



Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Anderson acceleration based on the \mathcal{H}^{-s} Sobolev norm for contractive and noncontractive fixed-point operators

Yunan Yang^{a,*}, Alex Townsend^a, Daniel Appelö^{b,c}^a Department of Mathematics, Cornell University, Ithaca, NY 14850, United States of America^b Department of Computational Mathematics, Science, and Engineering, Michigan State University, East Lansing, MI 48824, United States of America^c Department of Mathematics, Michigan State University, East Lansing, MI 48824, United States of America

ARTICLE INFO

Article history:

Received 26 February 2020

Received in revised form 11 September 2021

MSC:

65B99

65F10

65N15

65F08

65K10

46E39

Keywords:

Anderson acceleration

Fixed-point iteration

Sobolev space

Iterative methods

Optimization

Helmholtz equation

ABSTRACT

Anderson acceleration (AA) is a technique for accelerating the convergence of fixed-point iterations. In this paper, we apply AA to a sequence of functions and modify the norm in its internal optimization problem to the \mathcal{H}^{-s} norm, for some positive integer s , to bias it towards low-frequency spectral content in the residual. We analyze the convergence of AA by quantifying its improvement over Picard iteration. We find that AA based on the \mathcal{H}^{-2} norm is well-suited to solve fixed-point operators derived from second-order elliptic differential operators, including the Helmholtz equation.

© 2021 Elsevier B.V. All rights reserved.

1. Introduction

Anderson acceleration (AA) or Anderson mixing is an acceleration method for fixed-point iterations. Given a continuous operator $G : \mathcal{X} \rightarrow \mathcal{X}$, where $\mathcal{X} \subseteq L^2(\Omega)$ is a Hilbert space and $\Omega \subseteq \mathbb{R}^n$, a basic method for finding a fixed-point of G , i.e., $x = G(x)$, is Picard iteration

$$x_{k+1} = G(x_k), \quad k \geq 1, \quad x_0 \in \mathcal{X} \text{ given.} \quad (1.1)$$

AA can be used to speed up the convergence of x_0, x_1, \dots , to a fixed-point of G , or even calculate a fixed point when the Picard iterates diverge [1, Thm. 4.1]. While Picard iteration only uses the current iterate to calculate the next one, x_{k+1} in AA is a weighted sum of the previous $\min(k, m) + 1$ iterates and residuals, where m is a memory parameter. The weighted sum is chosen so that it minimizes a linearized residual [2, (4.16)] in the next iteration.

The application of AA includes flow problems [3], solving nonlinear radiation–diffusion equations [4], and accelerating certain optimization algorithms [5–8]. It is closely related to Pulay mixing [9] and DIIS (direct inversion on the iterative

* Corresponding author.

E-mail addresses: yy837@cornell.edu (Y. Yang), townsend@cornell.edu (A. Townsend), appeloda@msu.edu (D. Appelö).

subspace) [10,11], which are prominent methods in self-consistent field theory (SCFT) [12,13]. AA is also becoming popular in the numerical analysis community [1,14–17]. AA is related to many other iterative and acceleration methods. When $m = 0$, AA collapses to Picard iteration and when $m = \infty$, AA is essentially equivalent to GMRES (for Generalized Minimal RESidual method) when the fixed-point operator is linear [14]. For any m , AA can be viewed as a multiseccant quasi-Newton method [18,19] and is also related to traditional series acceleration methods [20].

As in Anderson’s paper [2], we primarily regard AA as an iteration performed on functions. As an acceleration method on functions, calculating the weighted sum to minimize the linearized residual involves an optimization problem that is often posed in the L^2 norm. The L^2 norm is convenient because the optimization problem is then a continuous least-squares problem. For practical computations, AA on functions must be discretized so that functions become vectors, and the L^2 norm becomes the discrete ℓ^2 norm. After discretization, the optimization problem becomes a classical least-squares problem, which can be solved using fast rank-updated QR factorizations [21, Sec. 6.5.1].

We follow a casual suggestion by Anderson [2, p. 554] and minimize the linearized residual at each iteration in a norm other than the L^2 norm. In particular, we seek further acceleration for fixed-point iterations involving second-order elliptic differential operators by selecting an \mathcal{H}^{-s} norm (see Section 2.1). After discretization, any \mathcal{H}^{-s} norm becomes a weighted least-squares norm. In certain situations, this can provide an implicit spectral bias to counterbalance the spectral biasing from a fixed-point operator. Other researchers have been motivated to modify the norm in AA based on the contraction properties, as opposed to spectral biasing, of G [3].

AA based on the \mathcal{H}^{-s} norm is equivalent to a multiseccant method in a weighted Frobenius norm (see Section 2.3). We use this viewpoint to analyze its convergence behavior by comparing it to AA based on the L^2 norm and Picard iteration (see Section 3.2). The improvement depends on the particular properties of the fixed-point operator. We present some analysis motivating the choice of the norm (see Theorem 3.2) as well as providing numerical experiments to demonstrate the benefit (see Section 4).

The paper is structured as follows. In Section 2, we provide background details on the AA method, introduce AA based on the \mathcal{H}^{-s} norm, and show how it can be discretized. In Section 3, we give a detailed analysis of the error reduction achieved by performing one step of the AA iteration (see Theorem 3.1). In Section 4, we provide numerical experiments, including both contractive and noncontractive fixed-point operators for the 1D Poisson equation and linear and non-linear Helmholtz equations. Concluding remarks can be found in Section 5.

2. Anderson acceleration

The convergence of Picard iteration in (1.1) is only guaranteed when certain assumptions hold on G as well as the initial iterate x_0 , and even then, its convergence is typically linear [22, Chap. 4.2]. To promote faster convergence, AA computes x_{k+1} using the previous $\min(k, m) + 1$ iterates and residuals. The original form of AA is given in Algorithm 1, and the main step is to take a linear combination of the past $\min(k, m) + 1$ iterates to minimize a linearized residual of the form $G(x_i) - x_i$.

Algorithm 1 The basic AA technique with the L^2 norm

Input: Given $x_0 \in \mathcal{X}$, mixing parameters $0 \leq \beta_k \leq 1$, and memory parameter $m \geq 1$, this algorithm computes a sequence x_0, x_1, \dots , intended to converge to a fixed-point of $G : \mathcal{X} \rightarrow \mathcal{X}$.

for $k = 0, 1, \dots$ until convergence **do**

$m_k = \min(m, k)$.

Compute $F_k = (f_{k-m_k}, \dots, f_k)$, where $f_i = G(x_i) - x_i$.

Solve

$$\alpha^{(k)} = \arg \min_{v \in \mathbb{C}^{m_k+1}, \sum_{i=0}^{m_k} v_i = 1} \|F_k v\|_{L^2}, \quad \alpha^{(k)} = (\alpha_0^{(k)}, \dots, \alpha_{m_k}^{(k)})^T.$$

Set

$$x_{k+1} = (1 - \beta_k) \sum_{i=0}^{m_k} \alpha_i^{(k)} x_{k-m_k+i} + \beta_k \sum_{i=0}^{m_k} \alpha_i^{(k)} G(x_{k-m_k+i}).$$

end for

The mixing parameter β_k at iteration k indicates how to combine the previous $m_k + 1$ iterates and residuals. The usual choice is to select $\beta_k = \beta$ for $k \geq 0$. AA with mixing parameter $\beta_k = \beta$ is the same as applying AA with $\beta_k = 1$ to the map $G_\beta(x) = (1 - \beta)x + \beta G(x)$ [23, p. 256]. Therefore, throughout this paper we use $\beta = 1$.

In Algorithm 1, the coefficient vector $\alpha^{(k)}$ is determined by a constrained optimization problem. To remove the constraint, and gain additional insight, one can set [14]

$$\gamma_i^{(k)} = \alpha_0^{(k)} + \dots + \alpha_i^{(k)}, \quad 0 \leq i \leq m_k - 1.$$

By carefully rewriting Algorithm 1 with $\beta_k = 1$ in terms of $\gamma_i^{(k)}$ for $0 \leq i \leq m_k - 1$, we obtain Algorithm 2. In this version of the algorithm, the coefficient vector $\gamma^{(k)}$ is determined by an unconstrained optimization problem, which can be computationally more convenient.

Algorithm 2 A reformulated AA technique for functions

Input: Given $x_0 \in \mathcal{X}$, memory parameter $m \geq 1$, and measure of distance $d : \mathcal{X} \times \mathcal{X} \rightarrow [0, \infty)$, this algorithm computes a sequence x_0, x_1, \dots , intended to converge to a fixed-point of $G : \mathcal{X} \rightarrow \mathcal{X}$.

Set $x_1 = G(x_0)$.

for $k = 1, 2, \dots$ until convergence **do**

$m_k = \min(m, k)$.

Set $D_k = (\Delta f_{k-m_k}, \dots, \Delta f_{k-1})$, where $\Delta f_i = f_{i+1} - f_i$ and $f_i = G(x_i) - x_i$.

