Oak Ridge National Laboratory

May 8, 2026

Two-Level Sketching Alternating Anderson Acceleration for Complex Physics Applications

Massimiliano (Max) Lupo-Pasini

Oak Ridge National Laboratory

U.S. DEPARTMENT of ENERGY

ORNL IS MANAGED BY UT-BATTELLE LLC
FOR THE US DEPARTMENT OF ENERGY

Contributors

- Paul Laiu, Computer Science and Mathematics Division, Oak Ridge National Laboratory

- Nicolas Barnafi, University of Santiago, Chile

Outline

Motivation and Narrative Roadmap

Anderson Acceleration and Alternating Anderson Acceleration

Linear Case

- Theoretical Results
- Main Result: Theorem
- Corollary (Heuristic to Use in Applications)
- Numerical Results

Linear-to-Nonlinear Bridge

Non-linear Case

- Two-Level Sketching and Adaptive Selection
- Numerical Results

Conclusions and Future Developments

Motivation: A Fixed-Point Solver Design Tension

Large-scale physics solvers need iterations that are simple enough to scale, but strong enough to converge.

Why fixed-point iterations?

- Picard/Richardson steps use local kernels such as residual evaluations and sparse matrix-vector products.
- They are attractive on distributed-memory systems because the operations are simple and parallel-friendly.
- **But the baseline iteration may converge slowly or fail when the map is weakly contractive or non-contractive.**

Why Anderson acceleration?

- It reuses recent residual/update history to compute a small least-squares mixing correction.
- It can reduce stagnation, improve robustness, and accelerate slow fixed-point sequences.
- **The price is a dense global least-squares problem tied to communication and memory movement.**

Why this talk?

- Alternation reduces how often Anderson mixing is applied.
- **Alternating multiple Picard iterations with one single Anderson update improves robustness of the scheme against stagnations**
- **Inexact/sketched least-squares reduces the cost of each mixing step.**
- **Physics-aware projection makes the reduction practical for nonlinear coupled PDEs.**

Central question: Can we keep Anderson's convergence benefit while paying only a small, stable, hardware-aware fraction of its cost?

Anderson Acceleration

[D. G. Anderson, 1965]

Given a fixed-point iteration $\mathbf{x}^{k+1} = G(\mathbf{x}^k), \quad k = 0, 1, \ldots$

AA computes a mixing of previous updates so that the new sequence $\overline{\mathbf{w}}_k$ converges faster.

(1) **New sequence** $\overline{\mathbf{x}}_k = \mathbf{x}_k - \sum_{i=1}^{p} g_j^k (\mathbf{x}_{k-p+j} - \mathbf{x}_{k-p+j-1})$ **Ensures better exploration of the loss function landscape**

Mixing to prevent stagnation

Formula (1) to compute AA can be recast as $\overline{\mathbf{x}}^k = \mathbf{x}^k + \mathbf{r}^k - (X_k + R_k)\mathbf{g}^k$

Using $X_k = [(\mathbf{x}^{k-m+1} - \mathbf{x}^{k-m}), (\mathbf{x}^{k-m+2} - \mathbf{x}^{k-m+1}), \ldots, (\mathbf{x}^k - \mathbf{x}^{k-1})] \in \mathbb{R}^{n\times m}$

$R_k = [(\mathbf{r}^{k-m+1} - \mathbf{r}^{k-m}), (\mathbf{r}^{k-m+2} - \mathbf{r}^{k-m+1}), \ldots, (\mathbf{r}^k - \mathbf{r}^{k-1})] \in \mathbb{R}^{n\times m}$

The vector of mixing coefficients $\mathbf{g}^k$ in AA can be computed by solving the following least-squares (LS) problem

(2) $\mathbf{g}^k = \underset{\mathbf{g}\in\mathbb{R}^m}{\operatorname{argmin}} \|\mathbf{r}^k - R_k\mathbf{g}\|_2^2$ The dimension m is a tunable parameter

Solving a least-squares at each iteration causes severe bottlenecks for parallelization

Alternating Anderson Acceleration

To reduce the number of least-squares to solve, a variant of AA called ***Alternating Anderson*** was proposed by Pratapa et al. 2016

$$\mathbf{x}^{k+1} = \mathbf{x}^k + C_k \mathbf{r}^k$$

$$C_k = \begin{cases} I, & k/p \notin \mathbb{N} \\ (1-\beta)I - \beta(W_k + R_k)(R_k^T R_k)^{-1} R_k^T, & k/p \in \mathbb{N} \end{cases}$$

Solving global least-squares, even if less frequently, is still expensive on large scale applications

Solution: replace solving the exact least-squares problem with solving an approximate one

[1] P. P. Pratapa, P. Suryanarayana, and J. E. Pask, “Anderson acceleration of the Jacobi iterative method: An efficient alternative to Krylov methods for large, sparse linear systems,” *Journal of Computational Physics*, vol. 306, pp. 43–54, 2016. DOI: 10.1016/j.jcp.2015.11.018.
[2] P. Suryanarayana, P. P. Pratapa, and J. E. Pask, “Alternating Anderson–Richardson method: An efficient alternative to preconditioned Krylov methods for large, sparse linear systems,” *Computer Physics Communications*, vol. 234, pp. 278–285, 2019. DOI: 10.1016/j.cpc.2018.07.007.

Narrative Roadmap: Three Questions, Three Contributions

The talk is organized as a progression, from reducing the frequency of global corrections to making each correction stable, cheaper, and physics-aware.

2019 **AAR foundation**

Why does reducing the frequency of global Anderson steps help?

?

→

2024 **Stable inexact AA/AAR**

How much least-squares inexactness is safe?

?

→

2026 **Two-level sketching**

How do we make this practical for nonlinear coupled physics?

?

Unifying principle: use problem structure first, then approximate only as much as stability indicators allow.

