Newton's Method or: Solving Nonlinear Equations Fast

Simple Words - October 2024

Matt Dallas
Dept. of Mathematics
University of Dallas, Irving, TX

October 29, 2024

Outline

Numerical Analysis

↓

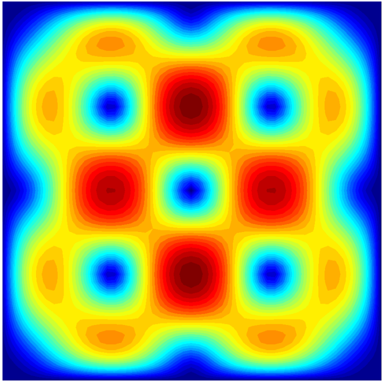
Applications

↓

Nonlinear Systems

↓

Newton's Method & Recent Work



What is Numerical Analysis?

- "Numerical Analysis is the branch of mathematics that provides tools and methods for solving mathematical problems in numerical form" - Walter Gautschi [Gau12].

What is Numerical Analysis?

- "Numerical Analysis is the branch of mathematics that provides tools and methods for solving mathematical problems in numerical form" - Walter Gautschi [Gau12].
- "Numerical analysis is a branch of mathematics that solves continuous problems using numeric approximation" - MathWorks.

What is Numerical Analysis?

- "Numerical Analysis is the branch of mathematics that provides tools and methods for solving mathematical problems in numerical form" - Walter Gautschi [Gau12].
- "Numerical analysis is a branch of mathematics that solves continuous problems using numeric approximation" - MathWorks.
- "Numerical analysis is the study of algorithms that use numerical approximation (as opposed to symbolic manipulations) for the problems of mathematical analysis" - Numerical Analysis Wikipedia page.

What is Numerical Analysis?

- “Numerical Analysis is the branch of mathematics that provides tools and methods for solving mathematical problems in numerical form" - Walter Gautschi
- “Numerical analysis is a branch of mathematics that solves continuous problems using numeric approximation" - MathWorks.
- “Numerical analysis is the study of algorithms that use numerical approximation (as opposed to symbolic manipulations) for the problems of mathematical analysis" - Numerical Analysis Wikipedia page.

Historical Applications

Figure: Babylonian Clay tablet 1800-1600 BC

Historical Applications

Figure: Rhind Papyrus, Egypt 1550 BC

Historical Applications

Figure: Left: Archimedes. Middle: Newton. Right: Gauss.

Historical Applications

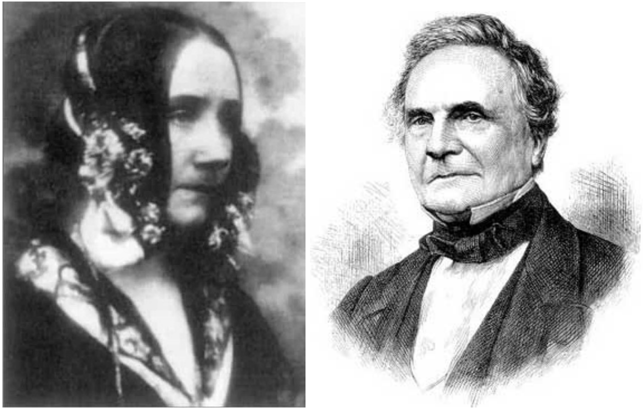
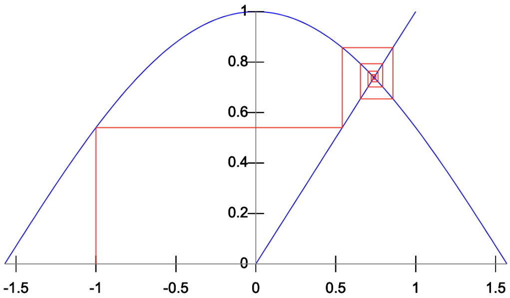


Figure: Left: Lovelace. Right: Babbage.

Modern Applications

Figure: Left : Turing. Right: Wilkinson.

Modern Applications



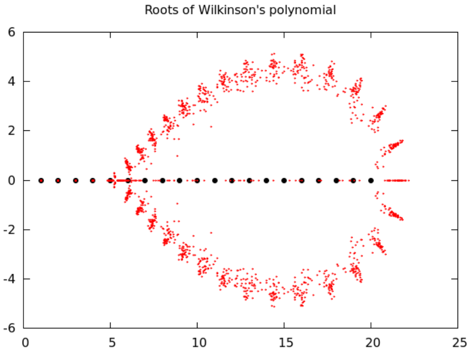


Figure: **Left: Nonlinear Solvers**. **Right: Conditioning and Stability** (courtesy of Kerlokiste).

Modern Applications

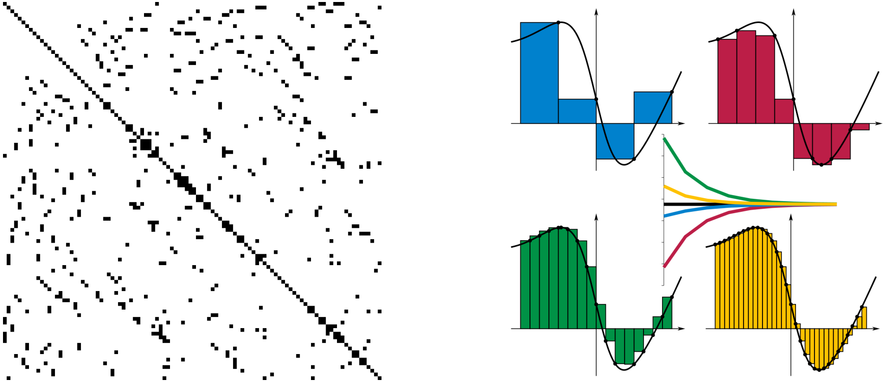


Figure: **Left: Linear Solvers**. **Right: Numerical Integration**.

My Area of Interest

Solving $\quad F(x) = 0 \quad$ where $F : \mathbb{R}^n \to \mathbb{R}^n$ is **nonlinear**.

Example:

$$\begin{aligned} x + y^2 &= 0 \\ \frac{3}{2}xy + y^2 + y^3 &= 0 \end{aligned}$$

In vector form,

$$F(x, y) = \begin{pmatrix} x + y^2 \\ \frac{3}{2}xy + y^2 + y^3 \end{pmatrix} = \mathbf{0}$$

Why?

