}^m} \left\| w_{k+1} - F_k \gamma \right\|, \text{ for } \gamma^{k+1} = \left(\gamma_k^{k+1}, \gamma_{k-1}^{k+1}, \ldots, \gamma_{k-m_k+1}^{k+1} \right)^\top. \tag{2.10}$$

The averages x_k^α and w_{k+1}^α used in the update (2.7) and the transformation between the two sets of optimization coefficients are related by

$$x_k^\alpha = x_k - E_k \gamma^{k+1}, \quad w_{k+1}^\alpha = w_{k+1} - F_k \gamma^{k+1}, \quad \gamma_j^{k+1} = \sum_{n=k-m_k}^{j-1} \alpha_n^{k+1}. \tag{2.11}$$

This form of the optimization problem is instrumental in the analysis by Evans *et al.* (2020), and its direct use in the practical implementation of Algorithm 2.2 is carefully discussed in Fang & Saad (2009); Walker & Ni (2011).

As commonly understood, the algorithm in its most general form does not identify the norm that should be used in the optimization. The minimization problem is usually taken in the l_2 (or weighted l_2) sense, whereby the least-squares problem can be solved efficiently by a (fast) QR method (see Toth & Kelley, 2015, for a discussion on minimizing in l_1 or l_∞). Throughout the rest of this manuscript the optimization problem (2.10) is considered the norm $\|\cdot\|$ induced by inner product $(\cdot, \cdot)$, which then falls under the least-squares setting. For example in Pollock *et al.* (2019), the optimization is done in the

H_0^1, sense as the nonlinear operator there is contractive in H_0^1; this is interpreted (and implemented) as a least-squares optimization of a (discrete) gradient.

The QR decomposition of F_k will be explicitly used in the analysis to extract relations between the optimization gain θ_k and optimization coefficients γ^k. A key repercussion of this approach is that assumptions on the boundedness of the optimization coefficients as used in Toth & Kelley (2015); Kelley (2018); Pollock *et al.* (2019); Evans *et al.* (2020) and for $m > 1$, are replaced by assumptions on the sufficient linear independence between columns of F_k. As discussed in Subsection 5.1, satisfaction of these assumptions can be easily verified and even enforced during the course of a numerical simulation.

3. Expansion of the residual

This section is summarized from Evans *et al.* (2020) and included here, both to make the paper more self-contained and to introduce a consistent notation. The novelty in the current paper is how the differences between consecutive iterates are bounded in terms of the nonlinear residuals under more general assumptions than contractiveness of the underlying fixed-point operator; and, without explicit assumptions on the boundedness of the optimization coefficients. The results of Sections 4 and 5 are applied to the residual expansion of this section to obtain the main results.

Starting with the definition of the residual by (2.4) and expanding the iterate x_k by the update (2.7), the nonlinear residual w_{k+1} can be written as

$$w_{k+1} = g(x_k) - x_k = (g(x_k) - x_{k-1}^{\alpha}) - \beta_{k-1} w_k^{\alpha}. \tag{3.1}$$

The first term on the right-hand side of (3.1) can be expanded by (2.6). Consistent with (2.11), the optimization coefficients α_j^k are collected into the coefficients γ_j^k by $\gamma_j^k := \sum_{n=k-m_{k-1}-1}^{j-1} \alpha_n^k$. Then

$$\begin{aligned} g(x_k) - x_{k-1}^{\alpha} &= \sum_{j=k-m_{k-1}-1}^{k-1} \alpha_j^k (g(x_k) - x_j) \\ &= \sum_{j=k-m_{k-1}-1}^{k-1} \alpha_j^k (g(x_j) - x_j) + \sum_{j=k-m_{k-1}}^{k} \left(\sum_{n=k-m_{k-1}-1}^{j-1} \alpha_n^k \right) (g(x_j) - g(x_{j-1})) \\ &= w_k^{\alpha} + \sum_{j=k-m_{k-1}}^{k} \gamma_j^k (g(x_j) - g(x_{j-1})). \end{aligned} \tag{3.2}$$

This equality shows the approximation to the fixed-point $g(x_k)$ is decomposed into the average of the previous iterates x_{k-1}^{α}, the average over previous updates w_k^{α} corresponding to the optimization problem from the last step and a weighted sum over the differences of consecutive approximations. Due to Assumption 2.1 each term $g(x_j) - g(x_{j-1})$ has a Taylor expansion $g(x_j) - g(x_{j-1}) = \int_0^1 g'(z_j(t)) e_j \, \mathrm{d}t$, where $z_j(t) = x_{j-1} + t e_j$. Rewriting (3.1) with (3.2) with this expansion yields

$$w_{k+1} = (1 - \beta_{k-1}) w_k^{\alpha} + \sum_{j=k-m_{k-1}}^{k} \gamma_j^k \int_0^1 g'(z_j(t)) e_j \, \mathrm{d}t. \tag{3.3}$$

Adding and subtracting consecutive averages each summand of the last term of (3.3) can be written as

$$\int_0^1 g'(z_j(t))e_j\,\mathrm{d}t = \int_0^1 g'(z_k(t))e_j\,\mathrm{d}t + \sum_{n=j}^{k-1}\int_0^1 g'(z_n(t))e_j - g'(z_{n+1}(t))e_j\,\mathrm{d}t. \tag{3.4}$$

Summing over the j's, the sum on the right-hand side of (3.3) may be expressed as

$$\begin{aligned}\sum_{j=k-m_{k-1}}^{k}\gamma_j^k\int_0^1 g'(z_j(t))e_j\,\mathrm{d}t = \int_0^1 g'(z_k(t))\sum_{j=k-m_{k-1}}^{k}\gamma_j^k e_j\,\mathrm{d}t \\ + \sum_{j=k-m_{k-1}}^{k-1}\sum_{n=j}^{k-1}\gamma_j^k\int_0^1 g'(z_n(t))e_j - g'(z_{n+1}(t))e_j\,\mathrm{d}t.\end{aligned} \tag{3.5}$$

From $\sum_{j=k-m_{k-1}}^{k}\gamma_j^k e_j = x_k - x_{k-1}^{\alpha}$ (see Evans *et al.*, 2020, Section 2 for details) and (2.7), it holds that

$$\sum_{j=k-m_{k-1}}^{k}\gamma_j^k e_j = x_k - x_{k-1}^{\alpha} = \beta_{k-1}w_k^{\alpha}. \tag{3.6}$$

Putting (3.6) together with (3.5) and (3.3) then yields

$$w_{k+1} = \int_0^1 (1-\beta_{k-1})w_k^{\alpha} + \beta_{k-1}g'(z_k(t))w_k^{\alpha}\,\mathrm{d}t + \sum_{j=k-m_{k-1}}^{k-1}\sum_{n=j}^{k-1}\int_0^1 \big(g'(z_n(t)) - g'(z_{n+1}(t))\big)e_j\gamma_j^k\,\mathrm{d}t. \tag{3.7}$$

Notice that the summands on the right of (3.7) are all zero if g is a linear operator, as g' is then constant. The terms summed over are next bounded using Assumption 2.1. It is worth noting here that for linear operators this will result in zero contribution from higher-order terms, whereas for nonlinear operators the higher-order terms are scaled by $\hat{\kappa}_g > 0$, the Lipschitz constant of g'. Intuitively, this expansion leads to a local result, as when the difference between iterates is sufficiently small, the graph of a function satisfying Assumption 2.1 at or between those iterates is nearly linear.

Taking norms in (3.7) and applying Assumption 2.1, then triangle inequalities applied to the terms of the final sum produces the expansion of $\|w_{k+1}\|$ in terms of $\|w_k^{\alpha}\|$ and $\|e_j\|, j = k-m_k,\ldots,k$, by

$$\begin{aligned}\|w_{k+1}\| &\leqslant \left((1-\beta_{k-1}) + \kappa_g\beta_{k-1}\right)\|w_k^{\alpha}\| + \frac{\hat{\kappa}_g}{2}\sum_{j=k-m_{k-1}}^{k-1}\|e_j\gamma_j^k\|\sum_{n=j}^{k-1}\big(\|e_{n+1}\| + \|e_n\|\big) \\ &= \left((1-\beta_{k-1}) + \kappa_g\beta_{k-1}\right)\|w_k^{\alpha}\| + \frac{\hat{\kappa}_g}{2}\sum_{n=k-m_{k-1}}^{k-1}\big(\|e_{n+1}\| + \|e_n\|\big)\sum_{j=n}^{k-1}\|e_j\gamma_j^k\|,\end{aligned} \tag{3.8}$$

where the last equality follows from reindexing the sums. The next step is to bound the $\|e_j\|$ terms by $\|w_j\|$ terms. Here the analysis departs from that in Evans *et al.* (2020). This will be shown first in the simpler case of depth $m = 1$ in Section 4 and then extended to more general depths $m > 1$ in Section 5.

4. Acceleration for depth $m = 1$

For depth $m = 1$ the matrix F_k has only one column, which removes several technicalities from the analysis. It is useful to use this case to overview the general framework and to introduce the extension to a noncontractive setting.

LEMMA 4.1 Let Assumption 2.1 hold, and let $m = 1$ in Algorithm 2.2. Assume there is a constant $\sigma > 0$ for which the residuals on stages $j + 1$ and j satisfy Assumption 2.3. Then the following bound holds on the difference between consecutive accelerated iterates:

$$\|e_{j+1}\| \leqslant \|w_{j+1}\| \left(\sigma^{-1}\sqrt{1-\theta_{j+1}^2} + \beta_j\theta_{j+1}\right). \tag{4.1}$$

Proof. The update (2.7) for the case $m = 1$ is

$$x_{j+1} = (1-\gamma_j^{j+1})x_j + \gamma_j^{j+1}(x_{j-1}) + \beta_j w_{j+1}^{\alpha}, \tag{4.2}$$

where consistent with (2.11), $\gamma_j^{j+1} = \alpha_{j-1}^{j+1}$. Taking norms and applying (2.8) allow

$$\|e_{j+1}\| = \|x_{j+1} - x_j\| \leqslant |\gamma_j^{j+1}|\|e_j\| + \beta_j\theta_{j+1}\|w_{j+1}\|. \tag{4.3}$$

Inequality (4.3) will be used to trade terms of the form $\|e_{j+1}\|$ for expressions in terms of $\|w_{j+1}\|$. The argument follows by relating the optimization coefficient γ_j^{j+1} to the optimization gain θ_{j+1}.