OAK RIDGE National Laboratory

AAR foundation

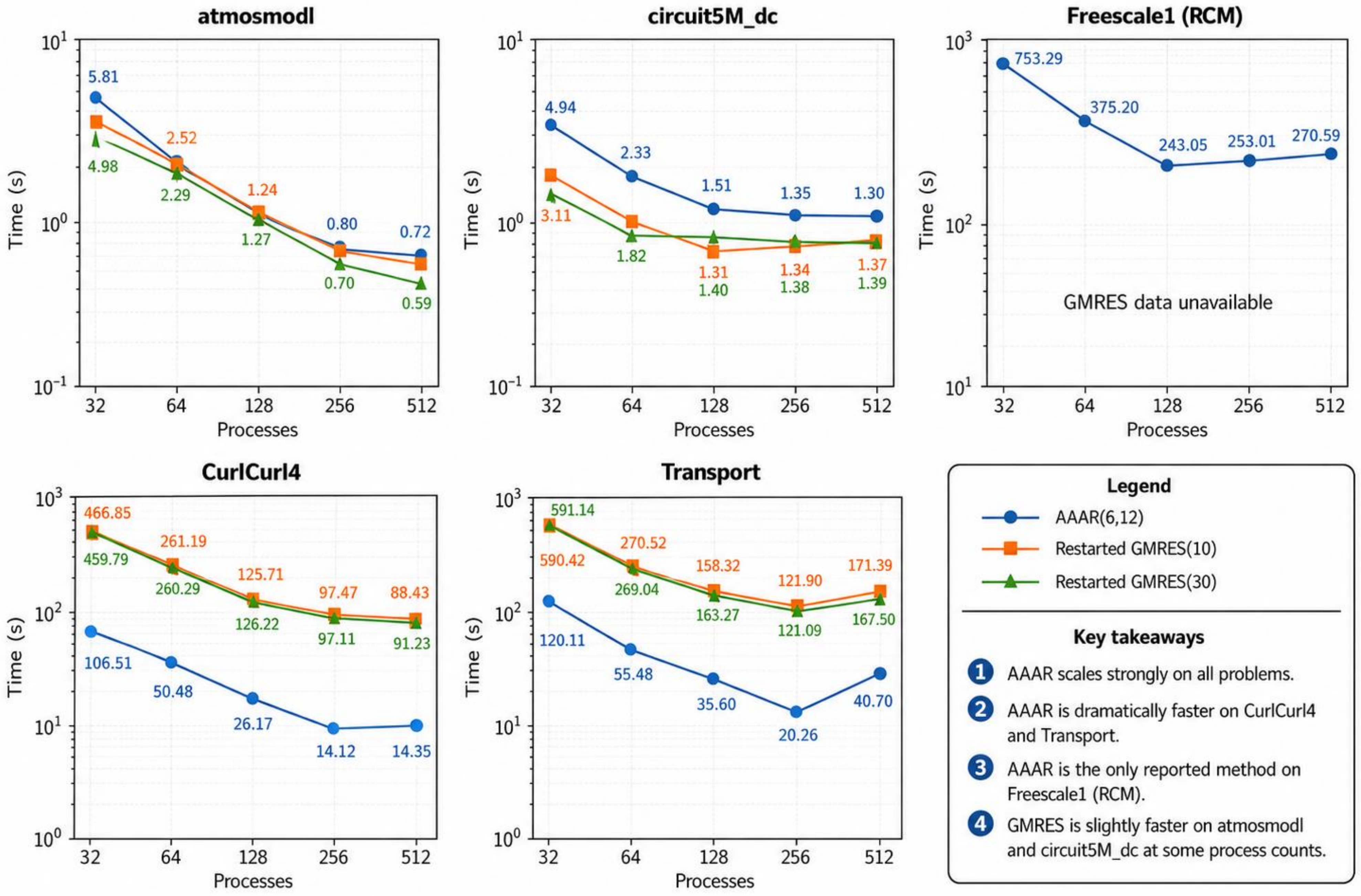
Why does reducing the frequency of global Anderson steps help?

M. Lupo Pasini, *Convergence analysis of Anderson-type acceleration of Richardson's iteration*, Numerical Linear Algebra with Applications, Volume 26, Issue 4 August 2019, e2241, 2019 https://onlinelibrary.wiley.com/doi/full/10.1002/nla.2241

Strong Scaling Comparison along Benchmark Problems

MPI experiments: Block-ILU(0) preconditioner with 1,024 diagonal blocks

Alternating Anderson Richardson reduces communication overhead compared to state-of-the-art iterative linear solvers



Oak Ridge National Laboratory

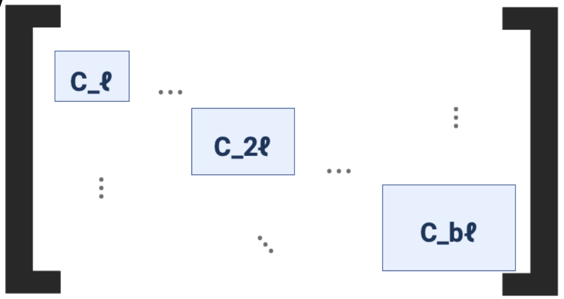
Block-circulant Test Matrix in the Stagnation Study

Matrix *A*: coefficient matrix *A* is block diagonal with b blocks. Each block of *A* has increasing size so that the kth block is an $\ell k \times \ell k$ circulant matrix, where ℓ represents an integer

Right-hand side *b*

Choose $x_0 = 0$ and set *b* so that only the first entry of each block is nonzero.

A = diag(C_ℓ, C_2ℓ, ..., C_bℓ)



Example circulant permutation block C_n

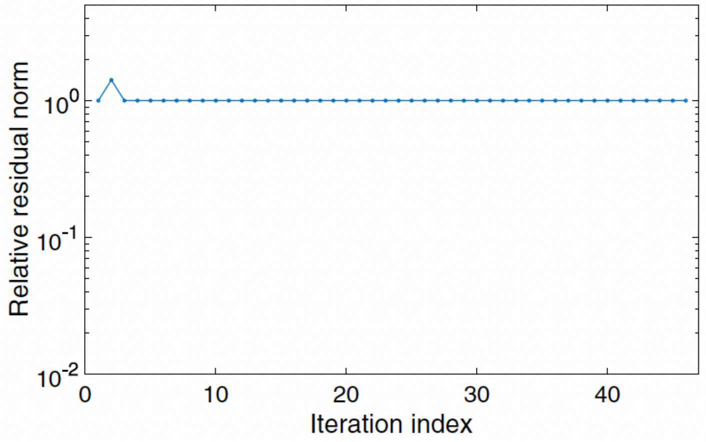
$$\begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}$$

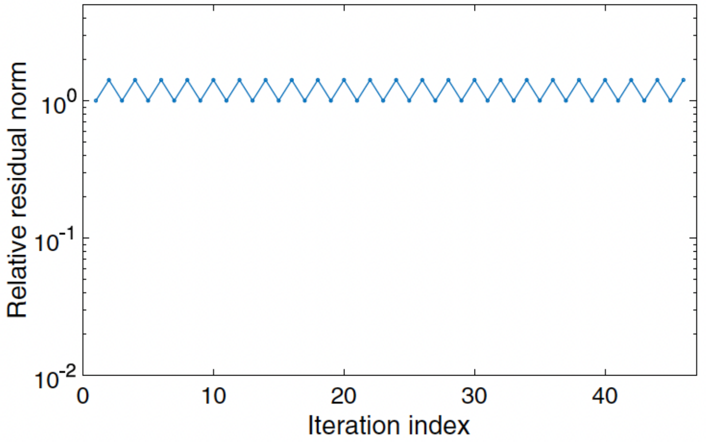
Key implication: this construction produces ℓ consecutive stagnation steps.
AR (p = 1) stagnates badly, while AAR converges when p is chosen large enough ($p > \ell - 1$).