In physics, many models are either differential or integral equations.

These equations are **discretized**, resulting in a nonlinear system of equations $F : \mathbb{R}^n \to \mathbb{R}^n$ of the form

$$F(x) = 0$$

Why?

In physics, many models are either differential or integral equations.

These equations are **discretized**, resulting in a nonlinear system of equations $F : \mathbb{R}^n \to \mathbb{R}^n$ of the form

$$F(x) = 0$$

- Specific application: **nonlinear partial differential equations**.

Why?

In physics, many models are either differential or integral equations.

These equations are **discretized**, resulting in a nonlinear system of equations $F : \mathbb{R}^n \to \mathbb{R}^n$ of the form

$$F(x) = 0$$

- Specific application: **nonlinear partial differential equations**.
  - The Wikipedia page titled "List of nonlinear partial differential equations" lists 103 PDEs.
  - Many of these have an entire Wikipedia page of their own. For example:

Incompressible Navier-Stokes

$$\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} - \Delta \mathbf{u} + \nabla p - \mathbf{g} = 0$$
$$\nabla \cdot \mathbf{u} = 0$$

Minimal Surface Equation

$$\nabla \cdot \left(Du / \sqrt{1 + |Du|^2} \right) = 0$$

Typical Steps in Computing Numerical Solutions

1. Start with $\mathcal{F}(u) = 0$, e.g., $\mathcal{F}$ may be a differential equation, integral equation, or integro-differential equation.

Typical Steps in Computing Numerical Solutions

1. Start with $\mathcal{F}(u) = 0$, e.g., $\mathcal{F}$ may be a differential equation, integral equation, or integro-differential equation.
2. (Optional) Modify $\mathcal{F}$ somehow to make later steps more convenient

Typical Steps in Computing Numerical Solutions

1. Start with $\mathcal{F}(u) = 0$, e.g., $\mathcal{F}$ may be a differential equation, integral equation, or integro-differential equation.
2. (Optional) Modify $\mathcal{F}$ somehow to make later steps more convenient
3. **Discretize** $\mathcal{F}$. This step turns $\mathcal{F}$ into a function $F : \mathbb{R}^n \to \mathbb{R}^n$.

Typical Steps in Computing Numerical Solutions

1. Start with $\mathcal{F}(u) = 0$, e.g., $\mathcal{F}$ may be a differential equation, integral equation, or integro-differential equation.
2. (Optional) Modify $\mathcal{F}$ somehow to make later steps more convenient
3. **Discretize** $\mathcal{F}$. This step turns $\mathcal{F}$ into a function $F : \mathbb{R}^n \to \mathbb{R}^n$.
4. Apply your favorite solver to the system of equations $F(x) = 0$.

Typical Steps in Computing Numerical Solutions

1. Start with $\mathcal{F}(u) = 0$, e.g., $\mathcal{F}$ may be a differential equation, integral equation, or integro-differential equation.
2. (Optional) Modify $\mathcal{F}$ somehow to make later steps more convenient
3. **Discretize** $\mathcal{F}$. This step turns $\mathcal{F}$ into a function $F : \mathbb{R}^n \to \mathbb{R}^n$.
4. Apply your favorite solver to the system of equations $F(x) = 0$.
5. Optimize solver given particular hardware.

Typical Steps in Computing Numerical Solutions

1. Start with $\mathcal{F}(u) = 0$, e.g., $\mathcal{F}$ may be a differential equation, integral equation, or integro-differential equation.
2. (Optional) Modify $\mathcal{F}$ somehow to make later steps more convenient
3. **Discretize** $\mathcal{F}$. This step turns $\mathcal{F}$ into a function $F : \mathbb{R}^n \to \mathbb{R}^n$.
4. Apply your favorite solver to the system of equations $F(x) = 0$.
5. Optimize solver given particular hardware.

- Mathematicians make entire careers out of focusing on just one of these steps.

Typical Steps in Computing Numerical Solutions

1. Start with $\mathcal{F}(u) = 0$, e.g., $\mathcal{F}$ may be a differential equation, integral equation, or even an integro-differential equation.
2. (Optional) Modify $\mathcal{F}$ somehow to make later steps more convenient
3. **Discretize** $\mathcal{F}$. This step turns $\mathcal{F}$ into a function $F : \mathbb{R}^n \to \mathbb{R}^n$.
4. Apply your favorite solver to the system of equations $F(x) = 0$.
5. Optimize solver given particular hardware.

- Each of these steps can be considered its own subfield of Numerical Analysis.
- My interests lie in step 4 when $F(x)$ is **nonlinear**. Specifically...

Newton's Method

Newton's method is a very old technique[1] for solving nonlinear equations, and is still a standard tool used my scientists today.

The bulk of my dissertation concerned analyzing a modification of Newton's method under certain tricky conditions.

[1] in the 16th century, a proto-Newton method was designed by Francois Vieta (1540–1603).

Newton(-Simpson-Raphson) Method

Joseph Raphson (1648–1715) and Thomas Simpson (1710–1761) also made significant contributions to Newton's method.

A concise discussion of the history of Newton's method can be found in Peter Deuflhard's *Newton's Method for Nonlinear Equations* [Deu05].

Newton's Method: Derivation

- Let $F : \mathbb{R} \to \mathbb{R}$ be nonlinear. We want to solve $F(x) = 0$.

Newton's Method: Derivation

- Let $F : \mathbb{R} \to \mathbb{R}$ be nonlinear. We want to solve $F(x) = 0$.
- The idea is to **linearize** F at an initial guess x_0, and then take x_1 to be the root of this linearization.

Newton's Method: Derivation

Let $F : \mathbb{R} \to \mathbb{R}$ be nonlinear. We want to solve $F(x) = 0$.

[2] There are ways to extend Newton's method when a function a not differentiable, e.g., by using subgradients or similar generalizations.

Newton's Method: Derivation

Let $F : \mathbb{R} \to \mathbb{R}$ be nonlinear. We want to solve $F(x) = 0$.

1. Write F as a Taylor Series[2] about some point x_0:

$$F(x) = F(x_0) + F'(x_0)(x - x_0) + \frac{1}{2}F''(x_0)(x - x_0)^2 + \cdots$$

[2] There are ways to extend Newton's method when a function a not differentiable, e.g., by using subgradients or similar generalizations.