For $m = 1$ the coefficient γ_j^{j+1} can be explicitly written as

$$\gamma_j^{j+1} = \frac{(w_{j+1}, w_{j+1} - w_j)}{\|w_{j+1} - w_j\|^2}. \tag{4.4}$$

In particular, this determines the decomposition of w_{j+1} into $w_R = \gamma_j^{j+1}(w_{j+1} - w_j)$, in the range of $(w_{j+1} - w_j)$ and $w_N = w_{j+1}^{\alpha} = w_{j+1} - \gamma_j^{j+1}(w_{j+1} - w_j)$, in the nullspace of $(w_{j+1} - w_j)^{\top}$. By the orthogonality of w_R and w_N it follows that

$$\|w_{j+1}\|^2 = \|w_R\|^2 + \|w_N\|^2 = \|\gamma_j^{j+1}(w_{j+1} - w_j)\|^2 + \|w_{j+1}^{\alpha}\|^2 = (\gamma_j^{j+1})^2\|w_{j+1} - w_j\|^2 + \theta_{j+1}^2\|w_{j+1}\|^2, \tag{4.5}$$

by which

$$|\gamma_j^{j+1}| = \sqrt{1-\theta_{j+1}^2}\,\frac{\|w_{j+1}\|}{\|w_{j+1} - w_j\|} \text{ and } \theta_{j+1} = \sqrt{1 - \frac{(w_{j+1}, w_{j+1} - w_j)^2}{\|w_{j+1}\|^2\|w_{j+1-w_j}\|^2}}, \tag{4.6}$$

where the expression for θ_{j+1} in (4.6) can be recognized as the (absolute value of the) direction sine between w_{j+1} and $w_{j+1} - w_j$. Applying the expression for γ_j^{j+1} in (4.6) to (4.3) yields

$$\|e_{j+1}\| \leqslant \sqrt{1-\theta_{j+1}^2}\,\frac{\|w_{j+1}\|}{\|w_{j+1}-w_j\|}\|e_j\| + \beta_j\theta_{j+1}\|w_{j+1}\|. \tag{4.7}$$

Applying now they key inequality (2.5) to (4.7) yields

$$\|e_{j+1}\| \leqslant \sigma^{-1}\sqrt{1-\theta_{j+1}^2}\|w_{j+1}\| + \beta_j\theta_{j+1}\|w_{j+1}\|, \tag{4.8}$$

establishing the result (4.1) □

REMARK 4.1 In the second case of Remark 2.1 where f' is nondegenerate, the results of Lemma 4.1 and $0 < \beta_j \leq 1$ show

$$\|e_{j+1}\| \leqslant \left(\frac{2}{\sigma_f}\sqrt{1-\theta_{j+1}^2} + \theta_{j+1}\right)\|w_{j+1}\| \leqslant \sqrt{1+4/\sigma_f^2}\|w_{j+1}\|,$$

where the last bound was obtained by maximizing the previous expression with respect to θ_{j+1}. Setting this expression no greater than $\sigma_f/\hat{\kappa}_g$, it follows that $\|w_{j+1}\| \leqslant \sigma_f^2/\left(\hat{\kappa}_g\sqrt{\sigma_f^2+4}\right)$ is sufficient to ensure $\|e_{j+1}\| \leqslant \sigma_f/\hat{\kappa}_g$, which implies satisfaction of Assumption 2.3 on the subsequent iteration.

Relation (4.8) is now used in the expansion of the residual (3.8) to bound $\|w_{k+1}\|$.

THEOREM 4.1 Suppose the hypotheses of Lemma 4.1 for $j = k-1$ and $j = k-2$. Then the following bound holds for the nonlinear residual $\|w_{k+1}\|$ generated by Algorithm 2.2 with depth $m = 1$.

$$\begin{aligned}\|w_{k+1}\| \leqslant \|w_k\| \Big\{ &\theta_k\big((1-\beta_{k-1}) + \kappa_g\beta_{k-1}\big) + \hat{\kappa}_g\sigma^{-1}\sqrt{1-\theta_k^2} \\ &\times \left(\|w_k\|\left(\sigma^{-1}\sqrt{1-\theta_k^2} + \beta_{k-1}\theta_k\right) + \|w_{k-1}\|\left(\sigma^{-1}\sqrt{1-\theta_{k-1}^2} + \beta_{k-2}\theta_{k-1}\right)\right)\Big\}. \end{aligned} \tag{4.9}$$

REMARK 4.2 Since $\hat{\kappa}_g$ represents the Lipschitz constant of g' if g is linear, then $\hat{\kappa}_g = 0$, and thus all of the higher order terms on the right side of (4.9) will vanish.

This result shows, not only how the first-order term is scaled by the optimization gain θ_k, but also that the higher order terms are scaled by $\sqrt{1-\theta_k^2}$. This explicitly establishes that if $\theta_k = 1$, then the higher order terms do not contribute to the total residual and the bound for the fixed-point iteration is recovered. This holds as well for the case $m > 1$, shown in the next section.

Proof. Expanding the residual by (3.8) yields for depth $m = 1$

$$\|w_{k+1}\| \leqslant \theta_k\big((1-\beta_{k-1}) + \kappa_g\beta_{k-1}\big)\|w_k\| + \hat{\kappa}_g\left(\|e_k\| + \|e_{k-1}\|\right)|\gamma_{k-1}^k|\,\|e_{k-1}\|,$$

where consistent with (2.11), $\gamma_{k-1}^k = \alpha_{k-2}^k$.

Applying (4.8) with $j = k-1$ and $j = k-2$ allows

$$\begin{aligned}\|w_{k+1}\| \leqslant \theta_k\big((1-\beta_{k-1}) + \kappa_g\beta_{k-1}\big)\|w_k\| + \hat{\kappa}_g\Bigg(\|w_k\|\left(\sigma^{-1}\sqrt{1-\theta_k^2} + \beta_{k-1}\theta_k\right)\\ + \|w_{k-1}\|\left(\sigma^{-1}\sqrt{1-\theta_{k-1}^2} + \beta_{k-2}\theta_{k-1}\right)\Bigg)|\gamma_{k-1}^k|\,\|e_{k-1}\|. \end{aligned} \tag{4.10}$$

Combining relation (4.6) with hypothesis (2.5) yields $|\gamma_{k-1}^k|\,\|e_{k-1}\| \leqslant \sigma^{-1}\sqrt{1-\theta_k^2}\,\|w_k\|$, by which

$$\begin{aligned}\|w_{k+1}\| \leqslant \theta_k\big((1-\beta_{k-1}) + \kappa_g\beta_{k-1}\big)\|w_k\| + \hat{\kappa}_g\Bigg(\|w_k\|\left(\sigma^{-1}\sqrt{1-\theta_k^2} + \beta_{k-1}\theta_k\right)\\ + \|w_{k-1}\|\left(\sigma^{-1}\sqrt{1-\theta_{k-1}^2} + \beta_{k-2}\theta_{k-1}\right)\Bigg)\sigma^{-1}\sqrt{1-\theta_k^2}\,\|w_k\|, \end{aligned} \tag{4.11}$$

establishing the result (4.9). □

The bound (4.9) shows for θ_k small the higher-order terms have a greater contribution, whereas for θ_k close to unity (the optimization did little) the residual is dominated by the first order term; and, $\hat{\kappa}_g$, the Lipschitz constant of g', has less influence on the residual.

In light of Remark 2.1, the two presented conditions under which the hypothesis (2.5) must hold are now discussed. First, if g is contractive on X, then (2.5) continues to hold on subsequent iterates without further conditions. Moreover, in that case it makes sense to run the iteration without damping ($\beta_j = 1$ for all j) and (4.11) reduces to

$$\|w_{k+1}\| \leqslant \|w_k\|\left\{\theta_k\kappa_g + \frac{\hat{\kappa}_g\sqrt{1-\theta_k^2}}{(1-\kappa_g)^2}\left(\|w_k\|\left(\sqrt{1-\theta_k^2} + \theta_k\right) + \|w_{k-1}\|\left(\sqrt{1-\theta_{k-1}^2} + \theta_{k-1}\right)\right)\right\}.$$

If instead, $\|f'(y)(x-y)\| \geqslant \sigma_f\|x-y\|$ for all $x, y \in X$, then at the next iteration $\|w_{k+1} - w_k\| \geqslant (\sigma_f/2)\|e_k\|$ continues to hold if $\|e_{k+1}\| \leqslant \|e_k\|$, which is guaranteed upon sufficient decrease of the sequence of residuals $\{\|w_k\|\}$. This explains the observation (demonstrated by the steady examples by Lott *et al.*, 2012, for instance) that Anderson accelerated noncontractive iterations can show rapid convergence. However, this does not guarantee convergence without some ability to enforce an inequality such as $\theta_k((1-\beta_{k-1}) + \kappa_g\beta_{k-1}) < 1-\varepsilon$, with sufficient frequency. As θ_k sufficiently less than one is essential to the success of the algorithm, this encourages the consideration of the theory for $m > 1$ in the next sections, as smaller gain factors can be obtained (to some extent) with greater algorithmic depth.

Finally, a corollary to (4.1) shows a simplified residual bound for contractive operators together with a condition for monotonic decrease of the residual. This result features tighter bounds on the higher order terms than in Evans *et al.* (2020), and without assumptions on the boundedness of the optimization coefficients.

COROLLARY 4.1 Suppose the hypotheses of Lemma 4.1 for $j = k-1$ and $j = k-2$, and the Lipschitz constant of g satisfies $\kappa_g < 1$. Then the following bound holds on the nonlinear residual $\|w_{k+1}\|$

generated by Algorithm 2.2 with $m = 1$ and $\beta_k = \beta = 1$:

$$\|w_{k+1}\| \leqslant \|w_k\| \left\{ \theta_k \kappa_g + \frac{\sqrt{2}\hat{\kappa}_g}{(1-\kappa_g)^2} \sqrt{1-\theta_k^2} \Big(\|w_k\| + \|w_{k-1}\| \Big) \right\}. \tag{4.12}$$

After the first two consecutive iterations $j = k-1, k$ where the following inequality is satisfied

$$\|w_j\| + \|w_{j-1}\| < \frac{\sqrt{1-\kappa_g^2}(1-\kappa_g)^2}{\sqrt{2}\hat{\kappa}_g}, \tag{4.13}$$

monotonic decrease of the residual is ensured.

Proof. From (4.9) with $\beta_k = 1$ and $\sigma = (1-\kappa_g)$, the residual $\|w_{k+1}\|$ satisfies

$$\|w_{k+1}\| \leqslant \|w_k\| \left\{ \theta_k \kappa_g + \frac{\hat{\kappa}_g \sqrt{1-\theta_k^2}}{(1-\kappa_g)^2} \Big(\|w_k\| \left(\sqrt{1-\theta_k^2} + \theta_k \right) + \|w_{k-1}\| \left(\sqrt{1-\theta_{k-1}^2} + \theta_{k-1} \right) \Big) \right\}. \tag{4.14}$$

The maximum of $\sqrt{1-\theta^2} + \theta$ on $0 \leqslant \theta \leqslant 1$ is $\sqrt{2}$, attained at $\theta = 1/\sqrt{2}$. Applying this to θ_{k-1}, θ_k within the higher order terms yields (4.12).

Following the same idea maximizing the bracketed term on the right-hand side of (4.12) over θ_k

$$\theta_k \kappa_g + \frac{\sqrt{2}\hat{\kappa}_g}{(1-\kappa_g)^2} \sqrt{1-\theta_k^2} \Big(\|w_k\| + \|w_{k-1}\| \Big) \leqslant \sqrt{\kappa_g^2 + \frac{2\hat{\kappa}_g^2}{(1-\kappa_g)^4} \big(\|w_k\| + \|w_{k-1}\| \big)^2}.$$

Setting (the square of) the right-hand side expression less than one, it follows that $\|w_{k+1}\| < \|w_k\|$ under condition (4.13). If this condition is satisfied for two consecutive iterates, then $\|w_{k+1}\| < \|w_k\|$ and $\|w_k\| < \|w_{k-1}\|$, which is sufficient to ensure monotonic decrease of the sequence. □

This corollary quantifies (in the contractive setting) the transition from the preasymptotic regime where the residuals may be large, to the asymptotic regime where the residuals are small enough that the higher order terms 'don't count', and previous convergence results such as those in Pollock *et al.* (2019) hold (see also Toth & Kelley, 2015; Kelley, 2018, for a different, but related approach). This will be generalized in Corollary 5.1 for algorithmic depths $m > 1$, where it will be sufficient for a similar condition to hold for $m+1$ consecutive iterates. However, the monotonicity result holds only for contractive operators.

5. Acceleration for depth $m > 1$

The analysis for $m > 1$ is somewhat more complicated than for $m = 1$, if only because in the optimization problem for $m = 1$, the matrix F_k has only one column. For $m > 1$ the columns of F_k are in general not orthogonal, and the estimates that follow show how detrimental this lack of orthogonality can be to the convergence rate. First, some standard results from numerical linear algebra are recalled. Then, Theorem 4.1 is generalized to $m > 1$.