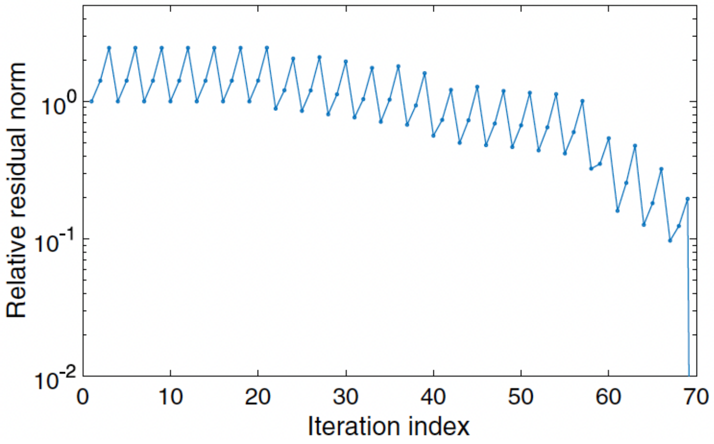
AR / p=1
residual never decreases

AAR p=2
oscillates; still fails

AAR p=3
recovers; finite convergence







Narrative Roadmap: Three Questions, Three Contributions

2019 **AAR foundation**

Why does reducing the frequency of global Anderson steps help?

- Reduced time to solution
- Improved scalability
- Improved robustness against stagnations

2024 **Stable inexact AA/AAR**

How much inexactness in Picard evaluations and/or least-squares is safe?

?

2026 **Two-level sketching**

How do we make this practical for nonlinear coupled physics?

?

Unifying principle: use problem structure first, then approximate only as much as stability indicators allow.

Oak Ridge National Laboratory

Linear Case

M.L.P., Paul Laiu, *Anderson acceleration with approximate calculations: Applications to scientific computing, Numerical Linear Algebra with Application*, Volume 31, Issue 5, October 2024, e2562, https://onlinelibrary.wiley.com/doi/full/10.1002/nla.2562

Main Result: Theorem (ensures that approximate calculations do not compromise convergence)

Consider the perturbed least-squares problem to compute the coefficients of the Anderson mixing

$$(R_k + \mathcal{E}_k)\mathbf{g}_k = \mathbf{r}^k$$

Perturbation on left-hand side (LHS)

with $R_k + \mathcal{E}_k = R_k + [E_0(\mathbf{r}^1 - \mathbf{r}^0), \ldots, E_k(\mathbf{r}^k - \mathbf{r}^{k-1})] = R_k - [E_0 A(\mathbf{x}^1 - \mathbf{x}^0), \ldots, E_k A(\mathbf{x}^k - \mathbf{x}^{k-1})]$

Let $\epsilon > 0, \quad \varepsilon = \epsilon \|A\|$

Possible due to assumption of linearity

Define the perturbation δ_k to the left-hand side of the least-squares as

$$\delta_k = \|[E_0 A(\mathbf{x}^1 - \mathbf{x}^0), \ldots, E_k A(\mathbf{x}^k - \mathbf{x}^{k-1})]\mathbf{g}^k\| \leq \sum_{i=1}^{k} |g_i^{(k)}| \|E_i\| \|A\| \|\mathbf{x}^i - \mathbf{x}^{i-1}\|$$

Define k as an iteration where AAR performs an Anderson mixing $R_k + \mathcal{E}_k = \hat{Q}_k \hat{T}_k$ (QR factorization)

If $\|E_i\| \leq \dfrac{\sigma_{\min}(\hat{T}_k)}{k} \dfrac{1}{\|\mathbf{r}^k\|} \dfrac{1}{\|\mathbf{x}^i - \mathbf{x}^{i-1}\|} \epsilon$

Then $\delta_k \leq \displaystyle\sum_{i=1}^{k} \frac{1}{\sigma_{\min}(\hat{T}_k)} \|\mathbf{r}^k\| \|A\| \|E_i\| \|\mathbf{x}^i - \mathbf{x}^{i-1}\| \leq \sum_{i=1}^{k} \frac{1}{k} \epsilon \|A\| = \varepsilon$

Proof of the Theorem: Lemma

Assume that $k = p\ell$ iterations of Full AAR have been carried out and the iteration has not stagnated for more than p consecutive steps.

Let $\mathbf{g}^k$ be the Anderson mixing computed by solving the perturbed least-squares problem

$$(R_k + \mathcal{E}_k)\mathbf{g}_k = \mathbf{r}_k$$

Then, the following inequality holds

$$|g_i^{(k)}| \leq \frac{1}{\sigma_{\min}(\hat{T}_k)} \|\mathbf{r}^k\| \quad \text{for any} \quad i = 1, \dots, k$$

with

$$R_k + \mathcal{E}_k = \hat{Q}_k \hat{T}_k \quad \text{(QR factorization)}$$

Proof of the Theorem (cont.'d)

The upper bound from the Lemma leads to the following chain of inequalities

Lemma

$$\delta_k \leq \sum_{i=1}^{k} |g_i^{(k)}| \|E_i\| \|\mathbf{x}^i - \mathbf{x}^{i-1}\| \leq \sum_{i=1}^{k} \frac{1}{\sigma_{\min}(\hat{T}_k)} \|\mathbf{r}^k\| \|A\| \|E_i\| \|\mathbf{x}^i - \mathbf{x}^{i-1}\|$$

IF for each i

Expensive to estimate

$$\|E_i\| \leq \frac{\sigma_{\min}(\hat{T}_k)}{k} \frac{1}{\|\mathbf{r}^k\|} \frac{1}{\|\mathbf{x}^i - \mathbf{x}^{i-1}\|} \epsilon$$

THEN

$$\delta_k \leq \sum_{i=1}^{k} \frac{1}{\sigma_{\min}(\hat{T}_k)} \|\mathbf{r}^k\| \|A\| \|E_i\| \|\mathbf{x}^i - \mathbf{x}^{i-1}\| \leq \sum_{i=1}^{k} \frac{1}{k} \epsilon \|A\| = \varepsilon$$

As a result, larger errors (systematic or noise induced) can be afforded as we approach convergence

OAK RIDGE National Laboratory

Type of Perturbations Affecting the LS Problem

Perturbation affecting only left-hand side (LHS)

$$(R_k + \mathcal{E}_k)\mathbf{g}_k = \mathbf{r}^k \quad \Rightarrow \quad \|E_i\| \leq \frac{\sigma_{\min}(\hat{T}_k)}{k} \frac{1}{\|\mathbf{r}^k\|} \frac{1}{\|\mathbf{x}^i - \mathbf{x}^{i-1}\|} \epsilon$$

Perturbation affecting both LHS and right-hand side (RHS)

$$(R_k + \mathcal{E}_k)\mathbf{g}_k = \mathbf{r}^k + \delta\mathbf{r}^k \quad \Rightarrow \quad \|E_i\| \leq \frac{\sigma_{\min}(\hat{T}_k)}{k} \frac{1}{\|\mathbf{r}^k\|} \frac{1}{\|\mathbf{x}^i - \mathbf{x}^{i-1}\|} \frac{\epsilon}{1+\epsilon}$$

Allowing for more sources of error leads to more stringent requirements on accuracy to maintain backward stability

How to Read the Theorem

PROS

Approximation error in the least-squares can be tolerated.

