Unconditionally convergent preconditioned Newton's method for Richards' equation

Konstantin Brenner

Laboratoire J.A. Dieudonné Inria & Univ. Côte d'Azur

IFPEN–Inria workshop

September 5, 2022

Outline

Introduction to Richards' equation

- Variably saturated sub-soil flow modeling

Monotone Newton's method

- Convergence proof $\neq$ performance

Preconditioned monotone Newton's method

- Convergence proof $+$ performance

Conclusion

Outline

Introduction to Richards' equation

- Variably saturated sub-soil flow modeling

Monotone Newton's method

- Convergence proof $\neq$ performance

Preconditioned monotone Newton's method

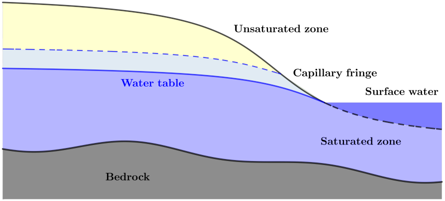
- Convergence proof $+$ performance

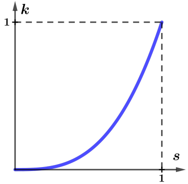
Conclusion

Richards equation

Unsaturated/saturated groundwater flow

$$\partial_t s - \operatorname{div}\left(k(s)\nabla\left(p+z\right)\right) = 0, \qquad s = S(p)$$

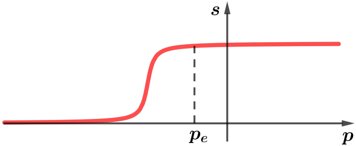




Relative permeability k as a function of saturation

Applications:

- Water resource estimation
- Irrigation
- Contaminant transport
- Interaction with surface water
- ...

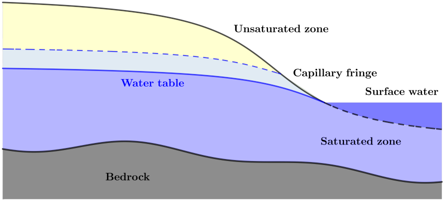


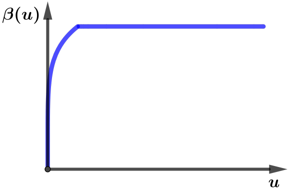
Retention curve $s = S(p)$

Richards equation

Unsaturated/saturated groundwater flow

$$\partial_t s - \operatorname{div}\left(\nabla u - k(s)\nabla z\right) = 0, \qquad s = \beta(u)$$





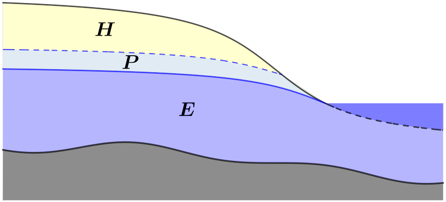
Applications:

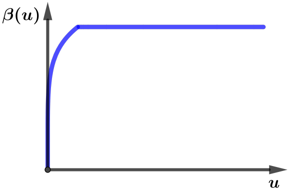
- Water resource estimation
- Irrigation
- Contaminant transport
- Interaction with surface water
- ...

Richards equation

Unsaturated/saturated groundwater flow

$$\partial_t s - \operatorname{div}\left(\nabla u - k(s)\nabla z\right) = 0, \qquad s = \beta(u)$$





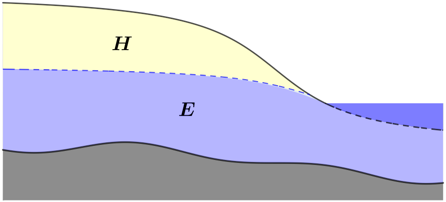
Applications:

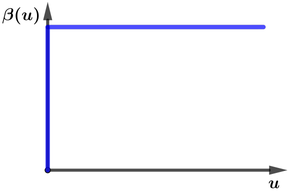
- Water resource estimation
- Irrigation
- Contaminant transport
- Interaction with surface water
- ...

Richards equation

Unsaturated/saturated groundwater flow

$$\partial_t s - \operatorname{div}\left(\nabla u - k(s)\nabla z\right) = 0, \qquad s \in \beta(u)$$





Applications:

- Water resource estimation
- Irrigation
- Contaminant transport
- Interaction with surface water
- ...

Why Richards' equation?

French Wikipedia: La résolution numérique de l'équation de Richards demeure l'un des problèmes d'analyse numérique les plus difficiles pour les sciences naturelles[1].

English Wikipedia: The numerical solution of the Richards' equation is one of the most challenging problems in earth science[1].

Original article[1]: Richards' equation is ... arguably one of the most difficult equations to reliably and accurately solve in all of hydrosciences.

[1] Farthing and Ogden, Numerical solution of Richards' Equation: a review of advances and challenges, 2017

Why Richards' equation?

French Wikipedia: La résolution numérique de l'équation de Richards demeure l'un des problèmes d'analyse numérique les plus difficiles pour les sciences naturelles[1].

English Wikipedia: The numerical solution of the Richards' equation is one of the most challenging problems in earth science[1].

Original article[1]: Richards' equation is . . . arguably one of the most difficult equations to reliably and accurately solve in all of hydrosciences.

Major numerical challenge: Robustness and efficiency of the nonlinear solvers.

[1] Farthing and Ogden, Numerical solution of Richards' Equation: a review of advances and challenges, 2017

Outline

Introduction to Richards' equation

- From saturated to unsaturated flow

Monotone Newton's method

- Convergence proof $\neq$ performance

Preconditioned monotone Newton's method

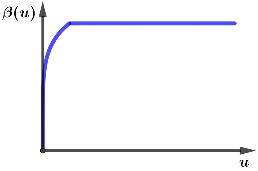
- Convergence proof $+$ performance

Conclusion

Model nonlinear system

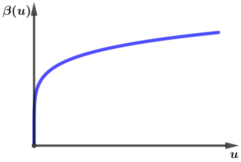
Find $\boldsymbol{u} \in \mathbb{R}_+^N$ satisfying

$$\beta(\mathbf{u}) + A\mathbf{u} = \boldsymbol{b}, \qquad \boldsymbol{b} \geq 0$$



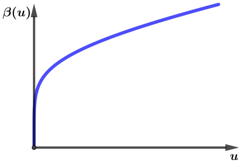
Richards' equation

$$\partial_t \min\left(u^{1/m}, 1\right) = \Delta u + \text{LOT}$$



Porous medium equation

$$\partial_t u^{1/m} = \Delta u$$



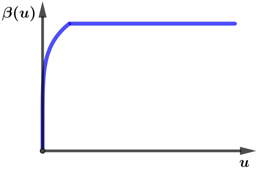
Transport with adsorption

$$\partial_t \left(u + u^{1/m}\right) = \operatorname{div}\left(\nabla u + u\mathbf{V}\right)$$

Model nonlinear system

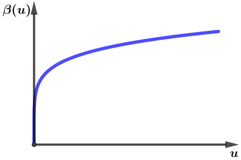
Find $\boldsymbol{u} \in \mathbb{R}_+^N$ satisfying

$$\beta(\mathbf{u}) + A\mathbf{u} = \boldsymbol{b}, \qquad \boldsymbol{b} \geq 0$$



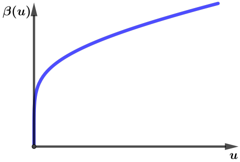
Richards' equation

$$\partial_t \min\left(u^{1/m}, 1\right) = \Delta u + \text{LOT}$$



Porous medium equation

$$\partial_t u^{1/m} = \Delta u$$



Transport with adsorption

$$\partial_t \left(u + u^{1/m}\right) = \operatorname{div}\left(\nabla u + u\mathbf{V}\right)$$

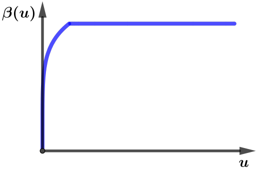
Assumptions for the analysis:

- $\beta_i : \mathbb{R}_+ \to \mathbb{R}_+$ diagonal, increasing and concave, $\beta_i'(0) \leq +\infty$

Model nonlinear system

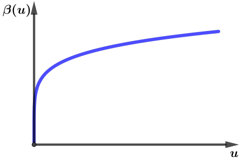
Find $\boldsymbol{u} \in \mathbb{R}_+^N$ satisfying

$$\beta(\mathbf{u}) + A\mathbf{u} = \boldsymbol{b}, \qquad \boldsymbol{b} \geq 0$$



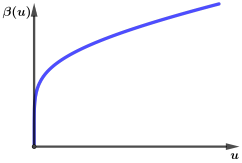
Richards' equation

$$\partial_t \min\left(u^{1/m}, 1\right) = \Delta u + \text{LOT}$$



Porous medium equation

$$\partial_t u^{1/m} = \Delta u$$



Transport with adsorption

$$\partial_t \left(u + u^{1/m}\right) = \operatorname{div}\left(\nabla u + u\mathbf{V}\right)$$

Assumptions for the analysis:

- $\beta_i : \mathbb{R}_+ \to \mathbb{R}_+$ diagonal, increasing and concave, $\beta_i'(0) \leq +\infty$
- $J(\boldsymbol{u}) = \beta'(\boldsymbol{u}) + A$ is M-matrix:
  $J(\boldsymbol{u})^{-1} \geq 0$ and $(J(\boldsymbol{u}))_{ij} \leq 0, i \neq j$

Monotone Newton's method

Notations

$$F(\boldsymbol{u}) = \beta(\boldsymbol{u}) + A\boldsymbol{u} - \boldsymbol{b}$$

Newton's method

$$F'(\boldsymbol{u}_k)\,(\boldsymbol{u}_{k+1} - \boldsymbol{u}_k) + F(\boldsymbol{u}_k) = 0$$

Monotone Newton's method

Notations

$$F(\boldsymbol{u}) = \beta(\boldsymbol{u}) + A\boldsymbol{u} - \boldsymbol{b}$$

Newton's method

$$F'(\boldsymbol{u}_k)\,(\boldsymbol{u}_{k+1} - \boldsymbol{u}_k) + F(\boldsymbol{u}_k) = 0$$

Monotone Newton Theorem (Baluev '52; Ortega & Rheinboldt '70)

Let $\boldsymbol{u}_0$ satisfy $F(\boldsymbol{u}_0) \leq 0$, then

- $F(\boldsymbol{u}_k) \leq 0$ and $\boldsymbol{u}_k \leq \boldsymbol{u}_{k+1} \leq \boldsymbol{u}_\star$ for all $k \geq 0$
- $\boldsymbol{u}_k$ converges to the unique solution $\boldsymbol{u}_\star$

Monotone Newton's method

Notations

$$F(\boldsymbol{u}) = \beta(\boldsymbol{u}) + A\boldsymbol{u} - \boldsymbol{b}$$

Newton's method

$$F'(\boldsymbol{u}_k)\,(\boldsymbol{u}_{k+1} - \boldsymbol{u}_k) + F(\boldsymbol{u}_k) = 0$$

Monotone Newton Theorem (Baluev '52; Ortega & Rheinboldt '70)

Let $\boldsymbol{u}_0$ satisfy $F(\boldsymbol{u}_0) \leq 0$, then

- $F(\boldsymbol{u}_k) \leq 0$ and $\boldsymbol{u}_k \leq \boldsymbol{u}_{k+1} \leq \boldsymbol{u}_\star$ for all $k \geq 0$
- $\boldsymbol{u}_k$ converges to the unique solution $\boldsymbol{u}_\star$

Sketch of proof:

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k - F'(\boldsymbol{u}_k)^{-1} F(\boldsymbol{u}_k) \qquad \Rightarrow \quad \boldsymbol{u}_{k+1} \geq \boldsymbol{u}_k$$

$$F(\boldsymbol{u}_{k+1}) - F(\boldsymbol{u}_k) \leq F'(\boldsymbol{u}_k)\,(\boldsymbol{u}_{k+1} - \boldsymbol{u}_k) = -F(\boldsymbol{u}_k) \quad \Rightarrow \quad F(\boldsymbol{u}_{k+1}) \leq 0.$$

