Fixed-Point Iteration

Fixed-point problem: Given $g : \mathbb{R}^n \to \mathbb{R}^n$, find $x_* \in \mathbb{R}^n$ such that $x_* = g(x_*)$.

Fixed-Point Iteration

Given x_0.

For $k = 0, 1, \ldots$

Set $x_{k+1} = g(x_k)$.

Fixed-point iterations occur widely in CS&E. Typically, ...

- Convergence is linear at best, often slow, often in doubt.
- "Globalization" is unavailable.
- The problem can be recast as $f(x_*) = 0$, where $f(x) \equiv g(x) - x$, for which there are many very effective algorithms and codes.

But there are often advantages, for example ...

- ease of implementation,
- Jacobian information unnecessary,
- desirable structure preserved,
- low cost per iteration,
- parallel advantages,
- constraints satisfied.

Anderson Acceleration

Anderson Acceleration

Given x_0 and $m \geq 1$.

Iterate: For $k = 1, 2, \ldots$

Set $m_k = \min\{m, k\}$.

Set $F_k = (f_{k-m_k}, \ldots, f_k)$, where $f_i = g(x_i) - x_i$.

Solve $\min_{\alpha \in R^{m_k+1}} \|F_k \alpha\|_2$ s. t. $\sum_{i=0}^{m_k} \alpha_i = 1$.

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

Rationale:

Suppose $g(x) = Ax + b$ for $A \in \boldsymbol{R}^{n \times n}$ and $b \in \boldsymbol{R}^n$.

Then $x_{k+1} = \sum_{i=0}^{m_k} \alpha_i g(x_{k-m_k+i}) = g(\sum_{i=0}^{m_k} \alpha_i x_{k-m_k+i})$. Thus

$$x_{k+1} = g(x_{min})$$

where $x_{min} = \sum_{i=0}^{m_k} \alpha_i x_{k-m_k+i}$ has minimal residual within the affine subspace containing $\{x_{k-m_k+i}\}_{i=0,\ldots,m_k}$.

Acceleration Methods — Perspectives

Anderson acceleration:

- Derived from a method of D. G. Anderson (1965).
- Used successfully for many years as *Anderson mixing* to accelerate the *self-consistent field* iteration in electronic structure computations; see C. Yang et al. (2008).
- Essentially the same method was independently described for particular applications by Washio & Oosterlee (1997) and Carlson & Miller (1998).

Mathematicians and numerical analysts have extensively studied . . .

- *vector-extrapolation* methods, especially polynomial methods: (reduced-rank, minimal-polynomial, modified minimal-polynomial);
- *vector* and *topological* ε*-algorithms*.

See Brezinski & Redivo-Zaglia (1991), Brezinski (2000), Jbilou–Sadok (2000), Smith–Ford–Sidi (1987), . . .

Anderson acceleration is in a distinct category:

- Eirola–Nevanlinna (1989) and U. Yang (1995) methods and related methods based on quasi-Newton updating;
- "charge-mixing" methods for electronic-structure computations: Pulay (1980, 1982), Kerker (1981), Thomas–Fermi (Raczkowski et al. 2001), Broyden (Kresse–Furthmüller 1996), . . .

See Fang–Saad (2008) and C. Yang et al. (2008).

Anderson Acceleration and GMRES

Joint work with Peng Ni (SINUM, 2011) ...

- *Anderson acceleration is "essentially equivalent" to GMRES on linear problems.*

Assume ...

- Anderson acceleration is not truncated, i.e., $m_k = k$ for each k.
- g is linear, i.e., $g(x) = Ax + b$ for $A \in \boldsymbol{R}^{n \times n}$ and $b \in \boldsymbol{R}^n$.
- $(I - A)$ is nonsingular.
- GMRES is applied to $(I - A)x = b$ with initial point x_0.

Theorem

Suppose these hold and that, for some $k > 0$, $r_{k-1}^{GMRES} \neq 0$ and also $\|r_{j-1}^{GMRES}\|_2 > \|r_j^{GMRES}\|_2$ for each j such that $0 < j < k$. Then

$$\sum_{i=0}^{k} \alpha_i x_i^{AA} = x_k^{GMRES} \text{ and } x_{k+1}^{AA} = g(x_k^{GMRES}).$$

Anderson Acceleration and GMRES (cont.)