The admissible error can grow as the residual decreases.

The residual target is preserved if perturbations stay within the bound.

CONS

Error injected cannot be arbitrarily large. Its magnitude must be judiciously controlled.

The theoretical constants computationally expensive.

This motivates computable heuristics plus a monotonicity check rather than direct use of every theoretical quantity.

Corollary (ensures that an appropriate use of heuristic still preserves convergence)

Assume that $k = p\ell$ iterations of Full AAR have been carried out and the iteration has not stagnated for more than p consecutive steps.

Define $$B_i = \frac{\gamma_i}{k} \frac{1}{\|\mathbf{r}^i\|} \frac{1}{\|\mathbf{x}^i - \mathbf{x}^{i-1}\|} \qquad i = 1, \ldots, k$$ $\gamma_i > 0$ for any iteration *i* with $i \leq k$.

If $\|E_i\| \leq B_i \epsilon$ and $\|\mathbf{r}^k\| < \|\mathbf{r}^{\ell(p-1)}\|$ **Monotonicity of the residual across successive Anderson steps**

Then $\delta_k < \epsilon$

The corollary is both practical and rigorous

The algorithm does not need to enforce the exact theoretical error bound at every step. Instead, it uses a computable surrogate bound plus a residual-monotonicity safeguard.

OAK RIDGE National Laboratory

Numerical Results: Linear Case

Richardson's scheme to iteratively solve linear systems

We consider the diagonal matrix $A = \mathrm{diag}(10^{-4}, 2, 3, \ldots, 100)$

The perturbation matrices E_k's, with $\|E_k\| = 1 \quad \text{for} \quad k > 0$ are random 100 ×100 matrices

generated with normally distributed values using the Matlab function randn.

$$\left(R_k + \epsilon_k \|R_k\| \|E_k\|\right) \mathbf{g} = \mathbf{r}^k$$

The quantity ϵ_k is used to tune the magnitude of the noise injected in the LHS of the linear system

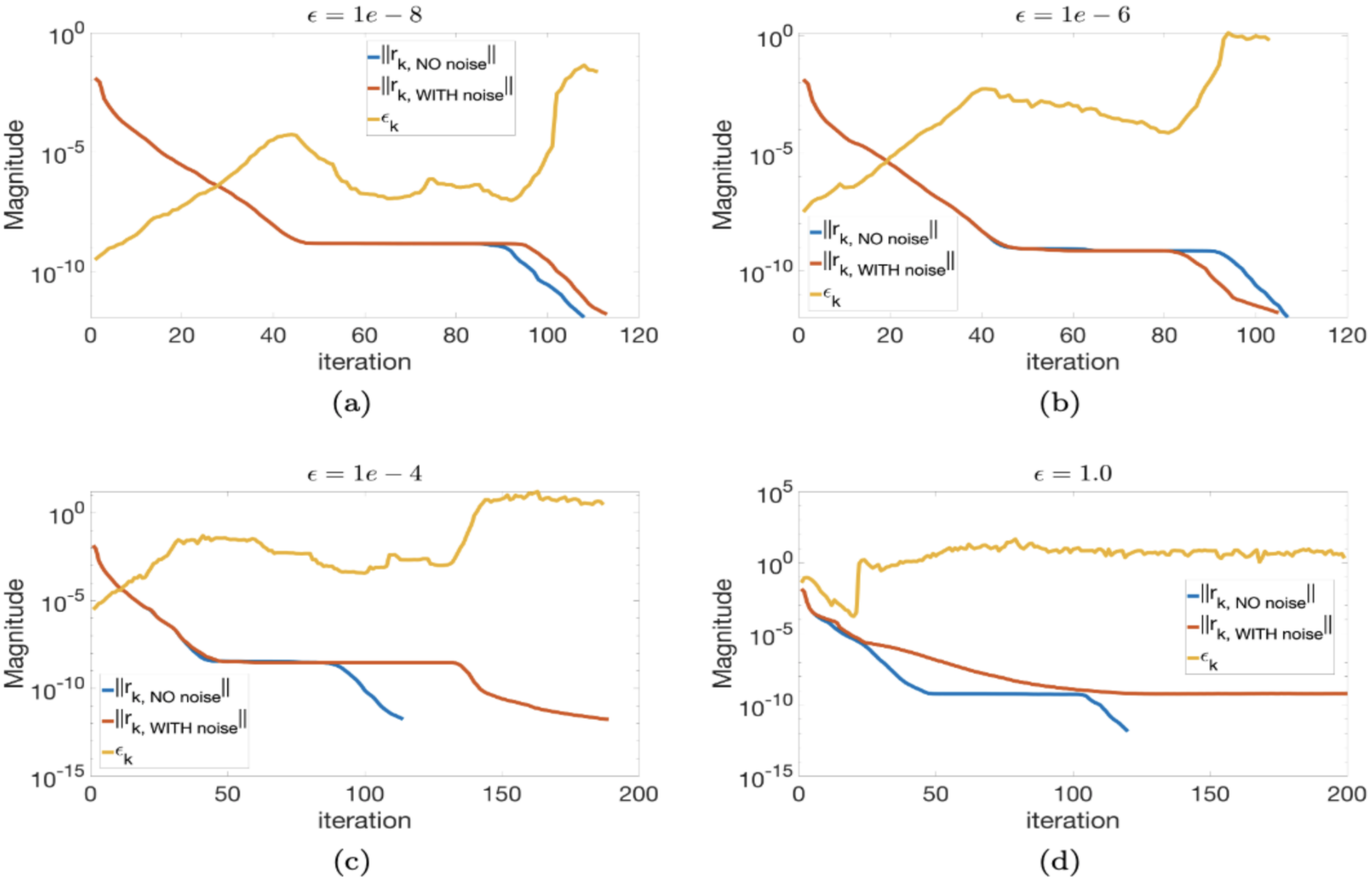
$$\epsilon_k = \frac{\epsilon}{k^*} \frac{\sigma_{\min}(T_k)}{\|\mathbf{r}^k\| \|\mathbf{x}^k - \mathbf{x}^{k-1}\|}$$

We set $k^* = 100$ accordingly to the size of the linear system we are solving.