Monotone Newton's method

Notations

$$F(\boldsymbol{u}) = \beta(\boldsymbol{u}) + A\boldsymbol{u} - \boldsymbol{b}$$

Newton's method

$$F'(\boldsymbol{u}_k)\,(\boldsymbol{u}_{k+1} - \boldsymbol{u}_k) + F(\boldsymbol{u}_k) = 0$$

Monotone Newton Theorem (Baluev '52; Ortega & Rheinboldt '70)

Let $\boldsymbol{u}_0$ satisfy $F(\boldsymbol{u}_0) \leq 0$, then

- $F(\boldsymbol{u}_k) \leq 0$ and $\boldsymbol{u}_k \leq \boldsymbol{u}_{k+1} \leq \boldsymbol{u}_\star$ for all $k \geq 0$
- $\boldsymbol{u}_k$ converges to the unique solution $\boldsymbol{u}_\star$

Sketch of proof:

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k - F'(\boldsymbol{u}_k)^{-1} F(\boldsymbol{u}_k) \qquad \Rightarrow \qquad \boldsymbol{u}_{k+1} \geq \boldsymbol{u}_k$$

$$F(\boldsymbol{u}_{k+1}) - F(\boldsymbol{u}_k) \leq F'(\boldsymbol{u}_k)\,(\boldsymbol{u}_{k+1} - \boldsymbol{u}_k) = -F(\boldsymbol{u}_k) \qquad \Rightarrow \qquad F(\boldsymbol{u}_{k+1}) \leq 0.$$

The method is semi-globally convergent. Is it efficient?

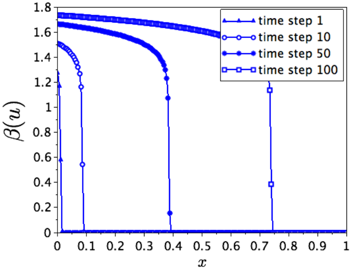
Numerical experiment

Porous medium equation on $(0,1) \times (0,T)$

$$\partial_t \beta(u) - \partial_{xx}^2 u = 0, \qquad \beta(u) = u^{1/m}$$

with Neumann boundary conditions

- Inflow at $x = 0$: $-\partial_x u(0,t) = q > 0$
- No-flow at $x = 1$
- Almost "dry" initial condition: $\beta(u(x,0)) = 10^{-10}$

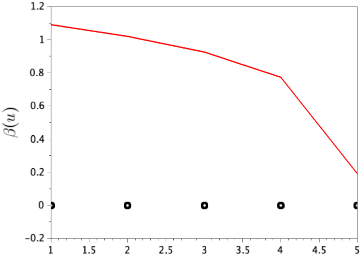


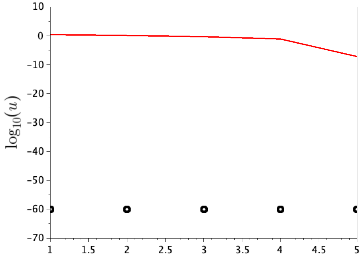
Solution profile at different time steps

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





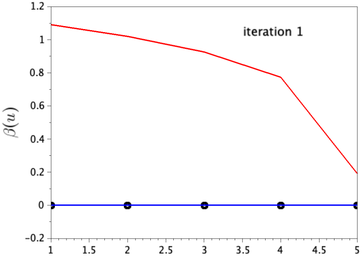
first last jacobi

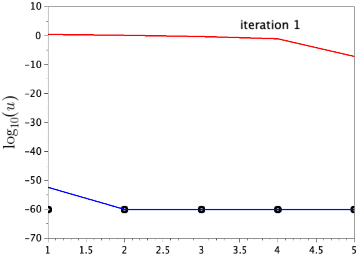
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





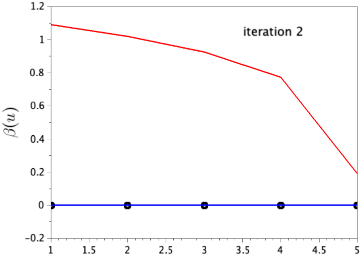
first last jacobi

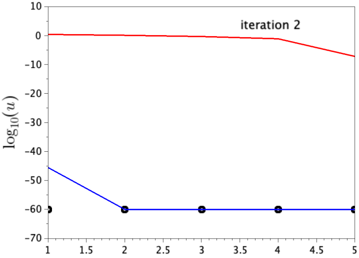
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





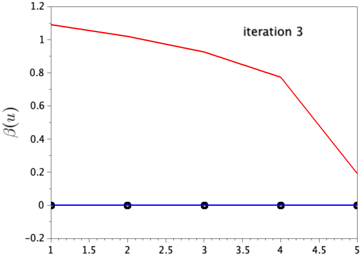
first last jacobi

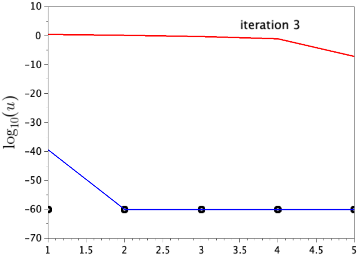
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





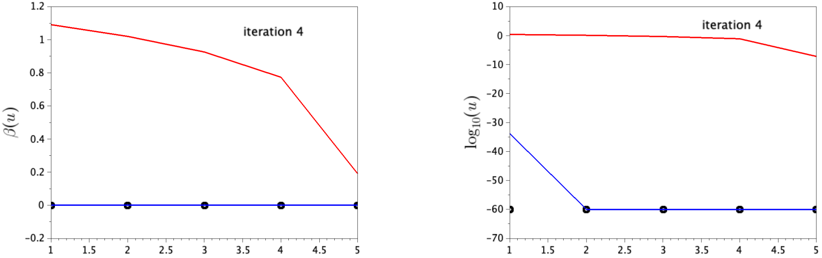
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



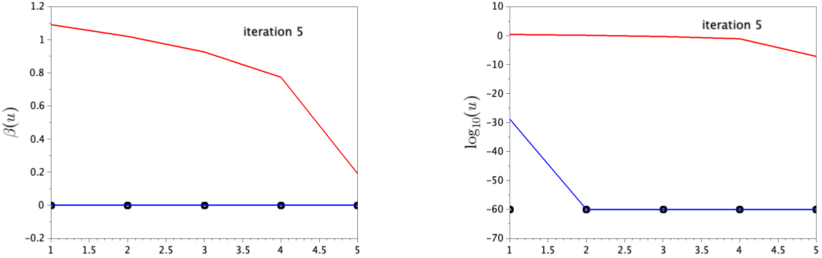
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



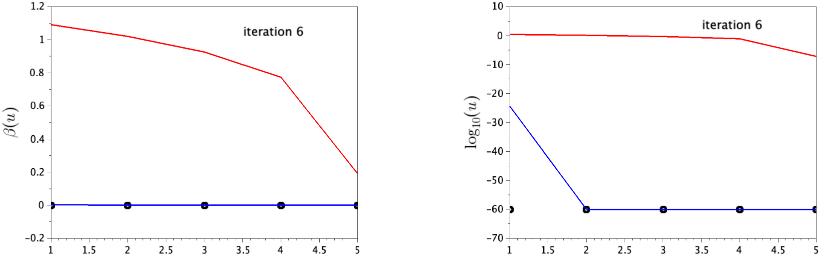
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



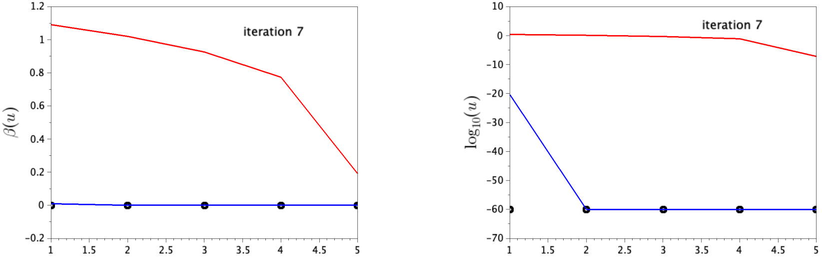
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



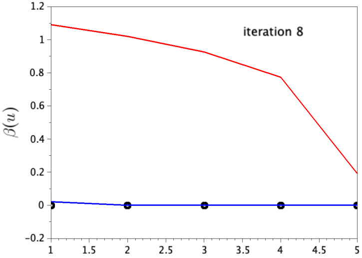
first last jacobi

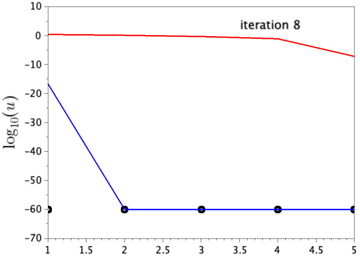
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





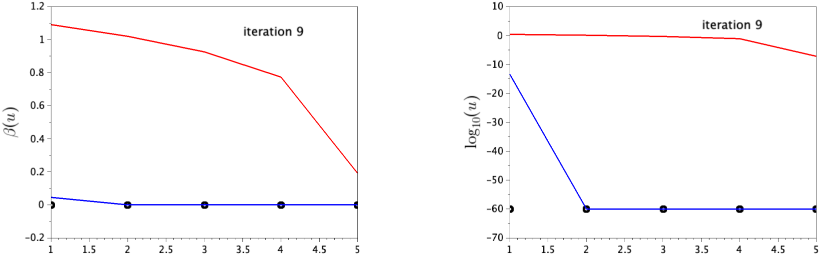
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



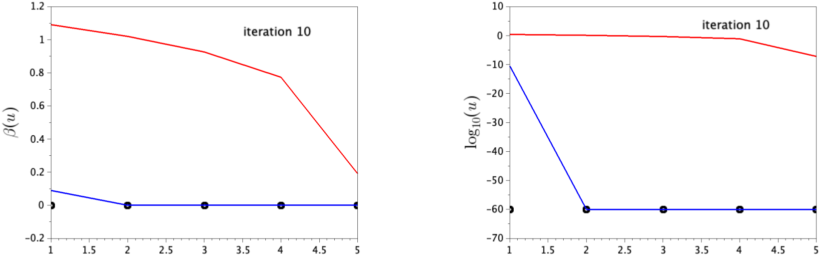
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



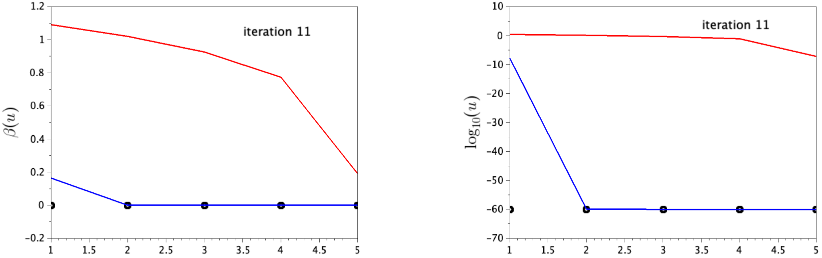
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



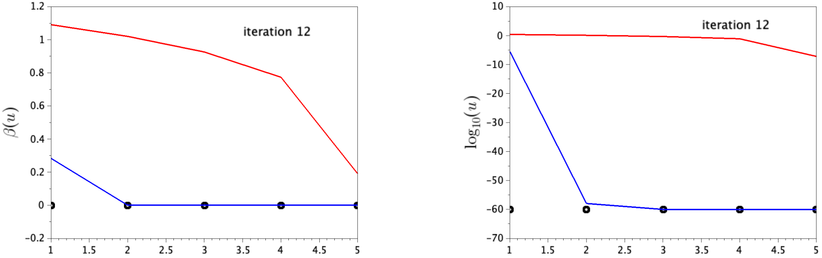
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