Newton's Method: Derivation

Let $F : \mathbb{R} \to \mathbb{R}$ be nonlinear. We want to solve $F(x) = 0$.

1. Write F as a Taylor Series[2] about some point x_0:

$$F(x) = F(x_0) + F'(x_0)(x - x_0) + \frac{1}{2}F''(x_0)(x - x_0)^2 + \cdots$$

2. If x is "close" to x_0, then we have

$$F(x) \approx F(x_0) + F'(x_0)(x - x_0).$$

i.e., a root of $L(x) := F(x_0) + F'(x_0)(x - x_0)$ should be close to a root of $F(x)$.

[2] There are ways to extend Newton's method when a function a not differentiable, e.g., by using subgradients or similar generalizations.

Newton's Method: Derivation

Let $F : \mathbb{R} \to \mathbb{R}$ be nonlinear. We want to solve $F(x) = 0$.

1. Write F as a Taylor Series[2] about some point x_0:

$$F(x) = F(x_0) + F'(x_0)(x - x_0) + \frac{1}{2}F''(x_0)(x - x_0)^2 + \cdots$$

2. If x is "close" to x_0, then we have

$$F(x) \approx F(x_0) + F'(x_0)(x - x_0).$$

   i.e., a root of $L(x) := F(x_0) + F'(x_0)(x - x_0)$ should be close to a root of $F(x)$.

3. Say $L(x_1) = 0$. Then

$$0 = F(x_0) + F'(x_0)(x_1 - x_0) \implies x_1 = x_0 - \frac{F(x_0)}{F'(x_0)}.$$

---

[2] There are ways to extend Newton's method when a function a not differentiable, e.g., by using subgradients or similar generalizations.

1. Write F as a Taylor Series about some point x_0:

$$F(x) = F(x_0) + F'(x_0)(x - x_0) + \frac{1}{2}F''(x_0)(x - x_0)^2 + \cdots$$

2. If x is "close" to x_0, then we have

$$F(x) \approx F(x_0) + F'(x_0)(x - x_0).$$

   i.e., a root of $L(x) := F(x_0) + F'(x_0)(x - x_0)$ should be close to a root of $F(x)$.

3. Say $L(x_1) = 0$. Then

$$0 = F(x_0) + F'(x_0)(x_1 - x_0) \implies x_1 = x_0 - \frac{F(x_0)}{F'(x_0)}.$$

4. Are we satisfied with x_1? If not, then go back through steps 1, 2, and 3 starting from x_0 replaced with x_1.

1. Write F as a Taylor Series about some point x_1:

$$F(x) = F(x_1) + F'(x_1)(x - x_1) + \frac{1}{2}F''(x_1)(x - x_1)^2 + \cdots$$

1. Write F as a Taylor Series about some point x_1:

$$F(x) = F(x_1) + F'(x_1)(x - x_1) + \frac{1}{2}F''(x_1)(x - x_1)^2 + \cdots$$

2. If x is “close" to x_1, then we have

$$F(x) \approx F(x_1) + F'(x_1)(x - x_1).$$

   i.e., a root of $L(x) := F(x_1) + F'(x_1)(x - x_0)$ should be close to a root of $F(x)$.

1. Write F as a Taylor Series about some point x_1:

$$F(x) = F(x_1) + F'(x_1)(x - x_1) + \frac{1}{2}F''(x_1)(x - x_1)^2 + \cdots$$

2. If x is "close" to x_1, then we have

$$F(x) \approx F(x_1) + F'(x_1)(x - x_1).$$

   i.e., a root of $L(x) := F(x_1) + F'(x_1)(x - x_0)$ should be close to a root of $F(x)$.

3. Say $L(x_2) = 0$. Then

$$0 = F(x_1) + F'(x_1)(x_2 - x_1) \implies x_2 = x_1 - \frac{F(x_1)}{F'(x_1)}.$$

1. Write F as a Taylor Series about some point x_1:

$$F(x) = F(x_1) + F'(x_1)(x - x_1) + \frac{1}{2}F''(x_1)(x - x_1)^2 + \cdots$$

2. If x is “close" to x_1, then we have

$$F(x) \approx F(x_1) + F'(x_1)(x - x_1).$$

   i.e., a root of $L(x) := F(x_1) + F'(x_1)(x - x_0)$ should be close to a root of $F(x)$.

3. Say $L(x_2) = 0$. Then

$$0 = F(x_1) + F'(x_1)(x_2 - x_1) \implies x_2 = x_1 - \frac{F(x_1)}{F'(x_1)}.$$

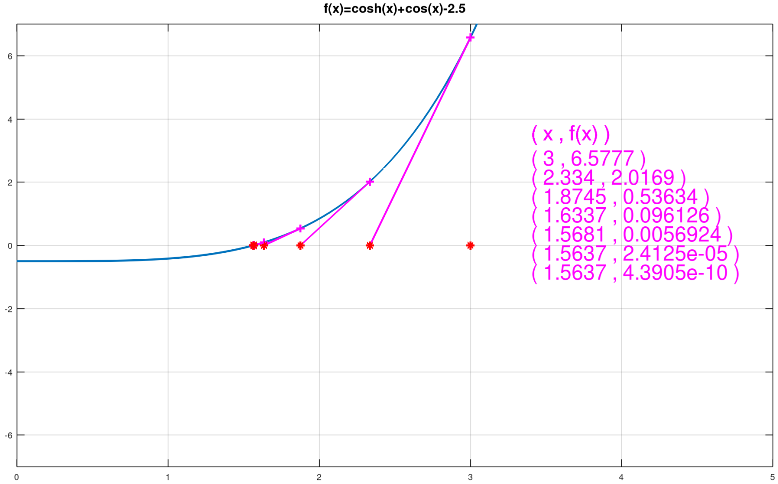
4. Are we satisfied with x_2? If not, then go back through steps 1, 2, and 3 starting from x_1 replaced with x_2.

Newton's Method: Derivation

Continuing this process generates Newton's method:

$$x_{n+1} = x_n - F(x_n)/F'(x_n).$$

Visualization



Example:

Use Newton's method to approximate $\sqrt{2}$.