Consider . . .

- $Ax = b$ for nonsingular $A \in \mathbb{R}^n$.
- Splitting $A = M - N$ for nonsingular $M \in \mathbb{R}^{n \times n}$.
- Stationary iteration $x_+ = g(x) \equiv M^{-1}Nx + M^{-1}b$.

Assume . . .

- Anderson acceleration is not truncated, i.e., $m_k = k$ for each k.
- GMRES is applied to $M^{-1}Ax = M^{-1}b$ with initial point x_0.

Corollary

Suppose these hold and that, for some $k > 0$, $r_{k-1}^{GMRES} \neq 0$ *and also* $\|r_{j-1}^{GMRES}\|_2 > \|r_j^{GMRES}\|_2$ *for each* j *such that* $0 < j < k$. *Then*

$$\sum_{i=0}^{k} \alpha_i x_i^{AA} = x_k^{GMRES} \text{ and } x_{k+1}^{AA} = g(x_k^{GMRES}).$$

NOTE: Applying Anderson acceleration to stationary iterations is *not* recommended as a general substitute for left-preconditioned GMRES. However, it may have advantages in some circumstances.

Implementation

The main implementational issue is *solving the least-squares problem*:

$$\min_{\alpha \in R^{m_k+1}} \|F_k \alpha\|_2 \quad s.t. \quad \sum_{i=0}^{m_k} \alpha_i = 1.$$

Note that ...

- F_{k+1} is obtained from F_k by adding a column on the right and, if $k \geq m$, also dropping a column on the left.
- In practice, additional columns may be dropped on the left to maintain acceptable conditioning.

These low-rank changes allow efficient QR solution of the least-squares problem by updating the Q and R factors at a cost of $O(m_k n)$ flops/iteration.

Good approach: As in Fang–Saad (2008), recast as an unconstrained problem

$$\min_{\gamma \in R^{m_k}} \|f_k - \mathcal{F}_k \gamma\|_2,$$

where $\mathcal{F}_k = (\Delta f_{k-m_k}, \ldots, \Delta f_{k-1})$ and $\Delta f_i = f_{i+1} - f_i$ for each i.

See Walker (2011) for details and a MATLAB code.

Example: The Expectation-Maximization (EM) Algorithm

Formalized by Dempster, Laird, and Rubin (1977).

Context: Statistical estimation (maximum-likelihood) using *incomplete data*.

General idea: Determine the next approximate MLE to *maximize* the *expectation* of the complete-data log-likelihood function, given the observed incomplete data and the current approximate MLE.

Marvelous property: The likelihood function *increases* at each iteration.

Particular application: Estimating the parameters in a *mixture density*

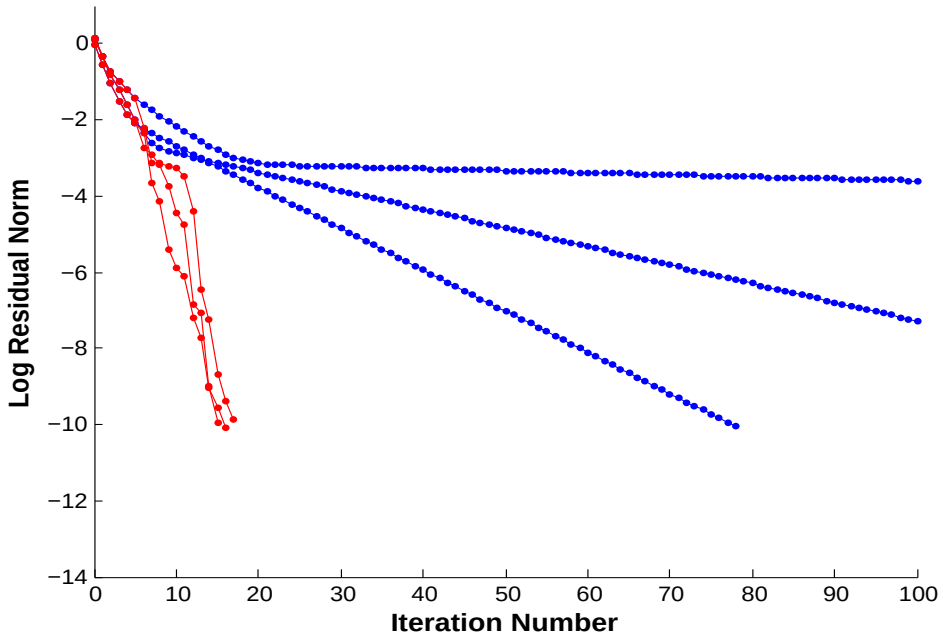
$$p(x|\Phi) = \sum_{i=1}^{m} \alpha_i p_i(x|\phi_i)$$