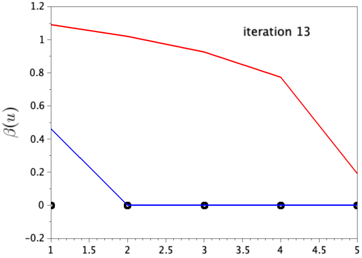
first last jacobi

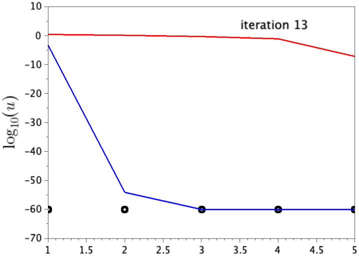
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





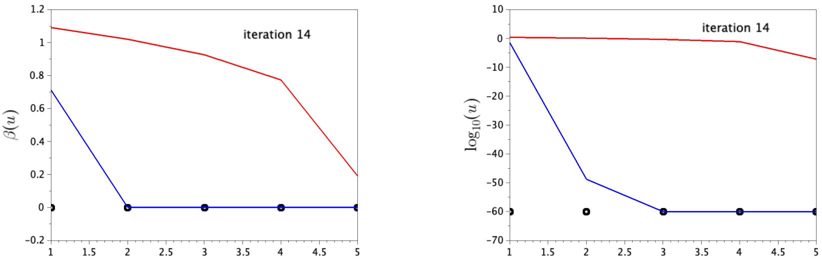
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



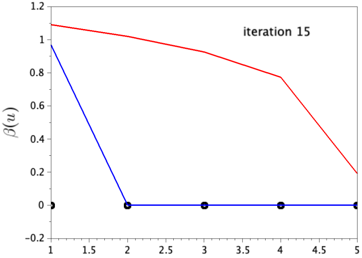
first last jacobi

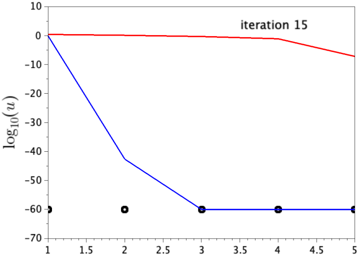
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





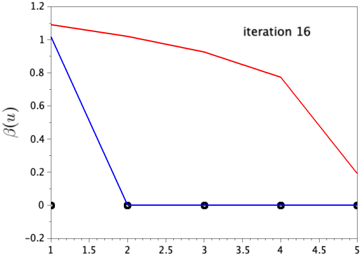
first last jacobi

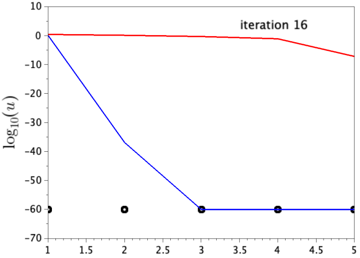
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





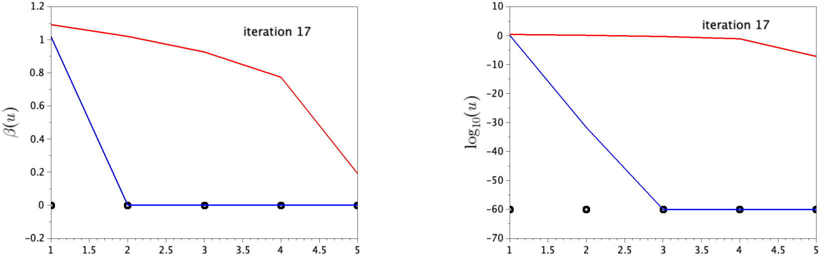
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



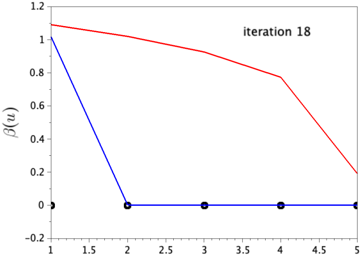
first last jacobi

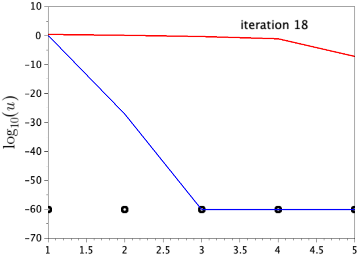
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





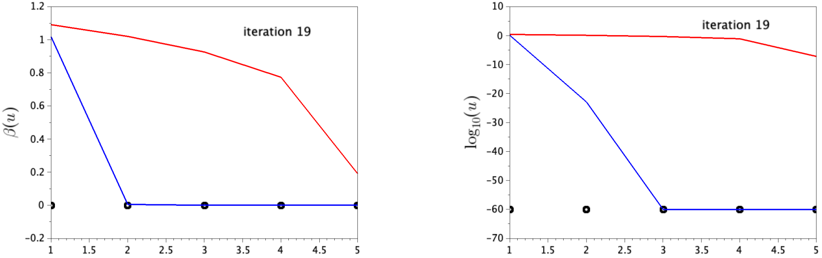
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



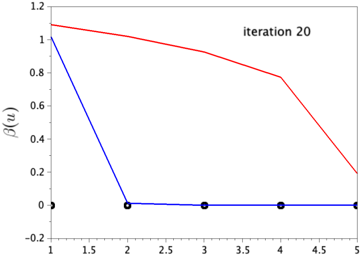
first last jacobi

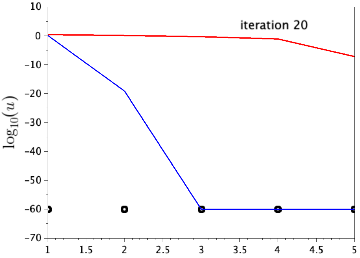
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





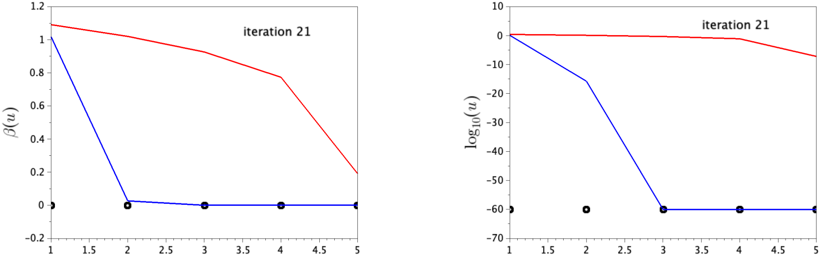
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



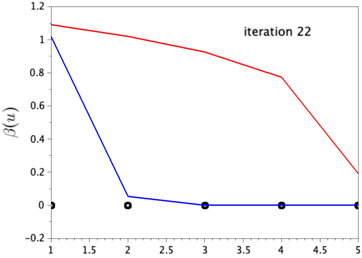
first last jacobi

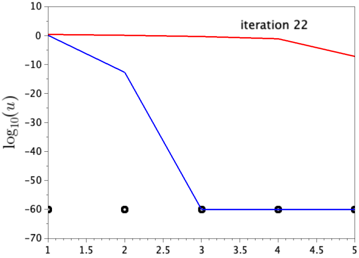
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





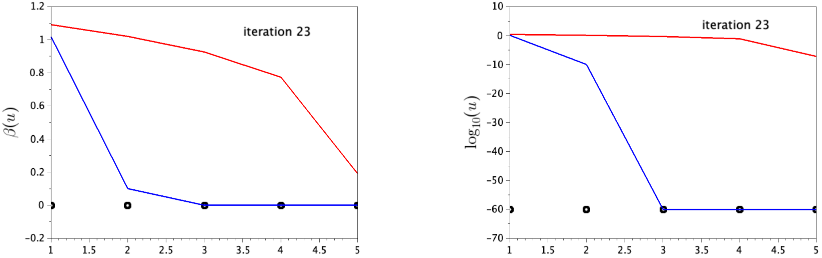
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



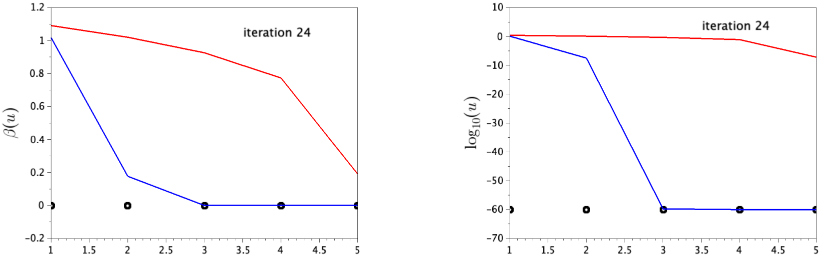
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



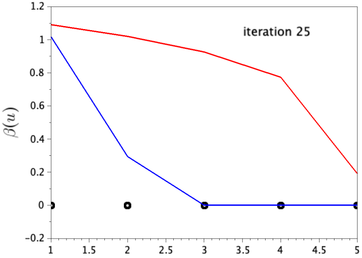
first last jacobi

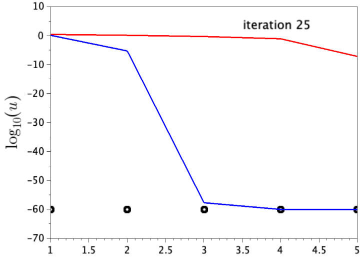
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





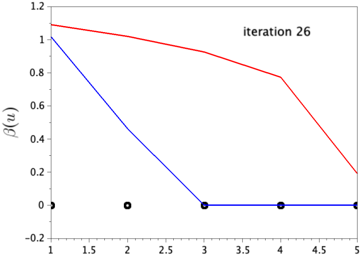
first last jacobi

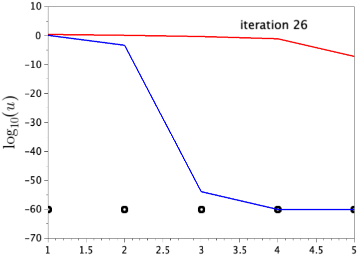
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





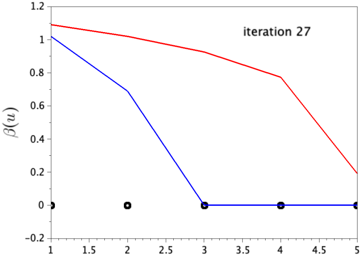
first last jacobi

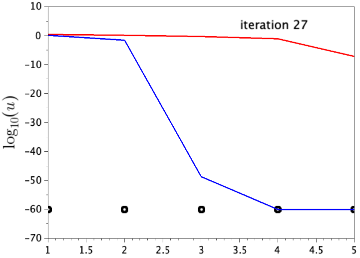
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





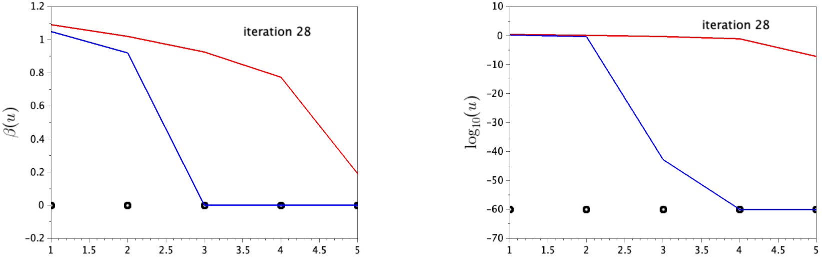
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