Solve

$$\gamma^{(k)} = \operatorname{argmin}_{v \in \mathbb{C}^{m_k}} d(f_k, D_k v), \quad \gamma^{(k)} = (\gamma_0^{(k)}, \dots, \gamma_{m_k-1}^{(k)})^T. \tag{2.1}$$

Set

$$x_{k+1} = G(x_k) - \sum_{i=0}^{m_k-1} \gamma_i^{(k)} [G(x_{k-m_k+i+1}) - G(x_{k-m_k+i})]. \tag{2.2}$$

end for

In Algorithm 2, one has a choice on $d : \mathcal{X} \times \mathcal{X} \rightarrow [0, \infty)$, which can be chosen as any distance function. A standard choice is to take $d(f, g) = \|f - g\|_{L^2(\mathbb{R}^n)}$ so that (2.1) can be efficiently solved. To see this, note that (2.1) becomes $\gamma^{(k)} = \operatorname{argmin}_{v \in \mathbb{C}^{m_k}} \|f_k - D_k v\|_{L^2(\mathbb{R}^n)}$. This means that, when D_k has linearly independent columns, we have

$$\gamma^{(k)} = (D_k^* D_k)^{-1} D_k^* f_k, \quad (D_k^* D_k)_{ij} = \langle \Delta f_i, \Delta f_j \rangle, \quad (D_k^* f_k)_i = \langle \Delta f_i, f_k \rangle, \tag{2.3}$$

where $\langle \cdot, \cdot \rangle$ is the standard L^2 inner-product. Moreover, (2.3) can be efficiently solved by a fast rank-updated QR factorization of quasimatrices [24].

When m is finite, AA is distinct from restarted GMRES in that it gradually phases out old residuals in favor of new ones while GMRES completely discards the history of the iterates every m iterations. We demonstrate, by numerical experiments with the Helmholtz equation, that the gradual replacement strategy used by AA exhibits better convergence properties than restarted GMRES (see Section 4.4).

The computational efficiency and convergence rate of AA is affected by the distance function. The majority of the literature focuses on the convergence of AA for vectors under the discrete ℓ^2 norm (Euclidean distance). In this setting, AA is known to have superlinear convergence [3,15] when accelerating fixed-point contraction operators. AA can also converge for sequences from noncontractive fixed-point operators [1].

2.1. The Hilbert space \mathcal{H}^{-s} and its norm

We select the distance function in Algorithm 2 to be $d(f, g) = \|f - g\|_{\mathcal{H}^{-s}}$ for some positive integer s . We observe that this can speed up the convergence of AA for fixed-point operators defined via second-order elliptic differential operators (see Section 4).

One can define $\mathcal{H}^{-s}(\mathbb{R}^n)$, for any real number s , as

$$\mathcal{H}^{-s}(\mathbb{R}^n) = \{f \in \mathcal{S}'(\mathbb{R}^n) : \mathcal{F}^{-1} [(1 + |\xi|^2)^{-s/2} \mathcal{F}f] \in L^2(\mathbb{R}^n)\},$$

where \mathcal{F} denotes the Fourier transform on \mathbb{R}^n and $\mathcal{S}'(\mathbb{R}^n)$ is the space of tempered distributions [25]. The Hilbert space $\mathcal{H}^{-s}(\mathbb{R}^n)$ can be equipped with the norm

$$\|f\|_{\mathcal{H}^{-s}(\mathbb{R}^n)} = \|\mathcal{F}^{-1} [(1 + |\xi|^2)^{-s/2} \mathcal{F}f]\|_{L^2(\mathbb{R}^n)}, \quad f \in \mathcal{H}^{-s}(\mathbb{R}^n). \tag{2.4}$$

It turns out that the solution to (2.1) with $d(f, g) = \|f - g\|_{\mathcal{H}^{-s}(\mathbb{R}^n)}$ can be expressed using a weighted projection formula. To see this, note that (2.1) becomes

$$\gamma^{(k)} = \operatorname{argmin}_{v \in \mathbb{C}^{m_k}} \|f_k - v\|_{\mathcal{H}^{-s}(\mathbb{R}^n)} = \operatorname{argmin}_{v \in \mathbb{C}^{m_k}} \|\mathcal{P}(f_k - v)\|_{L^2(\mathbb{R}^n)}, \quad \mathcal{P}f = \mathcal{F}^{-1} [(1 + |\xi|^2)^{-s/2} \mathcal{F}f]$$

and hence, when the columns of D_k are linearly independent, we have

$$\gamma^{(k)} = (D_k^* \mathcal{P}^2 D_k)^{-1} D_k^* \mathcal{P}^2 f_k. \tag{2.5}$$

Here, we have $(D_k^* \mathcal{P}^2 D_k)_{ij} = \langle \mathcal{P} \Delta f_{k-m_k+i}, \mathcal{P} \Delta f_{k-m_k+j} \rangle$ for $0 \leq i, j \leq m_k - 1$ and $(D_k^* \mathcal{P}^2 f_k)_i = \langle \mathcal{P} \Delta f_{k-m_k+i}, \mathcal{P} f_k \rangle$ for $0 \leq i \leq m_k - 1$.

The Hilbert space $\mathcal{H}^{-s}(\Omega)$ for a bounded Lipschitz-smooth domain $\Omega \subseteq \mathbb{R}^n$ is the set of restrictions of functions from $\mathcal{H}^{-s}(\mathbb{R}^n)$ equipped with the norm

$$\|f\|_{\mathcal{H}^{-s}(\Omega)} = \inf \{ \|g\|_{\mathcal{H}^{-s}(\mathbb{R}^n)} : g \in \mathcal{H}^{-s}(\mathbb{R}^n), g|_{\Omega} = f \}.$$

An equivalent and more explicit definition of $\|f\|_{\mathcal{H}^{-s}(\Omega)}$ is given via the Laplacian operator [26, p. 586]. That is, when Ω is a bounded domain with infinitely differentiable boundary, we have

$$\|f\|_{\mathcal{H}^{-s}(\Omega)} = \|u\|_{\mathcal{H}^s(\Omega)}, \tag{2.6}$$

where u is the solution to $(\sum_{r=0}^s (-1)^r \nabla^{2r})u = f$ with u satisfying zero Neumann boundary conditions.

We can begin to appreciate the purpose of the \mathcal{H}^{-s} norm from (2.4) and (2.5). The norm $\|f\|_{\mathcal{H}^{-s}}$ weights the low-frequency spectral content of f more than the high-frequency content. Thus, $\gamma^{(k)}$ is focused on making the low-frequency spectral content of the residual smaller, which can potentially counterbalance the spectral biasing of a fixed-point operator. One can select any type of frequency biasing—towards the low- or high-frequency spectral content of the residual—by choosing $d(f, g) = \|f - g\|_{\mathcal{H}^{-s}}$ for $s \in \mathbb{R}$. We suspect that a good choice of s depends on the spectral biasing of the fixed-point operator (see Section 4.1). While we focus on the benefits of the \mathcal{H}^{-s} norm, the idea of acceleration through changing the distance function is more general. One can select the distance function a priori or even modify it dynamically as the iteration proceeds.

The choice of distance function in AA is similar to preconditioning in an iterative method. The work in this paper began with the idea that integration reformulation [27,28] or integral preconditioning [29] might be a way to speed up the convergence of AA for certain fixed-point operators. We suspect that there are possible connections of our work to operator preconditioning [30], continuous Krylov methods [31], and Riesz operators [32].

2.2. Discretizing Anderson acceleration based on the \mathcal{H}^{-s} norm

One must first discretize Algorithm 2 before running it on a computer. In principle, any reasonable discretization scheme is appropriate. In this paper, we discretize functions and operators with finite difference schemes so that the iterates $x_k \in \mathcal{X}$ in Algorithm 2 are replaced by vectors that sample x_k at equispaced points.

The most subtle quantity to discretize in Algorithm 2 is $d(f, g)$. For example, if $\Omega = (0, 1)$, then a discrete analogue of $d(f, g) = \|f - g\|_{\mathcal{H}^{-1}(\Omega)}$ is given by

$$d(v, w) = \sqrt{h} \|(I_n - B_n)^{-1/2}(v - w)\|_2, \quad B_n = \frac{1}{h^2} \begin{bmatrix} -1 & 1 & & & & \\ 1 & -2 & \ddots & & & \\ & \ddots & \ddots & & & \\ & & & 1 & -2 & 1 \\ & & & & 1 & -1 \end{bmatrix}, \tag{2.7}$$

where $h = 1/(n - 1)$. Here, I_n is the $n \times n$ identity matrix and B_n is the $n \times n$ second-order finite difference matrix for the Laplacian with zero Neumann conditions. From (2.6), we know that $(d(f, g))^2 = \|u\|_{\mathcal{H}^{-1}(\Omega)}^2 = \langle u, u \rangle + \langle \nabla u, \nabla u \rangle = \langle u, u - \nabla^2 u \rangle$. Therefore, we take the discrete analogue as $d(v, w) = \sqrt{h} \|(I_n - B_n)^{1/2}(I_n - B_n)^{-1}(v - w)\|_2 = \sqrt{h} \|(I_n - B_n)^{-1/2}(v - w)\|_2$. Here, the integral in the definition of the L^2 norm is discretized by a low-order Riemann-like sum.