Note that $x = \sqrt{2}$ if and only if $F(x) = x^2 - 2 = 0$ on $x > 0$. Therefore Newton's method becomes

$$x_{n+1} = \frac{x_n}{2} + \frac{1}{x_n} =: \varphi(x_n)$$

Picking $x_0 = 1$ and iterating, the following sequence is generated:

Iteration	$\varphi(x_n)$	$\sqrt{2}$
1	1.5	1.414213562
2	1.416666667	1.414213562
3	1.414215686	1.414213562
4	1.414213562	1.414213562

$\|x_n - \sqrt{2}\| < 10^{-8}$ in 4 iterations starting from $x_0 = 1$. Compare with bisection method: $\|x_n - \sqrt{2}\| < 10^{-8}$ in 26 iterations.

Newton's Method: Properties

When $F'(x^*) \neq 0$, and F is twice continuously differentiable[3], the Newton sequence $\{x_n\}$ starting from x_0 satisfies

$$|x_{n+1} - x^*| \leq C|x_n - x^*|^2.$$

[3] This can be weakened to $F'(x)$ Lipschitz continuous.

Newton's Method: Properties

When $F'(x^*) \neq 0$, and F is twice continuously differentiable[3], the Newton sequence $\{x_n\}$ starting from x_0 satisfies

$$|x_{n+1} - x^*| \leq C|x_n - x^*|^2.$$

- Guaranteed **local** quadratic convergence
- Quadratic convergence means we're **doubling** our accuracy at each step (locally).

[3] This can be weakened to $F'(x)$ Lipschitz continuous.

Proof

The key is to use **Taylor's Series**.

Proof

The key is to use **Taylor's Series**.

Expand $F(x^*)$ about x_n:

$$0 = F(x_n) + F'(x_n)(x^* - x_n) + \frac{1}{2}F''(\eta)(x^* - x_n)^2.$$

Proof

Expand $F(x^*)$ about x_n:

$$0 = F(x_n) + F'(x_n)(x^* - x_n) + \frac{1}{2}F''(\eta)(x^* - x_n)^2.$$

Move $F(x_n)$ and $F'(x_n)(x^* - x_n)$ to the left-hand-side and divide by $F'(x_n)$.

$$\left| x_n - \frac{F(x_n)}{F'(x_n)} - x^* \right| = \left| \frac{F''(\eta)}{2F'(x_n)} \right| |x_n - x^*|^2$$

$$|x_{n+1} - x^*| \leq C |x_n - x^*|^2,$$

where $|F''(\eta)/(2F'(x_n))| \leq C$ in a neighborhood of x^*.

Concerns?

Newton's Method: Singular Case

If $F'(x^*) = 0$, then we can no longer bound $|F''(\eta)/(2F'(x_n))|$ in a neighborhood of x^*.

In this case, Newton's method only converges **linearly**.

If $F''(x^*) \neq 0$, then

$$\lim_{n\to\infty} \frac{|x_{n+1} - x^*|}{|x_n - x^*|} = \frac{1}{2}.$$

Newton's Method: Singular Case

If $F'(x^*) = 0$, then we can no longer bound $|F''(\eta)/(2F'(x_n))|$ in a neighborhood of x^*.

In this case, Newton's method only converges **linearly**.

If $F''(x^*) \neq 0$, then

$$\lim_{n\to\infty} \frac{|x_{n+1} - x^*|}{|x_n - x^*|} = \frac{1}{2}.$$

- This is **much** slower than when $F'(x^*) \neq 0$.
- The proof also makes heavy use of Taylor series, but is a bit longer, so we omit it here.

Newton's Method: Singular Case

If $F'(x^*) = 0$, then we can no longer bound $|F''(\eta)/(2F'(x_n))|$ in a neighborhood of x^*.

In this case, Newton's method only converges **linearly**.

If $F''(x^*) \neq 0$, then

$$\lim_{n\to\infty} \frac{|x_{n+1} - x^*|}{|x_n - x^*|} = \frac{1}{2}.$$

- This is **much** slower than when $F'(x^*) \neq 0$.
- The proof also makes heavy use of Taylor series, but is a bit longer, so we omit it here.
- **We can recover fast convergence though!**

Brief Tangent: Numerical Method in Pure Math

Numerical methods have uses beyond computing numerical solutions.

Brief Tangent: Numerical Method in Pure Math

Numerical methods have uses beyond computing numerical solutions.

- The bisection method $\xrightarrow{\text{proves}}$ Intermediate Value Theorem

Brief Tangent: Numerical Method in Pure Math

Numerical methods have uses beyond computing numerical solutions.

- The bisection method $\xrightarrow{\text{proves}}$ Intermediate Value Theorem
- Newton's method $\xrightarrow{\text{proves}}$ Inverse and Implicit Function Theorems

Brief Tangent: Numerical Method in Pure Math

Numerical methods have uses beyond computing numerical solutions.

- The bisection method $\xrightarrow{\text{proves}}$ Intermediate Value Theorem
- Newton's method $\xrightarrow{\text{proves}}$ Inverse and Implicit Function Theorems
- Fixed-point method $\xrightarrow{\text{proves}}$ Existence and Uniqueness of solutions to differential equations

Generalizing Newton's Method

Problems from $f : \mathbb{R} \to \mathbb{R}$ aren't a big problem today.

The problems keeping people at night are nonlinear systems of equations $F(x) : \mathbb{R}^n \to \mathbb{R}^n$, where n can be in the millions.

Calculus in $\mathbb{R}^n$

Given a function $F : \mathbb{R}^n \to \mathbb{R}^n$, how should we define its derivative?

In one variable, the derivative of $f : \mathbb{R} \to \mathbb{R}$ at p gives us the equation of the tangent line at p, i.e., the linearization of F at p:

$$f(x) \approx f(p) + f'(p)(x - p)$$

for x near p.

Calculus in $\mathbb{R}^n$

We can define the derivative of $F : \mathbb{R}^n \to \mathbb{R}^n$ at a vector $p \in \mathbb{R}^n$ by the same principal.

Using the norm $\|\cdot\|$, we want $F'(p)$ to be the **matrix** such that if $\|x - p\|$ is small, then

$$F(x) \approx F(p) + F'(p)(x - p).$$

- Note that $F(x)$, $F(p)$, and $x - p$ are **vectors**, and $F'(p)$ is a **matrix**.

