Journal of Scientific Computing (2023) 96:5
https://doi.org/10.1007/s10915-023-02231-4

Nonmonotone Globalization for Anderson Acceleration via Adaptive Regularization

Wenqing Ouyang[1,2,3] · Jiong Tao[4] · Andre Milzarek[1,2] · Bailin Deng[5]

Received: 22 February 2022 / Revised: 18 December 2022 / Accepted: 29 April 2023

Abstract
Anderson acceleration (AA) is a popular method for accelerating fixed-point iterations, but may suffer from instability and stagnation. We propose a globalization method for AA to improve stability and achieve unified global and local convergence. Unlike existing AA globalization approaches that rely on safeguarding operations and might hinder fast local convergence, we adopt a nonmonotone trust-region framework and introduce an adaptive quadratic regularization together with a tailored acceptance mechanism. We prove global convergence and show that our algorithm attains the same local convergence as AA under appropriate assumptions. The effectiveness of our method is demonstrated in several numerical experiments.

Keywords Anderson acceleration · Global convergence · Nonmonotone trust region · Adaptive regularization

Mathematics Subject Classification 65B05 · 65K10

1 Introduction

In applied mathematics, many problems can be reduced to solving a nonlinear fixed-point equation $g(x) = x$, where $x \in \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^n$ is a given function. If g is a contractive mapping, i.e.,

$$\|g(x) - g(y)\| \le \kappa \|x - y\| \quad \forall\, x, y \in \mathbb{R}^n, \tag{1}$$

✉ Bailin Deng
DengB3@cardiff.ac.uk

[1] School of Data Science, The Chinese University of Hong Kong, Shenzhen, China

[2] Shenzhen Research Institute of Big Data, Shenzhen, China

[3] Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen, China

[4] Department of Computer Science, University of Bath, Bath, UK

[5] School of Computer Science and Informatics, Cardiff University, Cardiff, UK

Published online: 18 May 2023

where $\kappa < 1$, then the iteration

$$x^{k+1} = g(x^k)$$

is ensured to converge to the fixed-point of g by Banach's fixed-point theorem. Anderson acceleration (AA) [2, 3, 49] is a technique for speeding up the convergence of such an iterative process. Instead of using the update $x^{k+1} = g(x^k)$, it generates x^{k+1} as an affine combination of the latest $m + 1$ steps:

$$x^{k+1} = g(x^k) + \sum\nolimits_{i=1}^{m} \alpha_i^* (g(x^{k-i}) - g(x^k)) \tag{2}$$

with the combination coefficients $\alpha^* = (\alpha_1^*, \ldots, \alpha_m^*) \in \mathbb{R}^m$ being computed via an optimization problem

$$\min_{\alpha} \left\| f(x^k) + \sum\nolimits_{i=1}^{m} \alpha_i (f(x^{k-i}) - f(x^k)) \right\|^2, \tag{3}$$

where $f(x^k) = g(x^k) - x^k$ denotes the *residual function*.

AA was initially proposed to solve integral equations [2] and has gained popularity in recent years for accelerating fixed-point iterations [49]. Examples include tensor decomposition [42], linear system solving [35], and reinforcement learning [18], among many others [1, 7, 24, 25, 27, 29, 30, 33, 52, 54].

On the theoretical side, it has been shown that AA is a quasi-Newton method for finding a root of the residual function [14, 16, 38]. When applied to linear problems (i.e., if $g(x) = Ax - b$), AA is equivalent to the generalized minimal residual method (GMRES), [34]. For nonlinear problems, AA is also closely related to the nonlinear generalized minimal residual method [50]. A local convergence analysis of AA for general nonlinear problems was first given in [45, 46] under the base assumptions that g is Lipschitz continuously differentiable and the AA mixing coefficients α, determined in (3), stay in a compact set. However, the convergence rate provided in [45, 46] is not faster than the one of the original fixed-point iteration. A more recent analysis in [13] shows that AA can indeed accelerate the local linear convergence of a fixed-point iteration up to an additional quadratic error term. This is further improved in [32] where q-linear convergence of AA is established. The convergence result in [32] requires sufficient linear independence of the columns of $[f(x^{k-1}) - f(x^k), \ldots, f(x^{k-m}) - f(x^{k-m+1})]$ which is typically stronger than the previously mentioned boundedness assumption on the coefficients α. By assuming the mixing coefficient α to be stationary during the iteration, an exact rate of AA is derived in [50].

One issue of classical AA is that it can suffer from instability and stagnation [34, 39]. Different techniques have been proposed to address this issue. For example, safeguarding checks were introduced in [30, 54] to only accept an AA step if it meets certain criteria, but without a theoretical guarantee for convergence. Another direction is to introduce regularization to the problem (3) for computing the combination coefficients. In [17], a quadratic regularization is used together with a safeguarding step to achieve global convergence of AA on Douglas–Rachford splitting, but there is no guarantee that the local convergence is faster than the original solver. In [39, 40], a similar quadratic regularization is introduced to achieve local convergence, although no global convergence proof is provided. A more detailed discussion of related literature and specific techniques connected to our algorithmic design and development is deferred to Sect. 2.1.

As far as we are aware, none of the existing approaches and modified versions of AA guarantee both global convergence and accelerated local convergence. In this paper, we propose a novel AA globalization scheme that achieves these two goals simultaneously. Specifically, we apply a quadratic regularization with its weight adjusted automatically according to the effectiveness of the AA step. We adapt the nonmonotone trust-region framework in [47] to update the weight and to determine the acceptance of the AA step. Our approach can not only achieve global convergence, but also attains the same local convergence rate established in [13] for AA without regularization. Furthermore, our local results also cover applications where the mapping g is nonsmooth and differentiability is only required at a target fixed-point of g. To the best of our knowledge, this is the first globalization technique for AA that achieves the same local convergence rate as the original AA scheme. Numerical experiments on both smooth and nonsmooth problems verify the effectiveness and efficiency of our method.

Notations Throughout this work, we restrict our discussion on the n-dimensional Euclidean space $\mathbb{R}^n$. For a vector x, $\|x\|$ denotes its Euclidean norm, and $\mathbb{B}_\epsilon(x) := \{y : \|y - x\| \le \epsilon\}$ denotes the Euclidean ball centered at x with radius ϵ. For a matrix A, $\|A\|$ is the operator norm with respect to the Euclidean norm. We use I to denote both the identity mapping (i.e., $I(x) = x$) and the identity matrix. For a function $h : \mathbb{R}^n \to \mathbb{R}^\ell$, the mapping $h' : \mathbb{R}^n \to \mathbb{R}^{\ell \times n}$ represents its derivative. h is called L-smooth if it is differentiable and $\|h'(x) - h'(y)\| \le L\|x - y\|$ for all $x, y \in \mathbb{R}^n$. An operator $h : \mathbb{R}^n \to \mathbb{R}^n$ is called nonexpansive if for all $x, y \in \mathbb{R}^n$ we have $\|h(x)-h(y)\| \le \|x-y\|$. We say that the operator h is ρ-averaged for some $\rho \in (0, 1)$ if there exists a nonexpansive operator $R : \mathbb{R}^n \to \mathbb{R}^n$ such that $h = (1-\rho)I+\rho R$. The set of fixed points of the mapping h is defined via $\mathrm{Fix}(h) := \{x : h(x) = x\}$. The interested reader is referred to [4] for further details on operator theory.

The AA formulation in Eqs. (2) and (3) assumes $k \ge m$. It can be adapted to account for the case $k < m$ by using $\hat{m}$ coefficients instead where $\hat{m} = \min\{m, k\}$. Without loss of generality, we use m to refer to the actual number of coefficients being used.

2 Algorithm and Convergence Analysis

2.1 Adaptive Regularization for AA

In the following, we set $f^k := f(x^k)$ and $g^k := g(x^k)$ to simplify notation. We first note that the accelerated iterate computed via (2) and (3) is invariant under permutations of the indices of $\{f^j\}$ and $\{g^j\}$. Concretely, let $\Pi_k := (k_0, k_1, \dots, k_m)$ be any permutation of the index sequence $(k, k-1, \dots, k-m)$. Then the point x^{k+1} calculated in Eq. (2) also satisfies

$$x^{k+1} = g^{k_0} + \sum\nolimits_{i=1}^{m} \bar{\alpha}_i^k (g^{k_i} - g^{k_0}), \tag{4}$$

with coefficients $\bar{\alpha}^k = (\bar{\alpha}_0^k, \dots, \bar{\alpha}_m^k)$ computed via

$$\bar{\alpha}^k \in \underset{\alpha}{\operatorname{argmin}} \left\| f^{k_0} + \sum\nolimits_{i=1}^{m} \alpha_i (f^{k_i} - f^{k_0}) \right\|^2, \tag{5}$$

which amounts to solving a linear system. In this paper, we use a particular class of permutations where

$$k_0 = \max\left\{ j \mid j \in \operatorname{argmin}_{i \in \{k,k-1,\dots,k-m\}} \|f^i\| \right\}, \tag{6}$$

i.e., k_0 is the largest index that attains the minimum residual norm among $\|f^{k-m}\|$, $\|f^{k-m+1}\|$, $\dots$, $\|f^k\|$. As we will see later, this type of permutation allow us to apply certain nonmonotone globalization techniques and to ultimately establish local and global convergence of our approach. An ablation study on the potential effect of the permutation strategy is presented in “Appendix D”.

One potential cause of instability of AA is the (near) linear dependence of the vectors $\{f^{k_i} - f^{k_0} : i = 1, \ldots, m\}$, which can result in (near) rank deficiency of the linear system matrix for the problem (5). To address this issue, we introduce a quadratic regularization to the problem (5) and compute the coefficients α^k via:

$$\alpha^k = \underset{\alpha}{\operatorname{argmin}} \ \|\hat{f}^k(\alpha)\|^2 + \lambda_k \|\alpha\|^2, \tag{7}$$

where $\lambda_k > 0$ is a regularization weight, and

$$\hat{f}^k(\alpha) := f^{k_0} + \sum\nolimits_{i=1}^{m} \alpha_i (f^{k_i} - f^{k_0}). \tag{8}$$

The coefficients α^k are then used to compute a trial step in the same way as in Eq. (4). In the following, we denote this trial step as $\hat{g}^k(\alpha^k)$ where

$$\hat{g}^k(\alpha) := g^{k_0} + \sum\nolimits_{i=1}^{m} \alpha_i (g^{k_i} - g^{k_0}). \tag{9}$$

The trial step is accepted as the new iterate if it meets certain criteria (which we will develop in the following in detail). Regularization such as the one in Eq. (7) has been suggested in [3] and is applied in [17, 40]. A major difference between our approach and the regularization in [17, 40] is the choice of λ_k: in [17] it is set in a heuristic manner, whereas in [40] it is either fixed or specified via grid search. We instead update λ_k adaptively based on the effectiveness of the latest AA step. Specifically, we observe that a larger value of λ_k can improve stability for the resulting linear system; it will also induce a stronger penalty for the magnitude of $\|\alpha^k\|$. In this case, the trial step $\hat{g}^k(\alpha^k)$ tends to be closer to g^{k_0}, which, according to Eq. (6), is the fixed-point iteration step with the smallest residual among the latest $m + 1$ iterates. On the other hand, a larger regularization weight may also hinder the fast convergence of AA if it is already effective in reducing the residual without regularization. Thus, λ_k is dynamically adjusted according to the reduction of the residual in the current step.

Our adaptive regularization scheme is inspired by the similarity between the problem (7) and the Levenberg-Marquardt (LM) algorithm [20, 26], a popular approach for solving nonlinear least squares problems of the form $\min_x \|F(x)\|^2$, where F is a vector-valued function. Each iteration of LM computes a variable update $d^k := x^{k+1} - x^k$ by solving a quadratic problem

$$\underset{d}{\operatorname{argmin}} \ \|F(x^k) + F'(x^k)d\|^2 + \bar{\lambda}_k \|d\|^2.$$

Here, the first term is a local quadratic approximation of the target function $\|F(x)\|^2$ using the first-order Taylor expansion of F, while the second term is a regularization with a weight $\bar{\lambda}_k > 0$. LM can be considered as a regularized version of the classical Gauss-Newton (GN) method for nonlinear least squares optimization [23]. In GN, each iteration computes an initial step d by minimizing the local quadratic approximation term only, i.e.,:

$$\underset{d}{\operatorname{argmin}} \ \|F(x^k) + F'(x^k)d\|^2, \tag{10}$$

which amounts to solving a linear system for d with the positive semidefinite matrix $(F'(x^k))^T F'(x^k)$. Similar to AA, the (near) linear dependence between the columns of $F'(x^k)$ can lead to (near) rank deficiency of the system matrix causing potential instability. To address this issue, LM introduces a quadratic regularization term for d, which adds a scaled identity matrix to the linear system matrix and prevents it from being singular. Furthermore, LM measures the effectiveness of the computed update using a ratio of the resulting reduction of the target function and a predicted reduction based on the quadratic approximation. The measure is utilized to determine the acceptance of the update, to enforce monotonic decrease of the target function, and to update the regularization weight for the next iteration. Such an adaptive regularization is an instance of a trust-region method [10].

Taking a similar approach as LM, we define two functions ared_k and pred_k that measure the actual and predicted reduction of the residual resulting from the solution α^k to (7):

$$\text{ared}_k := r_k - \|f(\hat{g}^k(\alpha^k))\|, \quad \text{pred}_k := r_k - c\|\hat{f}^k(\alpha^k)\|, \tag{11}$$

where $c \in (0, 1)$ is a constant. Here r_k measures the residuals from the latest $m + 1$ iterates via a convex combination:

$$r_k := (1 - m\gamma)\|f^{k_0}\| + \sum\nolimits_{i=1}^{m} \gamma \|f^{k_i}\|, \tag{12}$$

with $\gamma \in (0, \frac{1}{m+1})$ such that a higher weight is assigned to the smallest residual f^{k_0} among them. Note that $\hat{g}^k(\alpha^k)$ is the trial step, and $f(\cdot)$ is the residual function. Thus ared_k compares the latest residuals with the residual resulting from the trial step. This specific choice of r_k is inspired by the local descent properties of AA, see, e.g., [13, Theorem 4.4]. Moreover, note that $\hat{f}^k(\cdot)$ (see Eq. (8)) is a linear approximation of the residual function based on the latest residual values, and it is used in problem (7) to derive the coefficients α^k for computing the trial step. Thus $\hat{f}^k(\alpha^k)$ is a predicted residual for the trial step based on the linear approximation, and pred_k compares it with the latest residuals. The constant c guarantees that pred_k has a positive value (as long as a solution to the problem has not been found; see "Appendix A" for a proof). Similar to LM, we calculate the ratio

$$\rho_k = \frac{\text{ared}_k}{\text{pred}_k} \tag{13}$$

as a measure of effectiveness for the trial step $\hat{g}^k(\alpha^k)$ computed with Eqs. (7) and (9). In particular, if $\rho_k \geq p_1$ with a threshold $p_1 \in (0, 1)$, then from Eq. (13) and using the positivity of pred_k we can bound the residual of $\hat{g}^k(\alpha^k)$ via

$$\|f(\hat{g}^k(\alpha^k))\| \leq (1 - p_1)r_k + p_1 c\|\hat{f}^k(\alpha^k)\| < (1 - p_1)r_k + p_1\|f^{k_0}\|. \tag{14}$$

Like r_k, the last expression $(1 - p_1)r_k + p_1\|f^{k_0}\|$ in Eq. (14) is also a convex combination of the latest $m + 1$ residuals, but with a higher weight on the smallest residual f^{k_0} than on r_k. Hence, when $\rho_k \geq p_1$, we consider the decrease of the residual to be sufficient. In this case, we set $x^{k+1} = \hat{g}^k(\alpha^k)$ and say the iteration is *successful*. Otherwise, we discard the trial step and choose $x^{k+1} = g^{k_0} = g(x^{k_0})$, which corresponds to the fixed-point iteration step with the smallest residual among the latest $m + 1$ iterates. Thus, by permuting the indices $(k, k - 1, \ldots, k - m)$ according to Π_k, we can ensure to achieve the most progress in terms of reducing the residual when an AA trial step is rejected.