using an "unlabeled" sample on the mixture. Typically, the EM algorithm becomes a simple fixed-point iteration.

Example: EM Convergence and "Separation"

A Univariate Normal Mixture.

- $p_i(x|\phi_i) = \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-(x-\mu_i)^2/(2\sigma_i^2)}$ for $i = 1, \ldots, 3$.
- EM iterations on the means: $\mu_i^+ = \left\{ \sum_{k=1}^N x_k \frac{\alpha_i p_i(x_k|\phi_i)}{p(x_k|\Phi)} \right\} \Big/ \left\{ \sum_{k=1}^N \frac{\alpha_i p_i(x_k|\phi_i)}{p(x_k|\Phi)} \right\}$.
- Sample of 100,000 observations.
  - $[\alpha_1, \alpha_2, \alpha_3] = [.3, .3, .4]$, $[\sigma_1^2, \sigma_2^2, \sigma_3^2] = [1, 1, 1]$.
  - $[\mu_1, \mu_2, \mu_3] = [0, 2, 4]$, $[0, 1, 2]$, $[0, .5, 1]$.

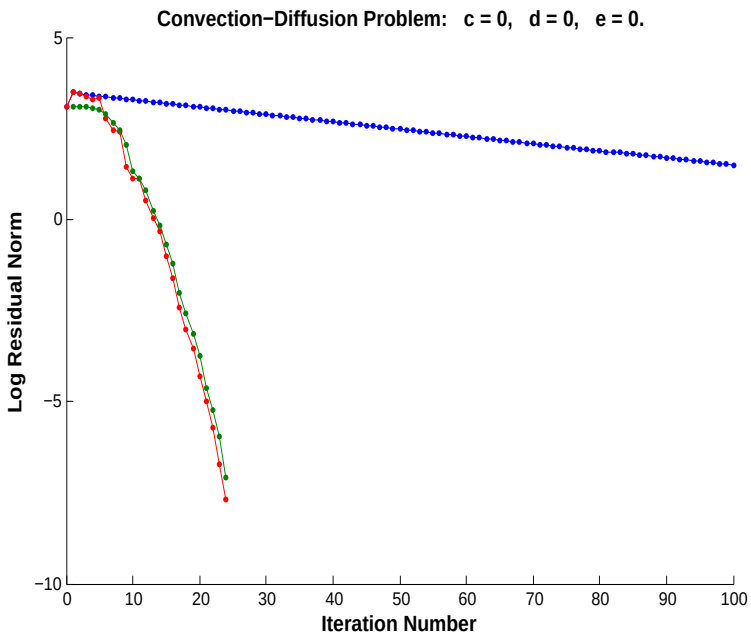


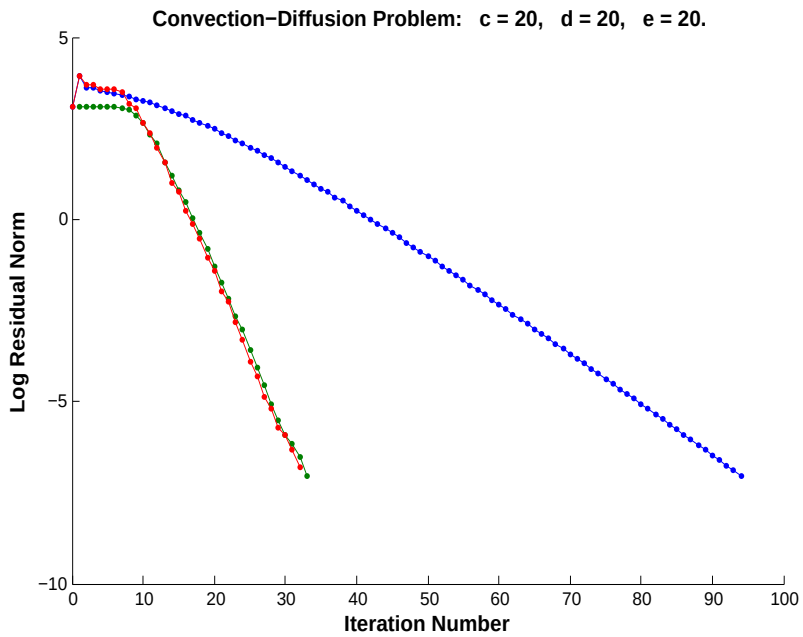