Numerical Results: Linear Case (cont’d)

This experiment is a numerical stress test of the paper’s main idea: **AA/AAR can tolerate approximate least-squares calculations, but the amount of tolerated inaccuracy must be tied to the residual size and the conditioning of the least-squares matrix.**

As guaranteed by the theorem, the figures show that while the error goes up at each iteration the process still converges



OAK RIDGE National Laboratory

Numerical Results: linear case (cont'd)

MM = Matrix Market collection
https://math.nist.gov/MatrixMarket/

SS = Suite Sparse Matrix Collection
https://sparse.tamu.edu

Matrix	Type	Size	Structure	Pos. def.	Source
fidap029	Real	2870	Nonsymmetric	Yes	MM
raefsky5	Real	6316	Nonsymmetric	Yes	SS
bcsstk29*	Real	13,992	Symmetric	No	SS
sherman3*	Real	5005	Nonsymmetric	No	MM
sherman5*	Real	3312	Nonsymmetric	No	MM
chipcool0	Real	20,082	Nonsymmetric	No	SS
e20r0000*	Real	4241	Nonsymmetric	No	MM
spmsrtls	Real	29,995	Nonsymmetric	No	SS
garon1	Real	3175	Nonsymmetric	No	SS
garon2	Real	13,535	Nonsymmetric	No	SS
memplus	Real	17,758	Nonsymmetric	No	SS
saylr4	Real	3564	Nonsymmetric	No	MM
xenon1*	Real	48,600	Nonsymmetric	No	SS
xenon2*	Real	157,464	Nonsymmetric	No	SS
venkat01	Real	62,424	Nonsymmetric	No	SS
QC2534	Complex	2534	Non-Hermitian	No	SS
mplate*	Complex	5962	Non-Hermitian	No	SS
light_in_tissue	Complex	29,282	Non-Hermitian	No	SS
kim1	Complex	38,415	Non-Hermitian	No	SS
chevron2	Complex	90,249	Non-Hermitian	No	SS

Oak Ridge National Laboratory

Numerical Results: linear case (cont'd)

Matrix	Total wall-clock time for least-squares / Total wall-clock time for Richardson's steps: Alternating AA($m = k$) ILU(0)	Alternating AA($m = k$) ILUT(τ)	Subselect. Altern. AA($m = k$) ILU(0)	Subselect. Altern. AA($m = k$) ILUT(τ)	Random. Altern. AA($m = k$) ILU(0)	Random. Altern. AA($m = k$) ILUT(τ)
fidap029	13.71	39.85	4.09	12.01	3.72	3.95
raefsky5	23.44	33.14	6.18	9.21	5.80	9.37
bcsstk29	2.02	2.17	2.23	1.81	1.49	1.21
sherman3	14.30	11.17	9.84	3.71	9.45	3.16
sherman5	36.98	35.79	9.1	10.07	8.19	13.23
chipcool0	33.74	8.41	9.53	2.60	6.34	2.18
e20r0000	24.77	7.37	12.96	3.11	15.33	3.02
spmsrtls	54.83	72.02	10.25	15.04	8.80	13.51
garon1	21.95	6.09	17.79	3.07	2.67	3.01
garon2	29.29	4.28	30.01	1.67	6.64	1.61
memplus	—	44.05	—	9.81	—	7.41
saylr4	38.31	33.04	12.17	10.43	10.19	6.56
xenon1	—	9.39	—	2.94	—	2.76
xenon2	—	10.02	—	2.19	—	1.91
venkat01	30.61	9.26	6.61	2.71	5.16	2.10
QC2534	5.94	7.14	5.79	3.48	5.71	3.42
mplate	—	7.15	—	7.61	—	6.11
light_in_tissue	47.23	20.36	10.29	4.30	10.22	4.30
kim1	37.23	31.92	8.40	6.62	8.97	6.21
chevron2	66.01	18.60	72.47	4.64	47.41	4.02

Take-Home Message from the Linear Case

- Theoretical bounds provide useful insights to judiciously inject inaccuracy in the calculations without compromising the final accuracy of the physics-based solver.

- Numerical results on indefinite linear systems show clear benefit of heuristics in drastically reducing the computational time compared to stat-of-the-art solvers.

Non-Linear Case

Nicolas Barnafi, M. L. P., *Two-Level Sketching Alternating Anderson acceleration for Complex Physics Applications*, SIAM Journal of Scientific Computing (SISC), **Accepted (04/14/2026)**, ArXiv preprint available at https://arxiv.org/abs/2505.08587

Anderson Acceleration

[D. G. Anderson, 1965]

Given a fixed-point iteration $\mathbf{x}^{k+1} = G(\mathbf{x}^k), \quad k = 0, 1, \ldots$

AA computes a mixing of previous updates so that the new sequence $\overline{\mathbf{x}}^k$ converges faster.

(1)
$$\overline{\mathbf{x}}_k = \mathbf{x}_k - \sum_{i=1}^{p} g_j^k (\mathbf{x}_{k-p+j} - \mathbf{x}_{k-p+j-1})$$

Formula (1) to compute AA can be recast as $\overline{\mathbf{x}}^k = \mathbf{x}^k + \mathbf{r}^k - (X_k + R_k)\mathbf{g}^k$

Using $X_k = [(\mathbf{x}^{k-m+1} - \mathbf{x}^{k-m}), (\mathbf{x}^{k-m+2} - \mathbf{x}^{k-m+1}), \ldots, (\mathbf{x}^k - \mathbf{x}^{k-1})] \in \mathbb{R}^{n \times m}$

$$R_k = [(\mathbf{r}^{k-m+1} - \mathbf{r}^{k-m}), (\mathbf{r}^{k-m+2} - \mathbf{r}^{k-m+1}), \ldots, (\mathbf{r}^k - \mathbf{r}^{k-1})] \in \mathbb{R}^{n \times m}$$

The vector of mixing coefficients $\mathbf{g}^k$ in AA can be computed by solving the following least-squares (LS) problem

(2) $$\mathbf{g}^k = \operatorname*{argmin}_{\mathbf{g} \in \mathbb{R}^m} \|\mathbf{r}^k - R_k \mathbf{g}\|_2^2$$

The dimension m is a tunable parameter

OAK RIDGE National Laboratory

Main Result: Theorem

Consider the perturbed least-squares problem to compute the coefficients of the Anderson mixing

$$(R_k + \mathcal{E}_k)\mathbf{g}_k = \mathbf{r}^k$$

with $R_k + \mathcal{E}_k = R_k + [E_0(\mathbf{r}^1 - \mathbf{r}^0), \ldots, E_k(\mathbf{r}^k - \mathbf{r}^{k-1})]$

Denote the Lipschitz constant of the fixed-point operator G with L such that: $\|\mathbf{r}^k - \mathbf{r}^{k-1}\| \leq L\|\mathbf{w}^k - \mathbf{w}^{k-1}\|$