Rigorous Definition of $F'(p)$

Definition 1

We say the function $F : \mathbb{R}^n \to \mathbb{R}^n$ is **differentiable at** $p \in \mathbb{R}^n$**, with derivative** $F'(p) \in \mathbb{R}^{n\times n}$, if[4]

$$\lim_{x\to p} \frac{\|F(x) - F(p) - F'(p)(x-p)\|}{\|x-p\|} = 0.$$

- One can show that $F'(p)$, if it exists, is unique.
- This definition leads to a generalization of Taylor's series.

[4] In more general settings, this is known as the Fréchet Derivative.

The Jacobian Matrix

Write $F(x) = (F_1(x), F_2(x), \dots, F_n(x))^T$, and $x = (x_1, x_2, \dots, x_n)^T$. Directly from the definition of $F'(p)$, one can show

$$F'(p) = \begin{bmatrix} \frac{\partial F_1(p)}{\partial x_1} & \frac{\partial F_1(p)}{\partial x_2} & \cdots & \frac{\partial F_1(p)}{\partial x_n} \\ \frac{\partial F_2(p)}{\partial x_1} & \frac{\partial F_2(p)}{\partial x_2} & \cdots & \frac{\partial F_2(p)}{\partial x_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial F_n(p)}{\partial x_1} & \frac{\partial F_n(p)}{\partial x_2} & \cdots & \frac{\partial F_n(p)}{\partial x_n} \end{bmatrix}$$

Multivariate Newton's Method

Given nonlinear $F : \mathbb{R}^n \to \mathbb{R}^n$, Newton's method takes the form

$$x_{k+1} = x_n - F'(x_n)^{-1} F(x_n).$$

- **Key Difference from 1D**: $F'(x_n)^{-1}$ is the **inverse matrix** of $F'(x_n)$.
- Thus questions and concerns when the derivative is zero now shift to when $F'(x)$ is **singular**.

Recent and Ongoing Projects

Accelerating Newton's Method at Singular points

Applications to Bifurcation Problems

Singular Points

Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a nonlinear function[5]. If x^* is a vector in $\mathbb{R}^n$ such that $F'(x^*)$ is **singular**, then we say that x^* is a **singular point**.

[5] The Newton theory presented here holds mutatis mutandis in a Hilbert or Banach space.

Newton's Method & Singular Points

When $F'(x^*)$ is nonsingular, Newton's method converges (locally) quadratically.

When $F'(x^*)$ is singular, Newton's method converges (locally) linearly.

Motivation: Bifurcation Points

- Consider a function $F : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$. We'll write the input vectors as (x, λ), where $x \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}$.

Motivation: Bifurcation Points

- Consider a function $F : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$. We'll write the input vectors as (x, λ), where $x \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}$.
- If we fix λ, then we have a function $F_\lambda : \mathbb{R}^n \to \mathbb{R}^n$.

Motivation: Bifurcation Points

- Consider a function $F : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$. We'll write the input vectors as (x, λ), where $x \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}$.
- If we fix λ, then we really just have a function $F_\lambda : \mathbb{R}^n \to \mathbb{R}^n$.

Question: Suppose $F_\lambda(x^*) = 0$. Is x^* the only "nearby" solution?

Motivation: Bifurcation Points

- Consider a function $F : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$. We'll write the input vectors as (x, λ), where $x \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}$.
- If we fix λ, then we really just have a function $F_\lambda : \mathbb{R}^n \to \mathbb{R}^n$.

Question: Suppose $F_\lambda(x^*) = 0$. Is x^* the only local solution?
Answer: If $F'_\lambda(x^*)$ is nonsingular, then yes (by the implicit function theorem).

Motivation: Bifurcation Points

- Consider a function $F : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$. We'll write the input vectors as (x, λ), where $x \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}$.
- If we fix λ, then we really just have a function $F_\lambda : \mathbb{R}^n \to \mathbb{R}^n$.

Question: Suppose $F_\lambda(x^*) = 0$. Is x^* the only local solution?
Answer: If $F'_\lambda(x^*)$ is nonsingular and continuous, then yes (by the implicit function theorem).

Necessary condition for bifurcation: $F'_\lambda(x^*)$ must be **singular**.

Figure 4. Train tracks in Canterbury, New Zealand, that buckled after an earthquake in 2010, in a shape that resembles the graph of $w_2(x)$.

Figure: A real life bifurcation. Image taken from December 2020 AMS Notices [BP20]

Improving Newton with Anderson Acceleration

Anderson is an **extrapolation** technique that combines previous guesses to construct a better one.

Extrapolation techniques can improve your guess at a lower computational cost than, for example, computing another Newton iterate.

Improving Newton with Anderson Acceleration

Suppose we seek a fixed point of g, and we have computed $m+1$ iterates $\{x_k, x_{k-1}, \ldots, x_{k-m}\}$ where $x_i = g(x_{i-1})$. Let $w_{k+1} = g(x_k) - x_k$.

$$E_k = \Big((x_k - x_{k-1}) \cdots (x_{k-m+1} - x_{k-m}) \Big), \qquad F_k = \Big((w_{k+1} - w_k) \cdots (w_{k-m+2} - w_{k-m+1}) \Big),$$

and $\gamma_{k+1} = \text{argmin}_{\gamma \in \mathbb{R}^m} \|w_{k+1} - F_k\gamma\|_2$. Then

$$x_{k+1}^{AA} = x_k + \beta w_{k+1} - (E_k + \beta F_k)\gamma_{k+1}. \tag{1}$$

Here $\beta \in (0,1]$ is a damping parameter. The results in this talk apply to $\beta = 1$ and $m = 1$.

We define the **optimization gain** [PR21]as

$$\theta_{k+1} := \frac{\|w_{k+1} - F_k\gamma_{k+1}\|}{\|w_{k+1}\|}. \tag{2}$$

History of Anderson Acceleration (AA)

(1965) First introduced by D.G. Anderson.

(1980) The closely related method, DIIS or Pulay Mixing, is introduced by Peter Pulay in *Convergence acceleration of iterative sequences. The case of SCF iteration*.

(2009) Fang and Saad prove that AA is a type of multisecant method in *Two classes of multisecant methods for nonlinear acceleration*.