Example: Domain Decomposition, Linear Problem

A Convection-Diffusion Problem.

$$\begin{aligned} \Delta u + cu + du_x + eu_y &= f \quad \text{in } \mathcal{D} = [0,1] \times [0,1], \\ u &= 0 \quad \text{on } \partial\mathcal{D}. \end{aligned}$$

- Centered differences, 128×128 grid.
- 4×4 array of subdomains, 32×32 grid per subdomain.
- Restricted additive Schwarz (Cai-Sarkis, 1999), 3 grid lines overlap, no coarse grid.
- $m = 50$ in Anderson acceleration.





Example: Domain Decomposition, Nonlinear Problem

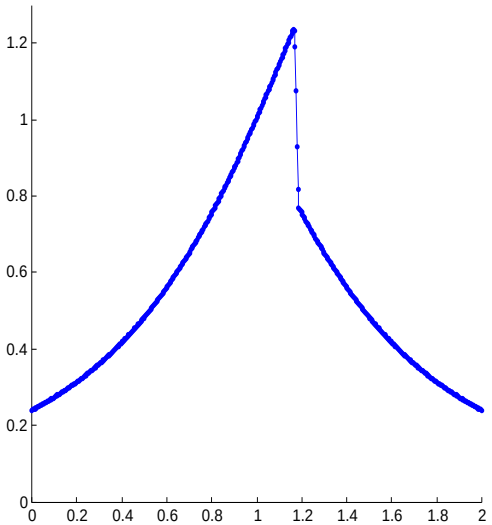
A Transonic Duct Flow Problem.

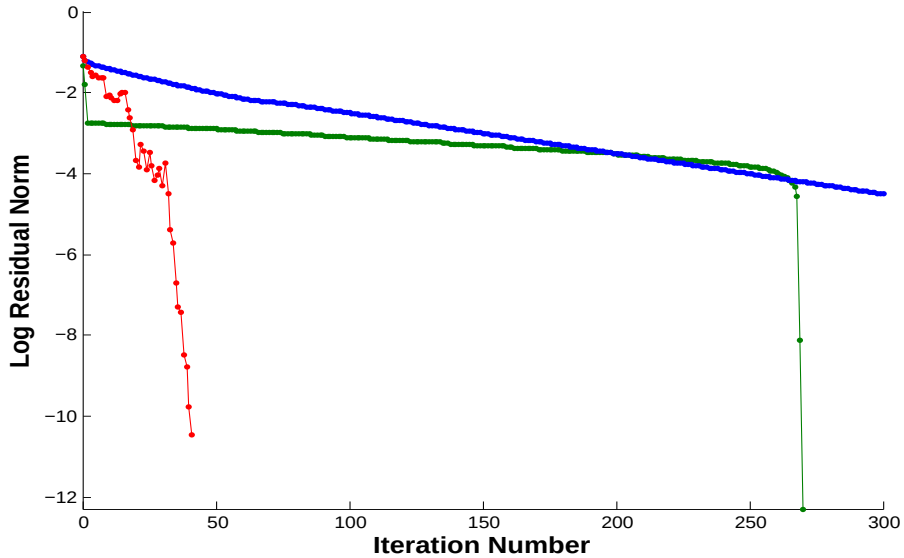
$$[A(x)\rho(u)u_x]_x = 0, \quad 0 < x < 2,$$
$$u(0) = 0, \quad u(2) = u_R,$$

where

$$\rho(u) = \left[1 + \tfrac{\gamma-1}{2}(1-u^2)\right]^{1/(\gamma-1)}$$
$$A(x) = 0.4 + 0.6(x-1)^2$$
$$u_R = 1.15, \ \gamma = 1.4$$