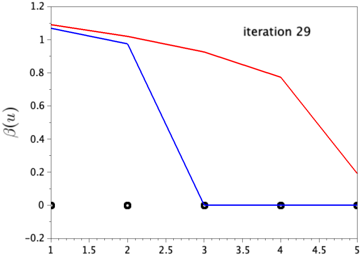
first last jacobi

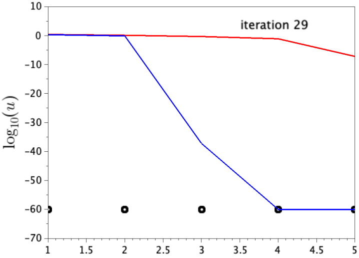
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





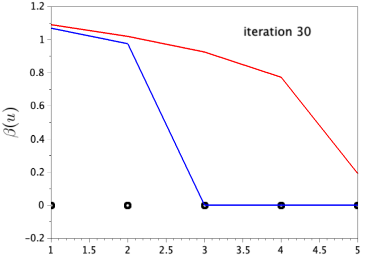
first last jacobi

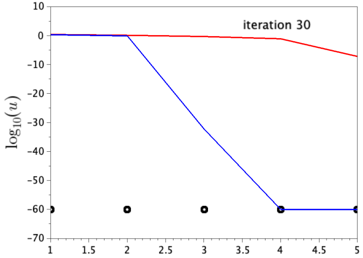
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





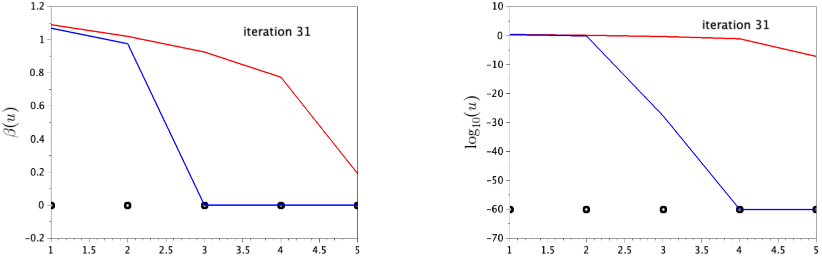
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



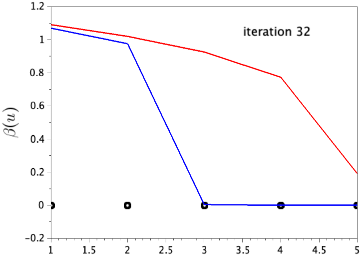
first last jacobi

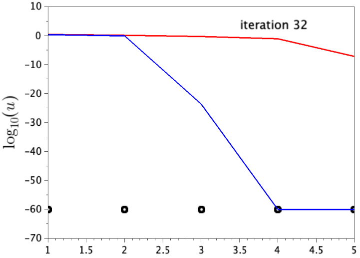
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





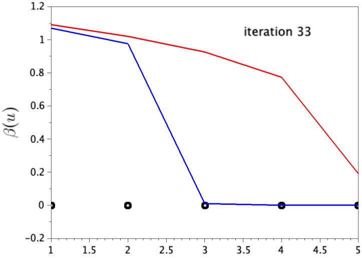
first last jacobi

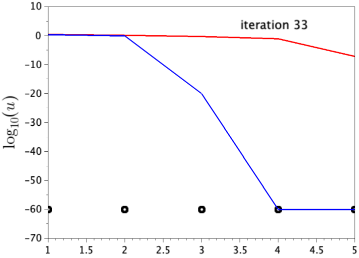
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





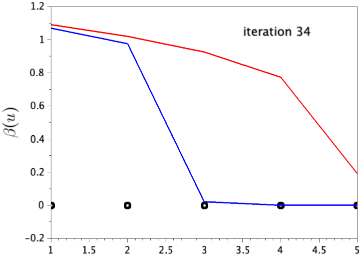
first last jacobi

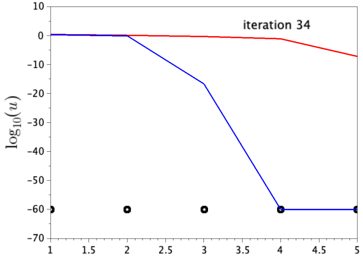
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





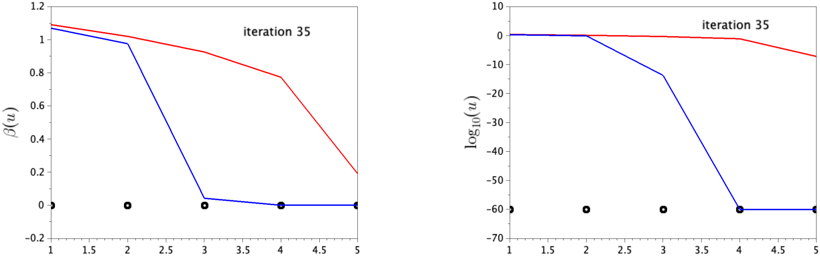
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



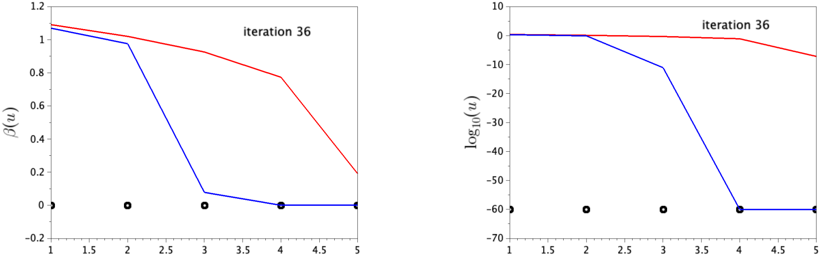
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



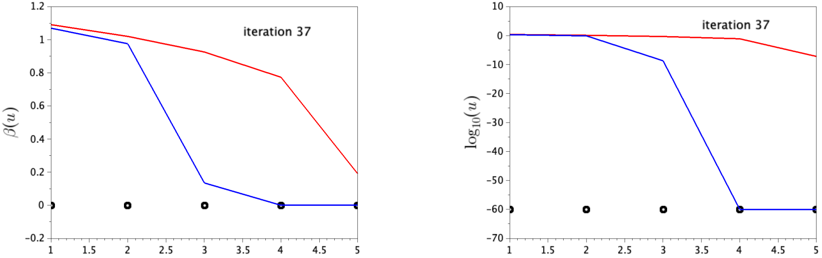
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



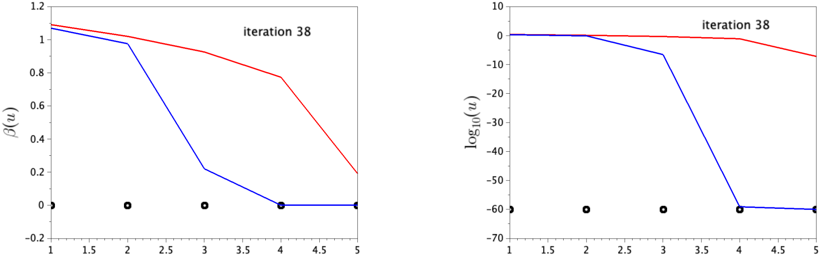
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



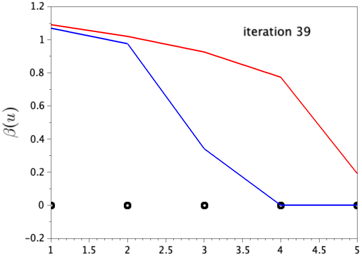
first last jacobi

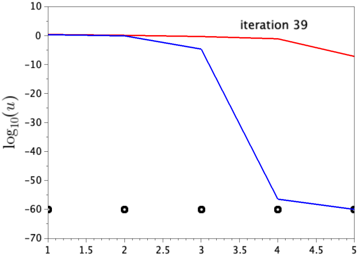
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





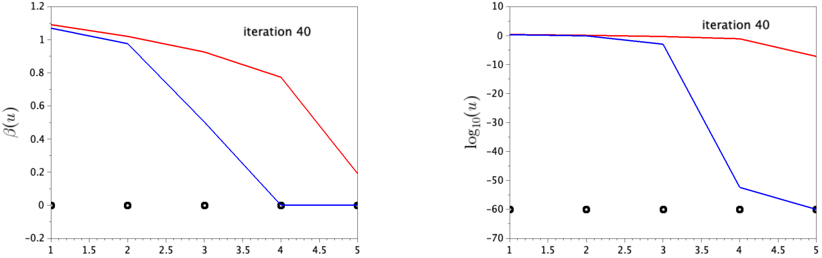
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



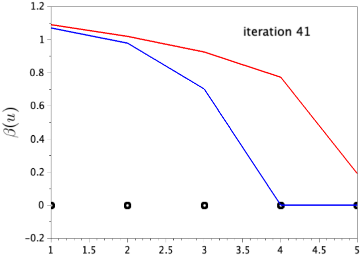
first last jacobi

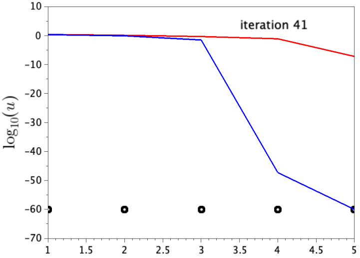
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





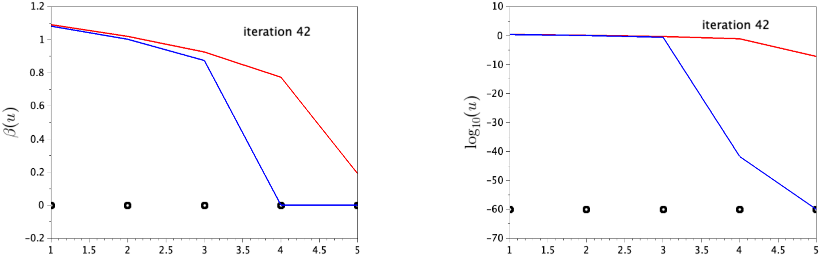
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



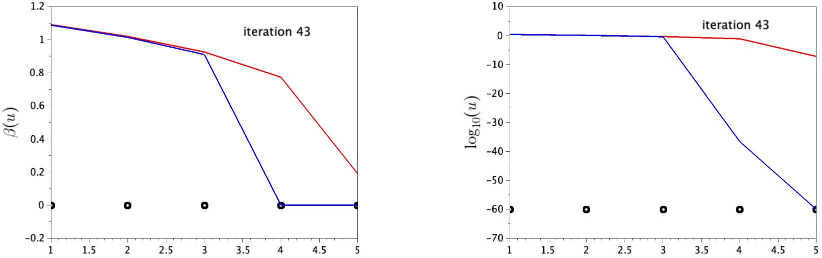
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



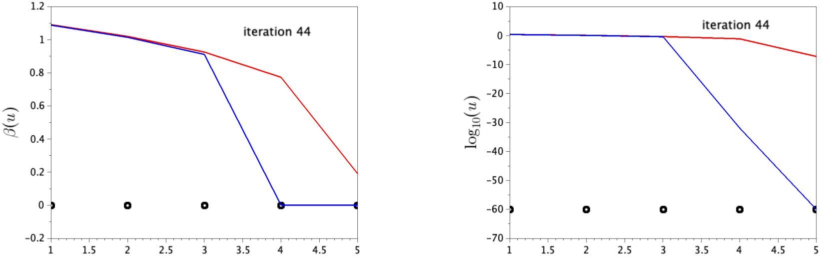
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



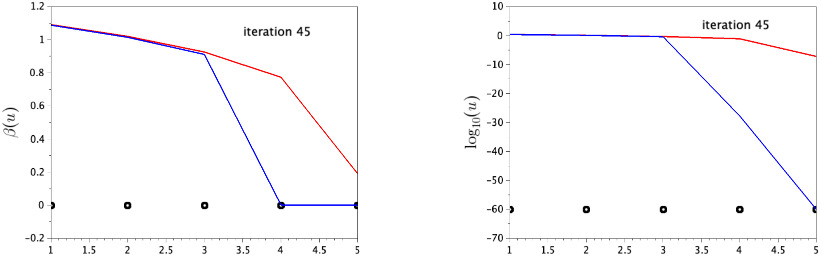
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