(2011) Walker and Ni show that, for linear problems, AA is equivalent to the well-known GMRES method in *Anderson Acceleration for Fixed-Point Iterations*.

(2015) Toth and Kelley provide first convergence proof for Anderson for contractive operators.

(2020) Evans, Pollock, Rebholz, and Xiao prove that Anderson improves rate of convergence for linearly convergent fixed-point iterations.

Open prior to 2023: why is Anderson effective when applied to **singular problems**?
Convergence?

Nonsingular Newton-Anderson

- In 2021, Sara Pollock (UF) and her collaborator Leo Rebholz (Clemson) published *Anderson acceleration for contractive and noncontractive operators*.
- A key result in their (and collaborators') 2020 paper is that a successful Anderson step at step $k+1$ is determined by (when $m=1$)

$$\theta_{k+1} := \frac{\|w_{k+1} - \gamma_{k+1}(w_{k+1} - w_k)\|_2}{\|w_{k+1}\|_2}.$$

Nonsingular Newton-Anderson

- In 2021, Sara Pollock (UF) and her collaborator Leo Rebholz (Clemson) published *Anderson acceleration for contractive and noncontractive operators*.
- A key result in their (and collaborators') 2020 paper is that a successful Anderson step at step $k+1$ is determined by (when $m = 1$)

$$\theta_{k+1} := \frac{\|w_{k+1} - \gamma_{k+1}(w_{k+1} - w_k)\|_2}{\|w_{k+1}\|_2}.$$

- Loosely speaking,

$$\begin{aligned} \theta_{k+1} << 1 &\implies \text{acceleration} \\ \theta_{k+1} \approx 1 &\implies \text{no acceleration} \end{aligned}$$

Singular Newton-Anderson (NA)

It had been observed numerically that Newton-Anderson can greatly improve convergence in singular problems.

Table 3
Results for degenerate problems.

Problem	Method	Iterations (k)	$\|f(x_k)\|$	$\|w_k\|$	q_k	Time (s)
D1. $n = 3$	Newton	14	3.991e−09	6.903e−05	1.077	0.000129
	N. Anderson(1)	5	1.656e−10	1.349e−05	1.493	7.152e−05
	KS-acc. N. (1.0, 0.9)	3	2.396e−09	0.0001877	1.993	4.476e−05
D2. $n = 1000$	Newton	16	2.628e−09	0.000382	1.075	0.9698
	N. Anderson(1)	6	1.236e−11	0.001947	1.663	0.3671
	N. Anderson(2)	6	1.912e−11	2.595e−06	1.399	0.3658
	KS-acc. N. (1.0, 0.9)	F	–	–	–	–
	KS-acc. N. (0.35, 0.1)	4	3.986e−09	0.002449	1.461	0.4772
D3. $n = 10$	Newton	46	4.339e−09	0.07587	1.057	0.001038
	N. Anderson(1)	17	7.899e−09	0.01347	1.45	0.0008288
	N. Anderson(2)	26	6.781e−11	0.0003507	1.404	0.002012
	N. Anderson(3)	6	3.964e−10	0.09431	5.87	0.000219
	N. Anderson(4)	5	7.289e−25	0.1056	23.05	0.0001861
	KS-acc. N. (1.0, 0.9)	F	–	–	–	–
	KS-acc. N. (0.35, 0.1)	F	–	–	–	–
	KS-acc. N. (0.7, 0.3)	20	1.758e−09	0.07474	1.165	0.0007143

Figure: Pollock, Schwarz, *Benchmarking results for the Newton-Anderson method*. 2020.[PS20].

Rayleigh-Bénard Convection

$$\begin{cases} -\mu\Delta\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla p - \mathrm{Ri}\, T\mathbf{e}_y = \mathbf{0} \\ \nabla\cdot\mathbf{u} = 0 \\ -\kappa\Delta T + \mathbf{u}\cdot\nabla T = 0 \end{cases} \qquad \begin{aligned} T = 1,\ \mathbf{x} \in \Gamma_1 &:= \{1\}\times(0,1), \\ T = 0,\ \mathbf{x} \in \Gamma_2 &:= \{0\}\times(0,1), \\ \nabla T\cdot\mathbf{n} = 0,\ \mathbf{x} \in \Gamma_3 &:= (0,1)\times\{0,1\}, \\ \mathbf{u} = \mathbf{0},\ \mathbf{x} \in \partial\Omega &= \Gamma_1\cup\Gamma_2\cup\Gamma_3. \end{aligned} \tag{3}$$

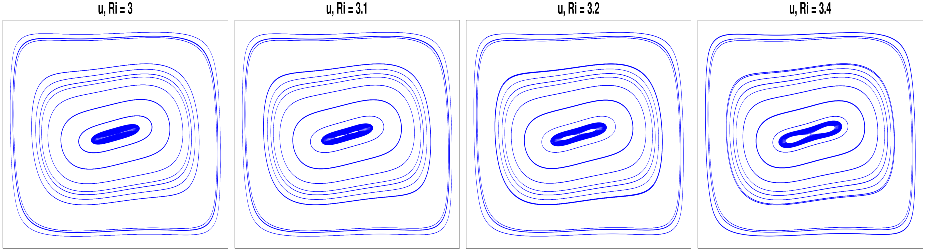


Figure: Velocity streamlines showing transition from one eddy to two eddies.

$\mathcal{P}_2 - \mathcal{P}_1^0$ Scott-Vogelius elements. 7,258 velocity dof and 5,346 pressure dof.

Numerical Results: Rayleigh-Bénard Convection

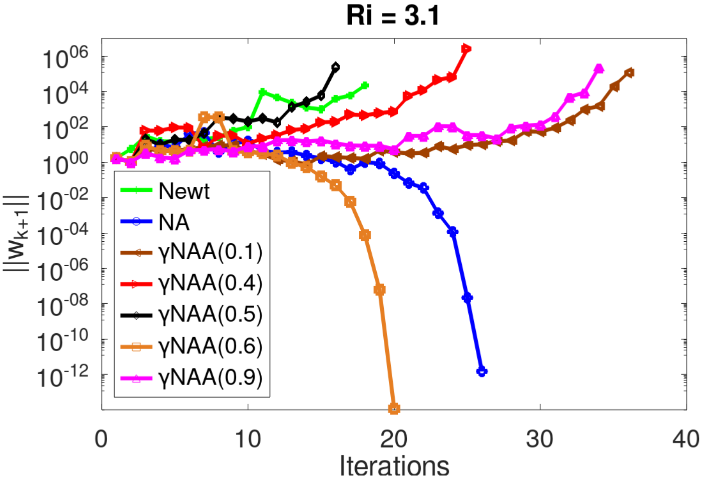


Figure: Residual history. Newton fails to find a solution, Anderson accelerated Newton finds a solution quickly.