- Finite-difference discretization, 512 equally spaced grid points, first-order density biasing stabilization (Cai–Keyes–Young 2000, Young et al. 2003).
- "Nonlinear" RAS, 64 grid points/subdomain, 8 grid points overlap.
- "Matrix-free" Newton-GMRES-backtracking method.
- $m = 20$ in Anderson acceleration.





Example: Picard Iteration, Variably Saturated Flow

Joint work with Aaron Lott, Carol Woodward, Ulrike Yang (2011) ...

$$\begin{aligned} \nabla[g(u)\nabla u] &= 0 \text{ in } \mathcal{D} = [0,1]\times[0,1] \\ u(0,z) &= -2,\ u(1,z) = 1 \\ u(x,0) &= u(x,1) = -2+3x \end{aligned} \qquad \textit{where} \quad \begin{aligned} g(u) &= \begin{cases} 1, & u \geq 0 \\ k_s\psi(|u|), & u < 0 \end{cases} \\ \psi(\theta) &= [1+\alpha\theta^n]^{-m/2}\left(1-\frac{(\alpha\theta)^{n-1}}{[1+(\alpha\theta)^n]m}\right)^2 \end{aligned}$$

- Van Genuchten – Mualem parametrization:
  - — α relates to mean pore size of soil;
  - — n relates to uniformity of pore distribution;
  - — $m = 1 - 1/n$;
  - — $k_s = 1$ here.
- Yavneh – Dardyk (2006): *"For a given domain and boundary conditions, this problem generally becomes more difficult for smaller n and larger α. For $n < 2$, g' fails to satisfy a Lipschitz condition at $u = 0$, and $g(u)$ actually tends to a step function for $n \to 1+$. For α that is not small compared to one, the solution exhibits a thin boundary layer near $x = 0$ ... which narrows quickly as α grows."*

Example: Picard Iteration, Variably Saturated Flow (cont.)

Popular solution methods:

- Newton's method.
  — fast (up to quadratic) convergence;
  — nonsymmetric linear systems;
  — requires Jacobian information.

- Picard iteration.
  — symmetric positive-definite linear systems;
  — no Jacobian information;
  — linear convergence, may be slow.

Numerical study:

- Applied Anderson acceleration to a modified Picard iteration (MPI) from Celia–Bouloutas–Zarba (1990).