We have selected the so-called half-sample discretization for $u'(0) = 0$ and $u'(1) = 0$ so that the matrix is symmetric [33, Sec. 3]. For $d(f, g) = \|f - g\|_{\mathcal{H}^{-2}(\Omega)}$, similar to (2.7), we take the discretization as

$$d(v, w) = \sqrt{h} \|(I_n - B_n + B_n^2)^{-1/2}(v - w)\|_2.$$

Therefore, once AA in an \mathcal{H}^{-s} norm is discretized for a nonzero s , it becomes AA in a weighted ℓ^2 norm.

2.3. The connection with the multisection method

A useful interpretation of AA for vectors is as a multisection method [18,34]. In particular, when $d(v, w) = \|v - w\|_2$, the update in (2.2) can be expressed in the following form:

$$x_{k+1} = x_k + (I - S_k)f_k,$$

where $S_k \in \mathbb{C}^{n \times n}$, x_k is the k th iterate from AA, and $f_k = G(x_k) - x_k$. It is shown in [18] that if one defines $\Delta x_i = x_{i+1} - x_i$, $\Delta f_i = f_{i+1} - f_i$, and

$$X_k = [\Delta x_{k-m_k}, \dots, \Delta x_{k-1}] \in \mathbb{C}^{n \times m_k}, \quad D_k = [\Delta f_{k-m_k}, \dots, \Delta f_{k-1}] \in \mathbb{C}^{n \times m_k}, \tag{2.8}$$

then S_k is the solution to the following constrained optimization problem:

$$\min_{S_k \in \mathbb{C}^{n \times n}} \|S_k\|_F, \quad \text{subject to } S_k D_k = X_k + D_k. \tag{2.9}$$

Here, $\|\cdot\|_F$ denotes the matrix Frobenius norm, i.e., $\|S_k\|_F^2 = \sum_{i,j=1}^n |(S_k)_{ij}|^2$. Furthermore, (2.9) has an explicit solution given by

$$S_k = (X_k + D_k) (D_k^* D_k)^{-1} D_k^*,$$

when the columns of D_k are linearly independent [18,19].

The \mathcal{H}^{-s} norm can be discretized to a weighted ℓ^2 norm when s is an integer, i.e., $d(v, w) = \|P(v - w)\|_2$, where P is a symmetric positive definite matrix. For example, $P = (I_n - B_n)^{-1/2}$ for \mathcal{H}^{-1} norm and $P = (I_n - B_n + B_n^2)^{-1/2}$ for \mathcal{H}^{-2} norm. Similar to the ℓ^2 norm, one can also write AA based on the \mathcal{H}^{-s} norm as a multiseccant method. From a weighted projection formula, we find that

$$x_{k+1} = x_k + (I - \tilde{S}_k)f_k,$$

where \tilde{S}_k solves the following constrained optimization problem

$$\min_{\tilde{S}_k} \|P\tilde{S}_kP^{-1}\|_F, \quad \text{subject to } \tilde{S}_kD_k = X_k + D_k. \tag{2.10}$$

Furthermore, (2.10) has an explicit solution given by

$$\tilde{S}_k = (X_k + D_k) (D_k^*P^2D_k)^{-1} D_k^*P^2, \tag{2.11}$$

which is derived in Lemma 2.1.

Lemma 2.1. *Let P be a positive definite matrix and let $D_k, X_k \in \mathbb{C}^{n \times m_k}$ such that D_k has linearly independent columns. Then, the solution to*

$$\min_{S_k} \|S_k\|_F, \quad \text{subject to } S_kPD_k = PX_k + PD_k,$$

is $S_k = P\tilde{S}_kP^{-1}$, where $\tilde{S}_k = (X_k + D_k) (D_k^*P^2D_k)^{-1} D_k^*P^2$ is the solution to (2.10).

Proof. The solution to

$$\min_{S'_k} \|S'_k\|_F, \quad \text{subject to } S'_kD'_k = X'_k + D'_k,$$

is given by $S'_k = (X'_k + D'_k) ((D'_k)^*D'_k)^{-1} (D'_k)^*$ [19]. The statement of the lemma follows by setting $D'_k = PD_k$ and $X'_k = PX_k$. \square

The interpretation of AA based on the \mathcal{H}^{-s} norm as a multiseccant method is useful for understanding its convergence behavior in Section 3.

3. Error analysis of Anderson acceleration

In this section, we analyze AA based on the \mathcal{H}^{-2} norm. Suppose that we have the following fixed-point iteration

$$x_{k+1} = G(x_k) = Ax_k + b, \quad k \geq 0,$$

where A is an $n \times n$ real symmetric matrix and b is an $n \times 1$ vector. Let x^* denote a fixed-point, i.e., $G(x^*) = x^*$. The dependence between the error $e_k = x_k - x^*$ and the residual $f_k = G(x_k) - x_k$ in any two consecutive iterates can be written as

$$e_{k+1} = Ae_k, \quad f_{k+1} = Af_k, \quad k \geq 0.$$

Since A is a real symmetric matrix, it has an orthogonal eigendecomposition given by

$$A = W\Lambda W^*, \quad \Lambda = \text{diag}(\lambda_1, \dots, \lambda_n),$$

where W is an orthogonal matrix. After k Picard iterations, $e_k = A^k e_0 = W\Lambda^k W^* e_0$ for $k \geq 0$, where $e_0 = x_0 - x^*$ is the initial error. There is an extensive literature on the convergence of AA [1,3,15], but an explicit convergence rate that depends on m is missing. It is nontrivial to derive an explicit convergence rate as a function of m since in every iteration the weights in AA are derived from an optimization problem.

In this paper, we are particularly interested in the relation between the convergence of AA and the memory parameter as well as the choice of distance function in (2.1). The setting of our analysis is the following. We first apply Picard iteration for k iterations (where $k \geq m$), and then we perform one step of the AA algorithm with memory parameter m to obtain the $(k+1)$ th iterate. We analyze the solution error e_{k+1} after one step of AA and compare it to the solution error after $k+1$ Picard iterations. We refer to this as the *one-step* analysis of AA. The improvement in the solution error is the *one-step* acceleration of AA.

Alternating between Picard and AA is proposed as the alternating Anderson–Jacobi method in [34]. Although it is convenient for the analysis, we do not advocate using an alternating scheme in practice since, in our experience, applying AA at every iteration usually has an improved convergence behavior.

3.1. Error analysis of Anderson acceleration in the ℓ^2 norm

In Section 2.3, AA is viewed as a multisection-type method with iterates defined as

$$x_{k+1} = x_k + (I - S_k)f_k, \quad S_k = (D_k + X_k)(D_k^* D_k)^{-1} D_k^*, \tag{3.1}$$

where the matrices D_k and X_k are defined in (2.8) and $f_k = G(x_k) - x_k$. We note that Picard iteration takes S_k to be the zero matrix, and we hope that (3.1) promotes faster convergence to a fixed-point of G . The error and residual between any two consecutive iterates satisfy the following recurrence:

$$e_{k+1} = (I - (I - S_k)(I - A)) e_k, \quad f_{k+1} = (I - (I - A)(I - S_k)) f_k. \tag{3.2}$$

The matrix S_k in (3.2) depends on the fixed-point operator as well as the previous m_k iterates and residuals. Therefore, we find it difficult to imagine a full and explicit convergence analysis of AA for general m . Instead, we analyze the acceleration effect when one runs Picard iteration for the first k iterations (where $k \geq m$) and then performs one step of AA to obtain the $(k + 1)$ th iterate. We start our error analysis by expressing the solution error explicitly in terms of a Krylov matrix. For an $n \times n$ matrix A and an $n \times 1$ vector b , a Krylov matrix is defined as

$$K_m(A, b) = [b \quad Ab \quad \dots \quad A^{m-1}b] \in \mathbb{C}^{n \times m}.$$

We have the following lemma that relates the error after doing Picard for $k + 1$ steps, denoted by Ae_k , to the error after doing Picard for k steps and then one step of AA with memory parameter m , denoted by e_{k+1} .

Lemma 3.1. *Let A be an $n \times n$ real symmetric matrix with eigenvalue decomposition $A = W \Lambda W^*$ and b be an $n \times 1$ vector. Suppose that $x_{j+1} = Ax_j + b$ for $0 \leq j \leq k - 1$, 0 and 1 are not eigenvalues of A , and that x_{k+1} is produced from AA based on the discrete ℓ^2 norm with memory parameter $1 \leq m \leq n$. Then, when K_H has linearly independent columns, we have*

$$e_{k+1} = WEW^* Ae_k, \quad E = D_\mu \left(I - K_H (K_H^* K_H)^{-1} K_H^* \right) D_\mu^{-1}, \tag{3.3}$$

where $D_\mu = \Lambda(\Lambda - I)^{-1}$, $K_H = K_m(\Lambda, HW^* e_0)$, and $H = (\Lambda - I)^2 \Lambda^{k-m}$.