Example from Incompressible Channel Flow

$$\begin{cases} -\mu\Delta\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla p = \mathbf{0} \\ \nabla\cdot\mathbf{u} = \mathbf{0} \end{cases} \qquad \begin{aligned} \mathbf{u} &= \mathbf{u}_{\text{in}}, \quad \Gamma_{\text{in}} \\ \mathbf{u} &= 0, \quad \Gamma_{\text{wall}} \\ -p\mathbf{n} + (\mu\nabla\mathbf{u})\mathbf{n} &= 0, \quad \Gamma_{\text{out}} \end{aligned} \tag{4}$$

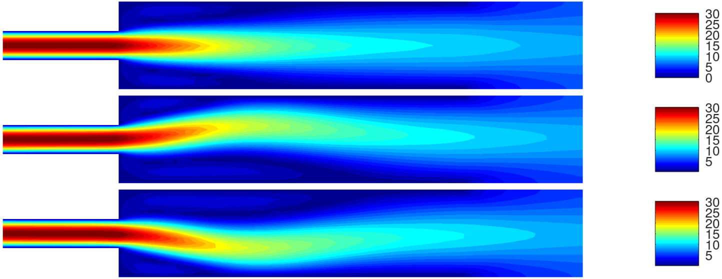


Figure: Stable Solutions to channel flow problem for different μ.

$\mathcal{P}_2 - \mathcal{P}_1$ Taylor-Hood elements. 12, 734 velocity dof and 1,672 pressure dof.

Example from Incompressible Channel Flow

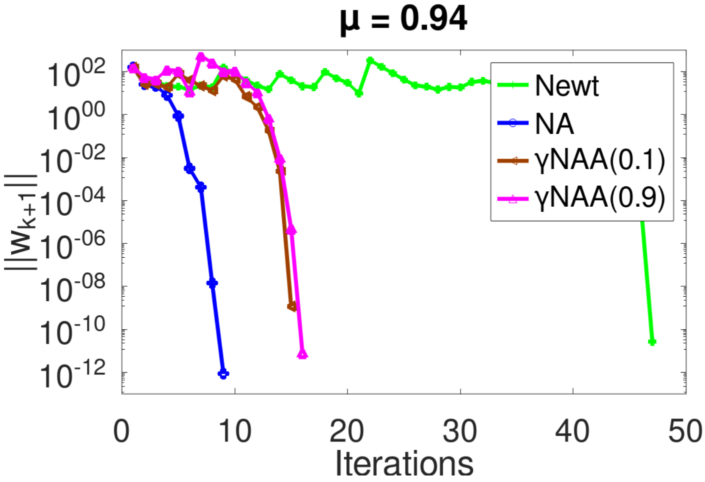


Figure: Residual history for Newton, Newton-Anderson, and γ-safeguarded Newton-Anderson.

It works...but why?

Our recent work[6,7] answers the questions

1. What's the mechanism behind singular NA acceleration?
2. Do the NA iterates remain well-defined and converge to x^*?

[6] M. Dallas and S. Pollock. Newton-Anderson at singular points. Int. J. Numer. Anal. and Mod., 20(5):667–692, 2023

[7] **Dallas**, S. Pollock, L. G. Rebholz, Analysis of an Adaptive Safeguarded Newton-Anderson Algorithm with Applications to Fluid Problems, ACSE, 2024

It works...but why?

Our recent work[6],[7] answers the questions

1. What's the mechanism behind singular NA acceleration?
   $\longrightarrow$ Within the region of invertibility, it's actually θ_{k+1}!
2. Do the NA iterates remain well-defined and converge to x^*?

[6] M. Dallas and S. Pollock. Newton-Anderson at singular points. Int. J. Numer. Anal. and Mod., 20(5):667–692, 2023

[7] **Dallas**, S. Pollock, L. G. Rebholz, Analysis of an Adaptive Safeguarded Newton-Anderson Algorithm with Applications to Fluid Problems, ACSE, 2024

It works...but why?

Our recent work[6,7] answers the questions

1. What's the mechanism behind singular NA acceleration?
   $\longrightarrow$ Within the region of invertibility, it's actually θ_{k+1}!
2. Do the NA iterates remain well-defined and converge to x^*?
   $\longrightarrow$ Sort of. We can prove convergence not of NA itself, but of a safeguarded version which we've called γ-safeguarded NA (γNA(r)).

[6] M. Dallas and S. Pollock. Newton-Anderson at singular points. Int. J. Numer. Anal. and Mod., 20(5):667–692, 2023

[7] **Dallas**, S. Pollock, L. G. Rebholz, Analysis of an Adaptive Safeguarded Newton-Anderson Algorithm with Applications to Fluid Problems, ACSE, 2024

Regions of Invertibility

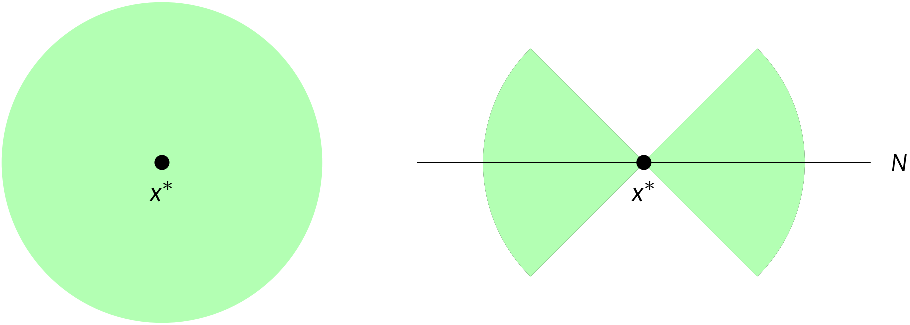


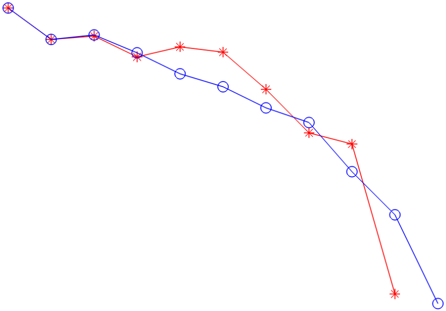
Figure: **Left**: Domain of convergence for Newton's method when $f'(x^*)$ is **nonsingular**. **Right**: Example domain of convergence when $f'(x^*)$ is **singular**. Note that these are also domains of invertibility for f'.