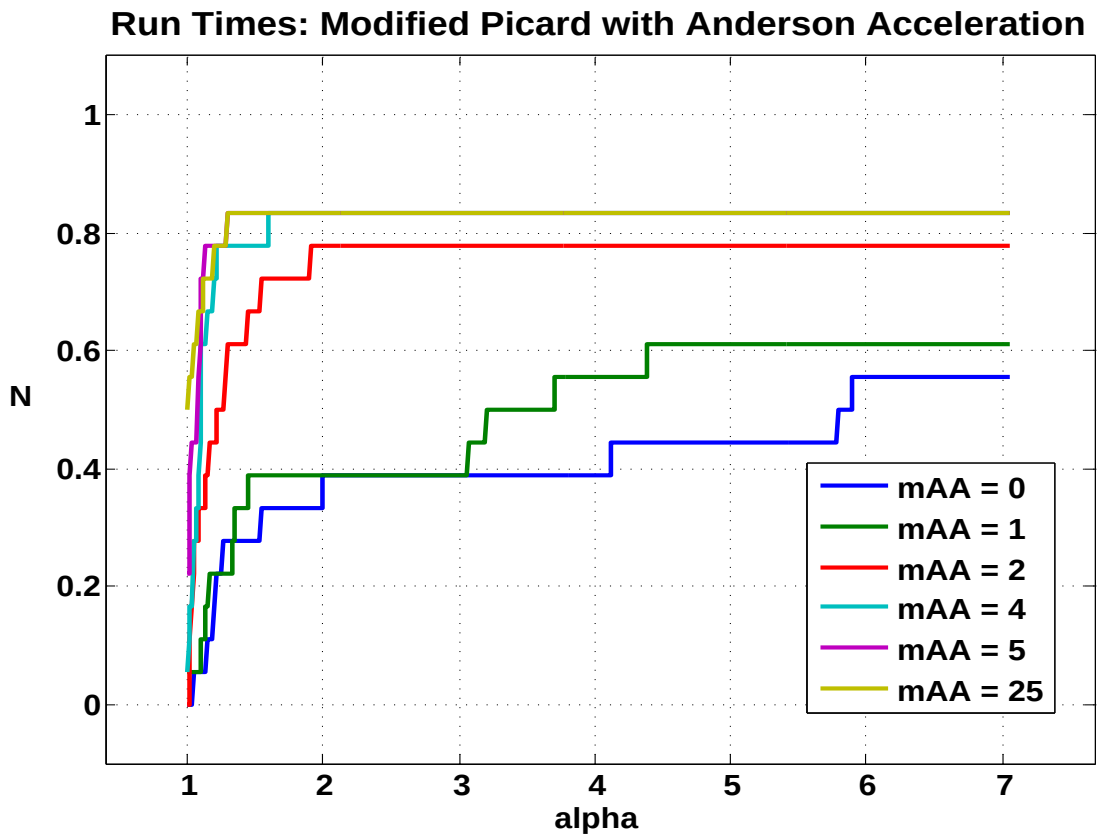
- Compared MPI, MPI with Anderson acceleration, and a Newton-GMRES-linesearch method within KINSOL.

- Finite-difference discretization; 1024×1024 grid.

Example: Picard Iteration, Variably Saturated Flow (cont.)

Run-time *performance profiles* (Dolan–Moré, 2002) for
- — modified Picard iteration (MPI),
- — MPI with Anderson acceleration using different mAA values.

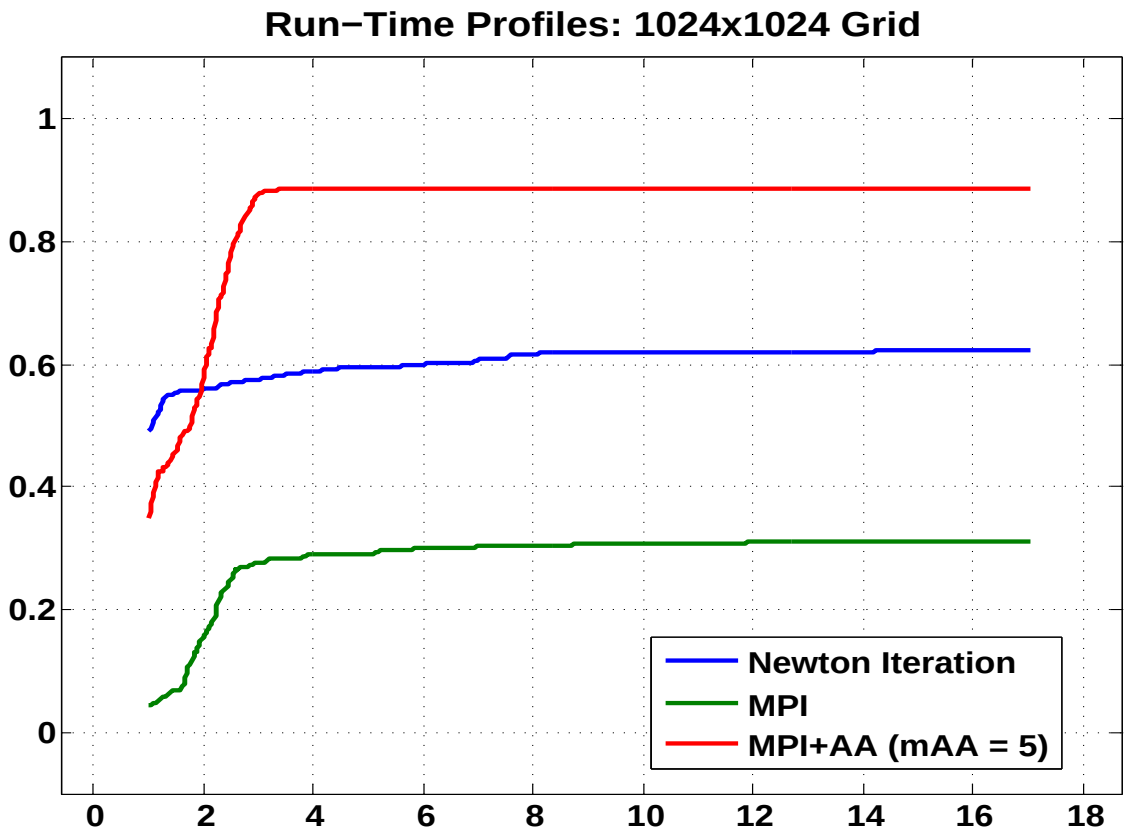
- 18 selected (α, n) pairs.