Proof. Note that $e_{j+1} - e_j = (A^{j+1} - A^j)e_0 = W \Lambda^j (\Lambda - I) W^* e_0$ and $f_{j+1} - f_j = (A - I)(e_{j+1} - e_j)$ for $0 \leq j \leq k - 1$. Thus, for D_k and X_k given in (2.8), we have

$$D_k = W \Lambda^{k-m} (\Lambda - I)^2 K_m(\Lambda, W^* e_0) \tag{3.4}$$

and $X_k = W \Lambda^{k-m} (\Lambda - I) K_m(\Lambda, W^* e_0)$. By substituting (3.4) into (3.1), we find that

$$S_k = W \Lambda (\Lambda - I)^{-1} K_H (K_H^* K_H)^{-1} K_H^*, \quad K_H = K_m(\Lambda, HW^* e_0), \tag{3.5}$$

where $H = (\Lambda - I)^2 \Lambda^{k-m}$. Furthermore, by substituting (3.5) into (3.2), we obtain the following:

$$e_{k+1} = W \Lambda (\Lambda - I)^{-1} \left(I - K_H (K_H^* K_H)^{-1} K_H^* \right) (\Lambda - I) W^* e_k.$$

The result follows by noting that $W^* e_k = W^* A^{-1} Ae_k = \Lambda^{-1} W^* Ae_k$. \square

The main observation from Lemma 3.1 is that E is a projection matrix. Furthermore, $D_\mu^{-1} E D_\mu$ is an orthogonal projection onto the space spanned by the column space of the Krylov matrix K_H . Since H is known explicitly, one can precisely quantify the difference between $\|D_\mu^{-1} W^* Ae_k\|_2$ and $\|D_\mu^{-1} W^* e_{k+1}\|_2$.

Theorem 3.1. *Under the same setup, notation, and assumptions of Lemma 3.1, suppose that the eigenvalues of A are contained in an interval $[a, b]$ that does not contain 0 or 1 . Then,*

$$\|D_\mu^{-1} W^* e_{k+1}\|_2 \leq C(a, b, m) \|D_\mu^{-1} W^* Ae_k\|_2, \quad C(a, b, m) = \left| T_m \left(\frac{2ab - a - b}{b - a} \right) \right|^{-1},$$

where $T_m(x)$ is the Chebyshev polynomial of degree m .

Proof. From Lemma 3.1 and since K_H is a Krylov matrix, we find that

$$\begin{aligned} \|D_\mu^{-1} W^* e_{k+1}\|_2 &= \|(I - K_H (K_H^* K_H)^{-1} K_H^*) D_\mu^{-1} W^* Ae_k\|_2 \\ &= \min_{c \in \mathbb{C}^m} \|D_\mu^{-1} W^* Ae_k - K_H c\|_2 \\ &= \min_{p \in \mathcal{P}_{m-1}} \|D_\mu^{-1} W^* Ae_k - p(\Lambda) HW^* e_0\|_2, \end{aligned}$$

where \mathcal{P}_{m-1} is the space of polynomials of degree $\leq m - 1$. Since $H = (\Lambda - I)^2 \Lambda^{k-m}$ and $D_\mu^{-1} = (\Lambda - I)\Lambda^{-1}$, we know that $HW^*e_0 = \Lambda^{k-m}(\Lambda - I)^2 W^*e_0 = \Lambda^{-m}(\Lambda - I)D_\mu^{-1}W^*Ae_k$. Therefore, we find that

$$\begin{aligned} \|D_\mu^{-1}W^*e_{k+1}\|_2 &\leq \min_{p \in \mathcal{P}_{m-1}} \|I - p(\Lambda)\Lambda^{-m}(\Lambda - I)\|_2 \|D_\mu^{-1}W^*Ae_k\|_2 \\ &\leq \min_{p \in \mathcal{P}_{m-1}} \max_{x \in [a,b]} |1 - p(x)x^{-m}(x - 1)| \|D_\mu^{-1}W^*Ae_k\|_2. \end{aligned}$$

We note that

$$\begin{aligned} \min_{p \in \mathcal{P}_{m-1}} \max_{x \in [a,b]} |1 - p(x)x^{-m}(x - 1)| &= \min_{p \in \mathcal{P}_{m-1}} \max_{x \in [\frac{1}{b}, \frac{1}{a}]} |1 - p(\frac{1}{x})x^{m-1}(1 - x)| \\ &= \min_{p \in \mathcal{P}_{m-1}} \max_{x \in [\frac{1}{b}, \frac{1}{a}]} |1 - p(x)(1 - x)| \\ &= \min_{q \in \mathcal{P}_m, q(1)=1} \max_{x \in [\frac{1}{b}, \frac{1}{a}]} |q(x)|. \end{aligned}$$

For any $|x_*| > 1$, we know that $|T_m(x_*)| \geq |p(x_*)|$ for any polynomial p of degree $\leq m$ such that $|p(x)| \leq 1$ for $x \in [-1, 1]$, where $T_m(x)$ is the Chebyshev polynomial of degree m [35]. Therefore, since $1 \notin [1/b, 1/a]$, we have

$$\min_{q \in \mathcal{P}_m, q(1)=1} \max_{x \in [\frac{1}{b}, \frac{1}{a}]} |q(x)| = \max_{x \in [\frac{1}{b}, \frac{1}{a}]} \left| \frac{T_m\left(\frac{2(x-\frac{1}{b})}{\frac{1}{a}-\frac{1}{b}} - 1\right)}{T_m\left(\frac{2(1-\frac{1}{b})}{\frac{1}{a}-\frac{1}{b}} - 1\right)} \right| \leq \left| T_m\left(\frac{2(1-\frac{1}{b})}{\frac{1}{a}-\frac{1}{b}} - 1\right) \right|^{-1},$$

where in the last inequality we used the fact that $|T_m(x)| \leq 1$ for $x \in [-1, 1]$. The result now follows as $2(1 - 1/b)/(1/a - 1/b) - 1 = (2ab - a - b)/(b - a)$. \square

The vector given by Ae_k is the solution error after $k + 1$ Picard iterations, while e_{k+1} is the solution error after k Picard iterations and then one step of AA. Therefore, Theorem 3.1 provides a bound on the acceleration effect by performing one-step AA. If $0 < a < b < 1$, the weighting matrix $D_\mu^{-1} = \Lambda(I - \Lambda)^{-1}$ enforces large weights on components of the error related to eigenvalues that are close to 1. Therefore, this particular weighting suggests that one-step AA is improving precisely the components of the residual that are making Picard iteration converge slowly.

The number $C(a, b, m)$ in Theorem 3.1 only depends on a, b , and m , where $[a, b]$ is an interval containing the eigenvalues of the fixed-point iteration matrix and m is the memory parameter in AA. For example, $C(0.3, 0.9, 10) \leq 0.024$ and $C(2, 100, 10) \leq 3.84 \times 10^{-8}$. For any interval $[a, b]$, not containing 0 and 1, the number $|(2ab - a - b)/(b - a)| > 1$, and hence $C(a, b, m)$ is a monotonically decreasing function of m (for fixed a and b). In fact, as a function of m , $C(a, b, m)$ decays exponentially to zero as $m \rightarrow \infty$.

3.2. Error analysis of Anderson acceleration in a weighted ℓ^2 norm

One can derive explicit formulas for AA when performed in a weighted ℓ^2 norm. That is, the distance function in (2.1) is $d(v, w) = \|P(v - w)\|_2$ for some positive definite matrix P . From (2.11), we find that

$$\tilde{S}_k = (X_k + D_k)(D_k^*P^2D_k)^{-1}D_k^*P^2.$$

Since $D_k = W\Lambda^{k-m}(\Lambda - I)^2K_m(\Lambda, W^*e_0)$ and $X_k = W\Lambda^{k-m}(\Lambda - I)K_m(\Lambda, W^*e_0)$, we have

$$\tilde{S}_k = JWK_H(K_H^*W^*P^2WK_H)^{-1}K_H^*W^*P^2, \tag{3.6}$$

where $H = (\Lambda - I)^2\Lambda^{k-m}$, $J = W\Lambda(\Lambda - I)^{-1}W^*$, and $K_H = K_m(\Lambda, HW^*e_0)$. Eq. (3.6) allows us to derive an analogous formula to (3.3) for AA in a weighted ℓ^2 norm.

Lemma 3.2. *Under the same setup, notation, and assumptions of Lemma 3.1, except that x_{k+1} is produced from AA with $d(v, w) = \|P(v - w)\|_2$ for some positive definite matrix. Then, we have*

$$e_{k+1} = W\tilde{E}W^*Ae_k, \quad \tilde{E} = D_\mu [I - K_H(K_H^*W^*P^2WK_H)^{-1}K_H^*W^*P^2W]D_\mu^{-1}.$$

Proof. The proof is essentially identical to the proof of Lemma 3.1. \square

Here, \tilde{E} is a projection matrix and hence we know that $\|e_{k+1}\|_2 \leq \|Ae_k\|_2$. However, now $D_\mu^{-1}\tilde{E}D_\mu$ is not an orthogonal projection (unless $P = I$). This makes it very difficult to do the analysis of one-step AA with a weighted ℓ^2 norm.

3.2.1. When P and A share the same eigenvectors

To make progress here, we make a strong assumption that the fixed-point operator and the matrix P share the same eigenvectors. That is, we suppose that $A = W\Lambda W^*$ and $P = W\Sigma W^*$ for the same orthogonal matrix W . Under this assumption, we can analyze the acceleration effect of one-step AA when performed with $d(v, w) = \|P(v - w)\|_2$. The following theorem is a generalization of [Theorem 3.1](#).