PROPOSITION 5.1 Let R_j be a $j \times j$ upper triangular matrix given by

$$R_j = \begin{pmatrix} R_{j-1} & b_j \\ 0 & r_{jj} \end{pmatrix},$$

where R_{j-1} is an invertible $j-1 \times j-1$ upper triangular matrix, b_j is a $j-1 \times 1$ vector of values and $r_{jj} \neq 0$. Then R_j is invertible and the inverse matrix satisfies

$$R_j^{-1} = \begin{pmatrix} R_{j-1}^{-1} & c_j \\ 0 & r_{jj}^{-1} \end{pmatrix}.$$

The next two results are specific to the economy (or thin) QR decomposition of $n \times m$ matrix A (see, for instance Golub & Van Loan, 1996, Chapter 5). The following notation will be used throughout the remainder of this section. For $u, v \in \mathbb{R}^n$, let $\cos(u, v) = (u, v)/(\|u\| \, \|v\|)$ be the usual direction cosine between vectors u and v, with the corresponding direction sine satisfying $\sin^2(u, v) = 1 - \cos^2(u, v)$. Let $\mathcal{A}_j$ be the subspace of $\mathbb{R}^n$ given by $\mathcal{A}_j = \text{span}\{a_1, \ldots, a_j\}$, with orthogonal basis $\{q_1, \ldots, q_j\}$; let $\sin^2(u, \mathcal{A}_j) = 1 - \sum_{i=1}^{j} \cos^2(u, q_i)$, denote the square of the direction sine between vector u and $\mathcal{A}_j$.

PROPOSITION 5.2 Let $\hat{Q}\hat{R}$ be the economy QR decomposition of a matrix $A \in \mathbb{R}^{n \times m}$, $n \geqslant m$, where A has columns $a_1, \ldots a_m$ and $\hat{Q}$ has orthonormal columns $q_1, \ldots q_m$. Then

$$r_{jj}^2 = \|a_j\|^2 \sin^2(a_j, \mathcal{A}_{j-1}), \; j = 1, \ldots, m. \tag{5.1}$$

The proof is standard and follows from writing the columns of $\hat{Q}$ as $q_j = v_j/\|v_j\|$ with $v_j = a_j - \sum_{i=1}^{j-1}(q_i, a_j)q_i$. Then $r_{jj}^2 = {}^2 = \|a_j\|^2 - \sum_{i=1}^{j-1}(q_i, a_j)^2$ from orthogonality. Factoring out $\|a_j\|^2$ from each term yields the result.

The next technical lemma gives a bound on the elements of $\hat{R}^{-1}$; it is proven here (in the appendix) to make the manuscript more self-contained.

LEMMA 5.1 Let $\hat{Q}\hat{R}$ be the economy QR decomposition of matrix $A \in \mathbb{R}^{n \times m}$, $n \geqslant m$, where A has columns $a_1, \ldots a_m$, $\hat{Q}$ has orthonormal columns $q_1, \ldots q_m$ and $\hat{R} = (r_{ij})$ is an invertible upper-triangular $m \times m$ matrix. Let $\hat{R}^{-1} = (s_{ij})$.

Suppose there is a constant $0 < c_s \leqslant 1$ such that $|\sin(a_j, \mathcal{A}_{j-1})| \geqslant tc_s$, $j = 2, \ldots, m$, which implies another constant $0 \leqslant tc_t < 1$ with $|\cos(a_j, q_i)| \leq c_t$, $j = 2, \ldots, m$ and $i = 1, \ldots, j-1$. Then it holds that

$$s_{11} = \frac{1}{\|a_1\|}, \qquad s_{ii} \leqslant \frac{1}{\|a_i\| \, c_s}, \; i = 2, \ldots, m, \tag{5.2}$$

$$|s_{1j}| \leqslant \frac{c_t(c_t + c_s)^{j-2}}{\|a_1\| \, c_s^{j-1}} \text{ and} \qquad |s_{ij}| \leqslant \frac{c_t(c_t + c_s)^{j-i-1}}{\|a_i\| \, c_s^{j-i+1}}, \text{ for} \tag{5.3}$$

$i = 2, \ldots, m-1$ and $j = i+1, \ldots, m$.

The constant $c_s > 0$ ensures the full rank of A and essentially bounds $\hat{Q}$ away from degeneracy, assuring sufficient linear independence of its columns. While the results are simpler in form if the second constant is taken as $c_t = 1$, the condition $c_s > 0$ implies $c_t < 1$. By taking this second constant into account, the results reflect that if the columns of A are close to (or actually) orthogonal, then c_t and the off-diagonal elements are close to (or actually) zero.

The next lemma generalizes Lemma 4.1 to $m > 1$. The technical difficulty of the more complicated relationship between the optimization coefficients and optimization gain is handled by expressing both in terms of a QR decomposition and then making use of Lemma 5.1.

LEMMA 5.2 Let Assumption 2.1 hold. Let $v_{n+1} = w_{n+1} - w_n$ and let Assumption 2.3 hold with constant σ for $n = j - m, \ldots, j$. Further, assume there is a constant $c_s > 0$ such that

$$\left|\sin\left(v_{n+1}, \operatorname{span}\{v_{n+2}, \ldots v_{j+1}\}\right)\right| \geqslant c_s,\ n = j - m + 1, \ldots, j - 1, \tag{5.4}$$

which implies there is a constant $0 \leqslant c_t < 1$ that satisfies

$$\left|\cos\left(v_{n+1}, v_p\}\right)\right| \leqslant c_t,\ n = j - m + 1, \ldots, j - 1 \text{ and } p = n + 2, \ldots, j + 1.$$

Then the following bound holds for the difference between consecutive iterates $e_{j+1} = x_{j+1} - x_j$:

$$\left\|e_{j+1}\right\| \leqslant \left\|w_{j+1}\right\| \left(C_{F,j+1}\sqrt{1 - \theta_{j+1}^2} + \beta_j\theta_{j+1}\right), \tag{5.5}$$

where the constant $C_{F,j+1}$ is given by

$$C_{F,j+1} := \sigma^{-1}\left(1 + \frac{(1 + c_t)\sum_{l=1}^{m_j-1}\binom{m_j-1}{l}c_t^{l-1}c_s^{m_j-l-1}}{c_s^{m_j-1}}\right). \tag{5.6}$$

Additionally, the following bounds hold for terms of the form $\left\|e_n\gamma_n^{j+1}\right\|$.

$$\left\|e_n\gamma_n^{j+1}\right\| \leqslant C_{n,j+1}\beta_j\sqrt{1 - \theta_{j+1}^2}\left\|w_{j+1}\right\|,\ n = j - m_j, \ldots, j, \tag{5.7}$$

where the constants $C_{n,j+1}$ are given by

$$C_{n,j+1} := \begin{cases} \frac{1}{\sigma}\left(\frac{c_t+c_s}{c_s}\right)^{m_j-1}, & n = j \\ \frac{1}{\sigma c_s}\left(\frac{c_t+c_s}{c_s}\right)^{m_j-(j-n+1)}, & n = j - m_j, \ldots, j - 1. \end{cases} \tag{5.8}$$

The additional assumption of (5.4) not found in the $m = 1$ case requires that the columns of the matrix used in the least squares problem (2.10), $v_{j+1}, \ldots, v_{j-m+2}$, maintain sufficient linear independence. See Subsection 5.1 on ensuring this assumption holds during a simulation.

Proof. Throughout this proof depth m_j will be denoted by m, for simplicity. Starting with the update for x_{j+1} from (2.7) and (2.11), defined for optimization coefficients γ^{j+1} from (2.10), and the matrix E_j given by (2.9) shows $x_{j+1} - x_j = -E_j\gamma^{j+1} + \beta_k w_{j+1}^{\alpha}$. Taking norms and applying (2.8) yield

$$\|e_{j+1}\| \leqslant \|E_j\gamma^{j+1}\| + \beta_j\theta_{j+1}\|w_{j+1}\|. \tag{5.9}$$

By (2.10) the coefficients γ^{j+1} are the least-squares solution to $F_j\gamma^{j+1} = w_{j+1}$, where F_j is given by (2.9). Using an economy QR-decomposition provides $\hat{R}\gamma^{j+1} = \hat{Q}^{\top}w_{j+1}$, by which (5.9) may be written

$$\|e_{j+1}\| \leqslant \|E_j\hat{R}^{-1}\hat{Q}^{\top}w_{j+1}\| + \beta_j\theta_{j+1}\|w_{j+1}\|. \tag{5.10}$$

The first term on the right of (5.10) can be bounded in terms of $\|w_{j+1}\|$ by considering an explicit expression for the optimization gain θ_{j+1}, as first discussed in Evans *et al.* (2020). From (2.8) and the unique decomposition $w_{j+1} = w_R + w_N$ with $w_R \in \text{Range}\,(F_j)$ and $w_N \in \text{Null}\,((F_j)^{\top})$, the null-space component w_N is the least-squares residual satisfying $\|w_N\| = \|F_j\gamma^{j+1} - w_{j+1}\| = \|w_{j+1}^{\alpha}\| = \theta_{j+1}\|w_{j+1}\|$, meaning $\theta_{j+1} = \sqrt{1 - \|\hat{Q}^{\top}w_{j+1}\|^2/\|w_{j+1}\|^2}$, or, by rearranging

$$\|w_{j+1}\|\sqrt{1-\theta_{j+1}^2} = \|\hat{Q}^{\top}w_{j+1}\|. \tag{5.11}$$

The first term on the right-hand side of (5.10) can now be controlled by (5.11), yielding

$$\|E_j\hat{R}^{-1}\hat{Q}^{\top}w_{j+1}\| \leqslant \|E_j\hat{R}^{-1}\|\|\hat{Q}^{\top}w_{j+1}\| \leqslant \|E_j\hat{R}^{-1}\|\|w_{j+1}\|\sqrt{1-\theta_{j+1}^2}. \tag{5.12}$$

It remains to bound $\|E_j\hat{R}^{-1}\|$. Writing $\hat{R}^{-1} = S = (s_{ij})$,

$$\begin{aligned}\|E_j\hat{R}^{-1}\| &= \left\|\begin{pmatrix} e_j\sum_{n=1}^{m}s_{1n} & e_{j-1}\sum_{n=2}^{m}s_{2n} & \cdots & e_{j-m+1}s_{mm}\end{pmatrix}\right\| \\ &\leqslant \left\|e_j\sum_{n=1}^{m}s_{1n}\right\| + \left\|e_{j-1}\sum_{n=2}^{m}s_{2n}\right\| + \cdots + \|e_{j-m+1}s_{mm}\|,\end{aligned} \tag{5.13}$$

where the last inequality follows from the standard bound of the matrix 2-norm by the Frobenius norm. Apply now the results of the technical Lemma 5.1.

For the first term in the sum of vector norms in (5.13), applying (5.2)–(5.3) of Lemma 5.1 then taking the finite geometric sum produces the bound

$$\begin{aligned}\left\| e_j \sum_{n=1}^{m} s_{1n} \right\| &\leqslant \|e_j\| \left| \sum_{n=1}^{m} s_{1n} \right| \\ &\leqslant \frac{\|e_j\|}{\|w_{j+1} - w_j\|} \left(1 + \sum_{n=2}^{m} \frac{c_t (c_t + c_s)^{n-2}}{c_s^{n-1}} \right) \\ &= \frac{\|e_j\|}{\|w_{j+1} - w_j\|} \left(\frac{c_t + c_s}{c_s} \right)^{m-1} \\ &\leqslant \sigma^{-1} \left(\frac{c_t + c_s}{c_s} \right)^{m-1}, \end{aligned} \tag{5.14}$$

where the last inequality follows from the hypothesis (2.5).