We also adjust the regularization weight λ_k according to the ratio ρ_k. Specifically, we set

$$\lambda_k = \mu_k\|f^{k_0}\|^2, \tag{15}$$

where the factor $\mu_k > 0$ is automatically updated based on ρ_k as follows:

- If $\rho_k < p_1$, then we consider the decrease of the residual to be insufficient and we increase the factor in the next iteration via $\mu_{k+1} = \eta_1\mu_k$ with a constant $\eta_1 > 1$.
- If $\rho_k > p_2$ with a threshold $p_2 \in (p_1, 1)$, then we consider the decrease to be high enough and reduce the factor via $\mu_{k+1} = \eta_2\mu_k$ with a constant $\eta_2 \in (0, 1)$. This will relax the regularization so that the next trial step will tend to be closer to the original AA step.

Algorithm 1 Anderson acceleration with adaptive regularization.

```
Require: $x^0 \in \mathbb{R}^n$, $\mu_0 > 0$, $k = 0$, $0 < p_1 < p_2 < 1$, $0 < \eta_2 < 1 < \eta_1$, $0 < c < 1$.
for $k = 0, 1, \ldots$ do
    Compute $g^k = g(x^k)$, $f^k = g^k - x^k$.
    if $\|f^k\|$ is smaller than a threshold $\epsilon_f$ then
        Return $g^k$ as the result.
    end if
    Construct a permutation $\Pi_k = (k_0, k_1, \ldots, k_m)$ according to Eq. (6).
    Compute the regularization weight $\lambda_k$ via Eq. (15).
    Solve the problem (7) to obtain the coefficients $\alpha^k$.
    Compute the ratio $\rho_k$ with Eq. (13).
    Update the regularization factor $\mu_{k+1}$ via:

        $\mu_{k+1} = \begin{cases} \eta_1 \mu_k & \text{if } \rho_k < p_1, \\ \mu_k & \text{if } \rho_k \in [p_1, p_2], \\ \eta_2 \mu_k & \text{if } \rho_k > p_2. \end{cases}$

    if $\rho_k \geq p_1$ then
        Set $x^{k+1} = \hat{g}^k(\alpha^k)$ via Eq. (9).
    else
        Set $x^{k+1} = g^{k_0}$.
    end if
end for
```

- Otherwise, in the case $\rho_k \in [p_1, p_2]$, the factor remains the same in the next iteration.

Here the choice of the parameters p_1, p_2 where $0 < p_1 < p_2 < 1$ follows the convention of basic trust-region methods [10].

Our setting of λ_k in Eq. (15) is inspired by [15] which relates the LM regularization weight to the residual norm. For our method, this setting ensures that the two target function terms in problem (7) are of comparable scales, so that the adjustment of the factor μ_k is meaningful. This choice of λ_k and the update rule of μ_k are quite standard in LM methods. However, the classical convergence analysis in [15] is not directly applicable here. In the LM method, the decrease of the residual can be predicted via its linearized model $\|F(x^k) + F'(x^k)d\|^2$. For AA, the linearized residual $\hat{f}^k(\alpha^k)$ is not a model for the update $x^{k+1} = \hat{g}^k(\alpha^k)$ but for $\hat{x}^k(\alpha^k)$ instead. Since a linearized residual of $\hat{g}^k$ is not readily available, we use an upper bound for such a linearization of $\hat{g}^k(\alpha^k)$ which is exactly given by $c\|\hat{f}^k(\alpha^k)\|$. The whole method is summarized in Algorithm 1.

Unlike LM which enforces a monotonic decrease of the target function, our acceptance strategy allows the residual for x^{k+1} to increase compared to the previous iterate x^k. Therefore, our scheme can be considered as a nonmonotone trust-region approach and follows the procedure investigated in [47]. In the next subsections, we will see that this nonmonotone strategy allows us to establish unified global and local convergence results. In particular, besides global convergence guarantees, we can show transition to fast local convergence and an acceleration effect similar to the original AA scheme can be achieved.

The main computational overhead of our method lies in the optimization problem (7), which amounts to constructing and solving an $m \times m$ linear system $(J^T J + \lambda_k I)\alpha^k = -J^T f^{k_0}$ where $J = [f^{k_1} - f^{k_0}, f^{k_2} - f^{k_0}, \ldots, f^{k_m} - f^{k_0}] \in \mathbb{R}^{n \times m}$. A naïve implementation that computes the matrix J from scratch in each iteration will result in $O(m^2 n)$ time for setting up the linear system, whereas the system itself can be solved in $O(m^3)$ time. Since we typically have $m \ll n$, the linear system setup will become the dominant overhead. To reduce the overhead, we note that each entry of $J^T J$ is a linear combination of inner products between $f^{k_0}, \ldots, f^{k_m}$. If we pre-compute and store these inner products, then it only requires additional $O(m^2)$ time to evaluate all entries. Moreover, the pre-computed inner products can be updated in $O(mn)$ time in each iteration, so we only need $O(mn)$ total time to evaluate $J^T J$. Similarly, we can evaluate $J^T f^{k_0}$ in $O(m)$ time. In this way, the linear system setup only requires $O(mn)$ time in each iteration. Moreover, as the parameter m is often a small value independent of n (and significantly smaller than n), the complexity $O(mn)$ is effectively linear with respect to n and only incurs a small computational overhead.

2.2 Global Convergence Analysis

We now present our main assumptions on g and f that allow us to establish global convergence of Algorithm 1. Our conditions are mainly based on a monotonicity property and on pointwise convergence of the iterated functions $g^{[k]} : \mathbb{R}^n \to \mathbb{R}^n$ defined as $g^{[k]}(x) := (\underbrace{g \circ \cdots \circ g}_{k \text{ times}})(x)$, for $k \in \mathbb{N}$.

Assumption 1 The functions g and f satisfy the following conditions:

(A.1) $\|f(g(x))\| \leq \|f(x)\|$ for all $x \in \mathbb{R}^n$.
(A.2) $\lim\limits_{k\to\infty} \|f(g^{[k]}(x))\| = \nu$ for all $x \in \mathbb{R}^n$, where $\nu = \inf_{x\in\mathbb{R}^n} \|f(x)\|$.

It is easy to see that Assumption 1 holds for any contractive function with $\nu = 0$. In particular, if g satisfies (1), we obtain

$$\begin{aligned}\|f(g^{[k]}(x))\| &= \|g(g^{[k]}(x)) - g(g^{[k-1]}(x))\| \\ &\leq \kappa \|f(g^{[k-1]}(x))\| \leq \ldots \leq \kappa^k \|f(x)\| \to 0 \end{aligned} \tag{16}$$

as $k \to \infty$. In the following, we will verify that Assumption 1 also holds for ρ-averaged operators which define a broader class of mappings than contractions.

Proposition 1 *Let $g : \mathbb{R}^n \to \mathbb{R}^n$ be a ρ-averaged operator with $\rho \in (0, 1)$. Then g satisfies Assumption 1.*

Proof By definition the ρ-averaged operator g is also nonexpansive and thus, (A.1) holds for g. To prove (A.2), let us set $y^0 := x$ and $y^{k+1} := g^{[k+1]}(x) = g(y^k)$ for all k. By (A.1), the sequence $\{\|f(y^k)\|\}_k$ is monotonically decreasing. Therefore, we can assume that $\lim_{k\to\infty} \|f(y^k)\| = \vartheta$. If $\vartheta > \nu$, then we may select $x^0 \in \mathbb{R}^n$ such that $\|f(x^0)\| < \nu + \frac{1}{2}(\vartheta - \nu)$. Defining $x^{k+1} = g^{[k+1]}(x^0) = g(x^k)$ and applying [4, Proposition 4.25(iii)], we have

$$\|x^{k+1} - y^{k+1}\|^2 \leq \|x^k - y^k\|^2 - \frac{1-\rho}{\rho} \|f(x^k) - f(y^k)\|^2.$$

This yields

$$\|x^{k+1} - y^{k+1}\|^2 \leq \|x^0 - y^0\|^2 - \frac{1-\rho}{\rho} \sum_{i=0}^{k} \|f(x^i) - f(y^i)\|^2.$$

By the reverse triangle inequality and (A.1), we have

$$\|f(x^i) - f(y^i)\| \geq \|f(y^i)\| - \|f(x^i)\| \geq \vartheta - \|f(x^0)\| \geq \frac{1}{2}(\vartheta - \nu).$$

Combining with the previous inequality, we obtain

$$\|x^{k+1} - y^{k+1}\|^2 \leq \|x^0 - y^0\|^2 - \frac{1-\rho}{2\rho}(k+1)(\vartheta - \nu).$$

Taking the limit $k \to \infty$, we reach a contradiction. So, we must have $\nu = \vartheta$, as desired. □

Remark 1 Setting κ (the Lipschitz constant of g) to 1 in (16), we see that (A.1) is always satisfied if g is a nonexpansive operator. However, nonexpansiveness is not a necessary condition for (A.1). In fact, we can construct an operator g that is not nonexpansive but satisfies (A.1) and (A.2), e.g.,

$$g : \mathbb{R} \to \mathbb{R}, \quad g(x) := \begin{cases} 0.5x & \text{if } x \in [0, 1], \\ 0 & \text{otherwise.} \end{cases}$$

For any $x \in [0, 1]$, we have $g^{[k]}(x) = 2^{-k}x$, $f(g^{[k]}(x)) = -2^{-(k+1)}x$ and it is not hard to verify (A.1) and (A.2). For any $x \notin [0, 1]$ it follows $f(g(x)) = f(0) = 0$, thus (A.1) and (A.2) also hold in this situation. However, since g is not continuous, it can not be nonexpansive.

Because of Proposition 1, our global convergence theory is applicable to a large class of iterative schemes. As an example, we show in the following that Assumption 1 is satisfied by forward-backward splitting, a popular optimization solver in machine learning.

Example 1 Let us consider the nonsmooth optimization problem:

$$\min_{x \in \mathbb{R}^n} \; r(x) + \varphi(x), \tag{17}$$

where both $r, \varphi : \mathbb{R}^n \to (-\infty, \infty]$ are proper, closed, and convex functions, and r is L-smooth. It is well known that x^* is a solution to this problem if and only if it satisfies the nonsmooth equation:

$$x^* - G_\mu(x^*) = 0, \quad G_\mu(x) := \text{prox}_{\mu\varphi}(x - \mu\nabla r(x)),$$

where $\text{prox}_{\mu\varphi}(x) := \text{argmin}_y \, \varphi(y) + \frac{1}{2\mu}\|x - y\|^2$, $\mu > 0$, is the proximity operator of φ, see also Corollary 26.3 of [4]. We can then compute x^* via the iterative scheme

$$x^{k+1} = G_\mu(x^k). \tag{18}$$

G_μ is known as the forward-backward splitting operator and it is a ρ-averaged operator for all $\mu \in (0, \frac{2}{L})$, see [8]. Hence, Assumption 1 holds and our theory can be used to study the global convergence of Algorithm 1 applied to (18).

Remark 2 For problem (17), it can be shown that Douglas–Rachford splitting, as well as its equivalent form of ADMM, can both be written as a ρ-averaged operator with $\rho \in (0, 1)$ (see, e.g., [22]). Therefore, the applications considered in [17] are also covered by Assumption 1.

We can now show the global convergence of Algorithm 1:

Theorem 1 *Suppose Assumption 1 is satisfied and let $\{x^k\}$ be generated by Algorithm 1 with $\epsilon_f = 0$. Then*

$$\lim_{k \to \infty} \|f^k\| = \nu,$$

where $\nu = \inf_{x \in \mathbb{R}^n} \|f(x)\|$.

Proof In the following, we will use $\mathcal{S}$ to denote the set of indices for all successful iterations, i.e., $\mathcal{S} := \{k : \rho_k \geq p_1\}$. To simplify the notation, we introduce a function $\mathcal{P} : \mathbb{N} \to \mathbb{N}$ defined as

$$\mathcal{P}(k) := \max \left\{ j \mid j \in \operatorname{argmin}_{i \in \{k, k-1, \ldots, k-m\}} \|f^i\| \right\}.$$

Notice that the number $\mathcal{P}(k)$ coincides with k_0 for fixed k.

If Algorithm 1 terminates after a finite number of steps, the conclusion simply follows from the stopping criterion. Therefore, in the following, we assume that a sequence of iterates of infinite length is generated. We consider two different cases:

Case 1: $|\mathcal{S}| < \infty$. Let $\bar{k}$ denote the index of the last successful iteration in $\mathcal{S}$ (we set $\bar{k} = 0$ if $\mathcal{S} = \emptyset$). We first show that $\mathcal{P}(k) = k$ for all $k \geq \bar{k} + 1$. Due to $\bar{k} + 1 \notin \mathcal{S}$, it follows $x^{\bar{k}+1} = g(x^{\mathcal{P}(\bar{k})})$ and by (A.1), this implies $\|f(x^{\bar{k}+1})\| \leq \|f(x^{\mathcal{P}(\bar{k})})\|$. From the definition of $\mathcal{P}$, we have $\|f(x^{\mathcal{P}(\bar{k})})\| \leq \|f^{\bar{k}-i}\|$ for every $0 \leq i \leq \min\{m, \bar{k}\}$ and hence $\mathcal{P}(\bar{k}+1) = \bar{k}+1$. An inductive argument then yields $\mathcal{P}(k) = k$ for all $k \geq \bar{k}+1$. Notice that for any $k \geq \bar{k}+1$, we have $k \notin \mathcal{S}$ and $x^{k+1} = g(x^{\mathcal{P}(k)}) = g(x^k)$. Utilizing (A.2), it follows that $\|f^k\| = \|f(g^{[k-\bar{k}]}(x^{\mathcal{P}(\bar{k})}))\| \to \nu$ as $k \to \infty$.

Case 2: $|\mathcal{S}| = \infty$. Let us denote

$$W_k := \max_{k-m \leq i \leq k} \|f^i\|.$$

We first show that the sequence $\{W_k\}$ is non-increasing.