Theorem 3.2. Under the same setup, notation, and assumptions as [Lemma 3.2](#), with eigenvalue decompositions $A = W\Lambda W^*$ and $P = W\Sigma W^*$, we have

$$\frac{\|\Sigma D_\mu^{-1} W^* e_{k+1}\|_2}{\max_{1 \leq i \leq n} |\Sigma_{ii}|} \leq C(a, b, m) \|D_\mu^{-1} W^* A e_k\|_2, \quad C(a, b, m) = \left| T_m \left(\frac{2ab - a - b}{b - a} \right) \right|^{-1}.$$

Proof. Since $P = W\Sigma W^*$, that statement of [Lemma 3.2](#) becomes

$$e_{k+1} = W D_\mu [I - K_H (K_H^* \Sigma^2 K_H)^{-1} K_H^* \Sigma^2] D_\mu^{-1} W^* A e_k.$$

Therefore, we have

$$\begin{aligned} \|\Sigma D_\mu^{-1} W^* e_{k+1}\|_2 &= \|\Sigma (I - K_H (K_H^* \Sigma^2 K_H)^{-1} K_H^* \Sigma^2) D_\mu^{-1} W^* A e_k\|_2 \\ &= \min_{c \in \mathbb{C}^m} \|\Sigma (D_\mu^{-1} W^* A e_k - K_H c)\|_2 \\ &= \min_{p \in \mathcal{P}_{m-1}} \|\Sigma (D_\mu^{-1} W^* A e_k - p(\Lambda) H W^* e_0)\|_2, \end{aligned}$$

where the second equality follows from the formula for weighted projection [[21](#), Sec. 6.1.1]. Here, \mathcal{P}_{m-1} is the space of polynomials of degree $\leq m - 1$. Since $H W^* e_0 = \Lambda^{-m} (\Lambda - I) D_\mu^{-1} W^* A e_k$ (see the proof of [Theorem 3.1](#)), we have

$$\begin{aligned} \|\Sigma D_\mu^{-1} W^* e_{k+1}\|_2 &\leq \min_{p \in \mathcal{P}_{m-1}} \|\Sigma (I - p(\Lambda) \Lambda^{-m} (\Lambda - I))\|_2 \|D_\mu^{-1} W^* A e_k\|_2 \\ &\leq \max_{1 \leq i \leq n} |\Sigma_{ii}| \min_{p \in \mathcal{P}_{m-1}} \|(I - p(\Lambda) \Lambda^{-m} (\Lambda - I))\|_2 \|D_\mu^{-1} W^* A e_k\|_2 \\ &\leq \max_{1 \leq i \leq n} |\Sigma_{ii}| \min_{p \in \mathcal{P}_{m-1}} \max_{x \in [a, b]} |(I - p(x) x^{-m} (x - I))| \|\Sigma\|_2 \|D_\mu^{-1} W^* A e_k\|_2. \end{aligned}$$

The result follows as the polynomial optimization problem is identical to the one in the proof of [Theorem 3.1](#). \square

To get a sense of [Theorem 3.2](#), suppose that $\Sigma_{ij} = 1/j^2$. Then, $\max_{1 \leq i \leq n} |\Sigma_{ii}| = 1$ so that the inequalities in [Theorems 3.1](#) and [3.2](#) are almost identical. The only difference is that [Theorem 3.2](#) is bounding $\|\Sigma D_\mu^{-1} W^* e_{k+1}\|_2$, not $\|D_\mu^{-1} W^* e_{k+1}\|_2$. This means that AA in the weighted ℓ^2 norm is penalizing the first entry of $D_\mu^{-1} W^* e_{k+1}$ more than the last entry. Since $\Sigma D_\mu^{-1} W^* e_{k+1}$ contains the term $W^* e_{k+1}$, one can view this as biasing towards certain spectral content of A .

4. Numerical experiments

In this section, we present numerical experiments to demonstrate the acceleration effects of AA based on the \mathcal{H}^{-2} norm. We show this with both contractive and noncontractive fixed-point operators that involve second-order differential operators.

4.1. A contractive operator for solving Poisson's equation

Our first example illustrates the theorems in Section 3 and the convergence behavior of AA. For this example, we recommend solving (4.1) using direct methods as the linear system is a tridiagonal Toeplitz matrix. We only use this example to illustrate our theorems.

Consider 1D Poisson's equation with zero Dirichlet boundary conditions on $(0, 1)$, i.e.,

$$-u''(x) = f(x), \quad u(0) = u(1) = 0. \tag{4.1}$$

We discretize (4.1) by using a second-order finite difference scheme to obtain the $n \times n$ linear system

$$\frac{1}{h^2} \underbrace{\begin{bmatrix} -2 & 1 & & & \\ 1 & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & & 1 & -2 \end{bmatrix}}_{=M} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \underbrace{\begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{bmatrix}}_{=b}, \quad x_j = jh, \quad h = \frac{1}{n+1}. \tag{4.2}$$

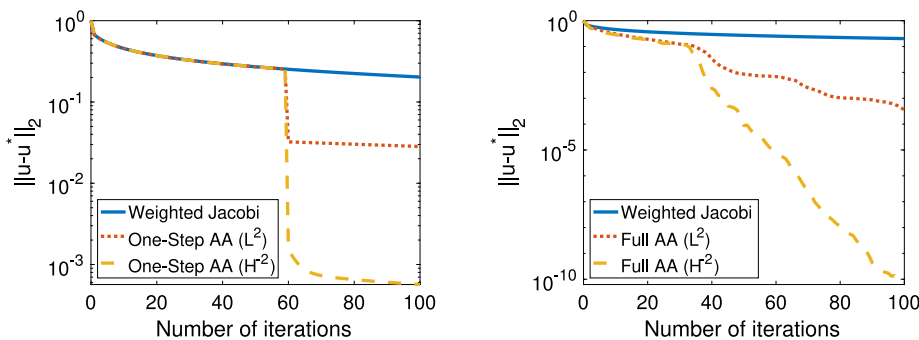


Fig. 1. Solving Poisson’s equation in (4.1) with the weighted Jacobi method, AA based on the L^2 norm, and AA based on the \mathcal{H}^{-2} norm. Left: The weighted Jacobi method compared to one-step AA. Right: The weighted Jacobi method compared to full AA.

As our fixed-point iteration, we consider the weighted Jacobi method given by

$$x_{n+1} = G(x_n) = \left(I - \frac{2}{3}D_M^{-1}M\right)x_n + \frac{2}{3}D_M^{-1}b, \quad D_M = \text{diag}(M).$$

Here, G is a contractive operator because $A = I - \frac{2}{3}D_M^{-1}M$ has eigenvalues

$$\lambda_j(A) = \frac{1}{3} + \frac{2}{3} \cos\left(\frac{j\pi}{n+1}\right), \quad 1 \leq j \leq n,$$

which satisfy $|\lambda_j(A)| < 1$ for $1 \leq j \leq n$. For each j , the eigenvector corresponding to $\lambda_j(A)$ is also known in closed form as

$$(v_j)_i = \sin\left(\frac{ij\pi}{n+1}\right), \quad 1 \leq i \leq n,$$

and the eigenvector components of the residual corresponding to $\lambda_j(A)$ have size $\mathcal{O}(|\lambda_j(A)|^k)$ after k iterations [36, Chapter 2]. If $e_0 = \sum_{j=1}^n c_j v_j$ is the initial error in the weighted Jacobi method, then

$$e_k = A^k e_0 = \sum_{j=1}^n c_j \lambda_j(A)^k v_j, \quad f_k = (I - A)e_k = \frac{4}{3} \sum_{j=1}^n \sin^2\left(\frac{j\pi}{2(n+1)}\right) c_j \lambda_j(A)^k v_j.$$

Thus, one can see a spectral biasing in the weighted Jacobi method: after a few iterations, $\|f_k\|_2$ might be small while $\|e_k\|_2$ is not (due to the $\sin^2((j\pi)/(2(n+1)))$ term). In particular, the eigenvector components associated with large j are more heavily weighted in f_k than in e_k . In Fig. 1, we show the convergence of the weighted Jacobi method and illustrate its poor convergence.

A natural idea is to use the choice of distance function in AA to counterbalance the spectral biasing in the weighted Jacobi method. In this case, the \mathcal{H}^{-2} norm is a good choice because the eigenvalues of $P = (I_n - B_n + B_n^2)^{-1/2}$ (see Section 2.2) are given by

$$\lambda_j(P) = \left(1 + \frac{4}{h^2} \sin^2\left(\frac{\pi(j-1)}{2n}\right) + \frac{16}{h^4} \sin^4\left(\frac{\pi(j-1)}{2n}\right)\right)^{-1/2}.$$

Therefore, the spectral biasing in the \mathcal{H}^{-2} norm approximately counterbalances the spectral biasing in the residual. This is only heuristic reasoning because the eigenvectors of P and A are not the same. Still, in practice, we observe that AA in the \mathcal{H}^{-2} norm converges rapidly.