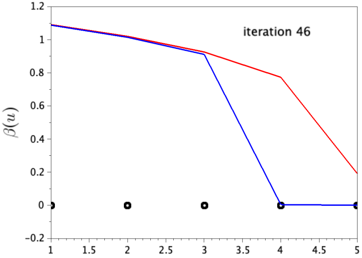
first last jacobi

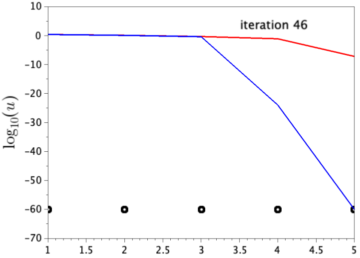
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





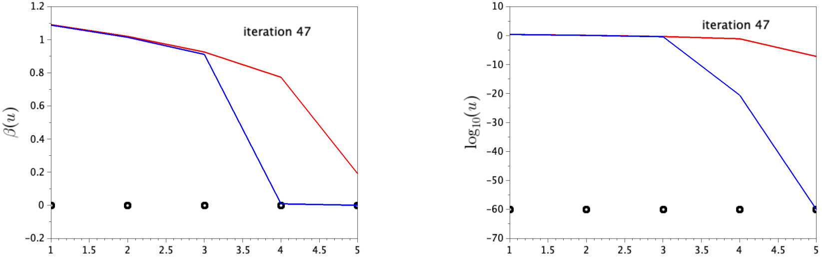
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



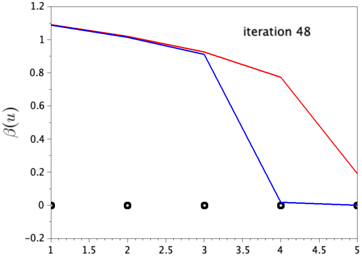
first last jacobi

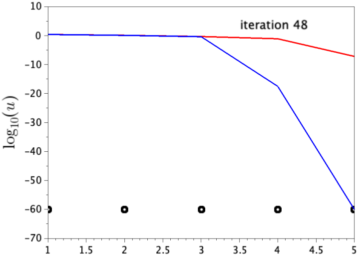
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





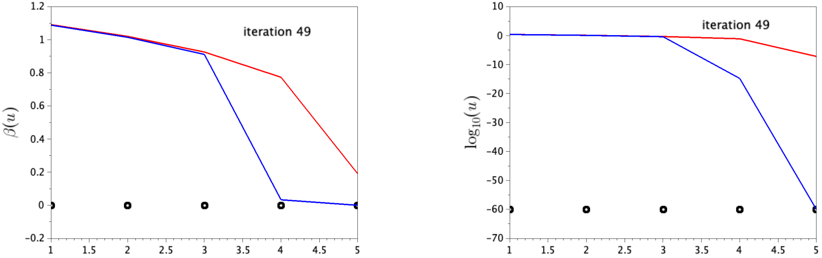
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



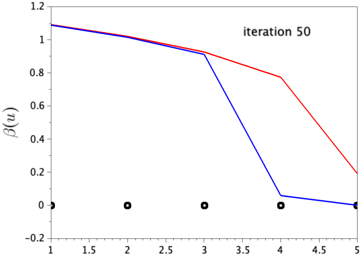
first last jacobi

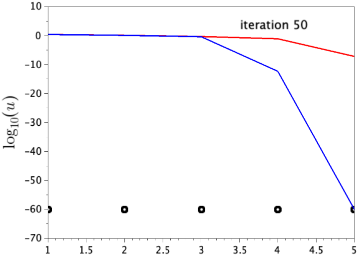
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





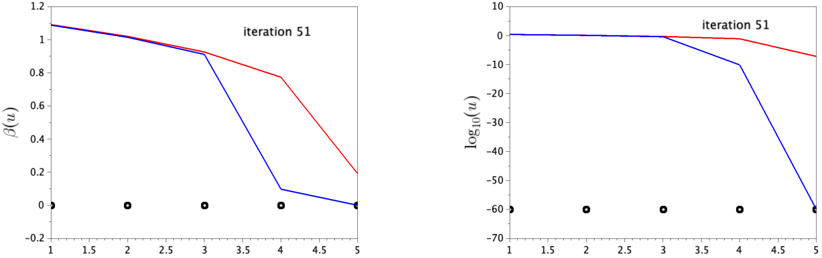
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



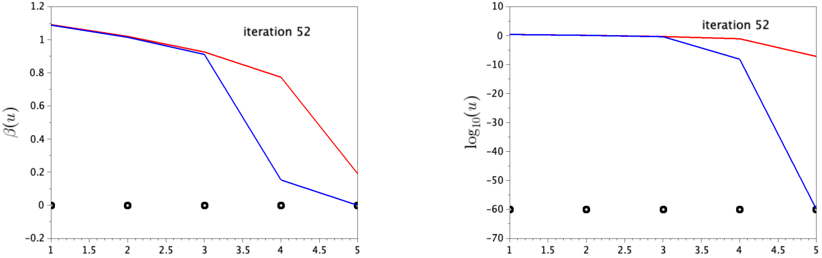
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



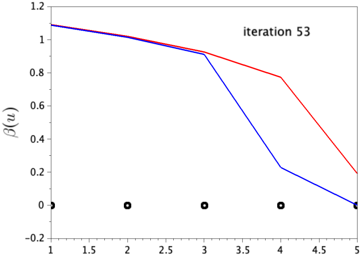
first last jacobi

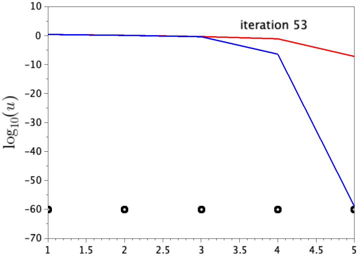
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





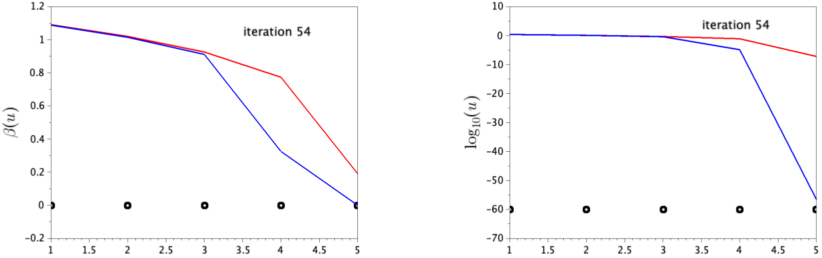
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



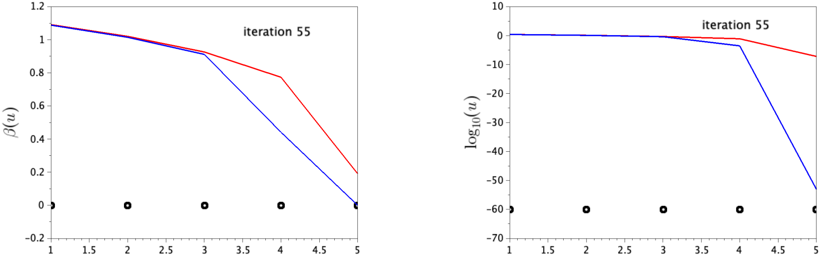
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



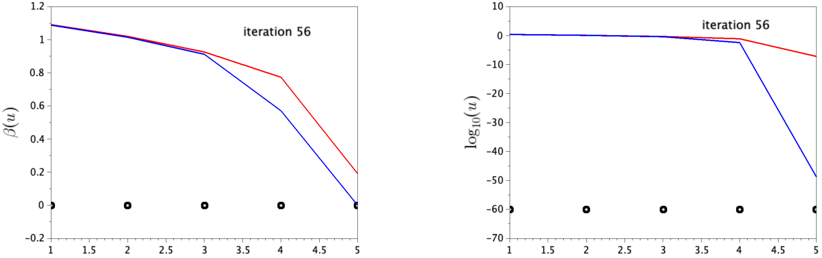
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



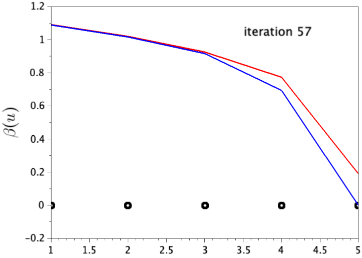
first last jacobi

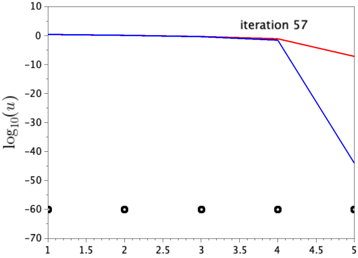
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





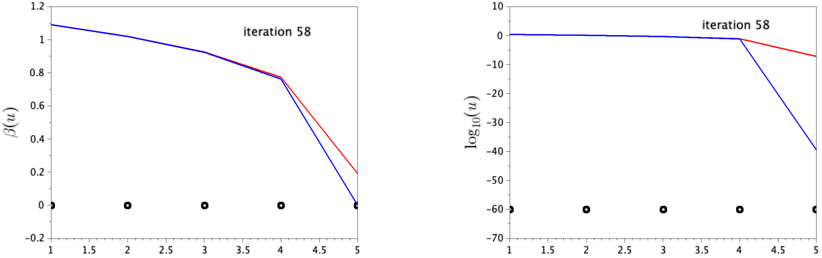
first   last   jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



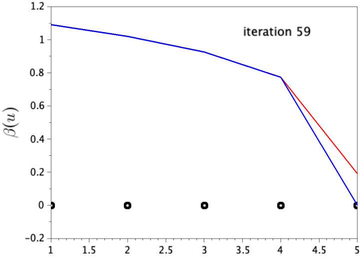
first last jacobi

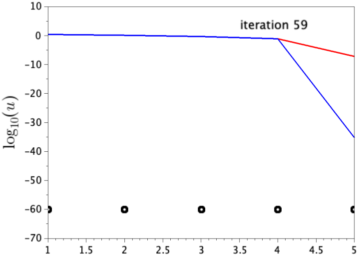
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





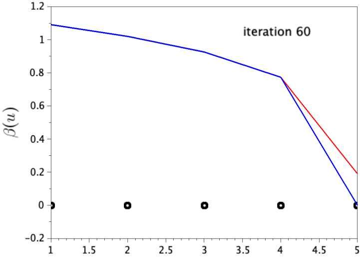
first last jacobi

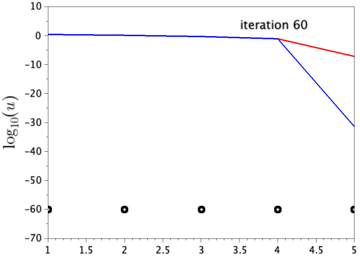
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





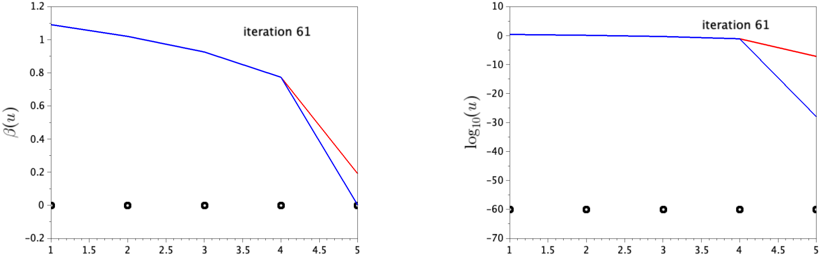
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