Proceed similarly for the remaining vector norms of (5.13), indexed by $p = 2, \ldots, m$, noting the additional factor of c_s in the denominator, to get

$$\left\| e_{j-p+1} \sum_{n=p}^{m} s_{pn} \right\| \leqslant \frac{1}{\sigma c_s} \left(1 + \sum_{n=p+1}^{m} \frac{(c_t + c_s)^{n-(p+1)}}{c_s^{n-p}} \right) \leqslant \frac{1}{\sigma c_s} \left(\frac{c_t + c_s}{c_s} \right)^{m-p}. \tag{5.15}$$

Finally, adding the contributions from $p = 1$ to $p = 2, \ldots, m$ from (5.14) and (5.15), and applying the total to (5.13) yield, assuming $c_t \neq 0$,

$$\left\| E_j \hat{R}^{-1} \right\| \leqslant \sigma^{-1} \left(\frac{(c_t + c_s)^{m-1}(c_t + 1) - c_s^{m-1}}{c_s^{m-1} c_t} \right) = \sigma^{-1} \left(1 + \frac{(1 + c_t) \sum_{j=1}^{m-1} \binom{m-1}{j} c_t^{j-1} c_s^{m-j-1}}{c_s^{m-1}} \right). \tag{5.16}$$

If it so happens that $c_t = 0$, meaning the columns of F_k are orthogonal, then $c_s = 1$ and (5.16) is in agreement with summing the terms directly from (5.14) and (5.15) yields $\left\| E_j \hat{R}^{-1} \right\| \leqslant m/\sigma$, in agreement in (5.16). Putting (5.16) together with (5.11) yields

$$\|e_{j+1}\| \leqslant C_{F,j+1} \sqrt{1 - \theta_{j+1}^2} \|w_{j+1}\| + \beta_j \theta_{j+1} \|w_{j+1}\|,$$

with $C_{F,j+1}$ given by (5.6), hence the result (5.5).

For the second result, (5.7), expanding (5.10), shows

$$\begin{pmatrix} e_j \gamma_j^{j+1} & e_{j-1} \gamma_{j-1}^{j+1} & \cdots & e_{j-m+1} \gamma_{j-m+1}^{j+1} \end{pmatrix} = E_j \gamma^{j+1} = E_j \hat{R}^{-1} \hat{Q}^{\top} w_{j+1}. \tag{5.17}$$

Accordingly, $e_{j-p+1}\gamma_{j-p+1}^{j+1} = e_{j-p+1}s^p\hat{Q}^\top w_{j+1}$, where s^p is row p of $\hat{R}^{-1}$. Hence, following (5.14) and applying (5.11) produces for the first column of (5.17):

$$\left\|e_j\gamma_j^{j+1}\right\| \leqslant \left\|e_j\sum_{n=1}^{m}s_{1n}\right\|\left\|w_{j+1}^{\alpha}\right\| \leqslant \sigma^{-1}\left(\frac{c_t+c_s}{c_s}\right)^{m-1}\beta_j\sqrt{1-\theta_{j+1}^2}\left\|w_{j+1}\right\|.$$

For the remaining columns following now (5.15) allows

$$\left\|e_{j-p+1}\gamma_{j-p+1}^{j+1}\right\| \leqslant \left\|e_{j-p+1}\sum_{n=1}^{m}s_{pn}\right\|\left\|w_{j+1}^{\alpha}\right\| \leqslant \frac{1}{\sigma c_s}\left(\frac{c_t+c_s}{c_s}\right)^{m_j-p}\beta_j\sqrt{1-\theta_{j+1}^2}\left\|w_{j+1}\right\|,$$

which establishes the second result (5.7) with $n = j-p+1$. □

Lemma (5.2) is now used to establish one-step residual bounds for general depths m.

THEOREM 5.1 Suppose the hypotheses of Lemma 5.2 for $j = k-m,\ldots,k-1$. Then the following bound holds for the nonlinear residual $\left\|w_{k+1}\right\|$ generated by Algorithm 2.2 with depth m:

$$\begin{aligned}\left\|w_{k+1}\right\| \leqslant \left\|w_k\right\|\Bigg\{&\theta_k((1-\beta_{k-1})+\kappa_g\beta_{k-1})+\frac{\hat{\kappa}_g}{2}\Bigg(\left\|w_k\right\|h(\theta_k)h_{k-1}(\theta_k)\\&+2\sum_{n=k-m_{k-1}+1}^{k-1}\left(\left\|w_n\right\|h(\theta_n)\sum_{j=n}^{k-1}h_j(\theta_k)\right)+\left\|w_{k-m_{k-1}}\right\|h(\theta_{k-m_{k-1}})\sum_{j=k-m_{k-1}}^{k-1}h_j(\theta_k)\Bigg)\Bigg\},\end{aligned} \tag{5.18}$$

where

$$h(\theta_j) = C_{F,j}\sqrt{1-\theta_j^2}+\beta_{j-1}\theta_j, \qquad h_j(\theta_k) = C_{j,k}\beta_{k-1}\sqrt{1-\theta_k^2}, \tag{5.19}$$

and the constants $C_{F,j}$ and $C_{j,k}$ are given by (5.6) and (5.8), respectively.

REMARK 5.1 As in Remark 4.2 if g is linear then $\hat{\kappa}_g = 0$ and the higher-order terms do not appear.

REMARK 5.2 Theorem 5.1 gives three significant improvements for the higher order terms, compared to the results for general m in Evans *et al.* (2020). First, the results above show

$$\left\|w_{k+1}\right\| \leqslant \mathcal{O}\left(\left\|w_k\right\|^2\right)+\mathcal{O}\left(\left\|w_k\right\|\left\|w_{k-1}\right\|\right)+\ldots\mathcal{O}\left(\left\|w_k\right\|\left\|w_{k-m_{k-1}}\right\|\right),$$

whereas previous results show $\left\|w_{k+1}\right\| \leq \mathcal{O}\left(\left\|w_k\right\|^2\right)+\mathcal{O}\left(\left\|w_{k-1}\right\|^2\right)+\ldots\mathcal{O}\left(\left\|w_{k-m_{k-1}}\right\|^2\right)$. This helps to explain how the steady Navier–Stokes numerical test of Section 6 is able to converge with very large m.

Secondly, the theorem makes no assumptions on the boundedness of the optimization coefficients. Instead, a more practical assumption is made for how close the matrix F_k from the least-squares problem (2.10) comes to degeneracy. Thirdly, similar to the $m = 1$ case of Theorem 4.1, Theorem 5.1 shows the higher order terms do not contribute to the residual if there is no gain from the optimization problem

($\theta_k = 1$). To see this note that each $h_j(\theta_k)$ in (5.18) has $\sqrt{1-\theta_k^2}$ as a factor, so if there is no gain from the optimization problem, then all the higher order terms in (5.18) vanish.

More explicitly each $h_j(\theta_k)$ in (5.18) is bounded by $C\sqrt{1-\theta_k^2}$ for a constant C (given in (5.24), where the factor of $(1-\kappa_g)$ in the denominator can be replaced by σ for the general case). Applying these simplifications to (5.18) shows $\|w_{k+1}\|$ satisfies the bound

$$\|w_{k+1}\| \leqslant \|w_k\| \left(\theta_k((1-\beta_{k-1}) + \kappa_g\beta_{k-1}) + \frac{C\hat{\kappa}_g\sqrt{1-\theta_k^2}}{2}\left(\|w_k\| h(\theta_k) + 2\sum_{n=k-m_{k-1}+1}^{k-1} (k-n)\|w_n\| h(\theta_n) + m_{k-1}\|w_{k-m_{k-1}}\| h(\theta_{k-m_{k-1}})\right)\right).$$

The proof of Theorem 5.1 follows the same essential outline as Theorem 4.1. In contrast to the technique used in Evans *et al.* (2020), a direct rather than inductive approach will be taken, as the optimization gain (which depends on m) appears in both higher and lower order terms.

Proof. The expansion of the residual (3.8) from Section 3 shows

$$\|w_{k+1}\| \leqslant \left((1-\beta_{k-1}) + \kappa_g\beta_{k-1}\right)\|w_k^\alpha\| + \frac{\hat{\kappa}_g}{2}\sum_{n=k-m_{k-1}}^{k-1}\left(\|e_{n+1}\| + \|e_n\|\right)\sum_{j=n}^{k-1}\left\|e_j\gamma_j^k\right\|. \tag{5.20}$$

Opening up the first sum of (5.20) allows

$$\sum_{n=k-m_{k-1}}^{k-1}\left(\|e_{n+1}\| + \|e_n\|\right)\sum_{j=n}^{k-1}\left\|e_j\gamma_j^k\right\| = \left\|e_{k-m_{k-1}}\right\|\sum_{j=k-m_{k-1}}^{k-1}\left\|e_j\gamma_j^k\right\| + 2\sum_{n=k-m_{k-1}+1}^{k-1}\|e_n\|\sum_{j=n}^{k-1}\left\|e_j\gamma_j^k\right\| + \|e_k\|\left\|e_{k-1}\gamma_{k-1}^k\right\|. \tag{5.21}$$

Applying now (5.5) then (5.7), (5.21) above is bounded by

$$\begin{aligned}&\|w_k\| h(\theta_k)\left\|e_{k-1}\gamma_{k-1}^k\right\| + 2\sum_{n=k-m_{k-1}+1}^{k-1}\left(\|w_n\| h(\theta_n)\sum_{j=n}^{k-1}\left\|e_j\gamma_j^k\right\|\right) + \left\|w_{k-m_{k-1}}\right\| h(\theta_{k-m_{k-1}})\sum_{j=k-m_{k-1}}^{k-1}\left\|e_j\gamma_j^k\right\| \\ &\leqslant \|w_k\| h(\theta_k)h_{k-1}(\theta_k)\|w_k\| + 2\sum_{n=k-m_{k-1}+1}^{k-1}\left(\|w_n\| h(\theta_n)\sum_{j=n}^{k-1} h_j(\theta_k)\|w_k\|\right) \\ &+ \left\|w_{k-m_{k-1}}\right\| h(\theta_{k-m_{k-1}})\sum_{j=k-m_{k-1}}^{k-1} h_j(\theta_k)\|w_k\|. \end{aligned} \tag{5.22}$$

Putting the bound of (5.22) back into (5.21) then yields

$$\begin{aligned}\|w_{k+1}\| \leqslant \|w_k\| \Bigg(&\theta_k((1-\beta_{k-1}) + \kappa_g\beta_{k-1}) \\ &+ \frac{\hat{\kappa}_g}{2}\Bigg(\|w_k\| h(\theta_k)h_{k-1}(\theta_k) + 2\sum_{n=k-m_{k-1}+1}^{k-1}\Bigg(\|w_n\| h(\theta_n)\sum_{j=n}^{k-1} h_j(\theta_k)\Bigg) \\ &+ \left\|w_{k-m_{k-1}}\right\| h(\theta_{k-m_{k-1}})\sum_{j=k-m_{k-1}}^{k-1} h_j(\theta_k)\Bigg)\Bigg),\end{aligned}$$

hence the result. □

The next corollary gives conditions to assure the monotonic decrease of the residual, in the contractive setting.

COROLLARY 5.1 Suppose the hypotheses of Lemma 5.2 for $j = k-m, \ldots, k-1$, and the Lipschitz constant κ_g satisfies $\kappa_g < 1$. Then the following bound holds for the nonlinear residual $\|w_{k+1}\|$ generated by Algorithm 2.2 with $\beta_k = \beta = 1$.

$$\begin{aligned}\|w_{k+1}\| \leqslant \|w_k\| \Bigg\{&\theta_k\kappa_g + \frac{C\sqrt{1+C^2}\sqrt{1-\theta_k^2}\hat{\kappa}_g}{2} \\ &\times \Bigg(\|w_k\| + 2\sum_{n=k-m_{k-1}+1}^{k-1}(k-n)\|w_n\| + m_{k-1}\left\|w_{k-m_{k-1}}\right\| \Bigg)\Bigg\},\end{aligned} \tag{5.23}$$

where

$$C = \max\left\{ \frac{1}{\sigma c_s}\left(\frac{c_t + c_s}{c_s}\right)^{m_k-1}, C_{F,k+1}\right\}, \text{ with } \sigma = (1-\kappa_g). \tag{5.24}$$

After the first $m+1$ consecutive iterations $j = k-m, \ldots, k$ (assuming here for simplicity that $k \geqslant 2m$, so the subscript on m may be dropped), such that the following inequality is satisfied

$$\|w_j\| + 2\sum_{n=j-m+1}^{k-1}\|w_n\| + \left\|w_{j-m}\right\| < \frac{2(1-\kappa_g)}{C\sqrt{1+C^2}\hat{\kappa}_g}, \tag{5.25}$$

monotonic decrease of the residual is assured.