In Fig. 1, we compare the convergence of the weighted Jacobi method to one-step AA as well as the full AA algorithm. For these tests we use $n = 63$, the memory parameter $m = 10$, and use the initial solution error of

$$(e_0)_j = \sum_{i=1}^{20} \sin(2ij\pi), \quad 1 \leq j \leq n.$$

As can be seen in the figures, applying AA successively at every iteration is preferred over one-step AA.

4.2. Noncontractive operator for solving Poisson’s equation

One can also attempt to solve (4.1) using Richardson iteration. That is,

$$x_{n+1} = G_R(x_n) = (I - M)x_n + b.$$

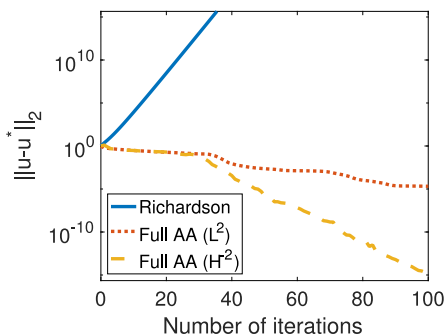


Fig. 2. Solving Poisson’s equation in (4.1) with Richardson iteration, AA based on the L^2 norm, and AA based on the \mathcal{H}^{-2} norm. This experiment highlights that AA can be used for noncontractive fixed-point operators.

Now, the fixed-point operator, G_R , is noncontractive as there are eigenvalues of $I - M$ larger than one in absolute value. Generally speaking, the Richardson iteration computes a divergent sequence. Nonetheless, to illustrate the surprising acceleration effects of AA, we repeat the experiment from Section 4.1 with the weighted Jacobi method replaced by Richardson iteration (see Fig. 2).

4.3. Nonlinear Helmholtz equation

AA is highly useful for nonlinear fixed-point operators. In this example, we present a nonlinear fixed-point operator designed to find $u : [0, 1] \rightarrow \mathbb{C}$, which solves the following 1D nonlinear Helmholtz equation [1]

$$\begin{aligned} \frac{d^2u}{dx^2} + k_0^2(1 + \epsilon(x)|u|^2)u &= 0, \quad 0 < x < 1, \\ \frac{du}{dx} + ik_0u &= 2ik_0, \quad x = 0, \\ \frac{du}{dx} - ik_0u &= 0, \quad x = 1. \end{aligned} \tag{4.3}$$

The nonlinear Helmholtz equation governs the propagation of linearly-polarized, time-harmonic electromagnetic waves in Kerr-type dielectrics [37]. We set $\epsilon(x)$ to be a piecewise constant function on $[0, 1]$, which approximates a realistic graded Kerr medium [37, p. 3]:

$$\epsilon(x) = \begin{cases} 0, & 0 \leq x \leq 0.1, \\ 1, & 0.1 < x \leq 0.2, \\ 2, & 0.2 < x \leq 0.3, \\ 3, & 0.3 < x \leq 0.7, \\ 4, & x > 0.7. \end{cases}$$

In the numerical tests, the system (4.3) is discretized by the same second-order finite difference method as described in [1, Sec. 6.2]. The resulting iterative scheme can be seen as a fixed-point operator: $u_{k+1} = G_{\text{NHL}}(u_k)$. Following [1], we set the initial guess to be $u_0 = e^{ik_0x}$, where k_0 is the linear wavenumber and x is the discretized interval $[0, 1]$ with grid spacing $h = 0.002$.

Fig. 3 shows the numerical results of solving the nonlinear Helmholtz equation with $k_0 = 20$ (left) and $k_0 = 70$ (right), respectively. In both cases, the Picard iteration fails to converge, and the residual remains constant for 100 iterations, while AA in the L^2 , \mathcal{H}^{-1} , and \mathcal{H}^{-2} norm all decrease the residual rapidly. As the wavenumber k_0 increases, the nonlinear Helmholtz problem becomes more challenging, and \mathcal{H}^{-2} norm becomes more beneficial.

Although (4.3) is nonlinear, there is spectral biasing from the second-order spatial derivative. Thus, one expects that the \mathcal{H}^{-2} norm could be counterbalancing the spectral bias of the fixed-point operator in (4.3). Nonetheless, the spectral properties might change drastically with different $\epsilon(x)$, so one must be careful. Also, we note that the convergence behavior highly depends on the initial guess u_0 . We also observe interesting convergence behavior as m increases. In particular, there seems to be essentially no benefit in taking large m , which we believe could be related to the nonlinearity in (4.3).

4.4. Solving the Helmholtz equation using the WaveHoltz iteration

In this experiment, we explore AA for the WaveHoltz iteration [38]. In the WaveHoltz iteration, we consider the Helmholtz equation in a bounded open Lipschitz domain Ω , i.e.,

$$\nabla \cdot (c^2(x)\nabla u) + \omega^2u = f(x), \quad x \in \Omega, \tag{4.4}$$

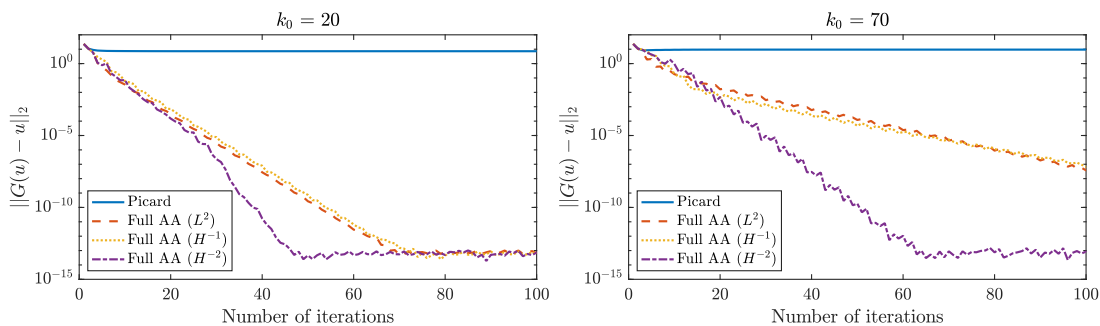


Fig. 3. Solving the nonlinear Helmholtz equation by Picard iteration, AA with $m = 1$ using the L^2 norm, the \mathcal{H}^{-1} norm, and the \mathcal{H}^{-2} norm with $k_0 = 20$ (left) and $k_0 = 70$ (right). The residuals are presented for the first 100 iterations.

together with the energy conserving homogeneous Dirichlet boundary conditions. That is,

$$i\omega u = 0, \quad x \in \partial\Omega. \tag{4.5}$$

As a result, the solution to (4.4) with the boundary condition (4.5) is a real-valued function. To find a solution to (4.4)–(4.5), we use the fixed-point iteration given by

$$u^{(M+1)} = \Pi u^{(M)}, \quad u^{(0)} \equiv 0,$$

where (see [38])

$$\Pi u = \frac{2}{T} \int_0^T \left(\cos(\omega t) - \frac{1}{4} \right) w(t, x) dt, \quad T = \frac{2\pi}{\omega}. \tag{4.6}$$

Here, $w(t, x)$ depends on $u(x)$ via the following wave equation:

$$\begin{aligned} w_{tt} - \nabla \cdot (c(x)^2 \nabla w) &= f(x) \cos(\omega t), \quad x \in \Omega, \quad 0 \leq t \leq T, \\ w(0, x) &= u(x), \quad w_t(0, x) = 0, \quad x \in \Omega, \\ w(t, x) &= 0, \quad x \in \partial\Omega, \quad 0 \leq t \leq T. \end{aligned} \tag{4.7}$$

First, we take $c(x)$ to be a variable wave speed and $\Omega = [0, 1]$. We discretize the problem with an equispaced grid $x_i = ih$ for $0 \leq i \leq n$ and $h = 1/(n + 1)$. We approximate the wave equation in (4.7) using a second-order finite difference scheme in space and time. If one takes $w_i^k \approx w(t_k, x_i)$ and $u_i^{(M)} \approx u^{(M)}(x_i)$, then we have the following discretization:

$$w_i^{-1} = u_i^{(M)} + \frac{\Delta t^2}{2} \left(\frac{(c_{i+1} + c_i)u_{i+1}^{(M)} - (c_{i+1} + 2c_i + c_{i-1})u_i^{(M)} + (c_i + c_{i-1})u_{i-1}^{(M)}}{2h^2} - f(x_i) \right), \quad \forall i,$$

$$w_i^0 = u_i^{(M)}, \quad \forall i,$$

$$\begin{aligned} \frac{w_i^{k+1} - 2w_i^k + w_i^{k-1}}{\Delta t^2} &= \frac{(c_{i+1} + c_i)w_{i+1}^k - (c_{i+1} + 2c_i + c_{i-1})w_i^k + (c_i + c_{i-1})w_{i-1}^k}{2h^2} \\ &\quad - \cos(\omega t_k) f(x_i), \quad i = 1, \dots, n - 1, \quad k \geq 0. \\ w_i^k &= 0, \quad i = 0, \quad i = n, \quad \forall k. \end{aligned}$$

Here, the integral in the projection (4.6) is discretized by the trapezoidal rule, and $u^{(M)}$ is the current iterate to the discretized WaveHoltz iteration. Since this is a linear problem, the discretized solution to the Helmholtz equation (4.4) can be shown to solve a linear system, i.e., $Au = b$.