Define the perturbation δ_k to the left-hand side of the least-squares as

$$\delta_k = \|[E_0(\mathbf{r}^1 - \mathbf{r}^0), \ldots, E_k(\mathbf{r}^k - \mathbf{r}^{k-1})]\mathbf{g}^k\| \leq \sum_{i=1}^{k} L\|\mathbf{g}^k\|_2\|E_i\|\|\mathbf{x}^k - \mathbf{x}^{k-1}\|$$

Define an arbitrary sequence $\{\eta_i\}_{i=1}^k, \quad \eta_i \geq 0 \quad \text{for} \quad i = 0, 1, \ldots, k$ such that $\sum_{i=1}^{k} \eta_i \leq \varepsilon$

Define k as an iteration where AAR performs an Anderson mixing

If $\|E_i\| \leq \eta_i \dfrac{\sigma_{\min}(\hat{T}^k)}{L\|\mathbf{r}^k\|\|(\mathbf{x}^i - \mathbf{x}^{i-1})\|}$ Expensive to estimate

$R_k + \mathcal{E}_k = \hat{Q}_k\hat{T}_k$ (QR factorization)

Then $\delta^k \leq \sum_{i=1}^{k} \dfrac{L}{\sigma_{min}(\hat{T}^k)}\|\mathbf{r}^k\|\|E_i\|\|\mathbf{x}^i - \mathbf{x}^{i-1}\| \leq \sum_{i=1}^{k} \eta_j \leq \varepsilon$

Type of Perturbations Affecting the LS Problem

Perturbation affecting only left-hand side (LHS)

$$(R_k + \mathcal{E}_k)\mathbf{g}_k = \mathbf{r}^k \quad \Rightarrow \quad \|E_i\| \leq \eta_i \frac{\sigma_{\min}(\hat{T}^k)}{L\|\mathbf{r}^k\|\|(\mathbf{x}^i - \mathbf{x}^{i-1})\|}$$

Perturbation affecting both LHS and right-hand side (RHS)

$$(R_k + \mathcal{E}_k)\mathbf{g}_k = \mathbf{r}^k + \delta\mathbf{r}^k \quad \Rightarrow \quad \|E_i\| \leq \eta_i \frac{\sigma_{min}(\hat{T}^k)}{L\|\mathbf{r}^k\|\|(\mathbf{x}^i - \mathbf{x}^{i-1})\|} \frac{1}{1+\varepsilon}$$

Allowing for more sources of error leads to more stringent requirements on accuracy to maintain backward stability

Bridge: What Changes from Linear to Nonlinear?

Linear case

Residual differences satisfy exact algebraic identities with the error.

Perturbations can be related directly to matrix quantities.

The stability story is cleaner and more explicit.

Nonlinear case

Exact identities between residuals and errors are replaced by Lipschitz-type control.

Local behavior of the fixed-point map must be estimated.

The same backward-stability philosophy still guides sketching.

The nonlinear theory keeps the same message, but the implementation must estimate stability thresholds adaptively.

Heuristics for the Non-Linear Case

Since calculating $\sigma_{min}(\hat{\mathbf{R}}^k)$ and L is computationally expensive, we approximate them as follows :

- Estimate $\sigma_{min}(\hat{\mathbf{R}}^k)$ by performing a few (e.g., between 1 and 5) iterations of the inverse power method to the matrix $[\hat{\mathbf{R}}^k]^T\hat{\mathbf{R}}^k$ (CT Kelley. Iterative methods for linear and nonlinear equations. SIAM, 1995.)

- Estimate L by keeping track of the increment norms so that $L \approx L_k = \max\left(L_{k-1}, \frac{\|\mathbf{r}^k - \mathbf{r}^{k-1}\|}{\|\mathbf{x}^k - \mathbf{x}^{k-1}\|}\right)$

OAK RIDGE National Laboratory

Narrative Roadmap: Three Questions, Three Contributions

2019 **AAR foundation**

Why does reducing the frequency of global Anderson steps help?

- Reduced time to solution
- Improved scalability
- Improved robustness against stagnations

2024/2026 **Stable inexact AA/AAR**

How much inexactness in Picard evaluations and/or least-squares is safe?

- Larger errors are tolerated as convergence is reached
- Theoretical guidelines allow reduction of computational cost without affecting final accuracy

2026 **Two-level sketching**

How do we make this practical for nonlinear coupled physics?

?

Unifying principle: use problem structure first, then approximate only as much as stability indicators allow.

2026

Physics-Informed Two-Level Sketching

Nicolas Barnafi, M. L. P., *Two-Level Sketching Alternating Anderson acceleration for Complex Physics Applications,* SIAM Journal of Scientific Computing (SISC), **Accepted (04/14/2026)**, ArXiv preprint available at https://arxiv.org/abs/2505.08587

Challenges to Apply AAP to Multi-Physics

1. Different physical quantities can attain values of largely different orders of magnitude

2. Combining these physical quantities can affect the conditioning of the LS problems solved to compute the Anderson mixing, which in turn affects the convergence of the AAP scheme

Proposed Solution: Two-Level Projection for LS Problem

Multi-physics problems can be recast as a system of only one of their variables without losing information (e.g., Schur complement in linear systems)

First Level:
- Physics-driven projection of the LS onto subspace identified by one physics field (e.g., projection onto pressure/velocity subspace for Stokes problem)

- Projection subspace is static throughout iterative process → **allows to conveniently pre-allocate the memory needed once at the beginning of the execution of the algorithm**

Second Level (nested within the first level):
- Adaptive selection of the rows in the LS problem using heuristics from backward stability

OAK RIDGE National Laboratory

Second Level - Adaptive Selection of the Rows in the LS Problem

Goal: use theoretical bound $\|E_i\| \leq \eta_i \frac{\sigma_{min}(\hat{\mathbf{R}}^k)}{L\|\mathbf{r}^k\|\|(\mathbf{x}^i - \mathbf{x}^{i-1})\|}$ to guide adaptive sketching

When adaptive sketching is used for sub-selection of entries in the LS problem, the matrix E_i is a projection operator

Denote the projection operator onto the subspace of retained entries with P_i

$$E_i = I - P_i \Rightarrow \|E_i\| = 1$$

$$\|E_i\| \leq \eta_i \frac{\sigma_{min}(\hat{\mathbf{R}}^k)}{L\|\mathbf{r}^k\|\|(\mathbf{x}^i - \mathbf{x}^{i-1})\|} \frac{1}{1+\varepsilon} \quad \Rightarrow \quad \varepsilon \leq \eta_i \frac{\sigma_{min}(\hat{\mathbf{R}}^k)}{L\|\mathbf{r}^k\|\|(\mathbf{x}^i - \mathbf{x}^{i-1})\|} - 1$$

OAK RIDGE National Laboratory

Adaptive Selection of the Rows in the LS Problem (cont.'d)

Definition of backward error on LHS

$$\varepsilon \leq \eta_i \frac{\sigma_{min}(\hat{\mathbf{R}}^k)}{L\|\mathbf{r}^k\|\|(\mathbf{x}^i - \mathbf{x}^{i-1})\|} - 1 \rightarrow \varepsilon_{\mathrm{LHS}}$$

We impose the error on the RHS to be $\varepsilon_{\mathrm{RHS}} \leq \varepsilon$

Such that

$$0 \leq \varepsilon_{\mathrm{RHS}} \leq \varepsilon_{\mathrm{LHS}}$$

Adaptive sketching selectively projects LS problem only if this chain of inequalities is satisfied to ensure backward stability of the numerical scheme

The Two Sketching Levels Have Different Jobs

LS solved to compute Anderson mixing

$$g^k = \text{argmin} \, \| \Pi_2 \, \Pi_1 \, (R^k g - r^k) \|_2$$

Π_1 = Projection operator onto subspace identified by physical field
Π_2 = Projection subspace identified by algebraic sketching

Level 1: physics projection

Static choice of field/subspace.