- If $k \in \mathcal{S}$, then we have:

$$p_1 \leq \rho_k = \frac{\mathrm{ared}_k}{\mathrm{pred}_k}.$$

We already know from "Appendix A" that $\mathrm{pred}_k > 0$.
Since $p_1 > 0$, it also holds that $\mathrm{ared}_k > 0$. Hence, if $c\|\hat{f}^k(\alpha^k)\| \leq \|f^{k+1}\|$, then using $r_k \leq W_k$ and (41) from "Appendix A", we can derive:

$$\begin{aligned} p_1 \leq \frac{\mathrm{ared}_k}{\mathrm{pred}_k} &= 1 + \frac{c\|\hat{f}^k(\alpha^k)\| - \|f^{k+1}\|}{\mathrm{pred}_k} = 1 + \frac{c\|\hat{f}^k(\alpha^k)\| - \|f^{k+1}\|}{r_k - c\|\hat{f}^k(\alpha^k)\|} \\ &\leq 1 + \frac{c\|\hat{f}^k(\alpha^k)\| - \|f^{k+1}\|}{W_k - c\|\hat{f}^k(\alpha^k)\|} = \frac{W_k - \|f^{k+1}\|}{W_k - c\|\hat{f}^k(\alpha^k)\|} \leq \frac{W_k - \|f^{k+1}\|}{(1-c)W_k}, \end{aligned}$$

which implies

$$\|f^{k+1}\| \leq c_p W_k, \tag{19}$$

where $c_p := 1 - (1-c)p_1 < 1$. Otherwise, if $c\|\hat{f}^k(\alpha^k)\| > \|f^{k+1}\|$, then we have

$$\|f^{k+1}\| \leq c W_k \leq c_p W_k. \tag{20}$$

- If $k \notin \mathcal{S}$, we have $x^{k+1} = g^{\mathcal{P}(k)}$. By Assumption (A.1), it then follows that

$$\|f^{k+1}\| \leq \|f^{\mathcal{P}(k)}\| \leq W_k. \tag{21}$$

Eqs. (19), (20) and (21) show that $\|f^{k+1}\| \leq W_k$. By definition of W_k, we then have $W_{k+1} \leq \max\{\|f^{k+1}\|, W_k\} = W_k$. This shows that the sequence $\{W_k\}$ is non-increasing. Next, we verify

$$W_{k+m+1} \leq c_p W_k$$

for all $k \in \mathcal{S}$. It suffices to prove that for any i satisfying $k+1 \leq i \leq k+m+1$, we have $\|f^i\| \leq c_p W_k$. Since we consider a successful iteration $k \in \mathcal{S}$, our previous discussion has already shown that $\|f^{k+1}\| \leq c_p W_k$. We now assume $\|f^i\| \leq c_p W_k$ for some $k+1 \leq i \leq k+m$. If $i \in \mathcal{S}$, we obtain $\|f^{i+1}\| \leq c_p W_i \leq c_p W_k$. Otherwise, it follows that $\|f^{i+1}\| \leq \|f^{\mathcal{P}(i)}\| \leq \|f^i\| \leq c_p W_k$. Hence, by induction, we have $W_{k+m+1} \leq c_p W_k$ for all $k \in \mathcal{S}$. Since $\{W_k\}$ is non-increasing and we assumed $|\mathcal{S}| = \infty$, this establishes $W_k \to 0$ and $\|f^k\| \to 0$. In this case, we can infer $\nu = 0$ and the proof is complete. □

Remark 3 This global result does not depend on the specific update rule for the regularization weight λ_k. Indeed, global convergence mainly results from our acceptance mechanism and hence, as a consequence of our proof, different update strategies for λ_k can also be applied. Our choice of λ_k in (15), however, will be essential for establishing local convergence of the method.

2.3 Local Convergence Analysis

Next, we analyze the local convergence of our proposed approach, starting with several assumptions.

Assumption 2 The function $g : \mathbb{R}^n \to \mathbb{R}^n$ satisfies the following conditions:

(B.1) g is Lipschitz continuous with a constant $\kappa < 1$.
(B.2) g is differentiable at x^* where x^* is the fixed point of the mapping g.

Remark 4 (B.1) is a standard assumption widely used in the local convergence analysis of AA [13, 39, 40, 46]. The existing analyses typically rely on the smoothness of g. In contrast, (B.2) allows g to be nonsmooth and only requires it to be differentiable at the fixed point x^*, allowing our assumptions to cover a wider variety of methodologies such as forward-backward splitting and Douglas–Rachford splitting under appropriate assumptions, see “Appendix B”. We note that in [6] the Lipschitz differentiability of g is replaced by continuous differentiability around x^*, while we only assume differentiability at one point. This technical difference is based on the observation that an expansion of the residual f^k is only required at the point x^* and not at the iterates x^k which allows to work with weaker differentiability requirements. We further note that AA has been investigated for nonsmooth g in [17, 53] but without local convergence analysis. Recent convergence results of AA for a scheme related to the proximal gradient method discussed in Example 1 can also be found in [24]. While the local assumptions and derived convergence rates in [24] are somewhat similar to our local results, we want to highlight that the algorithm and analysis in [24] are tailored to convex composite problems of the form (17). Moreover, the global results in [24] are shown for a second, guarded version of AA and are based on the strong convexity of the problem. In contrast and under conditions that are not stronger than the local assumptions in [24], we will establish unified global–local convergence of our approach for general contractions. In Sect. 3, we verify the conditions (B.1) and (B.2) on the numerical examples, with a more detailed discussion in “Appendix B”.

Remark 5 (B.1) implies that the function g is contractive, which is a sufficient condition for (A.1) and (A.2). Thus, a function g satisfying Assumption 2 will also fulfill Assumption 1 with $\nu = 0$.

Similar to the local convergence analyses in [13, 46], we also work with the following condition:

Assumption 3 For the solution $\bar{\alpha}^k$ to the unregularized AA problem (5), there exists $M > 0$ such that $\|\bar{\alpha}^k\|_\infty \le M$ for all k sufficiently large.

Remark 6 The assumptions given in [13, 46] are formulated without permuting the last $m+1$ indices. We further note that we do not require the solution $\bar{\alpha}^k$ to be unique.

The acceleration effect of the original AA scheme has only been studied very recently in [13] based on slightly stronger assumptions. In particular, their result can be stated as

$$\|f(\hat{g}^k(\bar{\alpha}^k))\| \le \kappa\theta_k\|f^{k_0}\| + \sum\nolimits_{i=0}^{m} O(\|f^{k-i}\|^2), \tag{22}$$

where

$$\theta_k := \|\hat{f}^k(\bar{\alpha}^k)\|/\|f^{k_0}\| \tag{23}$$

is an acceleration factor. Since $\bar{\alpha}^k$ is a solution to the problem (5), we have $\|\hat{f}^k(\bar{\alpha}^k)\| \le \|\hat{f}^k(0)\| = \|f^{k_0}\|$ so that $\theta_k \in [0, 1]$. Then (22) implies that for a fixed-point iteration that converges linearly with a contraction constant κ, AA can improve the convergence rate locally. In the following, we will show that our globalized AA method possesses similar characteristics under weaker assumptions.

We first verify that after finitely many iterations, every step $x^{k+1} = \hat{g}^k(\alpha^k)$ is accepted as a new iterate. Thus, our method eventually reduces to a pure regularized AA scheme.

Theorem 2 *Suppose that Assumptions 2 and 3 hold and let the constant c in (11) be chosen such that $c \ge \kappa$. Then, the sequence $\{x^k\}$ generated by Algorithm 1 (with $\epsilon_f = 0$) either terminates after finitely many steps, or converges to the fixed point x^* and there exists some $\ell \in \mathbb{N}$ such that $\rho_k \ge p_2$ for all $k \ge \ell$. In particular, every iteration $k \ge \ell$ is successful with $x^{k+1} = \hat{g}^k(\alpha^k)$.*

Proof Our proof consists of three steps. We first show the convergence of the whole sequence $\{x^k\}$ to the fixed point x^*. Afterwards we derive a bound for the residual $\|f(\hat{g}^k(\alpha^k))\|$ that can be used to estimate the actual reduction ared_k. In the third step, we combine our observations to prove the transition to the full AA method, i.e., we show that there exists some ℓ with $k \in \mathcal{S}$ for all $k \ge \ell$.

Step 1: Convergence of $\{x^k\}$. By (B.1), g is a contraction, i.e., for all $x \in \mathbb{R}^n$ we have

$$\|x - x^*\| \le \|x - g(x)\| + \|g(x) - g(x^*)\| \le \|f(x)\| + \kappa\|x - x^*\| \tag{24}$$

and it follows $\|f^k\| = \|f(x^k)\| \ge (1-\kappa)\|x^k - x^*\|$ for all k. Theorem 1 and Remark 5 guarantee $\lim_{k\to\infty}\|f^k\| = 0$ and hence, we can infer $x^k \to x^*$.

Step 2: Bounding $\|f(\hat{g}^k(\alpha^k))\|$. Introducing

$$\hat{x}^k(\alpha^k) := x^{k_0} + \sum\nolimits_{i=1}^{m} \alpha_i^k(x^{k_i} - x^{k_0})$$

and using (B.1), we can bound the residual $\|f(\hat{g}^k(\alpha^k))\|$ directly as follows:

$$\begin{aligned}\|f(\hat{g}^k(\alpha^k))\| &= \|g(\hat{g}^k(\alpha^k)) - \hat{g}^k(\alpha^k)\| \\ &\le \|g(\hat{g}^k(\alpha^k)) - g(\hat{x}^k(\alpha^k))\| + \|g(\hat{x}^k(\alpha^k)) - \hat{g}^k(\alpha^k)\| \\ &\le \kappa\|\hat{g}^k(\alpha^k) - \hat{x}^k(\alpha^k)\| + \|g(\hat{x}^k(\alpha^k)) - \hat{g}^k(\alpha^k)\| \\ &= \kappa\|\hat{f}^k(\alpha^k)\| + \|g(\hat{x}^k(\alpha^k)) - \hat{g}^k(\alpha^k)\|.\end{aligned}$$

We now continue to estimate the second term $\|g(\hat{x}^k(\alpha^k)) - \hat{g}^k(\alpha^k)\|$. From the algorithmic construction and the definition of α^k and $\bar{\alpha}^k$, it follows that

$$\|\hat{f}^k(\alpha^k)\|^2 + \lambda_k\|\alpha^k\|^2 \le \|\hat{f}^k(\bar{\alpha}^k)\|^2 + \lambda_k\|\bar{\alpha}^k\|^2 \le \|\hat{f}^k(\alpha^k)\|^2 + \lambda_k\|\bar{\alpha}^k\|^2, \quad (25)$$

which implies $\|\alpha^k\|_\infty \le \|\alpha^k\| \le \|\bar{\alpha}^k\| \le \sqrt{m}\|\bar{\alpha}^k\|_\infty \le \sqrt{m}M$ for all k. Defining $\nu^k = (\nu_0^k, \dots, \nu_m^k) \in \mathbb{R}^{m+1}$ with $\nu_0^k = 1 - \sum_{i=1}^m \alpha_i^k$ and $\nu_j^k = \alpha_j^k$ for $1 \le j \le m$, we obtain

$$\hat{g}^k(\alpha) = \sum\nolimits_{i=0}^m \nu_i^k g(x^{k_i}), \quad \hat{x}^k(\alpha) = \sum\nolimits_{i=0}^m \nu_i^k x^{k_i},$$

$\|\nu^k\|_\infty \le 1 + m^{\frac{3}{2}}M$, and $\sum_{i=0}^m \nu_i^k = 1$.
Consequently, applying the estimate (24) derived in step 1, it follows

$$\begin{aligned}\|\hat{x}^k(\alpha^k) - x^*\| = \left\|\sum\nolimits_{i=0}^m \nu_i^k(x^{k_i} - x^*)\right\| &\le (1 + m^{\frac{3}{2}}M)\sum\nolimits_{i=0}^m \|x^{k_i} - x^*\| \\ &\le (1 + m^{\frac{3}{2}}M)(1-\kappa)^{-1}\sum\nolimits_{i=0}^m \|f^{k-i}\|\end{aligned}$$

which shows $\hat{x}^k(\alpha^k) \to x^*$ as $k \to \infty$. This also establishes

$$o(\|\hat{x}^k(\alpha^k) - x^*\|) = o\left(\sum\nolimits_{i=0}^m \|f^{k-i}\|\right) \quad k \to \infty. \quad (26)$$

Note that the differentiability of g at x^*—as stated in Assumption (B.2)—implies $\|g(y) - g(x^*) - g'(x^*)(y - x^*)\| = o(\|y - x^*\|)$ as $y \to x^*$. Applying this condition to different choices of y and the boundedness of ν^k, we can obtain

$$\begin{aligned}&\|g(\hat{x}^k(\alpha^k)) - \hat{g}^k(\alpha^k)\| \\ &\quad = \|g(\hat{x}^k(\alpha^k)) - g(x^*) + g(x^*) - \hat{g}^k(\alpha^k)\| \\ &\quad \le \|g'(x^*)(\hat{x}^k(\alpha^k) - x^*) + g(x^*) - \hat{g}^k(\alpha^k)\| + o(\|\hat{x}^k(\alpha^k) - x^*\|) \\ &\quad = \left\|\sum\nolimits_{i=0}^m \nu_i^k[g'(x^*)(x^{k_i} - x^*) + g(x^*) - g(x^{k_i})]\right\| + o(\|\hat{x}^k(\alpha^k) - x^*\|) \\ &\quad \le \sum\nolimits_{i=0}^m o(\|x^{k_i} - x^*\|) + o(\|\hat{x}^k(\alpha^k) - x^*\|) \le o\left(\sum\nolimits_{i=0}^m \|f^{k-i}\|\right). \quad (27)\end{aligned}$$

Here, we also used (24), (26), and $\sum_{i=0}^m o(\|f^{k-i}\|) = o\left(\sum_{i=0}^m \|f^{k-i}\|\right)$. Combining our results, this yields

$$\|f(\hat{g}^k(\alpha^k))\| \le \kappa\|\hat{f}^k(\alpha^k)\| + o\left(\sum\nolimits_{i=0}^m \|f^{k-i}\|\right) \quad k \to \infty. \quad (28)$$

Step 3: Transition to fast local convergence. As in the proof of Theorem 1, let us introduce

$$W_k := \max_{k-m \le i \le k} \|f^i\|.$$

Due to (28) and $o\left(\sum_{i=0}^m \|f^{k-i}\|\right) = o(W_k)$ there then exists $\ell \in \mathbb{N}$ such that

$$\|f(\hat{g}^k(\alpha^k))\| \le \kappa\|\hat{f}^k(\alpha^k)\| + (1 - p_2)\min\{\gamma, 1 - c\}W_k$$

for all $k \ge \ell$.
Hence, using $c \ge \kappa$, we have

$$\text{ared}_k = r_k - \|f(\hat{g}^k(\alpha^k))\| \ge \text{pred}_k - (1 - p_2)\min\{\gamma, 1 - c\}W_k.$$

Similarly, for the predicted reduction pred_k we can show

$$\begin{aligned}\mathrm{pred}_k &= (1-\gamma m)\|f^{k_0}\| + \gamma \sum\nolimits_{i=1}^{m} \|f^{k_i}\| - c\|\hat{f}^k(\alpha^k)\| \\ &\geq \gamma \sum\nolimits_{i=0}^{m} \|f^{k_i}\| + (1-\gamma(m+1))\|f^{k_0}\| - c\|f^{k_0}\| \\ &\geq \gamma W_k + (1-\gamma)\|f^{k_0}\| - c\|f^{k_0}\|.\end{aligned}$$

Thus, if $1-\gamma-c \geq 0$, we obtain $\mathrm{pred}_k \geq \gamma W_k$. Otherwise, it follows $\mathrm{pred}_k \geq (1-c)W_k$ and together this yields $\mathrm{pred}_k \geq \min\{\gamma, 1-c\}W_k$. Combining the last estimates, we can finally deduce

$$\frac{\mathrm{ared}_k}{\mathrm{pred}_k} \geq \frac{\mathrm{pred}_k - (1-p_2)\min\{\gamma, 1-c\}W_k}{\mathrm{pred}_k} \geq p_2,$$

which completes the proof. □

Remark 7 Our novel nonmonotone acceptance mechanism is the central component of our proof for Theorem 2, as it allows us to balance the additional error terms caused by an AA step.

Next, we show that our approach can enhance the convergence of the underlying fixed-point iteration and that it has a local convergence rate similar to the original AA method as given in [13].