The proof follows similarly to the $m = 1$ case in Corollary 4.1, with the additional steps of bounding the two types of h coefficients.

Proof. For each $\beta_j = 1$ and $\sigma = 1 - \kappa_g$, as in Remark 5.2 the coefficients $h_n(\theta_k)$ are each bounded by $C\sqrt{1-\theta_k^2}$, with C given by (5.24). Applying this to (5.18) allows

$$\|w_{k+1}\| \leqslant \|w_k\| \left(\theta_k \kappa_g + \frac{C\hat{\kappa}_g\sqrt{1-\theta_k^2}}{2}\left(\|w_k\| h(\theta_k) + 2\sum_{n=k-m_{k-1}+1}^{k-1} (k-n)\|w_n\| h(\theta_n) + m_{k-1}\left\|w_{k-m_{k-1}}\right\| h(\theta_{k-m_{k-1}})\right)\right). \tag{5.26}$$

The coefficients $h(\theta_j)$ are each bounded by $C\sqrt{1-\theta_j^2} + \theta_j \leqslant \sqrt{1+C^2}$. Applying this to (5.26) yields (5.23).

Maximizing in terms of θ_k the square of the bracketed terms on the right-hand side of (5.23) is bounded by

$$\kappa_g^2 + \frac{C^2(1+C^2)\hat{\kappa}_g^2}{4}\left(\|w_k\| + 2\sum_{n=k-m+1}^{k-1}(k-n)\|w_n\| + m\|w_{k-m}\|\right)^2. \tag{5.27}$$

Setting (5.27) less than one implies $\|w_{k+1}\| < \|w_k\|$ under the condition (5.25). Satisfaction of $\|w_{j+1}\| < \|w_j\|$ for $m+1$ consecutive iterates $j = k-m, \ldots, k$, then implies reduction in every subsequent residual. □

5.1 *Practical guidance based on the theory*

The results of Theorems 4.1 and 5.1, and Corollaries 4.1 and 5.1 indicate that the most effective choice of algorithmic depth $m = m_k$ may be to have it increase through the simulation based on the three following regimes. The different regimes below can depend on the scaling of data and choice of initial iterates. Damping (not explicitly discussed here; see for instance Evans *et al.*, 2020) may be necessary to see a reduction in the first order residual terms in noncontractive settings, particularly in the initial regime. It is assumed here that the problem dimension n is significantly larger than the number of iterations allowed to solve the problem, and a 'large' value of m_k is still small compared with n. The following three-phase and two-phase approaches are demonstrated in the numerical experiments that follow.

5.1.1 *Three-phase approach.* This approach is appropriate for problems where the initial residual is large or poorly scaled, such that an accumulation of higher-order terms can cause lack of convergence, or even an overflow. This method is demonstrated on the p-Laplacian in Section 6.

- Initial regime: The residual w_k and difference between iterates e_k may be large (in norm), for instance $\mathcal{O}(1)$ or greater. The depth m_k should be chosen small (for instance between 0 and 2), as the accumulation of higher order terms on the right-hand side of (5.18) (cf. (3.7)) can prevent convergence. Additionally, as shown in (4.9) of Theorem 4.1 and (5.18) of Theorem 5.1, a more successful optimization gives greater weight to the higher-order terms.

- Pre-asymptotic regime. The residual or difference between iterates is on the order of 10^{-1} or 10^{-2}. The depth m_k can safely be increased roughly logarithmically with $\|w_k\|$, either to convergence tolerance or until the asymptotic regime is reached.
- Asymptotic regime. The residual is sufficiently small so that the higher-order terms of (5.18) are negligible, regardless of their scaling with respect to the optimization gain θ_k. The depth m_k can be increased, but should be only to the point that it has an impact on decreasing the gain θ_k.

Notably, in the results shown below for the steady NSEs, simply choosing a large depth m (by which $m_k = k - 1$, either up to some maximum m, or up to convergence) can be effective in the case that the initial residual is not greater than $\mathcal{O}(1)$, and drops sufficiently rapidly through the initial iterations. This is essentially the strategy above, starting in the pre-asymptotic, rather than the initial regime. In this example it is also demonstrated that switching to Newton iterations upon sufficient decay of the residual can yield rapid convergence.

5.1.2 *Two-phase approach.* This method is appropriate for problems with a moderately scaled initial residual, on the order of $\mathcal{O}(1)$, and is demonstrated in Section 6 on a nonlinear Helmholtz equation.

- Pre-asymptotic regime. The depth m_k is kept at a small to moderate value (2 to 5), until the residual drops below a given threshold, on the order of 10^{-2} or 10^{-3}.
- Asymptotic regime. The depth m_k is increased to a higher steady level, for instance $m_k = 10$. This allows smaller factors of the optimization gain θ_k due to a better solution of the least-squares problem, at the point where the residual is small enough that the increased weight and accumulation of higher-order terms does not interfere with convergence.

5.1.3 *Safeguarding and verification of* (5.4) *on sufficient linear independence of the columns of* F_k. It makes sense both numerically and theoretically to solve the least squares problem at each stage by means of an economy QR decomposition. Given the large number of degrees of freedom n in comparison to the algorithmic depth m, forming the decomposition and solving the least squares system has a negligible effect on total solution time; in each of our examples given below the total runtime is dominated by the linear system solves. Let $F_k = \hat{Q}\hat{R}$, where $\hat{R}$ is $m_k \times m_k$. Denoting the columns of F_k by $\{v_1, \dots, v_{m_k}\}$ Proposition 5.2 shows the diagonal values of $\hat{R}$ are given by $r_{ii} = \|v_i\| \, |\sin(v_i, \operatorname{span}\{v_1, \dots, v_{i-1}\})|$, by which $|\sin(v_i, \operatorname{span}\{v_1, \dots, v_{i-1}\})| = r_{ii}/\|v_i\|$. If a practitioner wishes to enforce (5.4), any column i for which this quantity falls below a given threshold may be removed from F_k (and accordingly from E_k) and the QR decomposition recalculated (or dynamically updated), by which (5.4) is satisfied in accordance with that threshold (cf. the safeguarding strategy introduced in Pollock & Schwartz, 2020, Section 2.1.2, for AA with $m = 1$ applied to Newton iterations). The method is well defined for any threshold value in $c_T \in [0, 1)$, providing no safeguarding for $c_T = 0$ and otherwise enforcing $c_s = c_T$. This method ensures the most recent column of F_k is used since $r_{11}/\|v_i\| = 1$. This strategy is demonstrated below in Section 6 on a finite element discretization of the p-Laplace equation, with p close to one.

This safeguarding strategy may be compared with that used in Walker & Ni (2011, Section 4) and Yang *et al.* (2009), in which the condition of F_k is monitored by the condition of $\hat{R}$ (which is the same), and the oldest column of F_k is dropped if the condition exceeds a subscribed threshold. The main difference is the present method allows the efficient numerical determination of which column(s) to drop, yielding a theoretically sound update to this older heuristic method. An alternate strategy based on

monitoring the condition number of $\hat{R}$, as suggested in Fang & Saad (2009), is to compute the singular value decomposition (SVD) of $\hat{R}$ and compute the least-squares solution using the pseudoinverse of the truncated expansion to preserve the condition (see Chan, 1982; Sidi, 2016). However, as shown in the numerical examples by Toth & Kelley (2015, Sections 3.1–3.2), the deteriorating condition of the least-squares matrix does not necessarily interfere with convergence.

6. Numerical experiments

In this section the following test problems will illustrate the theory and the practical guidance given above demonstrating both safeguarding and dynamic depth selection strategies, extend the AA methodology to a new application in the nonlinear Helmholtz equation and improve on existing results for AA applied to the steady NSEs.

6.1 *p-Laplace equation*

The p-Laplace (or p-Poisson) equation arises in many physical applications, including non-Newtonian flows, e.g., in glaciology; turbulent flows and flows in porous media; see Diaz & De Thelin (1994); Glowinski & Rappaz (2003); Diening *et al.* (2020). The elliptic p-Laplace equation that is given by

$$-\operatorname{div}\left((|\nabla u|^2/2)^{(p-2)/2}\nabla u\right) = f,$$

is degenerate for $p > 2$ and singular for $1 < p < 2$. In Evans *et al.* (2020) the p-Laplace equation with $p > 2$ is used to demonstrate an approach to adaptively updating damping factors β_k, and it is used as a benchmark problem to demonstrate preconditioned nonlinear solvers in Brune *et al.* (2015). For this example consider a regularized version in the singular regime

$$-\operatorname{div}\left(\left(\varepsilon^2 + \frac{1}{2}|\nabla u|^2\right)^{(p-2)/2}\nabla u\right) = c, \tag{6.1}$$

with $\varepsilon = 10^{-14}$, $p = 1.06$ (cf. Diening *et al.*, 2020) and $c = \pi$, over domain $(0,2) \times (0,2)$, subject to homogeneous Dirichlet boundary conditions.

The results shown below use a P_1 finite element discretization over a 256×256 uniform triangulation of the domain, which produces a discrete nonlinear problem with 66,049 degrees of freedom. The simulations were run using a Python implementation of the FEniCS finite element library (Alnæs *et al.*, 2015). Each simulation was started from initial iterate $u_0 = xy(x-1)(y-1)(x-2)(y-2)$ (cf. Brune *et al.*, 2015), and run to a residual tolerance of $\|w_k\| < 10^{-10}$, where and l_2 norm is used for both the convergence tolerance and the optimization.

A Picard (fixed-point) iteration for the P_1 finite element discretization of the variational form of (6.1) is given by: Find $u_k \in V_h$, satisfying for all $v \in V_h$

$$\int_\Omega \left(\varepsilon^2 + \frac{1}{2}|\nabla u_{k-1}|^2\right)^{(p-2)/2}\nabla u_k \cdot \nabla v \, \mathrm{d}x = \int_\Omega cv \, \mathrm{d}x, \tag{6.2}$$

where V_h is the space of continuous piecewise linear functions that vanish on the boundary.

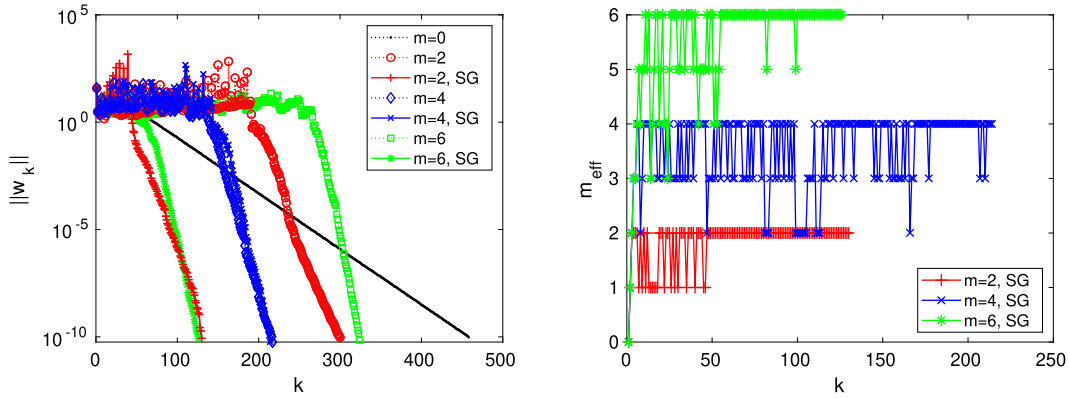

FIG. 1. Left: residual history for AA applied to (6.2) with depths $m = \{0, 2, 4, 6\}$, with and without safeguarding (SG), which selects columns of F_k to enforce (5.4), with threshold $c_s = 0.25$. Right: m_{eff}, the number of columns of F_k selected at each stage in the safeguarded simulations.