Improves scaling and conditioning.

Reduces memory allocation for the least-squares matrix.

Level 2: algebraic sketching

Dynamic row selection.

Adapts to the residual and stability threshold.

Controls least-squares perturbation to allow for an inexpensive QR/least-squares solve without destroying stability.

Why both are needed

Physics projection alone may still leave a large LS problem.

Algebraic sketching alone may ignore field scaling.

Together they separate modeling insight from stability control.

Numerical Results: Non-Linear Case

Baseline: standard AAP without any projection

Two Level projection

- **First Level Projection**: projection onto one field (we test all options)

- **Second Level Projection** (adaptive selection):

  - No adaptivity
  - Selection of rows corresponding to entries with large residual
  - Randomized selection

Numerical Results: One Time-Step for Boltzmann Equation

The non-relativistic Boltzmann equation is given by

$$\partial_t f + \mathcal{T}(f) = \mathcal{C}(f)$$ non-linear collision operator

Where $f(\mathbf{x}, \omega, \epsilon, t)$ denotes the density of particles at position $\mathbf{x} \in \mathbb{R}^3$ traveling along direction $\omega \in \mathbb{S}^2$ with energy $\epsilon \in \mathbb{R}^+$ at time $t \in \mathbb{R}^+$. Here the advection and collision operators are denoted by $\mathcal{T}$ and $\mathcal{C}$, respectively.

With a simple backward Euler method, the implicit stage of an implicit-explicit time integration scheme,

$$f(\mathbf{x}, \omega, \epsilon, t^{n+1}) = f(\mathbf{x}, \omega, \epsilon, t^n) + \Delta t\, \mathcal{C}(f(\mathbf{x}, \cdot, \cdot, t^{n+1}))\,, \quad \forall \mathbf{x} \in \mathbb{R}^3$$

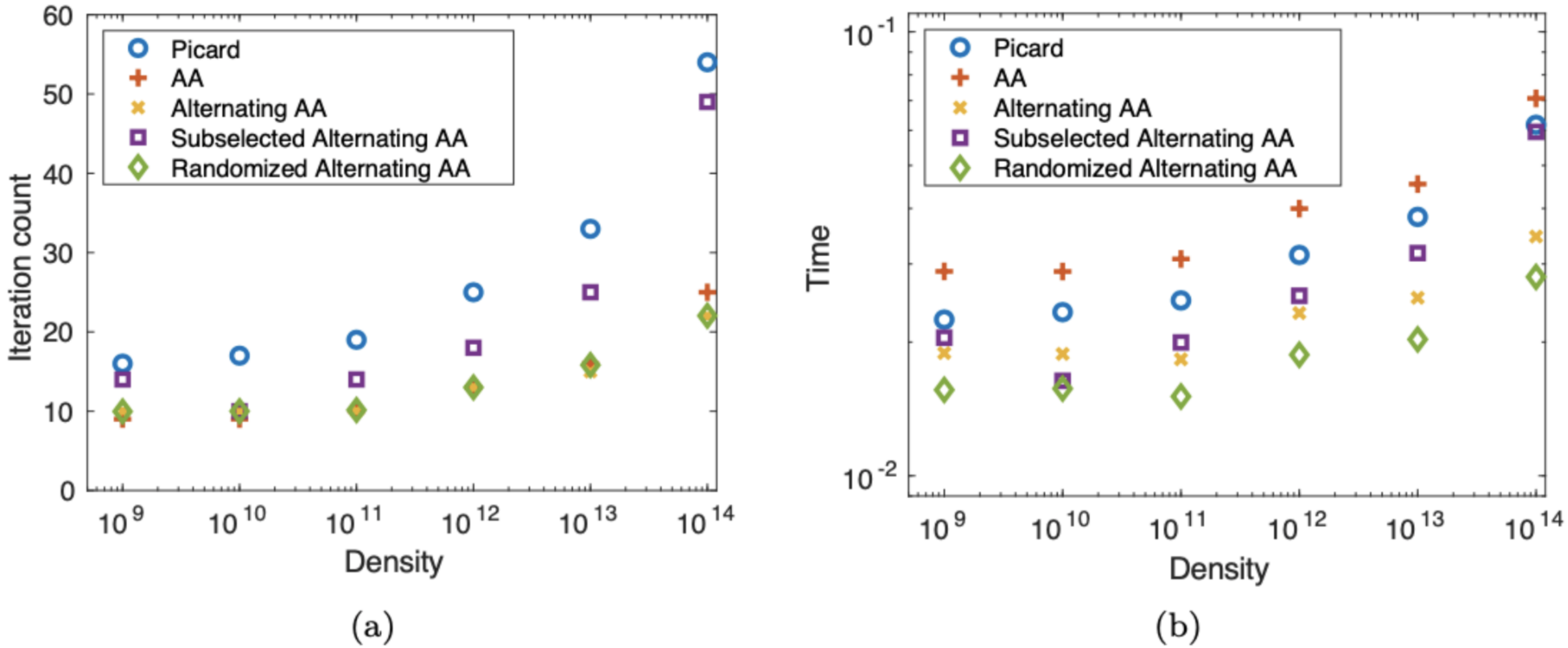
with $\mathcal{C}(f) = \eta_{tot}(f) - \chi_{tot}(f)\, f$, $\eta_{tot}(f)$ = total emissivity, $\chi_{tot}(f)$ = total opacity

The fixed-point iteration is $f^{n+1} = G(f^{n+1})$ where the fixed-point operator G is defined as

$$G(f^{n+1}) := (f^n + \Delta t \eta_{tot}(f^{n+1}))/(1 + \Delta t \chi_{tot}(f^{n+1}))$$

Oak Ridge National Laboratory

Numerical Results: Boltzmann Equation (cont.'d)



Iteration counts and computation time (in log scale) for various fixed-point solvers for solving the non-linear system at different matter density, which corresponds to the stiffness of the system.