Denoting the eigenpairs of the operator $u \mapsto -\nabla \cdot c(x)^2 \nabla u$ by $(\lambda_i, \phi_i(x))$, the eigenvalues of the fixed-point iteration satisfy the implicit relationship

$$\beta(\lambda_i) = \frac{2}{T} \int_0^T \left(\cos(\omega t) - \frac{1}{4} \right) \cos(\lambda_i t) dt,$$

where $\beta(\lambda)$ is a function such that $\beta(\lambda) \in [-1/2, 1)$. For the discretized problem one can show that A has eigenvalues in the interval $(0, 3/2]$ and shares its eigenvectors with the tridiagonal matrix in (4.2).

In our numerical experiments, we use $n = 513$ and set $\omega = 25\sqrt{2}$. The forcing term is zero everywhere, except at the 128th gridpoint. At the 128th gridpoint, we set it to be a constant scaled so that the numerical solution is ≈ 2 in magnitude. The solutions are displayed in Fig. 4.

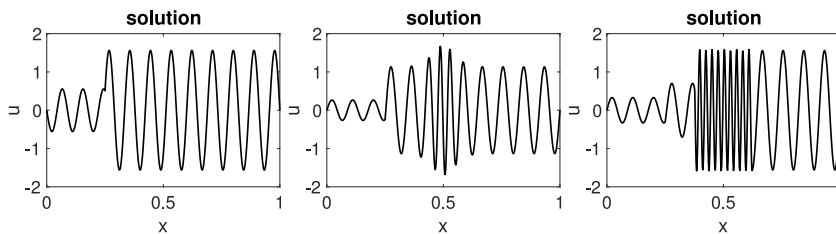


Fig. 4. Solutions for the three different wave speeds c_a (left), c_b (middle) and c_c (right); see (4.8).

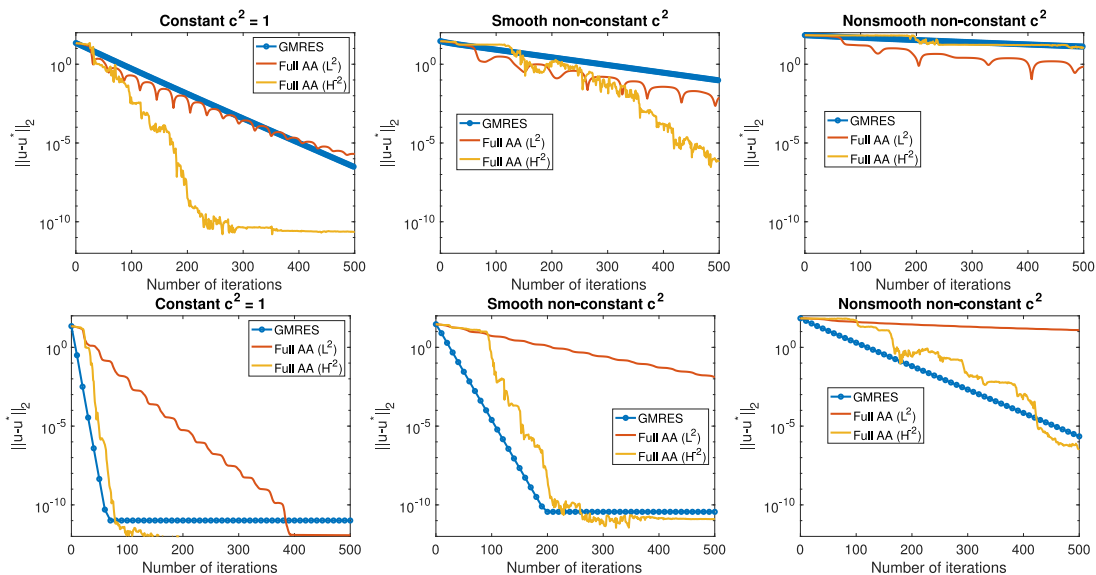


Fig. 5. Comparison of AA in the L^2 and \mathcal{H}^{-2} norm with GMRES. Top row ($m = 3$) and bottom row ($m = 10$). Left column, wave speed c_a , middle column, wave speed c_b , and bottom column, wave speed $c_c(x)$.

To compare AA in the L^2 norm with AA in the \mathcal{H}^{-2} norm, we consider three different wave speeds

$$c_a = 1, \quad c_b = 1 - 0.55e^{-144(x-0.5)^2}, \quad c_c = \begin{cases} 1 & |x - 0.5| > 0.125, \\ 0.3 & |x - 0.5| < 0.125. \end{cases} \tag{4.8}$$

We supply AA with the fixed-point operator associated to $Au = b$ obtained by performing Richardson iteration. We also compare with restarted GMRES, which restarts every m iterations (see Fig. 5). We find that AA in the \mathcal{H}^{-2} norm outperforms GMRES as well as AA in the L^2 norm for all cases except when the wave speed is c_c and $m = 3$. In this exceptional case, none of the methods manage to decrease the error substantially within 500 iterations.

The theory for the WaveHoltz iteration predicts that components of the solution that correspond to eigenpairs with eigenvalues closest to the Helmholtz frequency, ω , have the slowest converge rate. When the numerical solution has a small discretization error, these modes are typically well-resolved, and the spectral biasing due to the \mathcal{H}^{-2} norm appears to improve the convergence of AA. To compare the \mathcal{H}^{-1} and \mathcal{H}^{-2} norm more closely in AA, we compare the acceleration methods for $m = 3$ and $m = 10$ with the three different wave speeds (see Fig. 6). For small m , AA in the \mathcal{H}^{-2} norm converges faster than in the \mathcal{H}^{-1} norm.

4.4.1. WaveHoltz iteration in two dimensions

Next we consider an example in two dimensions with a wave speed given by

$$c^2(x, y) = \begin{cases} 0.3, & 0.4 \leq x \leq 0.6 \text{ or } 0.4 \leq y \leq 0.6, \\ 1, & \text{otherwise,} \end{cases}$$

on $\Omega = [0, 1]^2$. We, again, use second-order finite difference discretization with an 65×65 equispaced grid. We set the angular frequency to be $\omega = 11$. The forcing term is zero everywhere except at the gridpoint closest to $(0.25, 0.75)$ where it has an amplitude adjusted so that the solution is around 1 in magnitude.

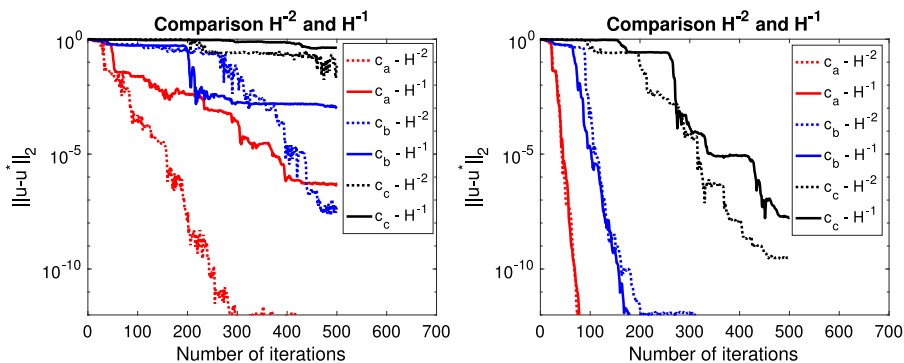


Fig. 6. Comparison of AA in the \mathcal{H}^{-1} norm and the \mathcal{H}^{-2} norm for $m = 3$ (left) and $m = 10$ (right).

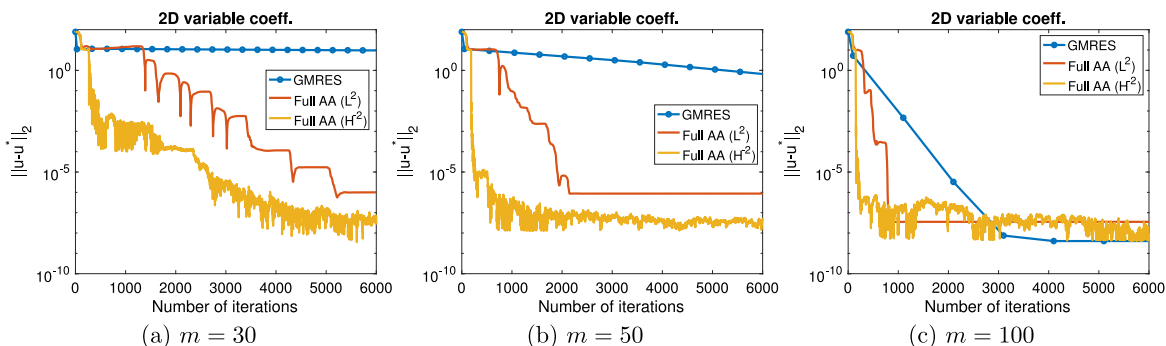


Fig. 7. Convergence histories for the two-dimensional Helmholtz example with (a) $m = 30$, (b) $m = 50$ and (c) $m = 100$.