Theorem 3 *Suppose that Assumptions 2, and 3 hold and let the parameters c, ϵ_f in Algorithm 1 satisfy $c \geq \kappa$ and $\epsilon_f = 0$. Then, for $k \to \infty$ it holds that:*

$$\|f^{k+1}\| \leq \kappa\theta_k \|f^{k_0}\| + o\left(\sum\nolimits_{i=0}^{m} \|f^{k-i}\|\right),$$

where $\theta_k := \|\hat{f}^k(\bar{\alpha}^k)\| / \|f^{k_0}\|$ is the corresponding acceleration factor. In addition, the sequence of residuals $\{\|f^k\|\}$ converges r-linearly to zero with a rate arbitrarily close to κ, i.e., for every $\eta \in (\kappa, 1)$ there exist $C > 0$ and $\hat{\ell} \in \mathbb{N}$ such that

$$\|f^k\| \leq C\eta^k \quad \forall\, k \geq \hat{\ell}.$$

Proof Theorem 2 implies $\rho_k \geq p_2$ for all $k \geq \ell$ and hence, from the update rule of Algorithm 1, it follows that

$$\mu_k = \eta_2 \mu_{k-1} \quad \forall\, k \geq \ell.$$

Then by (15), we can infer $\lambda_k = o(\|f^{k_0}\|^2)$. Using Eq. (25) and Assumption 3, this shows

$$\|\hat{f}^k(\alpha^k)\| \leq \|\hat{f}^k(\bar{\alpha}^k)\| + \sqrt{\lambda_k}\|\bar{\alpha}^k\| = \|\hat{f}^k(\bar{\alpha}^k)\| + o(\|f^{k_0}\|). \tag{29}$$

Thus, by (28), we obtain

$$\begin{aligned}\|f^{k+1}\| \leq\ & \kappa\|\hat{f}^k(\alpha^k)\| + o\left(\sum\nolimits_{i=0}^{m} \|f^{k-i}\|\right) \\ \leq\ & \kappa\|\hat{f}^k(\bar{\alpha}^k)\| + o(\|f^{k_0}\|) + o\left(\sum\nolimits_{i=0}^{m} \|f^{k-i}\|\right) \\ =\ & \kappa\theta_k\|f^{k_0}\| + o\left(\sum\nolimits_{i=0}^{m} \|f^{k-i}\|\right),\end{aligned} \tag{30}$$

as desired. In order to establish r-linear convergence, we follow the strategy presented in [46]. Let $\eta \in (\kappa, 1)$ be a given rate. Then, due to $\|f^k\| \to 0$ and using (30), there exists $\hat{\ell} \in \mathbb{N}$ such that

$$\|f^{k+1}\| \leq \kappa\|f^{k_0}\| + \bar{\nu} \cdot \sum\nolimits_{i=0}^{m} \|f^{k-i}\| \tag{31}$$

for all $k \geq \hat{\ell}$ where $\bar{\nu} := \frac{1-\eta}{1-\eta^{m+1}} \eta^m (\eta - \kappa)$. Defining $C := \eta^{-\hat{\ell}} \max_{\hat{\ell}-m \leq i \leq \hat{\ell}} \| f^i \| = W_{\hat{\ell}} \eta^{-\hat{\ell}}$, we then have

$$\| f^j \| \leq W_{\hat{\ell}} = (W_{\hat{\ell}} \eta^{-j}) \eta^j \leq (W_{\hat{\ell}} \eta^{-\hat{\ell}}) \eta^j = C \eta^j.$$

for all $\hat{\ell} - m \leq j \leq \hat{\ell}$. We now claim that the statement $\| f^k \| \leq C \eta^k$ holds for all $k \geq \hat{\ell}$. As just shown, this is obviously satisfied for the base case $k = \hat{\ell}$. As part of the inductive step, let us assume that the estimate $\| f^j \| \leq C \eta^j$ holds for all $j = \hat{\ell}, \hat{\ell}+1, \ldots, k$. (In fact, this bound also holds for $j = \hat{\ell} - m, \ldots, \hat{\ell} - 1$). By the definition of the index k_0, we have $\| f^{k_0} \| \leq C \eta^k$ and, due to (31), it follows

$$\begin{aligned} \| f^{k+1} \| &\leq \kappa \| f^{k_0} \| + \bar{\nu} \cdot \sum\nolimits_{i=0}^{m} \| f^{k-i} \| \leq C \kappa \eta^k + C \bar{\nu} \eta^k \sum\nolimits_{i=0}^{m} \left(\frac{1}{\eta} \right)^i \\ &= C \eta^k \left[\kappa + \bar{\nu} \cdot \frac{1 - \eta^{-(m+1)}}{1 - \eta^{-1}} \right] = C \eta^k \left[\kappa + \frac{\bar{\nu}}{\eta^m} \cdot \frac{1 - \eta^{m+1}}{1 - \eta} \right] = C \eta^{k+1}. \end{aligned}$$

Hence, our claim also holds for $k + 1$ which finishes the induction and proof. □

Under a stronger differentiability condition and stricter update rule for λ_k, we can recover the same local rate as in [13]:

Corollary 1 *Let the assumptions stated in Theorem 3 hold and let g satisfy the differentiability condition*

$$\| g(x) - g(x^*) - g'(x^*)(x - x^*) \| = O(\| x - x^* \|^2) \ \ as \ \ x \to x^*.$$

Suppose that the weight λ_k is updated via $\lambda_k = \mu_k \| f^{k_0} \|^4$. Then, for all k sufficiently large we have

$$\| f^{k+1} \| \leq \kappa \theta_k \| f^{k_0} \| + \sum\nolimits_{i=0}^{m} O(\| f^{k-i} \|^2).$$

Proof As mentioned in the remark after Theorem 1, our global results do still hold if a different update strategy is used for the weight parameter λ_k. Moreover, the proof of Theorem 2 does also not depend on the specific choice of λ_k. Consequently, we only need to improve the bound (28) for $\| f(\hat{g}^k(\alpha^k)) \|$ derived in step 2 of the proof of Theorem 2. Using the additional differentiability property $\| g(y) - g(x^*) - g'(x^*)(y - x^*) \| = O(\| y - x^* \|^2)$, $y \to x^*$, we can directly improve the estimate for $\| g(\hat{x}^k(\alpha^k)) - \hat{g}^k(\alpha^k) \|$ in (27) as follows:

$$\| g(\hat{x}^k(\alpha^k)) - \hat{g}^k(\alpha^k) \| \leq \sum\nolimits_{i=0}^{m} O(\| x^{k_i} - x^* \|^2) + O(\| \hat{x}^k(\alpha^k) - x^* \|^2).$$

Using the bound $(\sum_{i=0}^{m} y_i)^2 \leq (m+1) \sum_{i=0}^{m} y_i^2$ for $y \in \mathbb{R}^{m+1}$, we obtain $\| \hat{x}^k(\alpha^k) - x^* \|^2 = \sum_{i=0}^{m} O(\| f^{k-i} \|^2)$ and thus, mimicking and combining the derivations in step 2 of the proof of Theorem 2, we have

$$\| g(\hat{x}^k(\alpha^k)) - \hat{g}^k(\alpha^k) \| \leq \sum\nolimits_{i=0}^{m} O(\| f^{k-i} \|^2)$$

and

$$\| f(\hat{g}^k(\alpha^k)) \| \leq \kappa \| \hat{f}^k(\alpha^k) \| + \sum\nolimits_{i=0}^{m} O(\| f^{k-i} \|^2) \tag{32}$$

as $k \to \infty$. As in the previous proof, we can now infer $\mu_k \to 0$ (this follows from $\rho_k \geq p_2$ for all k sufficiently large) and $\lambda_k = o(\| f^{k_0} \|^4)$. Furthermore, as in (29), due to Eq. (25) and Assumption 3, it holds that $\| \hat{f}^k(\alpha^k) \| \leq \| \hat{f}^k(\bar{\alpha}^k) \| + o(\| f^{k_0} \|^2)$. Combining this result with (32), we can then establish the convergence rate stated in Corollary 1. □

Remark 8 The stronger differentiability condition, which was also used in [13] and other local analyses, is, e.g., satisfied when the derivative g' is locally Lipschitz continuous around x^*. More discussions of this property can also be found in “Appendix B”. We note that under this type of stronger differentiability, we can only improve the order of the remainder linearization error terms and not the linear rate of convergence.

3 Numerical Experiments

We verify the effectiveness of our method by applying it to several existing numerical solvers and comparing its convergence speed with the original solvers. We also include the acceleration approaches from [17, 40] for comparison. The regularized nonlinear acceleration (RNA) proposed in [40] computes an accelerated iterate via an affine combination of the previous k iterates, and it also introduces a quadratic regularization when computing the affine combination coefficients. Unlike our approach, it performs an acceleration step every k iterations instead of every iteration, and its regularization weight is determined by a grid search that finds the weight that leads to the lowest target function value at the accelerated iterate. The A2DR scheme proposed in [17] is a globalization of AA applied on Douglas–Rachford splitting, using a quadratic regularization together with an acceptance mechanism based on sufficient decrease of the residual. All experiments are carried out on a laptop with a Core-i7 9750 H at 2.6GHz and 16GB of RAM. The source code for the examples in this section is available at https://github.com/bldeng/Nonmonotone-AA.

Our method involves several parameters. The parameters p_1, p_2, η_1 and η_2, used for determining acceptance of the trial step and updating the regularization weight, are standard parameters for trust-region methods. We choose $p_1 = 0.01$, $p_2 = 0.25$, $\eta_1 = 2$, $\eta_2 = 0.25$ by default. The parameter γ affects the convex combination weights in computing r_k in Eq. (12), and we choose $\gamma = 10^{-4}$. For the parameter c in the definition of pred_k, we choose $c = \kappa$ where $\kappa < 1$ is a Lipschitz constant for the function g, to satisfy the conditions for Theorems 2 and 3. We will derive the value of κ in each experiment. The initial regularization factor μ_0 is set to $\mu_0 = 1$ unless stated otherwise. Concerning the number m of previous iterates used in an AA step, we can make the following observations: a larger m tends to reduce the number of iterations required for convergence, but also increases the computational cost per iteration; our experiments suggest that choosing $5 \leq m \leq 20$ often achieves a good balance. For each experiment below, we will include multiple choices of m for comparison. “Appendix C” provides some further ablation studies for the parameters p_1, p_2, η_1, η_2, and c.

3.1 Logistic Regression

First, to compare our method with the RNA scheme proposed by [40], we consider the following logistic regression problem from [40] that optimizes a decision variable $x \in \mathbb{R}^n$:

$$\min_x \; F(x), \tag{33}$$

where

$$F(x) = \frac{1}{N} \sum\nolimits_{i=1}^{N} \log(1 + \exp(-b_i a_i^T x)) + \frac{\tau}{2} \|x\|^2, \tag{34}$$

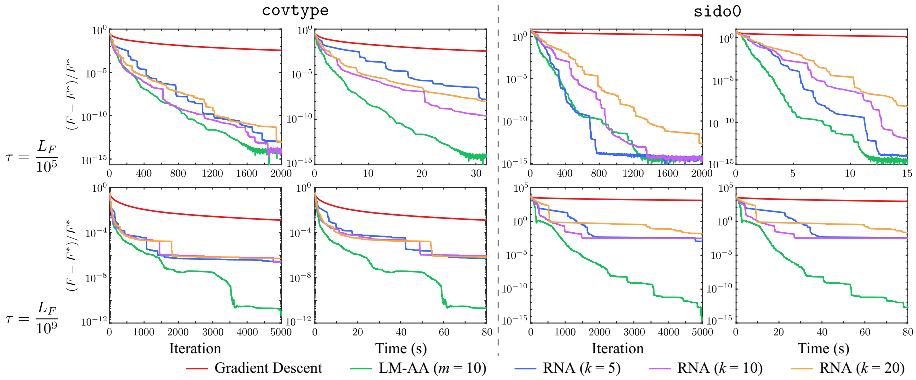

Fig. 1 Comparison between RNA [40] and our method on a gradient descent solver for the logistic regression problem (34) for the `covtype` and `sido0` datasets, with a different choice of parameter τ in each row

and $a_i \in \mathbb{R}^n$, $b_i \in \{-1, 1\}$ are the attributes and label of the data point i, respectively. Following [40], we consider gradient descent solver $x^{k+1} = g(x^k)$ with a fixed step size:

$$g(x) = x - \frac{2}{L_F + \tau}\nabla F(x),$$

where

$$L_F = \tau + \frac{\|A\|_2^2}{4N} \tag{35}$$

is the Lipschitz constant of ∇F, and $A = [a_1, \ldots, a_N]^T \in \mathbb{R}^{N \times n}$. Then g is Lipschitz continuous with modulus

$$\kappa = \frac{L_F - \tau}{L_F + \tau} < 1$$

and differentiable, which satisfies Assumption 2. We apply our approach (denoted by "LM-AA") and RNA to this solver, and compare their performance on two datasets: `covtype`[1] (54 features, 581,012 points), and `sido0`[2] (4932 features, 12,678 points). For each dataset, we normalize the attributes and solve the problem with $\tau = L_F/10^6$ and $\tau = L_F/10^9$, respectively. For the implementation of RNA, we use the source code released by the authors of [40].[3] We set $\mu_0 = 100$, and $m = 10, 15, 20$, respectively for our method. RNA performs an acceleration step every k iterations, and we test $k = 5, 10, 20$, respectively. All other RNA parameters are set to their default values as provided in the source code (in particular, with grid-search adaptive regularization weight and line search enabled). Figure 1 plots for each method the normalized target function value $(F(x^k) - F^*)/F^*$ with respect to the iteration count and computational time, where F^* is the ground-truth global minimum computed by running each method until full convergence and taking the minimum function value among all methods. All variants of LM-AA and RNA accelerate the decrease of the target function compared with the original gradient descent solver, with our methods achieving an overall faster decrease.

[1] https://archive.ics.uci.edu/ml/datasets/covertype.

[2] http://www.causality.inf.ethz.ch/data/sido0_matlab.zip.

[3] https://github.com/windows7lover/RegularizedNonlinearAcceleration/tree/master/Matlab/src.

3.2 Image Reconstruction

Next, we consider a nonsmooth problem proposed in [51] for total variation based image reconstruction:

$$\min_{w,u} \sum_{i=1}^{N^2} \|w_i\|_2 + \frac{\beta}{2}\sum_{i=1}^{N^2} \|w_i - D_i u\|_2^2 + \frac{\nu}{2}\|Ku - s\|_2^2, \tag{36}$$

where $s \in [0, 1]^{N^2}$ is an $N \times N$ input image, $u \in \mathbb{R}^{N^2}$ is the output image to be optimized, $K \in \mathbb{R}^{N^2 \times N^2}$ is a linear operator, $D_i \in \mathbb{R}^{2\times N^2}$ represents the discrete gradient operator at pixel i, $w = (w_1^T, \ldots, w_{N^2}^T)^T \in \mathbb{R}^{2N^2}$ are auxiliary variables for the image gradients, $\nu > 0$ is a fidelity weight, and $\beta > 0$ is a penalty parameter. The solver in [51] can be written as alternating minimization between u and w as follows:

$$u^{k+1} = \operatorname*{argmin}_u \frac{\beta}{2}\sum_{i=1}^{N^2} \|w_i^k - D_i u\|_2^2 + \frac{\nu}{2}\|Ku - s\|_2^2, \tag{37}$$

$$w^{k+1} = \operatorname*{argmin}_w \sum_{i=1}^{N^2} \|w_i\|_2 + \frac{\beta}{2}\sum_{i=1}^{N^2} \|w_i - D_i u^{k+1}\|_2^2. \tag{38}$$

The solutions to the subproblems (37) and (38) can both be computed in a closed form. When β and ν are fixed, this can be treated as a fixed-point iteration $w^{k+1} = g(w^k)$, and it satisfies Assumption 2 (see “Appendix B.2” for a detailed derivation of g and verification of Assumption 2). In the following, we consider the solver with $K = I$ and $\nu = 4$ for image denoising. In this case, condition (B.1) is satisfied with

$$\kappa = 1 - \left(1 + \frac{4\beta}{\nu}\right)^{-1}$$

(see “Appendix B.2” for the derivation). We apply this solver to a 1024 × 1024 image with added Gaussian noise that has a zero mean and a variance of $\sigma = 0.05$ (see Fig. 2).