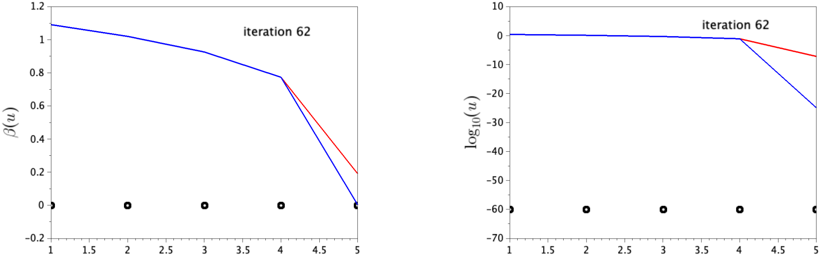
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



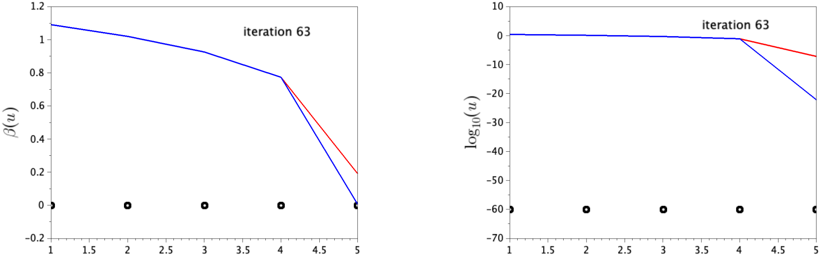
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



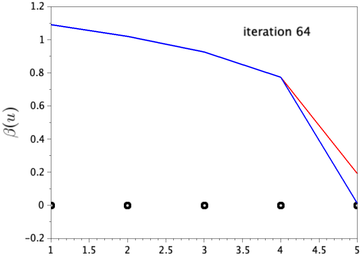
first last jacobi

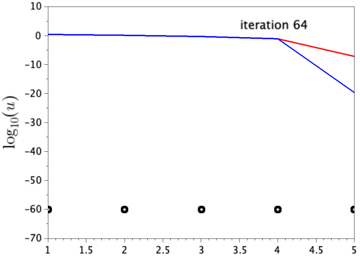
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





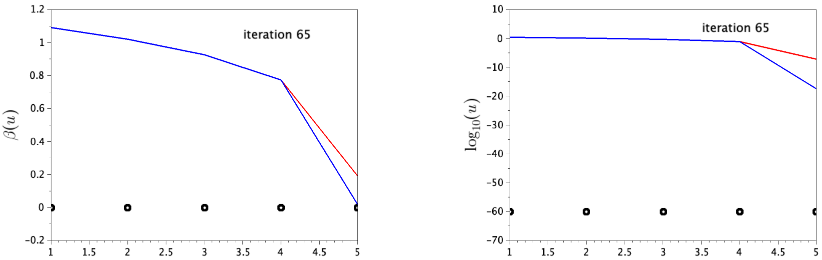
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



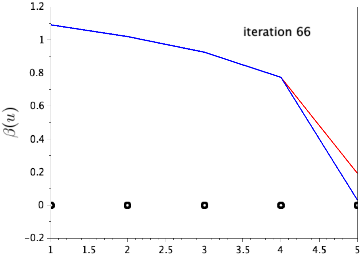
first last jacobi

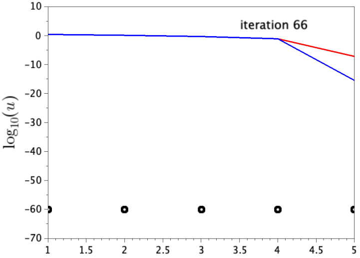
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





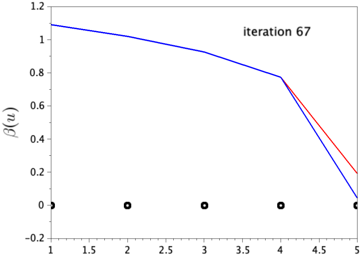
first last jacobi

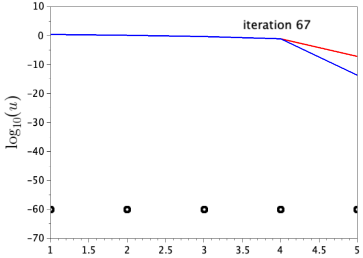
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





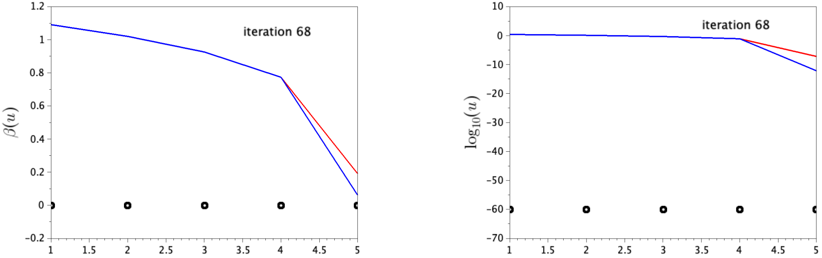
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



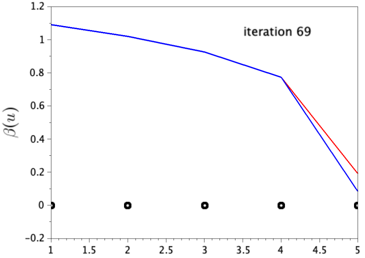
first last jacobi

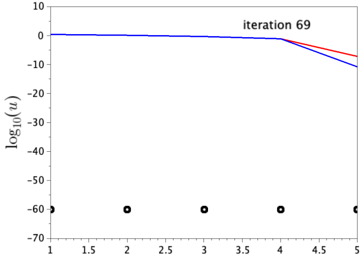
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





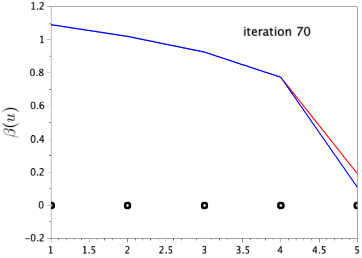
first last jacobi

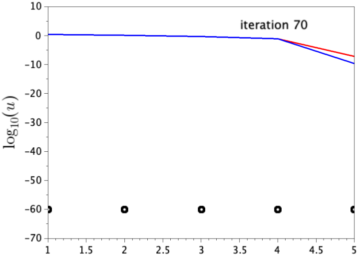
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





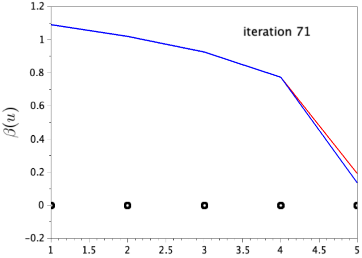
first last jacobi

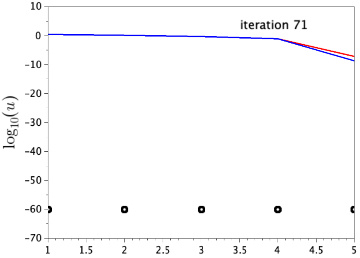
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





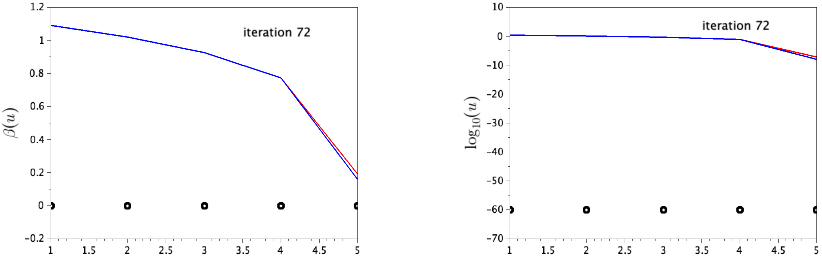
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



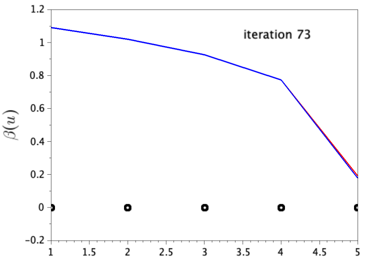
first last jacobi

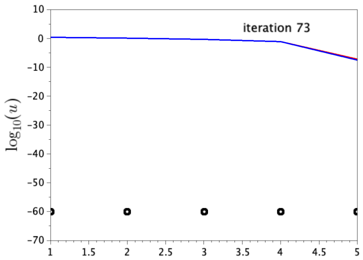
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





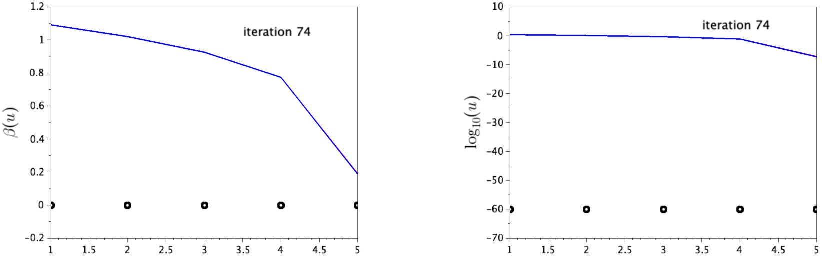
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



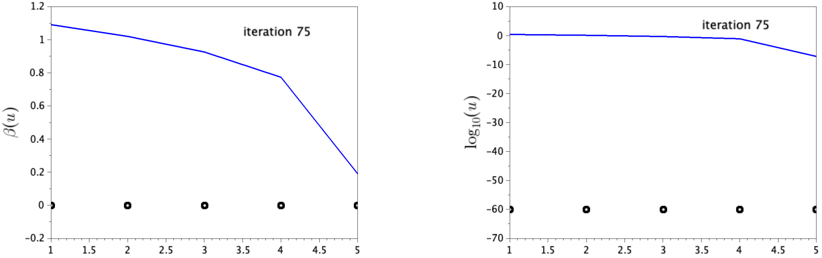
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



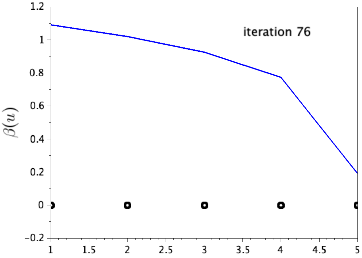
first last jacobi

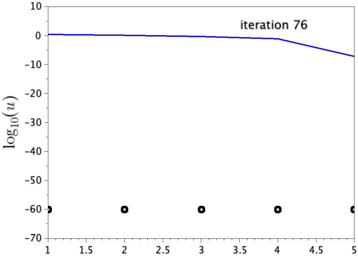
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





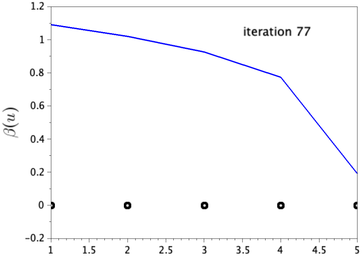
first last jacobi

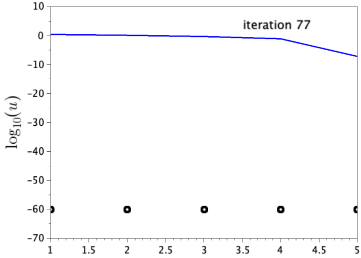
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





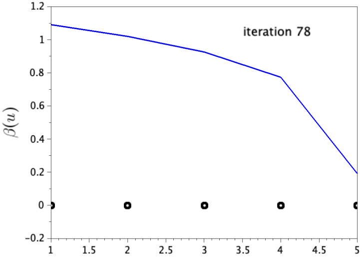
first last jacobi

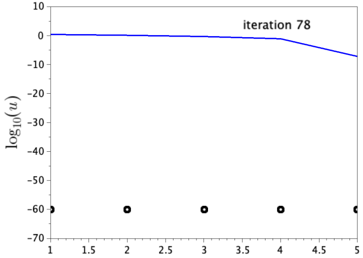
- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$