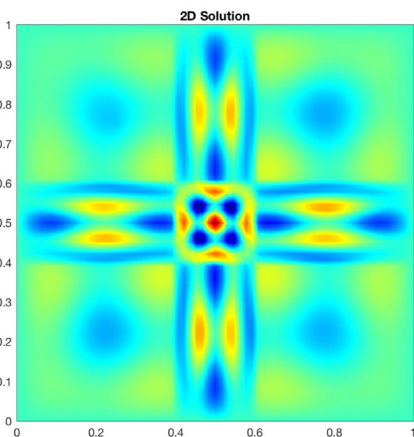


Fig. 8. The solution for the Helmholtz equation in two dimensions.

We employ Richardson iteration to generate a fixed-point operator and then use AA in the L^2 and \mathcal{H}^{-2} norm. Note that the degrees of freedom of the solution in this example are ordered as a long vector with lexicographical ordering and that we employ the one dimensional \mathcal{H}^{-2} norm to the two-dimensional data. We compare against restarted GMRES with restarts every m th iteration for $m = 30$, $m = 50$ and $m = 100$ (see Fig. 7). We observe that AA in the \mathcal{H}^{-2} norm outperforms restarted GMRES as well as AA in the L^2 norm. In this 2D setting, AA in the L^2 norm is also performing better than GMRES. Not surprisingly, the advantage of the “sliding memory” of AA is reduced as the restart depth increases. The solution to the problem is displayed in Fig. 8.

This experiment is encouraging, as the benefits of the \mathcal{H}^{-2} norm persist. Even though it might be possible to use a two-dimensional definition of the \mathcal{H}^{-2} for this simple geometry, its computation can be more costly and, in the case of complex geometry, quite cumbersome to compute.

5. Conclusion

In this paper, we propose the idea of using Anderson acceleration based on the \mathcal{H}^{-s} Sobolev norm. We observe that this can counterbalance the implicit spectral biasing in certain fixed-point operators. We rigorously analyze the convergence behavior of one-step AA, providing an explicit error bound using Chebyshev polynomials that decreases exponentially in the memory parameter m . Numerical experiments for both contractive, noncontractive, and non-linear operators demonstrate the acceleration effects of AA based on different norms. In practice, the choice of distance function in AA should be selected depending on the spectral biasing of the fixed-point operator.

Acknowledgments

This material is based upon work supported by the National Science Foundation, USA under Grant No. DMS-1818757, DMS-1913076 and DMS-1913129.

References

- [1] S. Pollock, L.G. Rebholz, Anderson acceleration for contractive and noncontractive operators, *IMA J. Numer. Anal.* (2021).
- [2] D.G. Anderson, Iterative procedures for nonlinear integral equations, *J. ACM* 12 (4) (1965) 547–560.
- [3] S. Pollock, L.G. Rebholz, M. Xiao, Anderson-accelerated convergence of Picard iterations for incompressible Navier–Stokes equations, *SIAM J. Numer. Anal.* 57 (2) (2019) 615–637.
- [4] H. An, X. Jia, H.F. Walker, Anderson acceleration and application to the three-temperature energy equations, *J. Comput. Phys.* 347 (2017) 1–19.
- [5] Y. Peng, B. Deng, J. Zhang, F. Geng, W. Qin, L. Liu, Anderson acceleration for geometry optimization and physics simulation, *ACM Trans. Graph.* 37 (4) (2018) 42.
- [6] A. Fu, J. Zhang, S. Boyd, Anderson accelerated Douglas–Rachford splitting, *SIAM J. Sci. Comput.* 42 (6) (2020) A3560–A3583.
- [7] V.V. Mai, M. Johansson, Anderson acceleration of proximal gradient methods, in: 37th Int. Conf. Mach. Learn. ICML, 2020, pp. 6620–6629.
- [8] Z. Li, J. Li, A fast Anderson–Chebyshev mixing method for nonlinear optimization, in: Inter. Conf. Art. Intel. Stats., 2020, pp. 1047–1057.
- [9] P. Pulay, Convergence acceleration of iterative sequences. The case of SCF iteration, *Chem. Phys. Lett.* 73 (2) (1980) 393–398.
- [10] K.N. Kudin, G.E. Scuseria, E. Cancès, A black-box self-consistent field convergence algorithm: One step closer, *J. Chem. Phys.* 116 (19) (2002) 8255–8261.
- [11] T. Rohwedder, R. Schneider, An analysis for the DIIS acceleration method used in quantum chemistry calculations, *J. Math. Chem.* 49 (9) (2011) 1889.
- [12] H. Cenicerós, G. Fredrickson, Numerical solution of polymer self-consistent field theory, *Multiscale Model. Simul.* 2 (3) (2004) 452–474.
- [13] F. Drolet, G.H. Fredrickson, Combinatorial screening of complex block copolymer assembly with self-consistent field theory, *Phys. Rev. Lett.* 83 (1999) 4317–4320.
- [14] H.F. Walker, P. Ni, Anderson acceleration for fixed-point iterations, *SIAM J. Numer. Anal.* 49 (4) (2011) 1715–1735.
- [15] A. Toth, C.T. Kelley, Convergence analysis for Anderson acceleration, *SIAM J. Numer. Anal.* 53 (2) (2015) 805–819.
- [16] C. Evans, S. Pollock, L.G. Rebholz, M. Xiao, A proof that Anderson acceleration increases the convergence rate in linearly converging fixed point methods (but not in quadratically converging ones), *SIAM J. Numer. Anal.* 58 (1) (2020) 788–810.
- [17] J. Zhang, B. O’Donoghue, S. Boyd, Globally convergent type-I Anderson acceleration for non-smooth fixed-point iterations, *SIAM J. Optim.* 30 (4) (2020) 3170–3197.
- [18] H.R. Fang, Y. Saad, Two classes of multisecond methods for nonlinear acceleration, *Numer. Linear Algebra Appl.* 16 (3) (2009) 197–221.
- [19] L. Lin, C. Yang, Elliptic preconditioner for accelerating the self-consistent field iteration in Kohn–Sham density functional theory, *SIAM J. Sci. Comput.* 35 (5) (2013) S277–S298.
- [20] E.J. Weniger, Nonlinear sequence transformations: Computational tools for the acceleration of convergence and the summation of divergent series, 2001, arXiv:math/0107080v1.
- [21] G.H. Golub, C.F. Van Loan, *Matrix Computations*, Johns Hopkins Univ. Press, 1996.
- [22] C.T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, Vol. 16, SIAM, 1995.
- [23] C.T. Kelley, Numerical methods for nonlinear equations, *Acta Numer.* 27 (2018) 207–287.
- [24] L.N. Trefethen, Householder triangularization of a quasimatrix, *IMA J. Numer. Anal.* 30 (4) (2009) 887–897.
- [25] G. Leoni, *A First Course in Sobolev Spaces*, Amer. Math. Soc., 2017.
- [26] M. Schechter, Negative norms and boundary problems, *Ann. Math.* 72 (1960) 581–593.
- [27] T.A. Driscoll, Automatic spectral collocation for integral, integro-differential, and integrally reformulated differential equations, *J. Comput. Phys.* 229 (17) (2010) 5980–5998.
- [28] L. Greengard, Spectral integration and two-point boundary value problems, *SIAM J. Numer. Anal.* 28 (4) (1991) 1071–1080.
- [29] J.S. Hesthaven, Integration preconditioning of pseudospectral operators. I. Basic linear operators, *SIAM J. Numer. Anal.* 35 (4) (1998) 1571–1593.
- [30] R. Hiptmair, Operator preconditioning, *Comput. Math. Appl.* 52 (5) (2006) 699–706.
- [31] M.A. Gilles, A. Townsend, Continuous analogues of Krylov subspace methods for differential operators, *SIAM J. Numer. Anal.* 57 (2) (2019) 899–924.
- [32] J. Málek, Z. Strakos, *Preconditioning and the Conjugate Gradient Method in the Context of Solving PDEs*, SIAM, 2014.
- [33] G. Strang, The discrete cosine transform, *SIAM Rev.* 41 (1) (1999) 135–147.
- [34] P.P. Pratapa, P. Suryanarayana, J.E. Pask, Anderson acceleration of the Jacobi iterative method: An efficient alternative to krylov methods for large, sparse linear systems, *J. Comput. Phys.* 306 (2016) 43–54.
- [35] L.N. Trefethen, D. Bau III, *Numerical Linear Algebra*, Vol. 50, SIAM, 1997.
- [36] W.L. Briggs, V.E. Henson, S.F. McCormick, *A Multigrid Tutorial*, Vol. 72, SIAM, 2000.
- [37] G. Baruch, G. Fibich, S. Tsynkov, High-order numerical method for the nonlinear Helmholtz equation with material discontinuities in one space dimension, *J. Comput. Phys.* 227 (1) (2007) 820–850.
- [38] D. Appelö, F. Garcia, O. Runborg, WaveHoltz: Iterative solution of the Helmholtz equation via the wave equation, *SIAM J. Sci. Comput.* 42 (4) (2020) A1950–A1983.