Example: Picard Iteration, Variably Saturated Flow (cont.)

Run-time performance profiles for
- — the Newton–GMRES-linesearch method.
- — modified Picard iteration (MPI),
- — MPI with Anderson acceleration (mAA = 5),

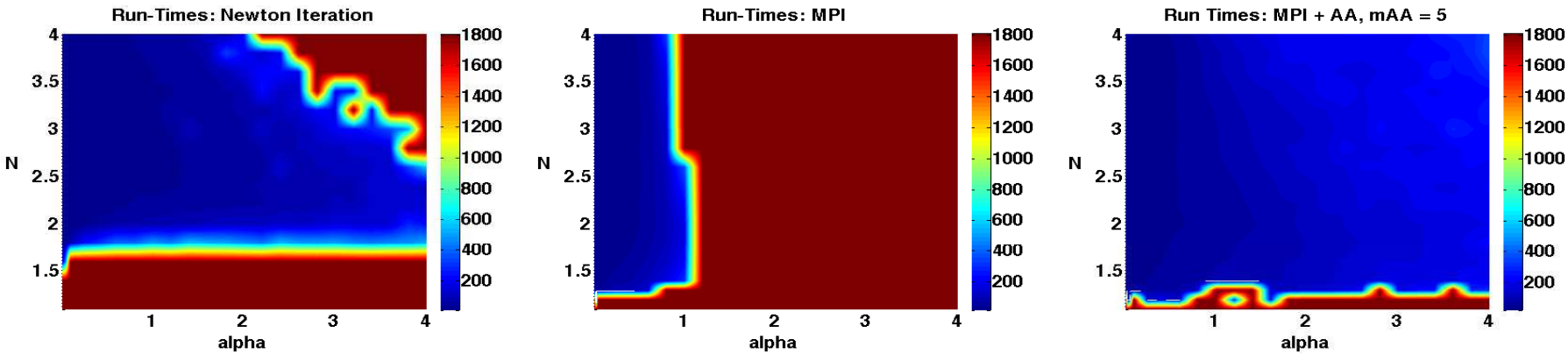
- 408 selected (α, n) pairs in $[0.01, 4.0] \times [1.1, 4.0]$.



Example: Picard Iteration, Variably Saturated Flow (cont.)

Run-time "maps" for
- — the Newton-GMRES-linesearch method,
- — modified Picard iteration (MPI),
- — MPI with Anderson acceleration (mAA = 5).

- 408 selected (α, n) pairs in $[0.01, 4.0] \times [1.1, 4.0]$.



Run times (in seconds) for the three methods.

References

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

C. Brezinski and M. Redivo-Zaglia, *Extrapolation Methods Theory and Practice*, North-Holland, Amsterdam, 1991

C. Brezinski, *Convergence acceleration during the 20th century*, J. Comput. Appl. Math., 122 (2000), 1-21

X.-C. Cai and D. E. Keyes and D. P. Young, *A nonlinear additive Schwarz preconditioned inexact Newton method for shocked duct flows*, Proc. 13th Int. Conf. on Domain Decomposition Methods (Lyon, France, Oct. 9-12, 2000), N. Débit, M. Garbey, R. Hoppe, J. Périaux, D. Keyes, and Y. Kuznetsov, eds., CIMNE, Barcelona, 2000.