With the given parameters, the defined fixed-point operator is not globally contractive, but the Picard iteration does converge essentially linearly as it approaches the solution. Here, AA is applied with $m_k = \min\{k-1, m\}$ for $m = 0, 2, 4, 8$, and these results are compared with dynamically updating the depth m_k by defining

$$\tilde{m}_k := \text{ceil}(-\log_{10}\|w_{k+1}\|) \quad \text{and} \quad m_k = \psi_{n,N}(\tilde{m}_k) := \begin{cases} n, & \tilde{m}_k \leqslant n \\ \tilde{m}_k, & n < \tilde{m}_k < N. \\ N, & \tilde{m}_k \geqslant N \end{cases} \tag{6.3}$$

Setting the depth by (6.3) implements the three-phase approach described in Subsection 5.1.1. Additionally, results are shown for enforcing the sufficient linear-independence condition (5.4) holds, with threshold $c_s = 0.25$, by the method described in Subsection 5.1.3. Runs where this condition is enforced are denoted as safeguarded (SG) in Figs 1 and 2.

Figure 1 (left) shows residual histories for AA with constant depths $m = \{0, 2, 4, 6\}$, where $m = 0$ (the Picard iteration) is included for reference. For $m = \{2, 6\}$ the safeguarding procedure reduces the number of iterations from over 300 to just over 100. This indicates that a near-linear dependence in the columns of F_k produces an undesirable accelerated step toward the beginning of those iterations. The safeguarding is seen to have little effect on any of the iterations once they begin their rapid convergence. It also has little effect on the simulation with $m = 4$, indicating that near-linear dependence is not always an issue. The right plot of Fig. 1 shows the number of columns of F_k (denoted m_{eff}) selected for use by the safeguarding strategy. It is interesting to notice how for $m = \{2, 6\}$ columns are deleted often toward the beginning of the simulations. For $m = 4$ columns are deleted throughout, but with little effect: the columns deleted caused little harm, but also contributed little to the convergence.

Figure 2 (left) shows residual histories using a constant depth of $m = 8$, together with the three-phase dynamic depth selection strategy suggested in Subsection 5.1.1, and given by (6.3), with depths ranging from 0 to 8 using $\psi_{0,8}$ and from 1 to 8 with $\psi_{1,8}$. In this figure it is observed that the safeguarding has little effect on convergence when it is used together with the dynamic depth selection; however, it has a substantial effect on convergence for the constant depth $m = 8$. The right plot of Fig. 2 shows the early intervention in removing certain columns from F_k, when $m_k = \min\{k-1, 8\}$ leads to the fast

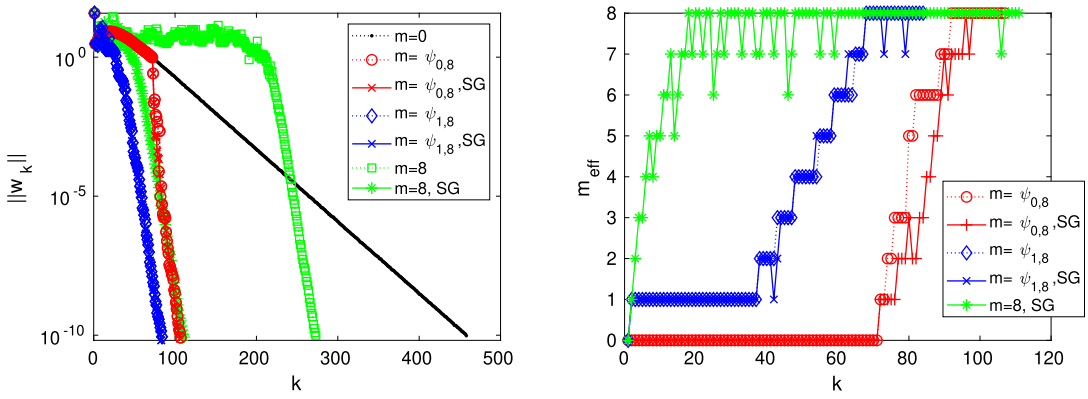

FIG. 2. Left: residual history for AA applied to (6.2) with depths $m = \{0, 8\}$, and dynamically selected depths $\psi_{0,8}$ and $\psi_{1,8}$ as given by (6.3), with and without safeguarding (SG), which selects columns of F_k to enforce (5.4), with threshold $c_s = 0.25$. Right: m_{eff}, the number of columns selected at each stage of the simulations.

convergence seen on the left. In contrast, while the safeguarding strategy does remove columns from F_k periodically using the dynamic $\psi_{1,8}$ and particularly for $\psi_{0,8}$, it leads to very little change in the convergence histories in either case.

This example shows that either dynamic depth selection or safeguarding can lead to improved convergence of AA. The early stages of simulations, particularly if they are started with poor initial iterates, as is the case here, can be sensitive to choice of depth without such interventions. Combining the two strategies did not lead to a noticeable advantage or disadvantage.

6.2 *Nonlinear Helmholtz equation*

The following one-dimensional nonlinear Helmholtz (NLH) equation arises in nonlinear optics, and describes the propagation of continuous-wave laser beams through transparent dielectrics. Following the formulation from Baruch *et al.* (2007), the system may be written as: Find $u : [0, 10] \to \mathbb{C}$, satisfying

$$\frac{\mathrm{d}^2 u}{\mathrm{d}x^2} + k_0^2\big(1 + \epsilon(x)|u|^2\big)u = 0, \quad 0 < x < 10,$$

$$\frac{\mathrm{d}u}{\mathrm{d}x} + ik_0 u = 2ik_0, \quad x = 0,$$

$$\frac{\mathrm{d}u}{\mathrm{d}x} - ik_0 u = 0, \quad x = 10.$$

Here, $\epsilon(x)$ is a given non-negative function of x representing a material constant at each point in space, and k_0 is the linear wave number in the surrounding medium. For simplicity $\epsilon(x) = \epsilon$ is taken as a non-negative constant.

Even in one dimension with constant $\varepsilon(x) > 0$ this is a very challenging problem, especially for larger values of ϵ and k_0, each of which increases the effect of the cubic nonlinearity. The system

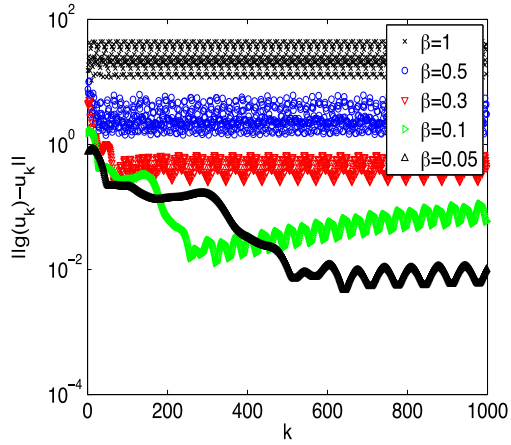

FIG. 3. Results of the NLH test with $\epsilon = 0.22$ and $m = 0$ demonstrating that decreasing a fixed damping factor β does not induce convergence of the fixed-point iteration.

is discretized by applying a second-order finite difference method (with uniform point spacing of $h = 0.01$) to the iteration

$$\frac{\mathrm{d}^2 u_{j+1}}{\mathrm{d}x^2} + k_0^2\left(1 + \epsilon|u_j|^2\right)u_{j+1} = 0, \quad 0 < x < 10, \tag{6.4}$$

$$\frac{\mathrm{d}u_{j+1}}{\mathrm{d}x} + ik_0 u_{j+1} = 2ik_0, \quad x = 0, \tag{6.5}$$

$$\frac{\mathrm{d}u_{j+1}}{\mathrm{d}x} - ik_0 u_{j+1} = 0, \quad x = 10. \tag{6.6}$$

This can be considered a fixed point iteration $u_{j+1} = g(u_j)$, with g defined to be the solution operator of the (discretized) systems (6.4)–(6.6). Following Baruch *et al.* (2007), $u_0 = e^{ik_0 x}$ is used as the initial iterate.

This NLH test uses $\epsilon = 0.22$, for which the fixed point iterations (6.4)–(6.6) do not converge. Figure 3 shows the fixed-point iteration ($m = 0$) with varying levels of relaxation (damping); this illustrates that (uniform) relaxation alone is not sufficient for convergence. In Fig. 4 results of AA applied to the iteration using relaxation parameter $\beta_k = \beta = 0.3$ are shown for $m = 1, 3, 5, 10$, all of which converge. The plot of k vs. θ_k shows a clear reduction in gain factors θ_k as the depth m increases. Comparing convergence histories for varying depths m, none of the depths tested show monotonic decrease, particularly in the preasymptotic regime. Depth $m = 10$, which becomes nearly monotone in the asymptotic regime, has gain values generally less than 0.6; whereas depth $m = 1$, which is far from monotone, has gain values that return to nearly one throughout the first 250 iterations shown in Fig. 4, on the right.

The next results, shown in Fig. 5, use a heuristic strategy for updating m. This strategy is based on the observation that depth $m = 3$ gives a faster initial decrease in the residual, and $m = 10$ gives the fastest

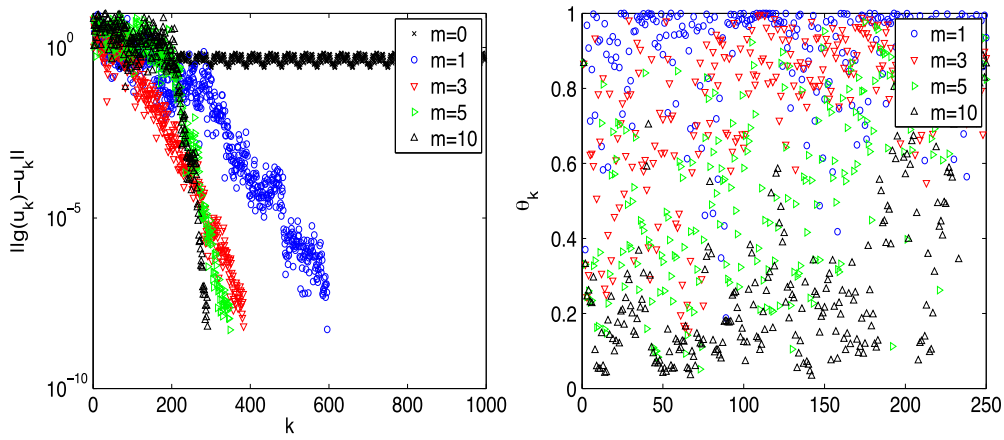

FIG. 4. Results of the NLH test with $\epsilon = 0.22$, as convergence of the nonlinear residual (left) for $\beta_k = \beta = 0.3$ and varying m, and θ_k for varying m (right).

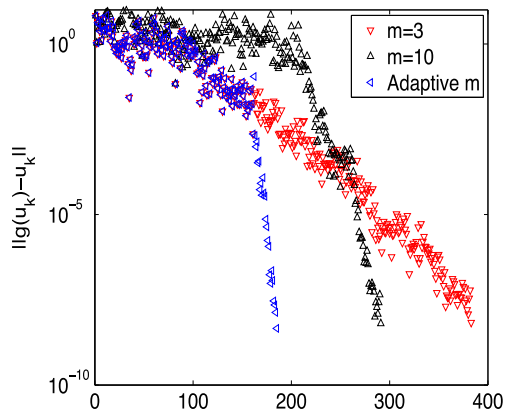

FIG. 5. Results of the NLH test with $\epsilon = 0.22$, as convergence of the nonlinear residual (left) for $\beta_k = \beta = 0.3$ and $m = 3$, $m = 10$, and a heuristic strategy where $m = 3$ at first, but switches to $m = 10$ once the nonlinear residual is sufficiently small.

eventual decrease. Here, depth m_k is switched from $m_k = 3$ to $m_k = 10$ on the condition of a sufficiently small residual, where the tolerance is set at 0.005. The depth-switching approach yields substantially faster convergence than either constant-depth strategy. This is again consistent with the theory, as larger higher order terms play a greater role earlier in the iteration history, and moreso at greater algorithmic depths. Once the higher order terms are sufficiently small (attained through a sequence of sufficiently small gain values), the decrease in gain θ_k for greater depths m yields better performance, as the residual is small enough to be dominated by the first-order term even as the number and weight of the higher order terms increase.