We use the source code released by the authors of [51][4] for the implementation of this solver, and apply our acceleration method with $m = 1, 3, 5$ respectively. For comparison, we also apply the RNA scheme with $k = 2, 5, 10$, respectively. Here we choose smaller values of m and k than the logistic regression example because both RNA and our method have a high relative overhead on this problem, which means that larger values of m or k may induce overhead that offsets the performance gain from acceleration. Similar to the logistic regression example, we use the released source code of RNA for our experiments, and set all RNA parameters to their default values. Figure 2 plots the residual norm $\|f(w)\|$ for all methods, with $\beta = 100$ and $\beta = 1000$, respectively. All instances of acceleration methods converge faster to a fixed point than the original alternating minimization solver, except for RNA with $k = 2$ which is slower in terms of the actual computational time due to its overhead for the grid search of regularization parameters. Overall, the two acceleration approaches achieve a rather similar performance on this problem.

3.3 Nonnegative Least Squares

Finally, to compare our method with [17], we consider a nonnegative least squares (NNLS) problem that is used in [17] for evaluation:

$$\min_x \; \psi(x) + \varphi(x), \tag{39}$$

[4] https://www.caam.rice.edu/~optimization/L1/ftvd/v4.1/.

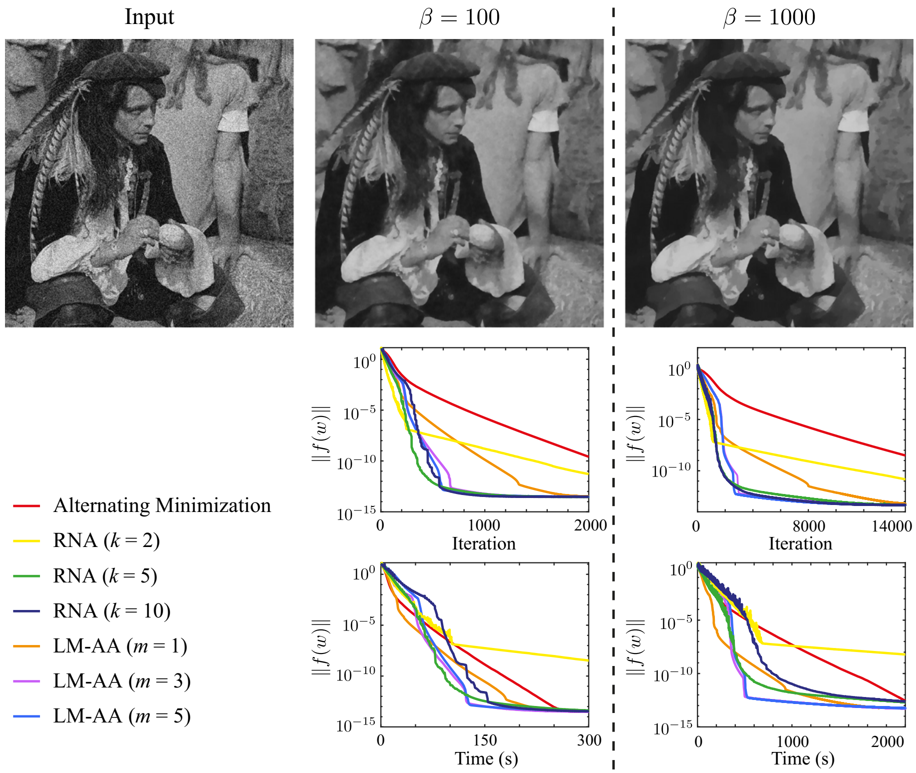

Fig. 2 Application of our method to an alternating minimization solver for an image denoising problem (36)

where $x = (x_1, x_2) \in \mathbb{R}^{2q}$, $\psi(x) = \|Hx_1 - t\|_2^2 + \mathcal{I}_{x_2 \geq 0}(x_2)$, and $\varphi(x) = \mathcal{I}_{x_1 = x_2}(x)$, with $\mathcal{I}_S$ being the indicator function of the set S. The Douglas–Rachford splitting (DRS) solver for this problem can be written as

$$v^{k+1} = g(v^k) = \tfrac{1}{2}((2\text{prox}_{\beta\varphi} - I)(2\text{prox}_{\beta\psi} - I) + I)v^k \tag{40}$$

where $v^k = (v_1^k, v_2^k) \in \mathbb{R}^{2q}$ is an auxiliary variable for DRS and β is the penalty parameter. In [17], the authors use their regularized AA method (A2DR) to accelerate the DRS solver (40). To apply our method, we verify in "Appendix B.3" that if H is of full column rank, then g satisfies condition (B.1) with

$$\kappa = \frac{\sqrt{3 + c_1^2}}{2} < 1$$

where $c_1 = \max\{\frac{\beta\sigma_1 - 1}{\beta\sigma_1 + 1}, \frac{1 - \beta\sigma_0}{1 + \beta\sigma_0}\}$, and σ_0, σ_1 are the minimal and maximal eigenvalues of $2H^T H$, respectively. Moreover, g is also differentiable under a mild condition. We compare our method with A2DR on the solver (40), with the same AA parameters $m = 10, 15, 20$. The methods are tested using a 600×300 sparse random matrix H with 1% nonzero entries and a random vector t. We use the source code released by the authors of [17][5] for the implementation of A2DR, and set all A2DR parameters to their default values. While A2DR and

[5] https://github.com/cvxgrp/a2dr.

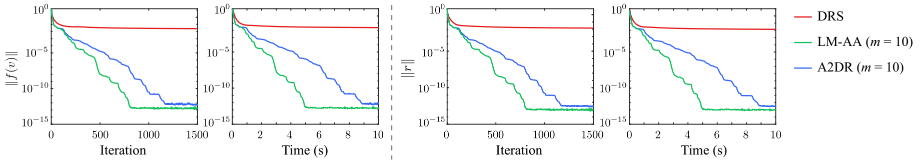

Fig. 3 Comparison between A2DR [17] and our method on the NNLS solver (40) with an 600 × 300 sparse random matrix H and a random vector t

DRS are implemented using parallel evaluation of the proximity operators in the released A2DR code, we implement our method as a single-threaded application for simplicity. Figure 3 plots the residual norm $\| f(v)\|$ for DRS and the two acceleration methods. It also plots the norm of the overall residual $r = (r_{\text{prim}}, r_{\text{dual}})$ used in [17] for measuring convergence, where r_{prim} and r_{dual} denote the primal and dual residuals as defined in Equations (7) and (8) of [17], respectively. For both residual measures, the original DRS solver converges slowly after the initial iterations, whereas the two acceleration methods achieve significant speedup. Moreover, the single-threaded implementation of our method outperforms the parallel A2DR with the same m parameter, in terms of both iteration count and computational time.

3.4 Statistics of Successful Steps

Our acceptance mechanism plays a key role in achieving global and local convergence of the proposed method. To demonstrate its behavior, we provide statistics of the successful steps in Figs. 1, 2 and 3. Specifically, for each instance of LM-AA, we count the total steps required to reach a certain level of accuracy and we compare it with the corresponding number of successful AA steps within these steps. Tables 1, 2, 3, and 4 show the statistics of successful steps for Figs. 1, 2 and 3, respectively. Here, besides the total number of steps, we report the success rate which is defined as the ratio between successful and total steps required to reach different levels of accuracy.

The results in Table 4 demonstrate that essentially all AA steps are accepted in the nonnegative least squares problem. This observation is also independent of the choice of the parameter m. More specifically, the success rate of AA steps only decreases and more alternative fixed-point iterations are performed when we seek to solve the problem with the highest accuracy $tol = 10^{-15}$. Table 3 illustrates that a similar behavior can also be observed for the image denoising problem (36) when setting $m = 1$. Notice that this high accuracy is close to machine precision and hence this effect is mainly caused by numerical errors and inaccuracies that affect the computation and quality of an AA step. The results in Table 3 also demonstrate a second typical effect: the success rate of AA step is often lower when the chosen accuracy is relatively low. With increasing accuracy, the rate then increases to around 70–80%. This general observation is also supported by our results for logistic regression, see Tables 1 and 2. (Here the maximum success rate is more sensitive to the choice of m, τ, and of the dataset).

In summary, the statistics provided in Tables 1, 2, 3, and 4 support our theoretical results. The success rate of AA steps gradually increases as the iteration gets closer to the fixed point, which indicates a transition to a pure regularized AA scheme. Furthermore, as more AA steps seem to be rejected at the beginning of the iterative procedure, our globalization mechanism effectively guarantees global progress and convergence of the approach.

Table 1 Statistics of successful steps of LM-AA for the logistic regression problem (34) and the dataset `covtype`

	m	*tol* (stopping criterion: $(F - F^*)/F^* \leq tol$)									
		10^{-3}		10^{-6}		10^{-9}		10^{-12}		10^{-15}	
		s-rate (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*
`covtype`											
$\tau = \frac{L_F}{10^6}$	10	72.3	83	94.3	599	98.3	2000	98.3	2000	98.3	2000
	15	57.3	82	88.0	541	93.8	1175	96.1	1887	96.4	2000
	20	49.0	98	73.5	434	87.8	1058	90.5	1351	92.9	1812
$\tau = \frac{L_F}{10^9}$	10	86.6	119	95.7	1083	96.3	3336	96.7	3755	96.7	3781
	15	55.4	83	86.0	783	32.0	5000	32.0	5000	32.0	5000
	20	50.5	91	73.2	593	18.8	5000	18.8	5000	18.8	5000

In each of the columns *iter*, we report the number of iterations required to satisfy $(F(x^k) - F^*)/F^* \leq tol$ with $tol \in \{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}, 10^{-15}\}$. The columns *s-rate* show the corresponding success rate of AA-steps, i.e., *s-rate* is the ratio between successful and total steps required to reach the different accuracies

Table 2 Statistics of successful steps of LM-AA for the logistic regression problem (34) and the dataset `sido0`

	m	*tol* (stopping criterion: $(F - F^*)/F^* \le tol$)									
		10^{-3}		10^{-6}		10^{-9}		10^{-12}		10^{-15}	
		s-rate (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*
`sido0`											
$\tau = \frac{L_F}{10^6}$	10	67.1	334	82.9	686	88.3	997	92.7	1621	93.8	2000
	15	55.6	315	77.7	676	84.8	994	89.7	1481	91.8	1992
	20	46.2	368	65.4	619	76.7	917	85.3	1453	88.8	2000
$\tau = \frac{L_F}{10^9}$	10	48.3	1208	64.6	2051	67.1	3006	53.5	4774	51.9	4950
	15	46.5	1815	59.5	2714	60.5	3788	50.9	5000	50.9	5000
	20	46.6	2205	57.8	3037	57.8	4090	51.7	5000	51.7	5000

In each of the columns *iter*, we report the number of iterations required to satisfy the criterion $(F(x^k) - F^*)/F^* \le tol$ with $tol \in \{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}, 10^{-15}\}$. The columns *s-rate* show the corresponding success rate of AA-steps, i.e., *s-rate* is the ratio between successful and total steps required to reach the different accuracies

Table 3 Statistics of successful steps of LM-AA for the image denoising problem (36)

	m	*tol* (stopping criterion: $\|f(w)\| \leq tol$)									
		10^{-3}		10^{-6}		10^{-9}		10^{-12}		10^{-15}	
		s-rate (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*
$\beta = 100$	1	93.7	190	96.6	378	98.4	812	99.0	1274	91.2	2000
	3	39.5	223	54.4	296	72.0	483	79.7	666	64.3	2000
	5	38.8	227	53.4	298	68.8	446	75.9	577	54.8	2000
$\beta = 1000$	1	92.1	1057	90.5	1489	95.9	3477	97.9	7677	75.7	15,000
	3	34.6	1458	39.6	1824	56.7	2554	68.8	3543	47.6	15,000
	5	35.0	1410	38.5	1776	60.6	2773	68.0	3408	43.0	15,000

In each of the columns *iter*, we report the number of iterations required to satisfy the criterion $\|f(w^k)\| \leq tol$ with $tol \in \{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}, 10^{-15}\}$. The columns *s-rate* show the success rate of AA-steps, i.e., *s-rate* is the ratio between successful and total steps required to reach the different accuracies

Table 4 Statistics of successful steps of LM-AA for the nonnegative least squares problem (39)

m	*tol* (stopping criterion: $\|f(v)\| \le tol$)									
	10^{-3}		10^{-6}		10^{-9}		10^{-12}		10^{-15}	
	s-rate (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*	*s-rate* (%)	*iter*
5	100	646	100	1500	100	1500	100	1500	100	1500
10	100	235	100	512	100	696	93.3	989	67.5	1500
15	100	151	100	437	100	680	100	792	58.0	1500
20	100	162	100	344	100	465	100	572	44.4	1500

In each of the columns *iter*, we report the number of iterations required to satisfy the criterion $\|f(w^k)\| \le tol$ with $tol \in \{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}, 10^{-15}\}$. The columns *s-rate* show the success rate of AA-steps, i.e., *s-rate* is the ratio between successful and total steps required to reach the different accuracies

4 Conclusions

We propose a novel globalization technique for Anderson acceleration which combines adaptive quadratic regularization and a nonmonotone acceptance strategy. We prove the global convergence of our approach under mild assumptions. Furthermore, we show that the proposed globalized AA scheme has the same local convergence rate as the original AA iteration and that the globalization mechanism does not hinder the acceleration effect of AA. This is one of the first AA globalization methods that achieves global convergence and fast local convergence simultaneously. Several numerical examples illustrate that our method is competitive and it can improve the efficiency of a variety of numerical solvers.

Funding A. Milzarek was partly supported by the Fundamental Research Fund - Shenzhen Research Institute for Big Data (SRIBD) Startup Fund JCYJ-AM20190601. B. Deng was partly supported by the Guangdong International Science and Technology Cooperation Project (No. 2021A0505030009).

Data Availability The datasets generated during and/or analysed during the current study are available in the GitHub repository https://github.com/bldeng/Nonmonotone-AA.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

A Proof that pred$_k$ in Eq. (12) is Positive

Proof By the definition of the minimization problem (7), we have

$$\|\hat{f}^k(\alpha^k)\|^2 \leq \|\hat{f}^k(\alpha^k)\|^2 + \lambda_k\|\alpha^k\|^2 \leq \|\hat{f}^k(0)\|^2 + \lambda_k\|0\|^2 = \|f^{k_0}\|^2.$$

Then Eqs. (12), (6) and $c \in (0, 1)$ imply that

$$r_k \geq \|f^{k_0}\| \geq \|\hat{f}^k(\alpha^k)\| \geq c\|\hat{f}^k(\alpha^k)\|. \tag{41}$$

By the algorithmic construction we know $\|f^{k_0}\| > 0$. So if $\|\hat{f}^k(\alpha^k)\| = 0$ then the second inequality is strict, otherwise the third inequality is strict. Overall we can deduce that $\text{pred}_k = r_k - c\|\hat{f}^k(\alpha^k)\|$ must be positive. □

B Verification of Local Convergence Assumptions

In this section, we briefly discuss different situations that allow us to verify and establish the local conditions stated in Assumption 2 and required for Corollary 1.

B.1 The Smooth Case

Clearly, assumption (B.2) is satisfied if g is a smooth mapping.

In addition, as mentioned at the end of Sect. 2.3, if the mapping g is continuously differentiable in a neighborhood of its associated fixed-point x^*, then the stronger differentiability condition

(C.1) $\|g(x) - g(x^*) - g'(x^*)(x - x^*)\| = O(\|x - x^*\|^2)$ for $x \to x^*$,

used in Corollary 1, holds if the derivative g' is locally Lipschitz continuous around x^*, i.e., for any $x, y \in \mathbb{B}_\epsilon(x^*)$ we have

$$\|g'(x) - g'(y)\| \leq L\|x - y\|. \tag{42}$$

Assumption (C.1) can then be shown via utilizing a Taylor expansion. Let us notice that for (C.1) it is enough to fix $y = x^*$ in (42). Such a condition is known as outer Lipschitz continuity at x^*. Furthermore, assumption (B.1) holds if $\sup_{x \in \mathbb{R}^n} \|g'(x)\| < 1$, see, e.g., Theorem 4.20 of [5].