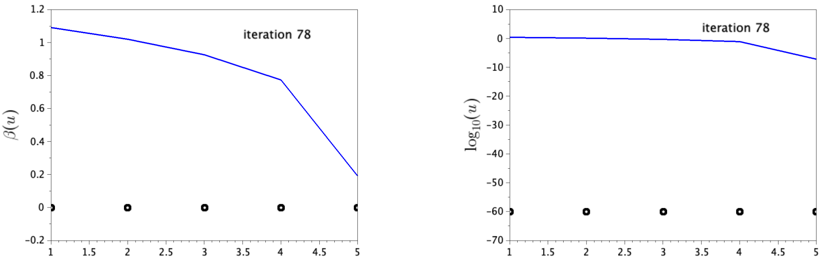
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Newton's iterates over a single time step

Newton's method applied to

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad \beta(\boldsymbol{u}) = \boldsymbol{u}^{1/10}$$



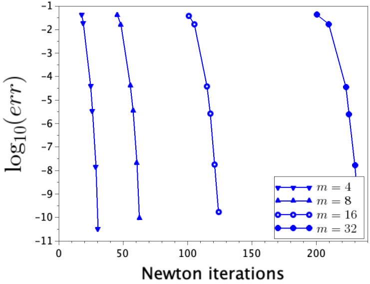
first last jacobi

- Slow convergence
- Iteration count grows with m and N

Alternative formulations

Original u-formulation:

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} - \boldsymbol{b} = 0$$



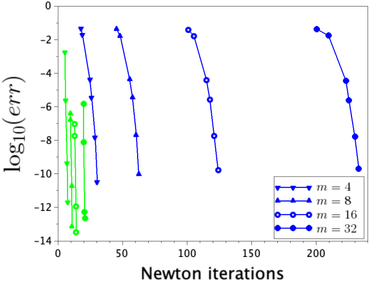
Alternative formulations

Original u-formulation:

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} - \boldsymbol{b} = 0$$

Alternative v-formulation:

$$\boldsymbol{v} + A\beta^{-1}(\boldsymbol{v}) - \boldsymbol{b} = 0$$



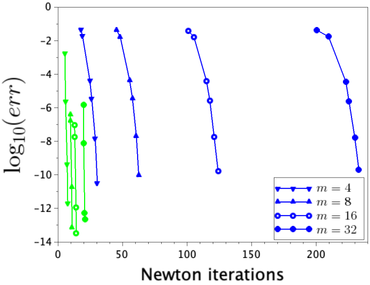
Alternative formulations

Original u-formulation:

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} - \boldsymbol{b} = 0$$

Alternative v-formulation:

$$\boldsymbol{v} + A\beta^{-1}(\boldsymbol{v}) - \boldsymbol{b} = 0$$



Blue: Original formulation is inefficient, manly because $\beta'(0) = +\infty$.

Green: Alternative formulation is more efficient, but concavity is lost: note that $(A)_{ii}(A)_{ij} \leq 0, i \neq j$

Outline

Introduction to Richards' equation

- From saturated to unsaturated flow

Monotone Newton's method

- Convergence proof $\neq$ performance

Preconditioned monotone Newton's method

- Convergence proof $+$ performance

Conclusion

Linear splitting methods

Model problem

$$A\boldsymbol{u} = \boldsymbol{b}$$

(Weak) regular splitting

$$A = M - N$$

such that

$$M^{-1} \geq 0, \quad N \geq 0 \quad \left(M^{-1}N,\, NM^{-1} \geq 0\right)$$

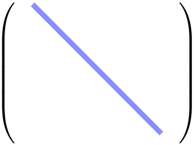
Stationary iterations

$$\boldsymbol{u}_{k+1} = M^{-1}\left(N\boldsymbol{u}_k + \boldsymbol{b}\right)$$

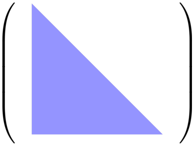
Preconditioned Krylov applied to

$$\left(I - M^{-1}N\right)\boldsymbol{u} = M^{-1}\boldsymbol{b}$$

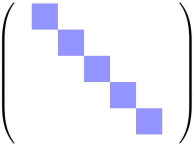
Jacobi



Gauss-Seidel



Block-Jacobi



Linear splitting methods

Model problem

$$A\boldsymbol{u} = \boldsymbol{b}$$

(Weak) regular splitting

$$A = M - N$$

such that

$$M^{-1} \geq 0, \quad N \geq 0 \quad \left(M^{-1}N,\, NM^{-1} \geq 0\right)$$

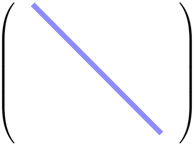
Stationary iterations

$$\boldsymbol{u}_{k+1} = M^{-1}\left(N\boldsymbol{u}_k + \boldsymbol{b}\right)$$

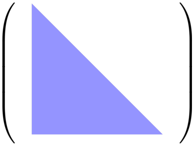
Preconditioned Krylov applied to

$$\left(I - M^{-1}N\right)\boldsymbol{u} = M^{-1}\boldsymbol{b}$$

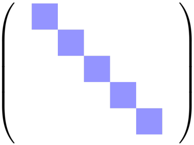
Jacobi



Gauss-Seidel



Block-Jacobi



If $A = M - N$ is a weak regular splitting, then

$$\rho\left(M^{-1}N\right) < 1 \quad \Leftrightarrow \quad A^{-1} \geq 0.$$

Nonlinear splitting methods

Model problem

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad A = P - Q$$

Splitting

$$\beta(\boldsymbol{u}) + P\boldsymbol{u} = Q\boldsymbol{u} + \boldsymbol{b}$$

K. B. On the monotone convergence of Jacobi-Newton method for mildly nonlinear systems. 2022

K. B. On global and monotone convergence of the preconditioned Newton's method for some mildly nonlinear systems. 2022

Nonlinear splitting methods

Model problem

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad A = P - Q$$

Splitting

$$\underbrace{\beta(\boldsymbol{u}) + P\boldsymbol{u}}_{M(u)} = \underbrace{Q\boldsymbol{u} + \boldsymbol{b}}_{N(u)}$$

K. B. On the monotone convergence of Jacobi-Newton method for mildly nonlinear systems. 2022

K. B. On global and monotone convergence of the preconditioned Newton's method for some mildly nonlinear systems. 2022

Nonlinear splitting methods

Model problem

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad A = P - Q$$

Splitting

$$\underbrace{\beta(\boldsymbol{u}) + P\boldsymbol{u}}_{M(u)} = \underbrace{Q\boldsymbol{u} + \boldsymbol{b}}_{N(u)}$$

Stationary iterations

$$\boldsymbol{u}_{k+1} = M^{-1}\left(N(\boldsymbol{u}_k)\right)$$

Newton's method applied to

$$\boldsymbol{u} - M^{-1}\left(N(\boldsymbol{u})\right) = 0$$

K. B. On the monotone convergence of Jacobi-Newton method for mildly nonlinear systems. 2022

K. B. On global and monotone convergence of the preconditioned Newton's method for some mildly nonlinear systems. 2022

Nonlinear splitting methods

Model problem

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad A = P - Q$$

Splitting

$$\underbrace{\beta(\boldsymbol{u}) + P\boldsymbol{u}}_{M(u)} = \underbrace{Q\boldsymbol{u} + \boldsymbol{b}}_{N(u)}$$

Stationary iterations

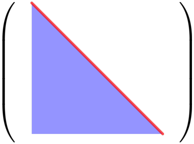
$$\boldsymbol{u}_{k+1} = M^{-1}\left(N(\boldsymbol{u}_k)\right)$$

Newton's method applied to

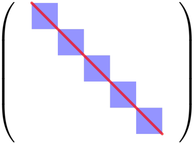
$$\boldsymbol{u} - M^{-1}\left(N(\boldsymbol{u})\right) = 0$$

Jacobi

Gauss-Seidel



Block-Jacobi



K. B. On the monotone convergence of Jacobi-Newton method for mildly nonlinear systems. 2022

K. B. On global and monotone convergence of the preconditioned Newton's method for some mildly nonlinear systems. 2022

Nonlinear splitting methods

Model problem

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}, \qquad A = P - Q$$

Splitting

$$\underbrace{\beta(\boldsymbol{u}) + P\boldsymbol{u}}_{M(u)} = \underbrace{Q\boldsymbol{u} + \boldsymbol{b}}_{N(u)}$$

Stationary iterations

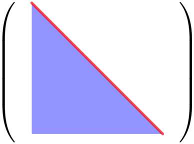
$$\boldsymbol{u}_{k+1} = M^{-1}\left(N(\boldsymbol{u}_k)\right)$$

Newton's method applied to

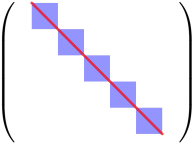
$$\boldsymbol{u} - M^{-1}\left(N(\boldsymbol{u})\right) = 0$$

Jacobi

Gauss-Seidel



Block-Jacobi



If $\beta'(\boldsymbol{u}) + A = M'(\boldsymbol{u}) - N'(u)$ is a weak regular splitting, then

$$\mathcal{F}(\boldsymbol{u}) = \boldsymbol{u} - M^{-1}(N(\boldsymbol{u}))$$

- satisfies the MNT;
- moreover, converges is global.

K. B. On the monotone convergence of Jacobi-Newton method for mildly nonlinear systems. 2022

K. B. On global and monotone convergence of the preconditioned Newton's method for some mildly nonlinear systems. 2022

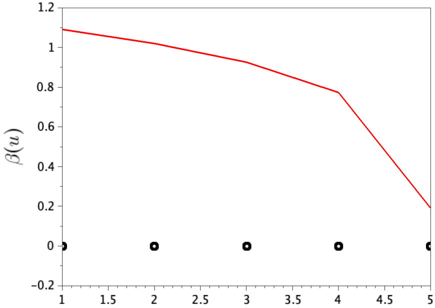
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\mathrm{unk} = 5$

baseline

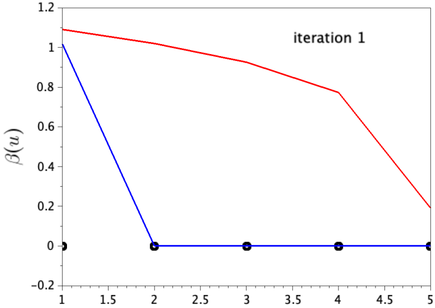
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\mathrm{unk} = 5$

baseline

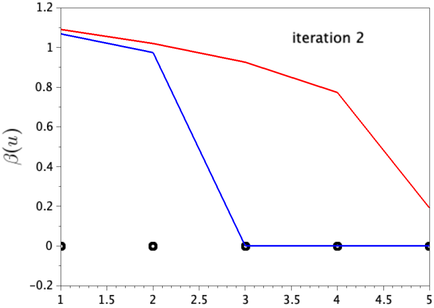
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\mathrm{unk} = 5$

baseline

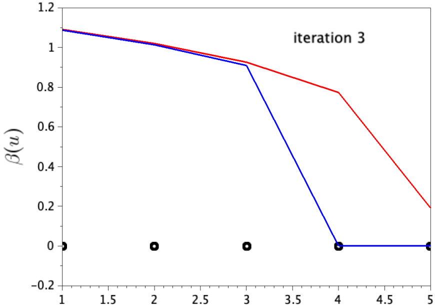
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 5$

baseline

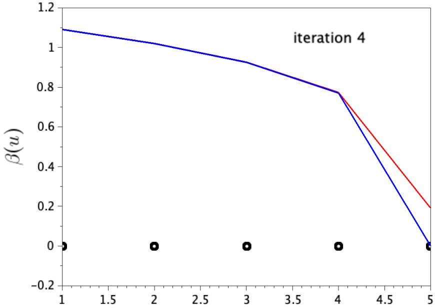
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#$unk $= 5$

baseline

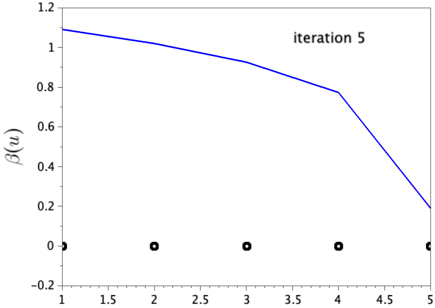
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 5$

baseline

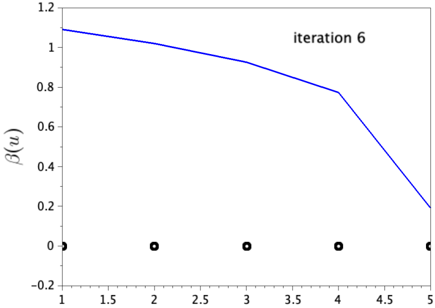
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 5$

baseline

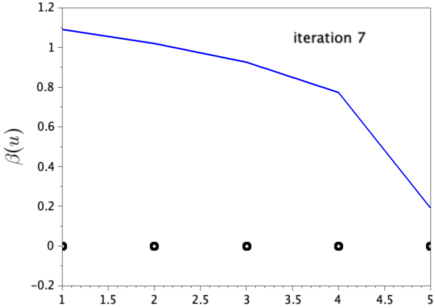
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