Numerical Results: Bidomain Equations

Given an ionic model $I_{\text{ion}} : \Omega \to \mathbb{R}$ and applied current $I_{\text{app}} : \Omega \to \mathbb{R}$,

$$F_e(u_e, u_i) := \frac{\partial(u_e - u_i)}{\partial t} - \operatorname{div} D_e \nabla u_e + I_{\text{ion}}(u_e - u_i) - I_{\text{app}} = 0 \quad \text{in } \Omega,$$

$$F_i(u_e, u_i) := \frac{\partial(u_i - u_e)}{\partial t} - \operatorname{div} D_i \nabla u_i - I_{\text{ion}}(u_e - u_i) + I_{\text{app}} = 0 \quad \text{in } \Omega,$$

$$\nabla u_{i,e} \cdot \boldsymbol{n} = 0 \quad \text{on } \partial\Omega,$$

$D_{e,i}$ are symmetric and positive definite conductivity tensors.

This is a degenerate parabolic problem that is commonly used to describe the propagation of an electric potential in soft tissue, such as the one that generates a heartbeat in cardiac tissue.

The domain is split into two parts: the space inside the cardiac muscle cells, and the space between the cardiac muscle cells.

Numerical Results: Bidomain Equations (cont.'d)

Picard iteration

$$\begin{bmatrix} u_e^{k+1} \\ u_i^{k+1} \end{bmatrix} = \begin{bmatrix} u_e^k \\ u_i^k \end{bmatrix} - \begin{bmatrix} F_e(u_e^k, u_i^k) \\ F_i(u_e^k, u_i^k) \end{bmatrix}$$

First-level physics-based projection

$$\boldsymbol{\Pi}_1\left(\begin{bmatrix} u_e \\ u_i \end{bmatrix}\right) = u_e,$$

Extra-cellular mask

OR

$$\boldsymbol{\Pi}_1\left(\begin{bmatrix} u_e \\ u_i \end{bmatrix}\right) = u_i,$$

Intra-cellular mask

DoFs	No mask		Best				
	Iters	Time	Iters	Time	Mask	Adaptivity	Alt (p)
9826	158	6.13	123	**5.27**	No mask	Subselect power	3
71874	71	18.27	69	**15.30**	No mask	Subselect power	3
549250	38	66.32	33	**53.75**	No mask	Random power	3
4293378	46	631.26	27	**343.26**	Extracellular mask	Random constant	3
15290746	45	2207.21	27	**1247.51**	No mask	Random power	3

Numerical Results: Navier-Stokes Equation

Given Reynolds number $Re > 0$

Find velocity $u : \Omega \to \mathbb{R}^2$
pressure $p : \Omega \to \mathbb{R}$ such that

$$
\begin{aligned}
-\frac{1}{Re}\Delta \boldsymbol{u} + [\nabla \boldsymbol{u}]\boldsymbol{u} + \nabla p &= 0 && \text{in } \Omega,\\
\operatorname{div} \boldsymbol{u} &= 0 && \text{in } \Omega,\\
\boldsymbol{u} &= \boldsymbol{u}_D && \text{on } \partial\Omega,\\
\int_\Omega p \, dx &= 0.
\end{aligned}
$$

Fixed-point formulation of the discretized system

$$
\begin{aligned}
-\frac{1}{Re}\Delta \boldsymbol{u}^{k+1} + [\nabla \boldsymbol{u}^{k+1}]\boldsymbol{u}^k + \nabla p^{k+1} &= 0 && \text{in } \Omega,\\
\operatorname{div} \boldsymbol{u}^{k+1} &= 0 && \text{in } \Omega,\\
\boldsymbol{u}^{k+1} &= \boldsymbol{u}_D && \text{on } \partial\Omega,\\
\int_\Omega p^{k+1} \, dx &= 0.
\end{aligned}
$$

$$(\boldsymbol{u}^{k+1}, p^{k+1}) = T(\boldsymbol{u}^k)$$

Numerical Results: Navier-Stokes Equation (cont.'d)

DoFs	No mask		Best				
	Iters	Time	Iters	Time	Mask	Adaptivity	Alt (p)
9026	26	2.81	26	**2.78**	Velocity mask	No adaptivity	1
36482	21	12.48	20	**11.67**	Velocity mask	No adaptivity	2
146690	20	65.39	18	**54.10**	Velocity mask	Subselect power	1
588290	19	327.59	18	**291.60**	Velocity mask	No adaptivity	2
1324802	19	979.72	18	**892.88**	Velocity mask	Subselect power	2

Practical Takeaways

When the approach helps most

Least-squares/history manipulation is a significant runtime component.

Field variables have very different scales.

A stable reduced LS problem preserves the iteration count.

When gains may be modest

Residual assembly or physics kernel dominates runtime.

The reduced LS problem does not change iteration count.

Aggressive masks hurt cache locality or trigger stability checks.

The method is most compelling when mathematical stability and hardware-aware reduction point in the same direction.

Narrative Roadmap: Three Questions, Three Contributions

2019 **AAR foundation**

Why does reducing the frequency of global Anderson steps help?

- Reduced time to solution
- Improved scalability
- Improved robustness against stagnations

2024/2026 **Stable inexact AA/AAR**

How much inexactness in Picard evaluations and/or least-squares is safe?

- Larger errors are tolerated as convergence is reached
- Theoretical guidelines allow reduction of computational cost without affecting final accuracy

2026 **Two-level sketching**

How do we make this practical for nonlinear coupled physics?

- Two-stage projection scheme to reduce the dimensionality of LS
- First static projection onto a physical field
- Second adaptive projection based on error estimates

Unifying principle: use problem structure first, then approximate only as much as stability indicators allow.

Conclusions and Future Work

- We presented backward stability analysis for Alternating Anderson acceleration applied to linear and non-linear fixed-point iterations
- Backward stability analysis suggest that AA tolerates larger approximation errors as convergence is reached
- Adaptive projections can be used to compress the LS problem to reduce the computational cost of the numerical scheme without affecting final convergence to desired accuracy
- Physics-based projection allows to retain sufficient physics information while still allowing for a significant reduction of the computational cost
- More theoretical analysis is needed to explain why physics-based projection works

Acknowledgements

M.L.P. and Paul Laiu’s contribution was supported in part by the Office of Science of the Department of Energy, by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration, and by the Laboratory Directed Research and Development (LDRD) Program of Oak Ridge National Laboratory managed by UT-Battelle, LLC for the US Department of Energy under contract DE-AC05-00OR22725.

Nicolas Barnafi’s contribution was supported by Centro de Modelamiento Matematico (CMM), Proyecto Basal FB210005, by ANID Postdoctoral Proyecto 3230326, and by the SIAM Postdoctoral grant.

Thank you!

Questions?

lupopasinim@ornl.gov