B.2 Total Variation Based Image Reconstruction

The alternating minimization solver for image reconstruction problem (36) can be written as a fixed-point iteration

$$w^{k+1} = g(w^k) := (\Phi \circ h)(w^k), \tag{43}$$

with

$$\Phi(w) := (s_\beta(w_1)^T, \ldots, s_\beta(w_{N^2})^T)^T \in \mathbb{R}^{2N^2}, \quad s_\beta(x) := \max\left\{\|x\| - \frac{1}{\beta}, 0\right\} \frac{x}{\|x\|},$$

and $h(w) := DM^{-1}(D^T w + (\nu/\beta)K^T f$, $D := (D_1^T, \ldots, D_{N^2}^T)^T$, and $M := D^T D + (\nu/\beta)K^T K$. Let us notice that the mapping h and the fixed point iteration (43) are well-defined when the null spaces of K and D have no intersection, see, e.g., Assumption 1 of [51].

Next, we verify Assumption 2 for the mapping g in (43).

Proposition 2 *Suppose that the operator $K^T K$ is invertible. Then, the spectral radius $\rho(T)$ of $T := DM^{-1}D^T$ fulfills $\rho(T) < 1$ and condition (B.1) is satisfied.*

Proof Utilizing the Sherman–Morrison–Woodbury formula and the invertibility of $K^T K$, we obtain

$$(I + \xi D(K^T K)^{-1} D^T)^{-1} = I - \xi D(K^T K)^{-1}(I + \xi D^T D(K^T K)^{-1})^{-1} D^T = I - T, \tag{44}$$

where $\xi := \beta/\nu$. Due to $\lambda_{\min}(I + \xi D(K^T K)^{-1} D^T) \geq 1$ and $\lambda_{\max}(I + \xi D(K^T K)^{-1} D^T) \leq 1 + \xi\|D(K^T K)^{-1} D^T\|$, it then follows that

$$\sigma((I + \xi D(K^T K)^{-1} D^T)^{-1}) \subset \left[\frac{1}{1 + \xi\|D(K^T K)^{-1} D^T\|}, 1\right],$$

where $\sigma(\cdot)$ denotes the spectral set of a matrix. Combining this observation with (44), we have

$$\sigma(T) \subset \left[0, 1 - \frac{1}{1 + \xi \|D(K^T K)^{-1} D^T\|}\right] \quad \Longrightarrow \quad \rho(T) < 1.$$

Furthermore, following the proof of Theorem 3.6 in [51], it holds that

$$\|g(w) - g(v)\| \leq \rho(T)\|w - v\| \quad \forall\, w, v \in \mathbb{R}^{2N^2}$$

and hence, assumption (B.1) is satisfied. □

Concerning assumption (B.2), it can be shown that g is twice continuously differentiable on the set

$$\mathcal{W} = \{w : \|[h(w)]_i\| \neq 1/\beta,\ \forall\, i = 1, \ldots, N^2\}.$$

(In this case the max-operation in the shrinkage operator s_β is not active). Moreover, since h is continuous, the set $\mathcal{W}$ is open. Consequently, for every point $w \in \mathcal{W}$, we can find a bounded open neighborhood $N(w)$ of w such that $N(w) \subset \mathcal{W}$. Hence, if the mapping g has a fixed-point w^* satisfying $w^* \in \mathcal{W}$, then we can infer that g is differentiable on $N(w^*)$ and assumption (B.2) has to hold at w^*. Furthermore, the stronger assumption (C.1) for Corollary 1 is satisfied as well in this case. Finally, if K is the identity matrix, then notice that the finite difference matrix D satisfies that $\|D\| \leq 2$ and we can infer $\rho(T) \leq 1-(1+4\beta/\nu)^{-1}$ which justifies the choice of c in our algorithm.

B.3 Nonnegative Least Squares

We first note that given the specific form of φ, we can calculate the proximity operator φ explicitly as

$$\text{prox}_{\beta\varphi}(v) = \frac{1}{2}((v_1 + v_2)^T, (v_1 + v_2)^T)^T,$$

where $v = (v_1^T, v_2^T)^T$. Consequently, we obtain $[2\text{prox}_{\beta\varphi} - I](v) = (v_2^T, v_1^T)^T$. Similarly, by setting $\psi_1(v_1) := \|Hv_1 - t\|_2^2$ and $\psi_2(v_2) := \mathcal{I}_{v_2 \geq 0}(v_2)$, we have

$$\begin{aligned}
\text{prox}_{\beta\psi}(v) &= \begin{pmatrix} \text{prox}_{\beta\psi_1}(v_1) \\ \text{prox}_{\beta\psi_2}(v_2) \end{pmatrix}, \\
\text{prox}_{\beta\psi_1}(v_1) &= (H^T H + (2\beta)^{-1} I)^{-1}(H^T t + v_1/(2\beta)), \\
\text{prox}_{\beta\psi_2}(v_2) &= \mathcal{P}_{[0,\infty)^q}(v_2),
\end{aligned}$$

where $\mathcal{P}_{[0,\infty)^q}$ denotes the Euclidean projection onto the set of nonnegative numbers $[0, \infty)^q$. In the next proposition, we give a condition to establish (B.1) for g.

Proposition 3 *Let σ_0 and σ_1 denote the minimum and maximum eigenvalue of $2H^T H$, respectively and suppose $\sigma_0 > 0$. Then, the mapping g is Lipschitz continuous with modulus $\sqrt{3 + c_1^2}/2$, where $c_1 = \max\{\frac{\beta\sigma_1 - 1}{\beta\sigma_1 + 1}, \frac{1 - \beta\sigma_0}{1 + \beta\sigma_0}\} < 1$.*

Proof We can explicitly calculate g as follows

$$g(v) = \frac{1}{2}((2\text{prox}_{\beta\varphi} - I)(2\text{prox}_{\beta\psi} - I) + I)v = \frac{1}{2}\begin{pmatrix} \mathcal{R}_\beta^2(v_2) + v_1 \\ \mathcal{R}_\beta^1(v_1) + v_2 \end{pmatrix}, \tag{45}$$

where $\mathcal{R}^i_\beta = 2\mathrm{prox}_{\beta\psi_i} - I$, $i = 1, 2$. By Proposition 4.2 of [4], the reflected operators $\mathcal{R}^1_\beta$ and $\mathcal{R}^2_\beta$ are nonexpansive. Moreover, since $\sigma_0 > 0$, ψ_1 is strongly convex with modulus σ_0 and σ_1-smooth. Then by Theorem 1 of [19], $\mathcal{R}^1_\beta$ is Lipschitz continuous with modulus c_1. Next, for any $v, \bar{v} \in \mathbb{R}^{2q}$, we have

$$\begin{aligned}
&\|g(v) - g(\bar{v})\| \\
&= \frac{1}{2}\left[\|\mathcal{R}^2_\beta(v_2) - \mathcal{R}^2_\beta(\bar{v}_2) + v_1 - \bar{v}_1\|^2 + \|\mathcal{R}^1_\beta(v_1) - \mathcal{R}^1_\beta(\bar{v}_1) + v_2 - \bar{v}_2\|^2\right]^{\frac{1}{2}} \\
&\le \frac{1}{2}\left[(c_1^2 + 1)\|v_1 - \bar{v}_1\|^2 + 2(c_1 + 1)\|v_1 - \bar{v}_1\|\|v_2 - \bar{v}_2\| + 2\|v_2 - \bar{v}_2\|^2\right]^{\frac{1}{2}} \\
&\le \frac{1}{2}\left[(c_1^2 + 1)\|v_1 - \bar{v}_1\|^2 + \frac{(c_1+1)^2}{c_1^2+1}\|v_1 - \bar{v}_1\|^2 + (c_1^2 + 1)\|v_2 - \bar{v}_2\|^2 + 2\|v_2 - \bar{v}_2\|^2\right]^{\frac{1}{2}} \\
&\le \frac{1}{2}\sqrt{(c_1^2 + 3)\|v_1 - \bar{v}_1\|^2 + (c_1^2 + 3)\|v_2 - \bar{v}_2\|^2} \le \frac{\sqrt{3 + c_1^2}}{2}\|v - \bar{v}\|,
\end{aligned}$$

where we used Cauchy's inequality, the nonexpansiveness of $\mathcal{R}^2_\beta$, and the Lipschitz continuity of $\mathcal{R}^1_\beta$. The estimate in the second to last line follows from Young's inequality. □

Hence, assumption (B.1) is satisfied if H has full column rank. Using the special form of the mapping g in (45), we see that g is twice continuously differentiable at $v = (v_1^T, v_2^T)^T$ if and only if $v \in \mathcal{V} := \mathbb{R}^q \times \prod_{i=1}^q \mathbb{R}\backslash\{0\}$, i.e., if none of the components of v_2 are zero. As before, we can then infer that assumption (B.2) and the stronger condition (C.1) have to hold at every fixed-point v^* of g satisfying $v^* \in \mathcal{V}$.

B.4 Further Extensions

We now formulate a possible extension of the conditions presented in Sect. B.1 to the nonsmooth setting.

If the mapping g has more structure and is connected to an underlying optimization problem like in forward-backward and Douglas–Rachford splitting, nonsmoothness of g typically results from the proximity operator or projection operators. In such a case, further theoretical tools are available and for certain function classes it is possible to fully characterize the differentiability of g at x^* via a so-called strict complementarity condition. In fact, the conditions $w^* \in \mathcal{W}$ and $v^* \in \mathcal{V}$ from Sects. B.2 and B.3 are equivalent to such a strict complementarity condition. In the case of forward-backward splitting, a related and in-depth discussion of this important observation is provided in [24, 41] and we refer the interested reader to [21, 28, 31, 41] for further background.

Concerning the stronger assumption (C.1), we can establish the following characterization: Suppose that g is locally Lipschitz continuous and let us consider the properties:

- The function g is differentiable at x^*.
- The mapping g is strongly (or 1-order) semismooth at x^* [36], i.e., we have

$$\sup_{M \in \partial g(x)} \|g(x) - g(x^*) - M(x - x^*)\| = O(\|x - x^*\|^2)$$

  when $x \to x^*$. Here, the multifunction $\partial g : \mathbb{R}^n \rightrightarrows \mathbb{R}^{n\times n}$ denotes Clarke's subdifferential of the locally Lipschitz continuous (and possibly nonsmooth) function g, see, e.g., [9, 36].

- There exists an outer Lipschitz continuous selection $M^* : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ of ∂g in a neighborhood of x^*, i.e., for all x sufficiently close to x^* we have $M^*(x) \in \partial g(x)$ and

  $$\|M^*(x) - M^*(x^*)\| \le L_M \|x - x^*\|$$

  for some constant $L_M > 0$.

Then, the mapping g satisfies the condition (C.1) at x^*.

Proof The combination of differentiability and semismoothness implies that g is strictly differentiable at x^* and as a consequence, Clarke's subdifferential at x^* reduces to the singleton $\partial g(x^*) = \{g'(x^*)\}$. We refer to [28, 36, 37] for further details. Thus, we can infer $M^*(x^*) = g'(x^*)$ and we obtain

$$\begin{aligned}\|g(x) - g(x^*) - g'(x^*)(x - x^*)\| &\le \|g(x) - g(x^*) - M^*(x)(x - x^*)\| \\ &\quad + \|[M^*(x) - M^*(x^*)](x - x^*)\| \\ &\le O(\|x - x^*\|^2) + L_M\|x - x^*\|^2,\end{aligned}$$

for $x \to x^*$, where we used the strong semismoothness and outer Lipschitz continuity in the last step. This establishes (C.1). □

The class of strongly semismooth functions is rather rich and includes, e.g., piecewise twice continuously differentiable (PC^2) functions [48], eigenvalue and singular value functions [43, 44], and certain spectral operators of matrices [11]. Let us further note that the stated selection property is always satisfied when g is a piecewise linear mapping. In this case, the sets $\partial g(x)$ are polyhedral and outer Lipschitz continuity follows from Theorem 3D.1 of [12].

C Ablation Study on Parameter Choices

This subsection provides more numerical experiments on the parameters of our method, using the examples given in Figs. 1, 2 and 3 of the paper. In each experiment, we run our method by varying a subset of the parameters while keeping all other parameters the same as in the original figures, to evaluate how the varied parameters influence the performance of our method. For the evaluation, we plot the same convergence graphs as in the original figures to compare the performance resulting from different parameter choices. The evaluation is performed on the parameters c, (p_1, p_2), (η_1, η_2), and μ_0.

We first consider the logistic regression problem in Fig. 1 with $m = 15$ and $\tau = L_F \times 10^{-6}$. The parameters used in Fig. 1 are: $p_1 = 0.01$, $p_2 = 0.25$, $\eta_1 = 2$, $\eta_2 = 0.25$, $\mu_0 = 100$, $c = \kappa = (L_F - \tau)/(L_F + \tau)$. Figures 4, 5, 6, and 7 show the results using varied values of c, (p_1, p_2), (η_1, η_2), and μ_0, respectively.

Next, we consider the image reconstruction problem in Fig. 2 with $m = 5$ and $\beta = 100$. The parameters used in Fig. 2 are: $p_1 = 0.01$, $p_2 = 0.25$, $\eta_1 = 2$, $\eta_2 = 0.25$, $\mu_0 = 1$, and $c = \kappa$ where κ is derived in "Appendix B.2". Figures 8, 9, 10, and 11 show the results using varied values of c, (p_1, p_2), (η_1, η_2), and μ_0, respectively.

Finally, we consider the nonnegative least squares problem in Fig. 3 with $m = 10$. The parameters used in Fig. 3 are: $p_1 = 0.01$, $p_2 = 0.25$, $\eta_1 = 2$, $\eta_2 = 0.25$, $\mu_0 = 1$, and $c = \kappa$ where κ is derived in "Appendix B.3". Figures 12, 13, 14, and 15 show the results using varied values of c, (p_1, p_2), (η_1, η_2), and μ_0, respectively.