γ-Safeguarding

To prove convergence, we need to ensure that the iterates remain within the region of invertibility. To achieve this, we created the γ-safeguarding algorithm.

Algorithm 2 γ-safeguarding

Given x_k, x_{k-1}, w_{k+1}, w_k, and γ_{k+1}. Set $r \in (0,1)$ and $\lambda = 1$.
$\beta \leftarrow r\|w_{k+1}\|/\|w_k\|$
if $\gamma_{k+1} = 0$ **or** $\gamma_{k+1} \geq 1$ **then**
  $x_{k+1} \leftarrow x_k + w_{k+1}$
else
  if $|\gamma_{k+1}|/|1-\gamma_{k+1}| > \beta$ **then**
    if $\gamma_{k+1} > 0$ **and** $\beta/(\gamma_{k+1}(1+\beta)) < 1$ **then**
      $\lambda \leftarrow \beta/(\gamma_{k+1}(1+\beta))$
    end if
    if $\gamma_{k+1} < 0$ **and** $0 \leq \beta/(\gamma_{k+1}(\beta-1)) < 1$ **then**
      $\lambda \leftarrow \beta/(\gamma_{k+1}(\beta-1))$
    end if
  end if
end if
$\gamma_{k+1} \leftarrow \lambda\gamma_{k+1}$
$x_{k+1} \leftarrow x_k + w_{k+1} - \gamma_{k+1}(x_k - x_{k-1} + w_{k+1} - w_k)$



Local Convergence

- We'll denote γ-safeguarded NA as γNa(r), where r is a parameter set by the user.
- Then, paraphrasing Theorem 6.1 in [DP23], we have

For x_0 sufficiently close to N and x^*, and $x_1 = x_0 + w_1$, γNA(r) remains well-defined and converges to x^* with

$$\begin{aligned} \|P_R e_{k+1}\| &\leq c_4 \max\{|1 - \lambda_{k+1}\gamma_{k+1}| \, \|e_k\|^2, |\lambda_{k+1}\gamma_{k+1}| \, \|e_{k-1}\|^2\} \\ \|P_N e_{k+1}\| &< \kappa \theta_{k+1}^{\lambda} \|P_N e_k\|. \end{aligned}$$

$\kappa \in (1/2, 1)$

What we have and haven't said

- What we've shown[8],[9]:
  - Under certain conditions, Newton-Anderson accelerates Newton iterates in the singular case by the same mechanism seen in the nonsingular case.
  - With γ-safeguarding, Newton-Anderson converges locally, and in general faster than Newton (never worse).
  - γ-safeguarding can help in both the asymptotic regime and preasymptotic regime.
  - with adaptive γ-safeguarding, convergence will not be slowed by NA if the problem is nonsingular.

[8] **Dallas**, S. Pollock, Newton-Anderson at Singular Points, IJNAM, 2023

[9] **Dallas**, S. Pollock, L. G. Rebholz, Analysis of an Adaptive Safeguarded Newton-Anderson Algorithm with Applications to Fluid Problems, ACSE, 2024

What we have and haven't said

- What we've shown[8],[9]:
  - Under certain conditions, Newton-Anderson accelerates Newton iterates in the singular case by the same mechanism seen in the nonsingular case.
  - With γ-safeguarding, Newton-Anderson converges locally, and in general faster than Newton (never worse).
  - γ-safeguarding can help in both the asymptotic regime and preasymptotic regime.
  - with adaptive γ-safeguarding, convergence will not be slowed by NA if the problem is nonsingular.
- Open questions
  - Do these results extend to dim $N > 1$ case?
  - What about for depth $m > 1$?
  - How small can θ_{k+1} be?
  - Preasymptotic regime?

[8] **Dallas**, S. Pollock, Newton-Anderson at Singular Points, IJNAM, 2023

[9] **Dallas**, S. Pollock, L. G. Rebholz, Analysis of an Adaptive Safeguarded Newton-Anderson Algorithm with Applications to Fluid Problems, ACSE, 2024

References I

D. G. Anderson.
Iterative procedures for nonlinear integral equations.
J. Assoc. Comput. Mach., 12(4):547–560, 1965.

Renato G. Bettiol and Paolo Piccione.
Instability and bifurcation.
Notices of the AMS, December:1679 – 1691, 2020.

P. Deuflhard.
Newton Methods for Nonlinear Problems.
Springer Series in Computational Mathematics. Springer, 2005.

M. Dallas and S. Pollock.
Newton-Anderson at singular points.
Int. J. Numer. Anal. and Mod., 20(5):667–692, 2023.

References II

C. Evans, S. Pollock, L. Rebholz, and M. Xiao.
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.

H. Fang and Y. Saad.
Two classes of multisecant methods for nonlinear acceleration.
Numer. Linear Algebra Appl., 16(3):197–221, 2008.

Walter Gautschi.
Numerical Analysis.
Springer, 2012.

Sara Pollock and Leo G. Rebholz.
Anderson acceleration for contractive and noncontractive operators.
IMA J. Numer. Anal., 41(4):2841–2872, 2021.

References III

S. Pollock and H. Schwartz.
Benchmarking results for the Newton–Anderson method.
Results Appl. Math., 8:100095, 2020.

Convergence acceleration of iterative sequences. the case of scf iteration.
Chemical Physics Letters, 73(2):393–398, 1980.

A. Toth and C. T. Kelley.
Convergence analysis for Anderson acceleration.
SIAM J. Numer. Anal., 53(2):805–819, 2015.

H. F. Walker and P. Ni.
Anderson acceleration for fixed-point iterations.
SIAM J. Numer. Anal., 49(4):1715–1735, 2011.

Thank you!

My team