6.3 *Three-dimensional steady Navier–Stokes equations*

The last example shown is for the three-dimensional driven cavity benchmark test problem for the steady NSE. The steady NSEs are given in a domain $\Omega \subset \mathbb{R}^d$ ($d = 2,3$) by

$$u \cdot \nabla u + \nabla p - \nu \Delta u = f, \tag{6.7}$$

$$\nabla \cdot u = 0, \tag{6.8}$$

$$u|_{\partial\Omega} = s, \tag{6.9}$$

where ν is the kinematic viscosity that is inversely proportional to the Reynolds number $Re := \nu^{-1}$, f is a forcing term, and u and p represent velocity and pressure. The NSEs are well known to be more difficult to solve with larger Reynolds number.

The three-dimensional driven cavity is a widely studied benchmark problem for the NSE, and typically with $Re \leqslant 1000$ (see Wong & Baker, 2002, and the reference therein). For this problem $\Omega = (0,1)^3$, and there is no forcing ($f = 0$). For boundary conditions $s = 0$ is enforced on the bottom and sides, and on the top, $s = \langle 1,0,0\rangle^\top$, by which the driving force is provided by the moving lid. Recently, higher Re have been considered, but as a *time dependent flow*, in an attempt to find the first Hopf bifurcation where the flow becomes oscillatory, and will not converge to a steady state (Feldman & Gelfgat, 2010; Chiu *et al.*, 2016). This bifurcation appears to occur around $Re \approx 2000$. Here, the systems (6.7)–(6.9) are solved by applying AA to the Picard iteration, given by Girault & Raviart (1986) as

$$u_k \cdot \nabla u_{k+1} + \nabla p_{k+1} - \nu \Delta u_{k+1} = f, \tag{6.10}$$

$$\nabla \cdot u_{k+1} = 0, \tag{6.11}$$

$$u_{k+1}|_{\partial\Omega} = s. \tag{6.12}$$

The system above defines a fixed-point iteration with $u_{k+1} = g(u_k)$, where g is the solution operator for a spatial discretization of (6.10)–(6.12). The system is discretized using (P_3, P_2^{disc}) Scott–Vogelius finite elements on a barycenter refined tetrahedral mesh that provides 1.3 million total degrees of freedom. The tetrahedral mesh was created using first a box mesh to subdivide all axes using Chebyshev points (to be more refined near the boundary), then splitting each box into six tetrahedra, then splitting each tetrahedron with a barycenter refinement. The initial guess for each of the NSE tests is $u_0 = 0$ (no continuation methods are applied).

In the paper by Pollock *et al.* (2019) AA applied to (6.10)–(6.12) (referred to here as AAPicard) was studied both theoretically and numerically. Under a small data condition that implies the underlying fixed-point iteration is contractive, it was shown that the method converges and that the linear convergence rate is improved by AA. It is remarked, however, that the techniques used in that analysis and the coefficients in front of the higher order terms differ significantly from those shown here.

For the current test problem, as shown in Pollock *et al.* (2019), with an initial guess of $u_0 = 0$, the Picard method does not converge when $Re = 400$. Hence, for $Re \geqslant 400$, Picard iterations for steady solutions are not globally contractive. In fact, AAPicard with $m = 1$ fails as well, although convergence is attained with depths $m = 2,3,4$ as demonstrated in Pollock *et al.* (2019). To show the effectiveness of AAPicard, considerably higher Reynolds numbers are considered here: results are

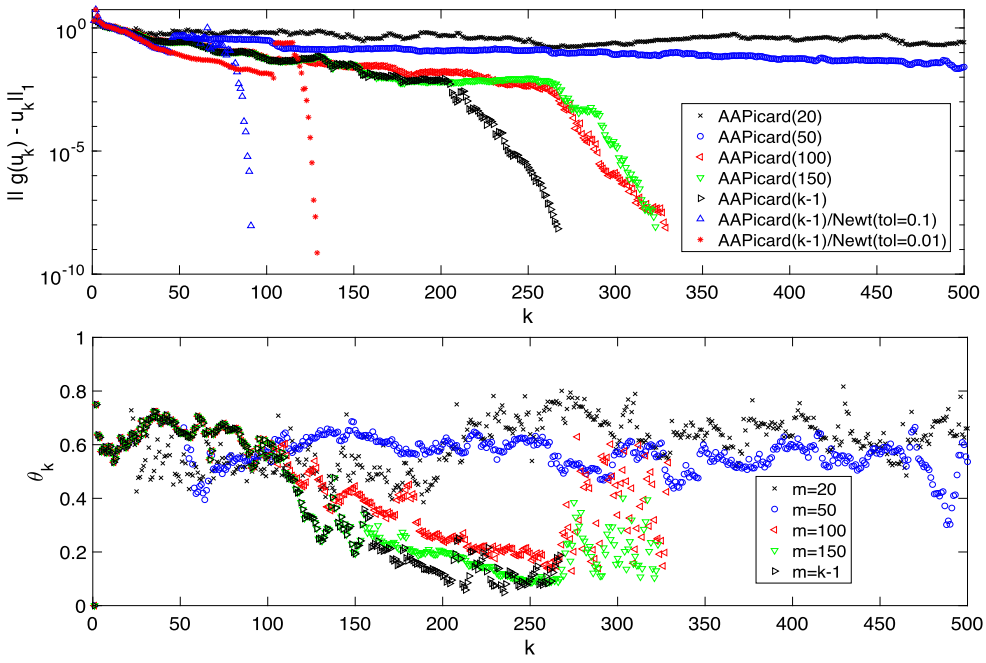

FIG. 6. Top: convergence of AAPicard with varying m with and without a switch to Newton. Bottom: gain factors θ_k for varying m.

presented for $Re = 2500$ and $Re = 3100$, far beyond the range where the Picard iteration is contractive; and moreover, well past the first Hopf bifurcation (Feldman & Gelfgat, 2010; Chiu *et al.*, 2016). Thus the method is converging to steady solutions in a time dependent regime, which from a mathematical point of view is interesting in itself. As discussed in Akervik *et al.* (2006), such solutions can serve as base-flow solutions in instability studies and flow control.

The $Re = 2500$ tests show different choices of the depth m, including the largest possible ($m_k = k - 1$), with no relaxation ($\beta_k = \beta = 1$). Results are shown in Fig. 6. For $m \leqslant 50$ convergence is not achieved (nor is it close to being achieved) after 500 iterations. For $m = 100, 150$ and $m = k - 1$, the method does converge. It appears that the stability of the NSE Picard iteration (Girault & Raviart, 1986) bounds the magnitude of any residual, and the improved analysis herein shows that higher order terms are all scaled by the latest residual, which together allows the method to benefit from the small gain factor θ_k that comes from a greater algorithmic depth m ($m \geqslant 100$ creates gain factors θ_k that get to 0.25 and below). Notably, choosing m as large as possible, $m_k = k - 1$, gives the fastest convergence.

Finally, a combination of AAPicard with Newton (cf. Fabien *et al.*, 2018) was tested. The Newton iteration differs from the Picard in that the term $(u_{k+1} - u_k) \cdot \nabla u_k$ is added to the left side of (6.10). Additionally, a line search was used in the Newton iterations. The results shown used $m_k = k - 1$ for the initial AAPicard iterations and switched to Newton once the nonlinear residual reached a sufficiently low tolerance. For an H_0^1-norm tolerance of 1, the method failed to converge. For tolerances of 0.1 and 0.01, the method converged, and much faster than AAPicard on its own (see the top plot in Fig. 6).

With this technique the solver attained convergence up to $Re = 3100$ (using AAPicard with $m_k = k - 1$ and $\beta_k = \beta = 0.5$, up to a residual tolerance of 0.03, then switching to Newton with

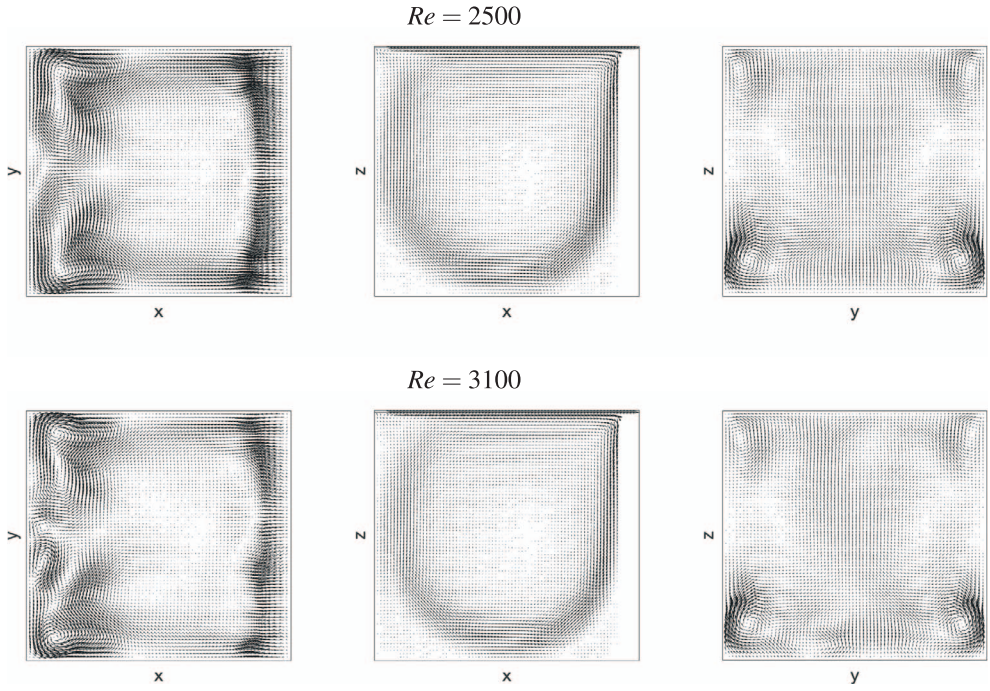

FIG. 7. Shown above are $Re = 2500$ and 3100 solutions, as midsliceplanes of the velocity fields.

a line search). With this method 217 iterations were needed to converge to a tolerance of 10^{-8} in the H_0^1-norm. With a continuation method that improves the initial guess solutions at even higher Re can be obtained. Plots of the $Re = 2500$ and 3100 solutions are shown in Fig. 7 as midsliceplanes of the velocity fields.

7. Conclusion

The presented one-step analysis of AA sharpens the previously developed residual bounds for contractive operators, and extends them to a class of potentially noncontractive operators that are important for the approximation of solutions to nonlinear PDEs. The new analysis shows how the relative scaling of the higher-order terms increases as the solution to the underlying optimization problem improves. Understanding the balance of the higher and lower order terms in the residual expansion is instrumental in the design of robust and efficient algorithms for challenging nonlinear problems. The current theory assumes that the latest difference between consecutive residuals sufficiently changes the span of the previous differences, up to the given algorithmic depth. An efficient safeguarding strategy to ensure this assumption holds is introduced and demonstrated, advancing the connection between theory and practice in a sense not accomplished with the usual assumption that the optimization coefficients are bounded. Practical advantages based on the present advances in theory are demonstrated in the numerical section, where AA is used to attain results for the nonlinear Helmholtz equation and three-dimensional steady Navier–Stokes past the first Hopf bifurcation, which cannot be attained by the usual combinations of Picard iterations, Newton iterations and relaxation techniques alone.

Acknowledgements

The authors would like to thank the anonymous referees for suggesting additional clarification on the connection between the theory and the examples, and on the value of assuming (5.4) instead the boundedness of the optimization coefficients.

Funding

National Science Foundation Division of Mathematical Sciences (1852876 and 2011519 to S.P.; 1522191 and 2011490 to L.R.).