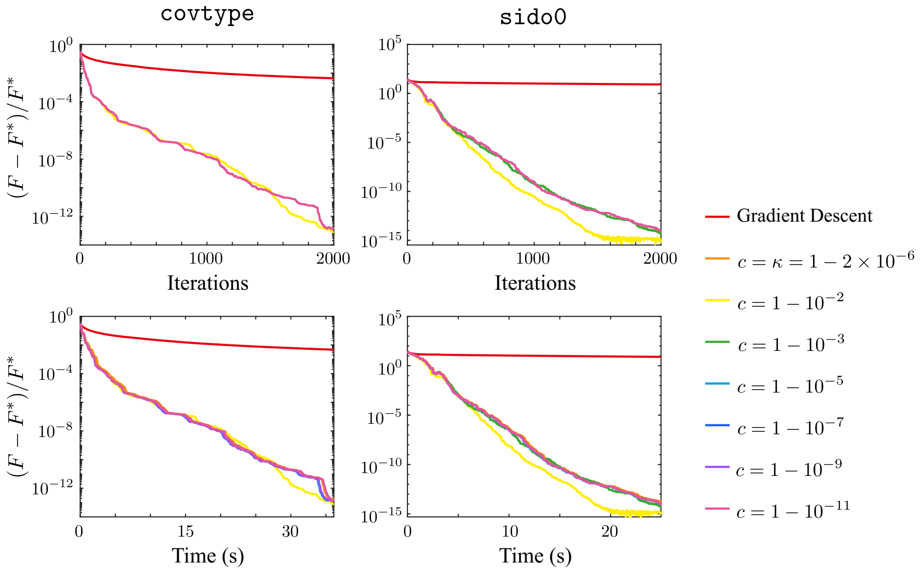

Fig. 4 Convergence plots for LM-AA ($m = 15$) on the gradient descent solvers in Fig. 1 with $\tau = L_F \times 10^{-6}$, using different values of the parameter c

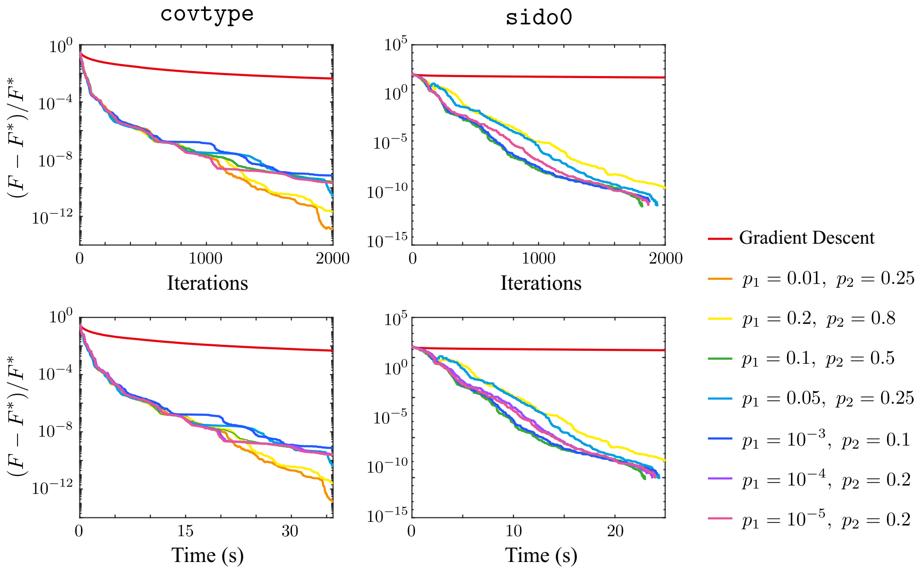

Fig. 5 Convergence plots for LM-AA ($m = 15$) on the gradient descent solvers in Fig. 1 with $\tau = L_F \times 10^{-6}$, using different values of the parameters p_1 and p_2

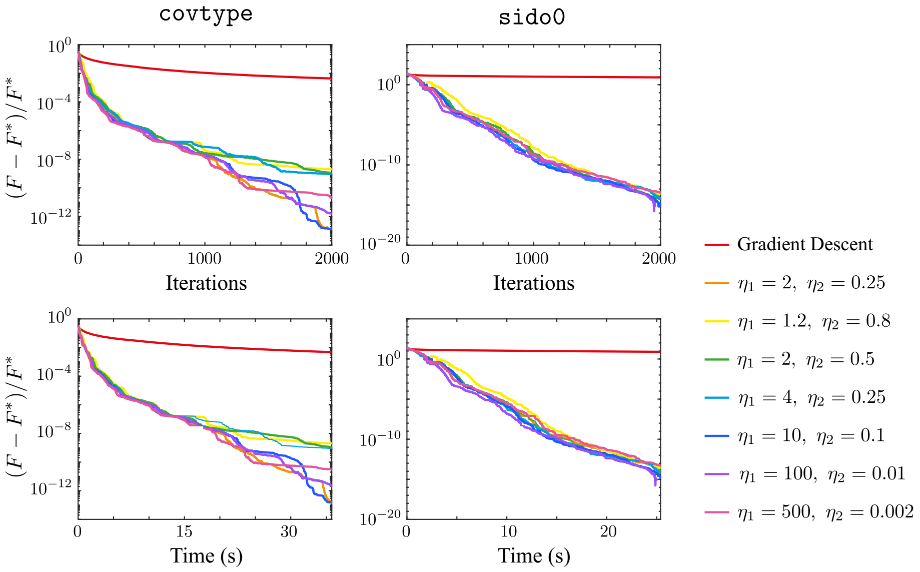

Fig. 6 Convergence plots for LM-AA ($m = 15$) on the gradient descent solvers in Fig. 1 with $\tau = L_F \times 10^{-6}$, using different values of the parameters η_1 and η_2

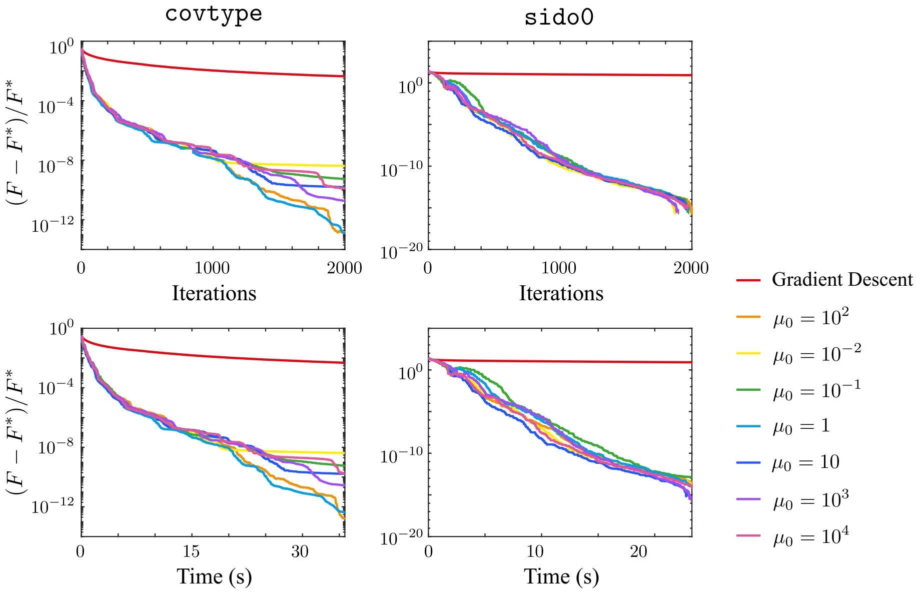

Fig. 7 Convergence plots for LM-AA ($m = 15$) on the gradient descent solvers in Fig. 1 with $\tau = L_F \times 10^{-6}$, using different values of the parameter μ_0

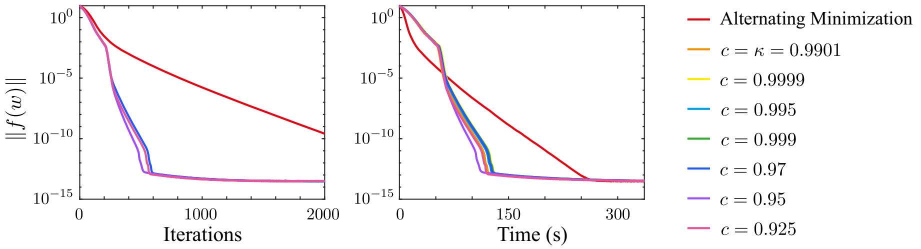

Fig. 8 Convergence plots for LM-AA ($m = 5$) on the alternating minimization solver in Fig. 2 with $\beta = 100$, using different values of the parameter c

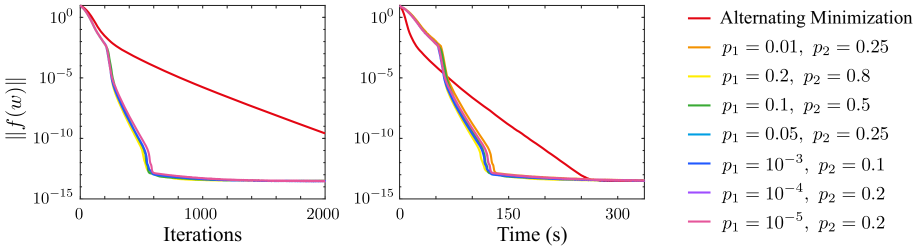

Fig. 9 Convergence plots for LM-AA ($m = 5$) on the alternating minimization solver in Fig. 2 with $\beta = 100$, using different values of the parameters p_1 and p_2

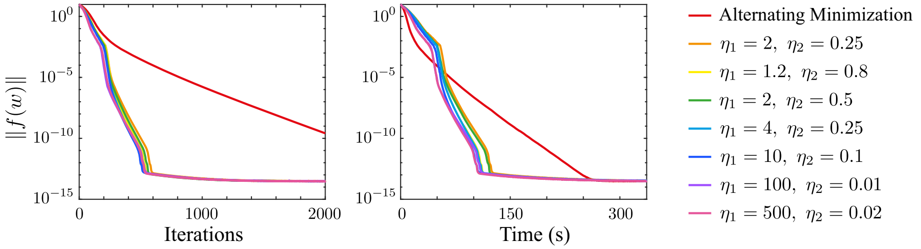

Fig. 10 Convergence plots for LM-AA ($m = 5$) on the alternating minimization solver in Fig. 2 with $\beta = 100$, using different values of the parameters η_1 and η_2

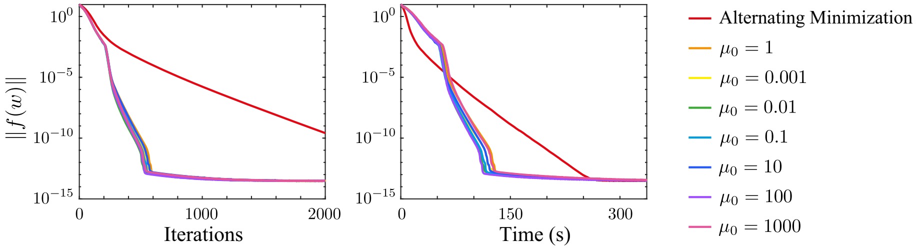

Fig. 11 Convergence plots for LM-AA ($m = 5$) on the alternating minimization solver in Fig. 2 with $\beta = 100$, using different values of the parameter μ_0

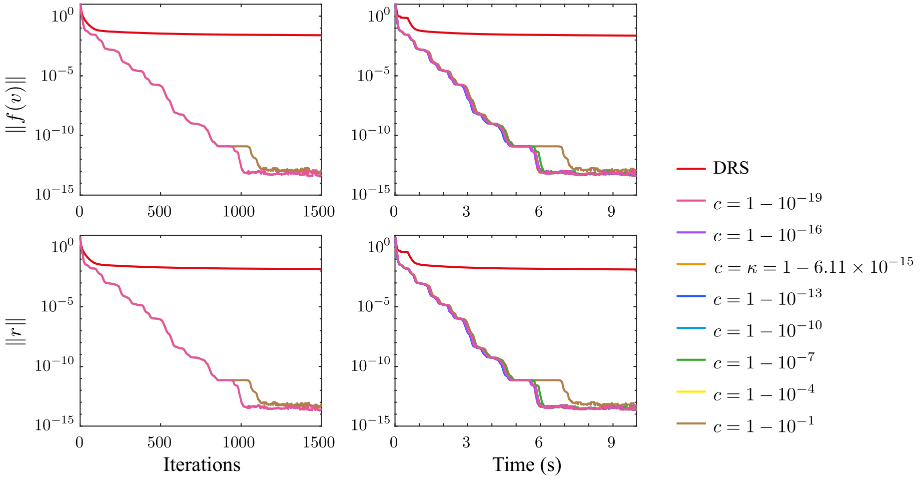

Fig. 12 Convergence plots for LM-AA ($m = 10$) on the DR splitting solver in Fig. 3, using different values of the parameter c

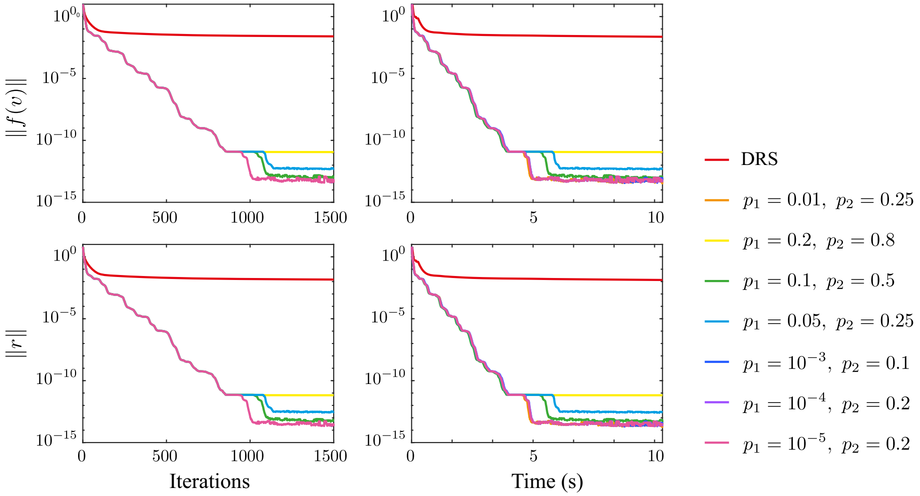

Fig. 13 Convergence plots for LM-AA ($m = 10$) on the DR splitting solver in Fig. 3, using different values of the parameters p_1 and p_2

As shown in Figs. 4, 8, and 12, our algorithm is not very sensitive w.r.t. the choice of c. Specifically, we still observe convergence if c is chosen smaller than the Lipschitz constant κ. This robustness is particularly important when good estimates of the constant κ are not available. Although the bound $c \geq \kappa$ is required in our theoretical results in Sect. 2.3, fast convergence can still be observed for other choices of c. This indicates that either we have over-estimated the Lipschitz constant κ or the acceleration effect of AA steps can be much faster than κ.

The results in Figs. 5, 9, and 13 demonstrate that the performance of LM-AA is not overly affected by the choice of the trust-region parameters p_1 and p_2 either. In general, good

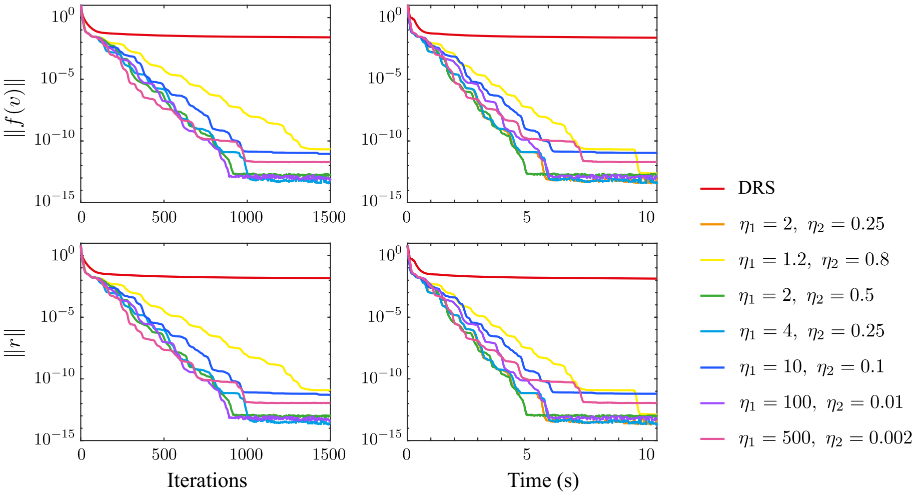

Fig. 14 Convergence plots for LM-AA ($m = 10$) on the DR splitting solver in Fig. 3, using different values of the parameters η_1 and η_2

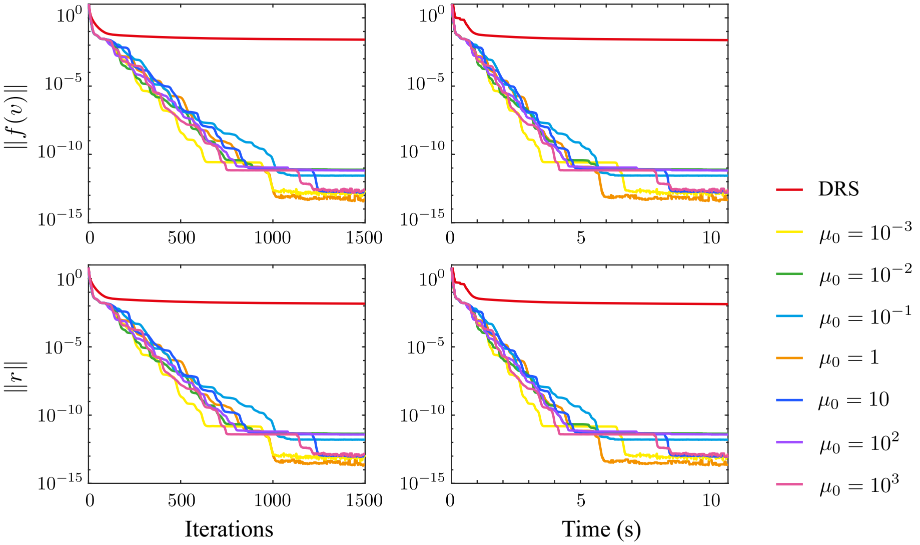

Fig. 15 Convergence plots for LM-AA ($m = 10$) on the DR splitting solver in Fig. 3, using different values of the parameter μ_0

performance can be achieved if p_1 is moderately small and p_2 is not too large. Thus, we decide to work with the standard choice $p_1 = 0.01$ and $p_2 = 0.25$.