X.-C. Cai and M. Sarkis, *A restricted additive Schwarz preconditioner for general sparse linear systems*, SIAM J. Sci. Comput., 21 (1999) 792-797

N. N. Carlson and K. Miller, *Design and application of a gradient-weighted moving finite element code I: in one dimension*, SIAM J. Sci. Comput., 19 (1998), 728-765

M. A. Celia, E. T. Bouloutas, R. L. Zarba, *A general mass-conservative numerical solution for the unsaturated ow equation*, Water Resour. Res. 26 (7) (1990), 1483-1496

A. P. Dempster and N. M. Laird and D. B. Rubin, *Maximum-likelihood from incomplete data via the EM algorithm*, J. Royal Statist. Soc. Ser. B (methodological), 39 (1977), 1-38

E. D. Dolan and J. J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program., Ser. A, 91 (2002), 201-213

T. Eirola and O. Nevanlinna, *Accelerating with rank-one updates*, Lin. Alg. Appl., 121 (1989), 511-520

H. Fang and Y. Saad, *Two classes of multisecant methods for nonlinear acceleration*, Numer. Linear Algebra Appl., 16 (2008), 197-221

K. Jbilou and H. Sadok, *Vector extrapolation methods. Applications and numerical comparison*, J. Comput. Appl. Math., 122 (2000), 149-165

G. P. Kerker, *Efficient iteration scheme for self-consistent pseudopotential calculations*, Phys. Rev. B, 23 (1981), 3082-3084

References

G. Kresse and J. Furthmüller, *Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set*, Phys. Rev. B, 54 (1996), 169-186
myrefP. A. Lott, H. F. Walker, C. S. Woodward, and U. M. YangAn accelerated Picard method for nonlinear systems related to variably saturated flowLawrence Livermore National Lab Report LLNL-JRNL-491543; submitted to Advances in Water Resources

P. Pulay, *Convergence acceleration of iterative sequences. The case of SCF iteration*, Chem. Phys. Letters, 73 (1980), 393-398

P. Pulay, *Improved SCF convergence*, J. Computational Chem., 3 (1982), 556-560

D. Raczkowski, A. Canning and L. W. Wang, *Thomas-Fermi charge mixing for obtaining self-consistency in density functional calculations.*, Phys. Rev. B, 64 (2001), 12101-1-4

D. A. Smith, W. F. Ford and A. Sidi, *Extrapolation methods for vector sequences*, SIAM Rev., 29 (1987), 199-233

H. F. Walker and P. Ni, *Anderson acceleration for fixed-point iterations*, SIAM J. Numer. Anal., 49 (2011), 1715-1735

H. F. Walker, , Worcester Polytechnic Institute Mathematical Sciences Department Rep. MS-6-15-50, June 2011

T. Washio and C. W. Oosterlee, *Krylov subspace acceleration for nonlinear multigrid schemes*, Electronic Trans. Numer. Anal., 6 (1997), 271-290

C. Yang, J. C. Meza, B. Lee, and L.-W. Wang, *KSSOLV — a MATLAB toolbox for solving the Kohn–Sham equations*, ACM Trans. Math. Softw., 36 (2008), 1-35

U. M. Yang, *A Family of Preconditioned Iterative Solvers for Sparse Linear Systems*, PhD Thesis, Department of Computer Science, University of Illinois at Urbana-Champaign, 1995

I. Yavneh and G. Dardyk, *A multilevel nonlinear method*, SIAM J. Sci. Comput., 28 (2006), 24-46

D. P. Young, W. P. Huffman, R. G. Melvin. C. L. Hilmes, and F. T. Johnson, *Nonlinear elimination in aerodynamic analysis and desigh optimization*, in *Large-Scale PDE-Constrained Optimization*, L. T. Biegler, O. Ghattas, M. Heinkenschloss, and B. van Bloemen Waanders, eds., Springer-Verlag, Berlin, Heidelberg, New York, 2003