References

Akervik, E., Brandt, L., Henningson, D., Hoepffner, J., Marxen, O. & Schlatter, P. (2006) Steady solutions of the Navier–Stokes equations by selective damping. *Phys. Fluids*, **18**, 1–4.

Alnæs, M. S., Blechta, J., Hake, J., Johansson, A., Kehlet, B., Logg, A., Richardson, C., Ring, J., Rognes, M. E. & Wells, G. N. (2015) The FEniCS project version 1.5. *Arch. Numer. Softw.*, **3**, 9–23.

An, H., Jia, X. & Walker, H. (2017) Anderson acceleration and application to the three-temperature energy equations. *J. Comput. Phys.*, **347**, 1–19.

Anderson, D. G. (1965) Iterative procedures for nonlinear integral equations. *J. Assoc. Comput. Mach.*, **12**, 547–560.

Baruch, G., Fibich, G. & Tsynkov, S. (2007) High-order numerical method for the nonlinear Helmholtz equation with material discontinuities in one space dimension. *J. Comput. Phys.*, **227**, 820–850.

Both, J. W., Kumar, K., Nordbotten, J. M. & Radu, F. A. (2019) Anderson accelerated fixed-stress splitting schemes for consolidation of unsaturated porous media. *Comput. Math. Appl.*, **77**, 1479–1502. 7th International Conference on Advanced Computational Methods in Engineering (ACOMEN 2017).

Brezinski, C., Redivo-Zaglia, M. & Saad, Y. (2018) Shanks sequence transformations and Anderson acceleration. *SIAM Rev.*, **60**, 646–669.

Brune, P. R., Knepley, M. G., Smith, B. F. & Tu, X. (2015) Composing scalable nonlinear algebraic solvers. *SIAM Rev.*, **57**, 535–565.

Chan, T. (1982) An improved algorithm for computing the singular value decomposition. *ACM Trans. Math. Softw.*, **8**, 72–83.

Chiu, S.-H., Pan, T.-W., He, J., Guo, A. & Glowinski, R. (2016) A numerical study of the transition to oscillatory flow in 3D lid-driven cubic cavity flows. arXiv preprint arXiv:1604.06926.

Diaz, J. I. & De Thelin, F. (1994) On a nonlinear parabolic problem arising in some models related to turbulent flows. *SIAM J. Math. Anal.*, **25**, 1085–1111.

Diening, L., Fornasier, M., Tomasi, R. & Wank, M. (2020) A relaxed Kačanov iteration for the p-Poisson problem. *Numer. Math.*, **145**, 1–34.

Evans, C., Pollock, S., Rebholz, L. & Xiao, M. (2020) 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.*, **58**, 788–810.

Eyert, V. (1996) A comparative study on methods for convergence acceleration of iterative vector sequences. *J. Comput. Phys.*, **124**, 271–285.

Fabien, M. S., Knepley, M. G. & Rivière, B. M. (2018) A hybridizable discontinuous Galerkin method for two-phase flow in heterogeneous porous media. *Int. J. Numer. Methods Eng.*, **116**, 161–177.

Fang, H. & Saad, Y. (2009) Two classes of multisecant methods for nonlinear acceleration. *Numer. Linear Algebra Appl.*, **16**, 197–221.

Feldman, Y. & Gelfgat, A. (2010) Oscillatory instability of a three-dimensional lid-driven flow in a cube. *Phys. Fluids*, **22**, 1–9.

Girault, V. & Raviart, P.-A. (1986) *Finite Element Methods for Navier–Stokes Equations: Theory and Algorithms*. Berlin, Heidelberg: Springer.
Glowinski, R. & Rappaz, J. (2003) Approximation of a nonlinear elliptic problem arising in a non-Newtonian fluid flow model in glaciology. *ESAIM Math. Model. Numer. Anal.*, **37**, 175–186.
Golub, G. H. & Van Loan, C. F. (1996) *Matrix Computations*, 3rd edn. Baltimore, MD, USA: Johns Hopkins University Press.
Higham, N. & Strabic, N. (2016) Anderson acceleration of the alternating projections method for computing the nearest correlation matrix. *Numer. Algorithms*, **72**, 1021–1042.
Kelley, C. T. (2018) Numerical methods for nonlinear equations. *Acta Numerica*, **27**, 207–287.
Lott, P. A., Walker, H. F., Woodward, C. S. & Yang, U. M. (2012) An accelerated Picard method for nonlinear systems related to variably saturated flow. *Adv. Water Resour.*, **38**, 92–101.
Peng, Y., Deng, B., Zhang, J., Geng, F., Qin, W. & Liu, L. (2018) Anderson acceleration for geometry optimization and physics simulation. *ACM Trans. Graphics*, **37**, 42:1–42:14.
Pollock, S., Rebholz, L. & Xiao, M. (2019) Anderson-accelerated convergence of Picard iterations for incompressible Navier–Stokes equations. *SIAM J. Numer. Anal.*, **57**, 615–637.
Pollock, S. & Schwartz, H. (2020) Benchmarking results for the Newton–Anderson method. *Results Appl. Math.*, **8**, 100095.
Sidi, A. (2016) SVD-MPE: an SVD-based vector extrapolation method of polynomial type. *Appl. Math.*, **7**, 1260–1278.
Smith, D. A., Ford, W. F. & Sidi, A. (1987) Extrapolation methods for vector sequences. *SIAM Rev.*, **29**, 199–233.
Stasiak, P. & Matsen, M. (2011) Efficiency of pseudo-spectral algorithms with Anderson mixing for the SCFT of periodic block-copolymer phases. *Eur. Phys. J. E*, **34:110**, 1–9.
Toth, A., Kelley, C., Slattery, S., Hamilton, S., Clarno, K. & Pawlowski, R. (2015) Analysis of Anderson acceleration on a simplified neutronics/thermal hydraulics system. *Proceedings of the ANS MC2015 Joint International Conference on Mathematics and Computation (M&C), Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method*, ANS MC2015 CD, 1–12.
Toth, A. & Kelley, C. T. (2015) Convergence analysis for Anderson acceleration. *SIAM J. Numer. Anal.*, **53**, 805–819.
Walker, H. F. & Ni, P. (2011) Anderson acceleration for fixed-point iterations. *SIAM J. Numer. Anal.*, **49**, 1715–1735.
Wong, K. & Baker, A. (2002) A 3D incompressible Navier–Stokes velocity–vorticity weak form finite element algorithm. *Int. J. Numer. Methods Fluids*, **38**, 99–123.
Yang, C., Meza, J. C., Lee, B. & Wang, L.-W. (2009) KSSOLV—a MATLAB toolbox for solving the Kohn–Sham equations. *ACM Trans. Math. Softw.*, **36**, 1–35.

A. Appendix

The proof of the technical Lemma 5.1 follows.

Proof. The proof follows by induction on the submatrix formed by the first p rows and columns of R, then by induction indexing up the entries of the right-most column. Let $R_p = R(1:p, 1:p)$, the upper-left $p \times p$ block of $\hat{R}$, with inverse S_p.

The off-diagonal entries r_{ij} of $\hat{R}$ are given by $r_{ij} = (q_i, a_j) = \|a_j\| \cos(q_i, a_j)$, and by Proposition 5.2 the diagonal entries are given by $r_{ii} = \|a_i\| \, |\sin(a_i, \mathcal{A}_{i-1})|$, following the convention that the columns of $\hat{Q}$ are chosen so the r_{ii} are positive.

For the trivial case of $p = 1$, $R_1 = r_{11}$ and $s_{11} = 1/r_{11} = 1/\|a_1\|$. By Proposition 5 to compute the inverse of R_2 it remains to compute s_{22} and s_{12}. It is useful here to state the inversion formula for entries

of the right-most column (index p) as

$$s_{pp} = \frac{1}{r_{pp}} \text{ and } s_{kp} = -\frac{1}{r_{kk}} \sum_{j=1}^{p-k} r_{k,k+j} s_{k+j,p} = -\frac{1}{r_{kk}} \sum_{j=1}^{p-k} \|a_{k+j}\| \cos(q_k, a_{k+j}) s_{k+j,p},\ k < p. \tag{A.1}$$

For $p = 2$ the inversion formula (A.1) and expression (5.1) for the diagonal entries yield $s_{22} = 1/r_{22} = 1/(\|a_2\| |\sin(a_2, q_1)|)$. Then by the hypotheses of the lemma $s_{22} \leqslant 1/(\|a_2\| c_s)$. Using (A.1) the off-diagonal entry then satisfies $s_{12} = -\|a_2\| \cos(q_1, a_2) s_{22}/r_{11}$, yielding $|s_{12}| \leqslant c_t/(\|a_1\| c_s)$. Hence, for $p = 2$ the result holds. Continue by induction on p, assuming the result holds for $q = 1, \ldots, p-1$. Then for $q = p$,

$$R_p = \begin{pmatrix} R_{p-1} & r_{1p} \\ & \vdots \\ 0 & r_{pp} \end{pmatrix}.$$

By (A.1), Proposition 5.2 and the hypotheses of the lemma $\|s_{pp}\| \leqslant 1/(\|a_p\| c_s)$.

Similarly by (A.1) $\|s_{p-1,p}\| \leqslant c_t/(\|a_{p-1}\| c_s^2)$. This satisfies the base step on the inner induction, up row p of S_p. Assuming the bound of (5.3) for s_{ip} holds for $i = p-1$ down to $i = k+1$, it suffices to show the result for $i = k$. By (A.1) and the inductive hypothesis,

$$|s_{kp}| = \left| \frac{1}{r_{kk}} \sum_{j=1}^{p-k} \|a_{k+j}\| \cos(q_k, a_{k+j}) s_{k+j,p} \right| \leqslant \frac{1}{r_{kk}} \left(\sum_{j=1}^{p-k-1} \frac{c_t^2 (c_t + c_s)^{p-(k+j)-1}}{c_s^{p-(k+j)+1}} + \frac{c_t}{c_s} \right).$$

Setting $n = p - k$

$$|s_{kp}| \leqslant \frac{1}{r_{kk}} \left(\sum_{j=1}^{n-1} \frac{c_t^2 (c_t + c_s)^{n-j-1}}{c_s^{n-j+1}} + \frac{c_t}{c_s} \right) = \frac{c_t}{c_s^n r_{kk}} \left(\sum_{j=1}^{n-1} c_t (1 + c_s)^{n-j-1} c_s^{j-1} + c_s^{n-1} \right). \tag{A.2}$$

Rearranging the terms in the sum shows

$$\begin{aligned}
&\sum_{j=1}^{n-1} c_t(c_t+c_s)^{n-j-1}c_s^{j-1} + c_s^{n-1} \\
&\quad = \sum_{j=1}^{n-2} c_t(c_t+c_s)^{n-j-1}c_s^{j-1} + (c_t+c_s)c_s^{n-2} \\
&\quad = \sum_{j=1}^{n-3} c_t(c_t+c_s)^{n-j-1}c_s^{j-1} + (c_t+c_s)^2c_s^{n-3} \\
&\quad \vdots \\
&\quad = c_t(c_t+c_s)^{n-2} + c_t(c_t+c_s)^{n-3}c_s + (c_t+c_s)^{n-3}c_s^2 \\
&\quad = c_t(c_t+c_s)^{n-2} + (c_t+c_s)^{n-2}c_s \\
&\quad = (c_t+c_s)^{n-1}. \qquad \text{(A.3)}
\end{aligned}$$

Applying (A.3) and (5.1) to (A.2) allows

$$|s_{1p}| \leqslant \frac{c_t(c_t+c_s)^{p-2}}{\|a_1\|\, c_s^{p-1}} \text{ and } |s_{kp}| \leqslant \frac{c_t(c_t+c_s)^{p-k-1}}{\|a_k\|\, c_s^{p-k+1}},\ k=2,\ldots,p-1,$$

which completes the inductive step on k and hence on p and establishes the result. □