In comparison, the trust-region parameters η_1 and η_2 can have a more significant impact on the performance of our algorithm. While the performance of LM-AA is not sensitive to the choice of η_1 and η_2 in the logistic regression problem using the dataset `sido0` (Fig. 6) and in the denoising problem (Fig. 10), more variation can be seen in the remaining two examples. In general, the performance seems to deteriorate when η_1 and η_2 are chosen to be very close to each other. The standard choice $\eta_1 = 2$ and $\eta_2 = 0.25$ again achieves convincing performance on all numerical examples and has a good balance when increasing and decreasing the weight parameter λ_k.

The convergence plot for different values of μ_0 are shown in Figs. 7, 11, and 15. Our observations are again somewhat similar: the performance of LM-AA on the logistic regression problem for `sido0` (Fig. 7) and on the denoising problem (Fig. 11) is very robust w.r.t. the choice of μ_0. In the nonnegative least squares problem, μ_0 appears to mainly affect the last convergence stage of the algorithm, i.e., different choices of μ_0 can lead to an earlier jump to a level with higher accuracy. Overall the parameters $\mu_0 = 1$ (for denoising and NNLS) and $\mu_0 = 100$ (for logistic regression) yield the most robust results.

D Ablation Study on Permutation Strategy

We also provide an ablation study for the permutation strategy in line 6 of Algorithm 1. In general, when the parameter γ is small, our nonmonotone globalization strategy is close to a monotone criterion on the residual. In this case, the minimal residual iteration k_0 mostly coincides with the current iteration number k and the permutation causes little difference. However, when γ is (relatively) large, then the usage of permutations can cause essential differences in the numerical performance. In particular, when the current trial step is rejected, then Algorithm 1 performs $x^{k+1} = g^k$ as next iteration if no permutation is used. In general,

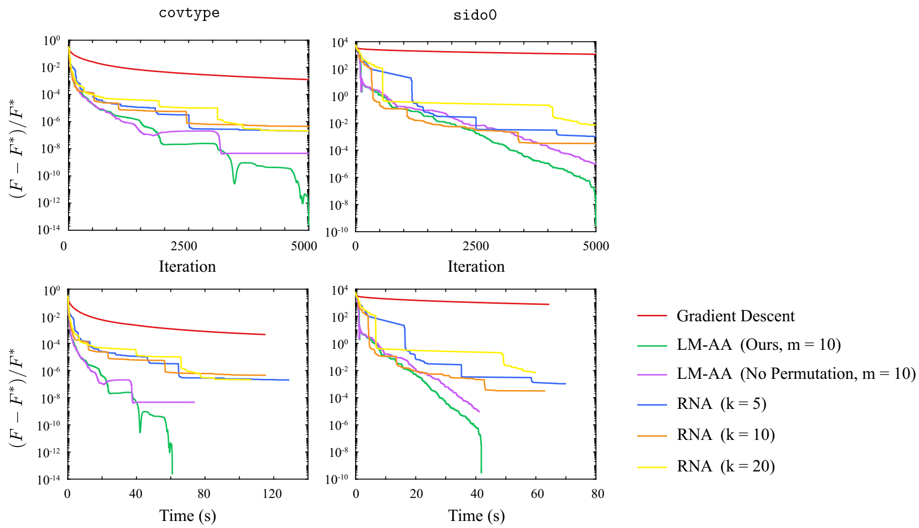

Fig. 16 The performance of LM-AA ($m = 10, \gamma = 0.05$) with or without the permutation strategy given in Eq. 6, for logistic regression on the `covtype` and `sido0` datasets

the update $x^{k+1} = g^k$ can be worse than $x^{k+1} = g^{k_0}$ since x^{k_0} has the smallest residual among the m latest iterations. We test Algorithm 1 without permutation in Fig. 16 for the logistic regression experiment. We set $\gamma = 0.05$ in Fig. 16 for LM-AA and keep other parameters unchanged. As can be seen from the figure, permutation improves the overall convergence and performance.

References

1. An, H., Jia, X., Walker, H.F.: Anderson acceleration and application to the three-temperature energy equations. J. Comput. Phys. **347**, 1–19 (2017)
2. Anderson, D.G.: Iterative procedures for nonlinear integral equations. J. ACM **12**(4), 547–560 (1965)
3. Anderson, D.G.M.: Comments on “Anderson acceleration, mixing and extrapolation”. Numer. Algorithms **80**(1), 135–234 (2019)
4. Bauschke, H.H., Combettes, P.L., et al.: Convex Analysis and Monotone Operator Theory in Hilbert Spaces. CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC. Springer, New York (2011)
5. Beck, A.: Introduction to nonlinear optimization. In: MOS-SIAM Series on Optimization, vol. 19. Society for Industrial and Applied Mathematics (SIAM); Mathematical Optimization Society, Philadelphia, PA. Theory, algorithms, and applications with MATLAB (2014)
6. Bian, W., Chen, X., Kelley, C.: Anderson acceleration for a class of nonsmooth fixed-point problems. SIAM J. Sci. Comput. **43**(5), S1–S20 (2021)
7. Both, J.W., Kumar, K., Nordbotten, J.M., Radu, F.A.: Anderson accelerated fixed-stress splitting schemes for consolidation of unsaturated porous media. Comput. Math. Appl. **77**(6), 1479–1502 (2019)
8. Byrne, C.: An elementary proof of convergence of the forward-backward splitting algorithm. J. Nonlinear Convex Anal. **15**(4), 681–691 (2014)
9. Clarke, F.H.: Optimization and nonsmooth analysis. In: Classics in Applied Mathematics, vol. 5, 2nd edn. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1990)
10. Conn, A.R., Gould, N.I.M., Toint, P.L.: Trust Region Methods. Society for Industrial and Applied Mathematics (SIAM), Mathematical Programming Society (MPS), Philadelphia (2000)
11. Ding, C., Sun, D., Sun, J., Toh, K.C.: Spectral operators of matrices: semismoothness and characterizations of the generalized Jacobian. SIAM J. Optim. **30**(1), 630–659 (2020)
12. Dontchev, A.L., Rockafellar, R.T.: Implicit Functions and Solution Mappings. Springer Series in Operations Research and Financial Engineering, 2nd edn. Springer, New York (2014). A view from variational analysis
13. Evans, C., Pollock, S., Rebholz, L.G., Xiao, M.: A proof that Anderson acceleration improves the convergence rate in linearly converging fixed-point methods (but not in those converging quadratically). SIAM J. Numer. Anal. **58**(1), 788–810 (2020)
14. Eyert, V.: A comparative study on methods for convergence acceleration of iterative vector sequences. J. Comput. Phys. **124**(2), 271–285 (1996)
15. Fan, J.Y.: A modified Levenberg-Marquardt algorithm for singular system of nonlinear equations. J. Comput. Math. **21**(5), 625–636 (2003)
16. Fang, H.R., Saad, Y.: Two classes of multisecant methods for nonlinear acceleration. Numer. Linear Algebra Appl. **16**(3), 197–221 (2009)
17. Fu, A., Zhang, J., Boyd, S.: Anderson accelerated Douglas–Rachford splitting. SIAM J. Sci. Comput. **42**(6), A3560–A3583 (2020)
18. Geist, M., Scherrer, B.: Anderson acceleration for reinforcement learning. arXiv preprint arXiv:1809.09501 (2018)
19. Giselsson, P., Boyd, S.: Linear convergence and metric selection for Douglas–Rachford splitting and ADMM. IEEE Trans. Autom. Control **62**(2), 532–544 (2016)
20. Levenberg, K.: A method for the solution of certain non-linear problems in least squares. Q. Appl. Math. **2**, 164–168 (1944)
21. Liang, J., Fadili, J., Peyré, G.: Activity identification and local linear convergence of forward-backward-type methods. SIAM J. Optim. **27**(1), 408–437 (2017)
22. Liang, J., Fadili, J., Peyré, G.: Local convergence properties of Douglas–Rachford and alternating direction method of multipliers. J. Optim. Theory Appl. **172**(3), 874–913 (2017)
23. Madsen, K., Nielsen, H., Tingleff, O.: Methods for Non-linear Least Squares Problems. Informatics and Mathematical Modelling, 2nd edn. Technical University of Denmark, Kongens Lyngby (2004)

24. Mai, V., Johansson, M.: Anderson acceleration of proximal gradient methods. In: Singh, A. H.D. III (ed.) Proceedings of the 37th International Conference on Machine Learning, Proceedings of Machine Learning Research, vol. 119, pp. 6620–6629. PMLR, Virtual (2020)
25. Mai, V.V., Johansson, M.: Nonlinear acceleration of constrained optimization algorithms. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4903–4907. IEEE (2019)
26. Marquardt, D.W.: An algorithm for least-squares estimation of nonlinear parameters. J. SIAM **11**, 431–441 (1963)
27. Matveev, S., Stadnichuk, V., Tyrtyshnikov, E., Smirnov, A., Ampilogova, N., Brilliantov, N.V.: Anderson acceleration method of finding steady-state particle size distribution for a wide class of aggregation-fragmentation models. Comput. Phys. Commun. **224**, 154–163 (2018)
28. Milzarek, A.: Numerical methods and second order theory for nonsmooth problems. Ph.D. thesis, Technische Universität München (2016)
29. Pavlov, A.L., Ovchinnikov, G.W., Derbyshev, D.Y., Tsetserukou, D., Oseledets, I.V.: AA-ICP: Iterative closest point with Anderson acceleration. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 1–6. IEEE (2018)
30. Peng, Y., Deng, B., Zhang, J., Geng, F., Qin, W., Liu, L.: Anderson acceleration for geometry optimization and physics simulation. ACM Trans. Graph. **37**(4), 42 (2018)
31. Poliquin, R.A., Rockafellar, R.T.: Generalized Hessian properties of regularized nonsmooth functions. SIAM J. Optim. **6**(4), 1121–1137 (1996)
32. Pollock, S., Rebholz, L.G.: Anderson acceleration for contractive and noncontractive operators. IMA J. Numer. Anal. **41**(4), 2841–2872 (2021)
33. Pollock, S., Rebholz, L.G., Xiao, M.: Anderson-accelerated convergence of Picard iterations for incompressible Navier–Stokes equations. SIAM J. Numer. Anal. **57**(2), 615–637 (2019)
34. Potra, F.A., Engler, H.: A characterization of the behavior of the Anderson acceleration on linear problems. Linear Algebra Appl. **438**(3), 1002–1011 (2013)
35. Pratapa, P.P., Suryanarayana, P., Pask, J.E.: Anderson acceleration of the Jacobi iterative method: an efficient alternative to Krylov methods for large, sparse linear systems. J. Comput. Phys. **306**, 43–54 (2016)
36. Qi, L.Q., Sun, J.: A nonsmooth version of Newton's method. Math. Program. **58**, 353–367 (1993)
37. Rockafellar, R.T., Wets, R.J.B.: Variational Analysis. Grundlehren der Mathematischen Wissenschaften, vol. 317, 3rd edn. Springer, Berlin (2009)
38. Rohwedder, T., Schneider, R.: An analysis for the DIIS acceleration method used in quantum chemistry calculations. J. Math. Chem. **49**(9), 1889–1914 (2011)
39. Scieur, D., d'Aspremont, A., Bach, F.: Regularized nonlinear acceleration. In: Advances in Neural Information Processing Systems, pp. 712–720 (2016)
40. Scieur, D., d'Aspremont, A., Bach, F.: Regularized nonlinear acceleration. Math. Program. **179**(1), 47–83 (2020)
41. Stella, L., Themelis, A., Patrinos, P.: Forward–backward quasi-Newton methods for nonsmooth optimization problems. Comput. Optim. Appl. **67**(3), 443–487 (2017)
42. Sterck, H.D.: A nonlinear GMRES optimization algorithm for canonical tensor decomposition. SIAM J. Sci. Comput. **34**(3), A1351–A1379 (2012)
43. Sun, D., Sun, J.: Strong semismoothness of eigenvalues of symmetric matrices and its application to inverse eigenvalue problems. SIAM J. Numer. Anal. **40**(6), 2352–2367 (2002)
44. Sun, D., Sun, J.: Strong semismoothness of the Fischer–Burmeister SDC and SOC complementarity functions. Math. Program. **103**(3), 575–581 (2005)
45. Toth, A., Ellis, J.A., Evans, T., Hamilton, S., Kelley, C., Pawlowski, R., Slattery, S.: Local improvement results for Anderson acceleration with inaccurate function evaluations. SIAM J. Sci. Comput. **39**(5), S47–S65 (2017)
46. Toth, A., Kelley, C.: Convergence analysis for Anderson acceleration. SIAM J. Numer. Anal. **53**(2), 805–819 (2015)
47. Ulbrich, M.: Nonmonotone trust-region methods for bound-constrained semismooth equations with applications to nonlinear mixed complementarity problems. SIAM J. Optim. **11**(4), 889–917 (2001)
48. Ulbrich, M.: Semismooth Newton Methods for Variational Inequalities and Constrained Optimization Problems in Function Spaces. MOS-SIAM Series on Optimization, vol. 11. Society for Industrial and Applied Mathematics (SIAM), Mathematical Optimization Society, Philadelphia (2011)
49. Walker, H.F., Ni, P.: Anderson acceleration for fixed-point iterations. SIAM J. Numer. Anal. **49**(4), 1715–1735 (2011)
50. Wang, D., He, Y., De Sterck, H.: On the asymptotic linear convergence speed of Anderson acceleration applied to ADMM. J. Sci. Comput. **88**(2), 1–35 (2021)

51. Wang, Y., Yang, J., Yin, W., Zhang, Y.: A new alternating minimization algorithm for total variation image reconstruction. SIAM J. Imaging Sci. **1**(3), 248–272 (2008)
52. Willert, J., Park, H., Taitano, W.: Using Anderson acceleration to accelerate the convergence of neutron transport calculations with anisotropic scattering. Nucl. Sci. Eng. **181**(3), 342–350 (2015)
53. Zhang, J., O'Donoghue, B., Boyd, S.: Globally convergent type-I Anderson acceleration for nonsmooth fixed-point iterations. SIAM J. Optim. **30**(4), 3170–3197 (2020)
54. Zhang, J., Peng, Y., Ouyang, W., Deng, B.: Accelerating ADMM for efficient simulation and optimization. ACM Trans. Graph. **38**(6), 1–21 (2019)