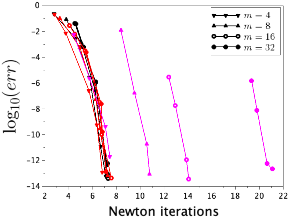
$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$

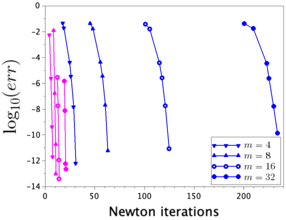


$\#\text{unk} = 5$

baseline

Data robustness





Original system

$$\beta(\boldsymbol{u}) + A\boldsymbol{u} = \boldsymbol{b}$$

Change of variable

$$\boldsymbol{v} + A\beta^{-1}(\boldsymbol{v}) = \boldsymbol{b}$$

Jacobi-Newton method

$$\boldsymbol{u} = M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u}))$$

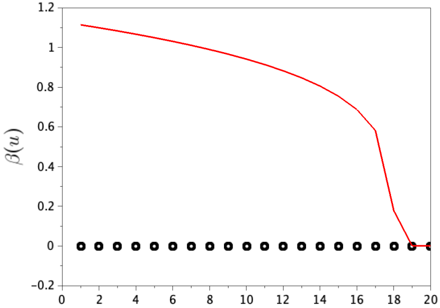
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 20$

baseline

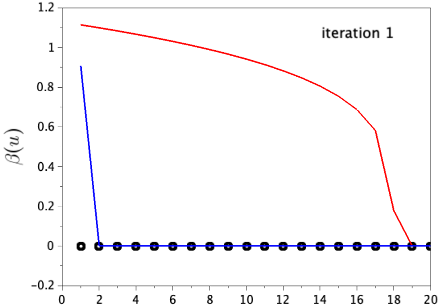
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 20$

baseline

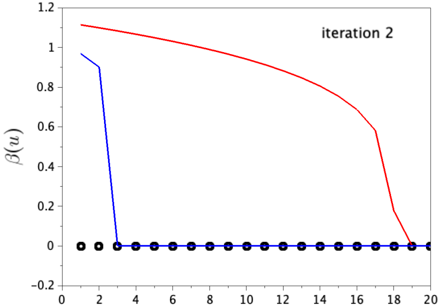
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#$unk $= 20$

baseline

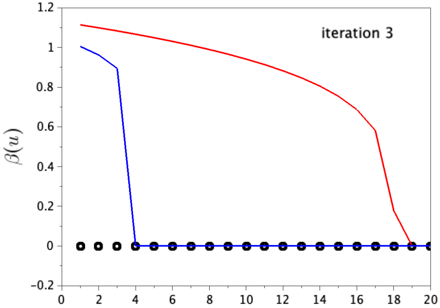
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#$unk $= 20$

baseline

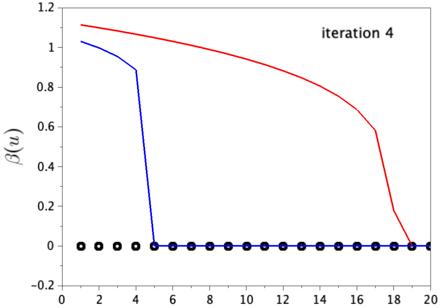
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



#unk = 20

baseline

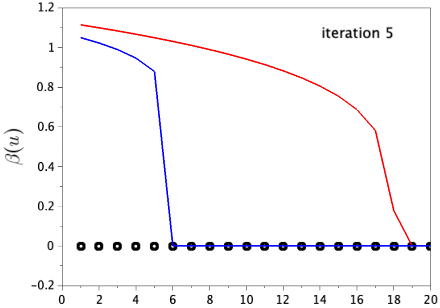
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 20$

baseline

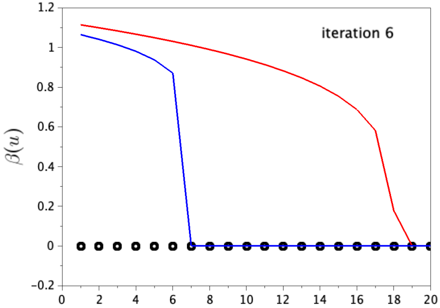
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#$unk $= 20$

baseline

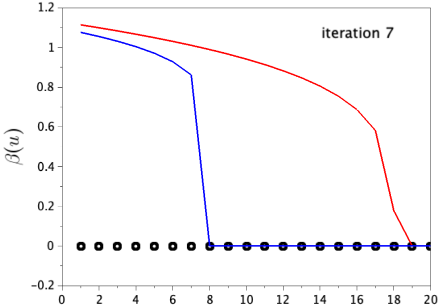
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#$unk $= 20$

baseline

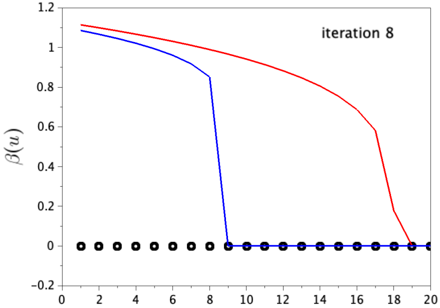
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 20$

baseline

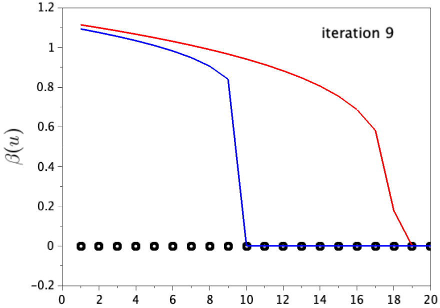
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 20$

baseline

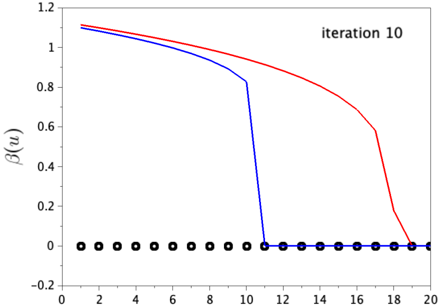
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 20$

baseline

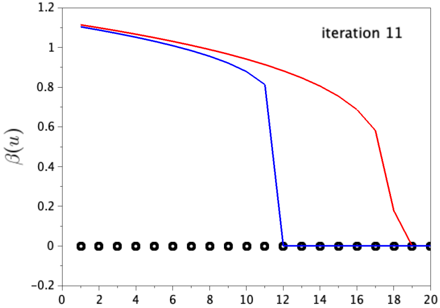
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 20$

baseline

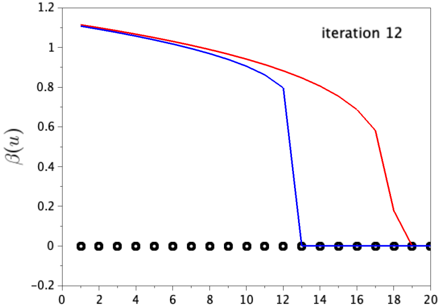
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 20$

baseline

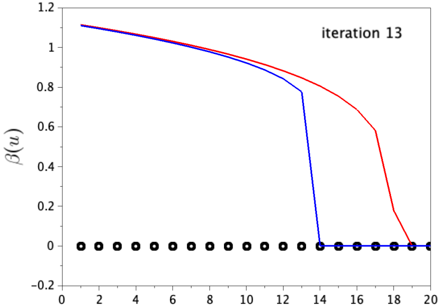
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 20$

baseline

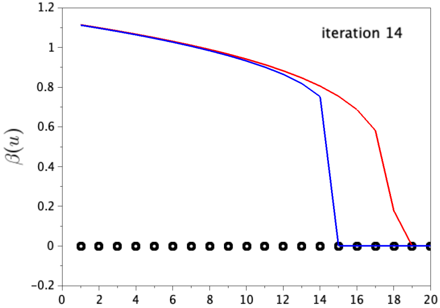
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#$unk $= 20$

baseline

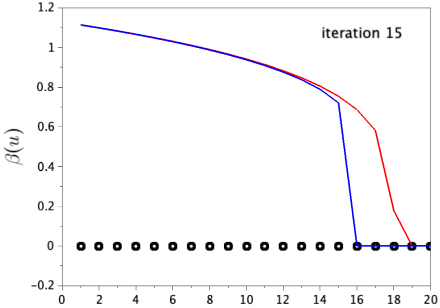
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



#unk = 20

baseline

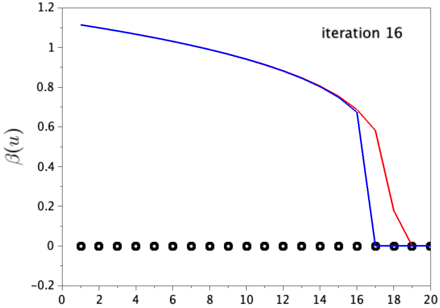
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\mathsf{unk} = 20$

baseline

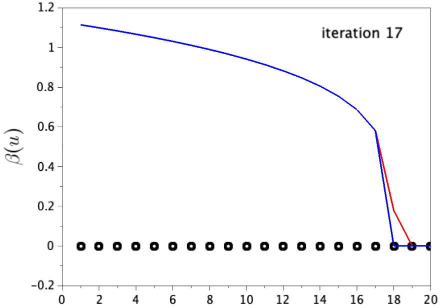
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



$\#\text{unk} = 20$

baseline

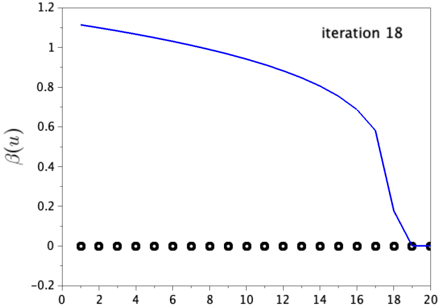
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



#unk = 20

baseline

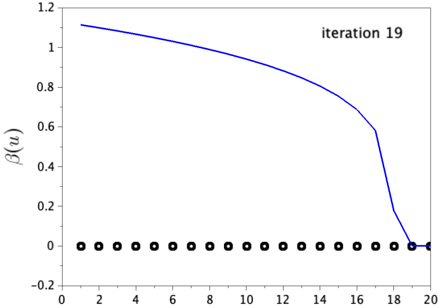
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



#unk = 20

baseline

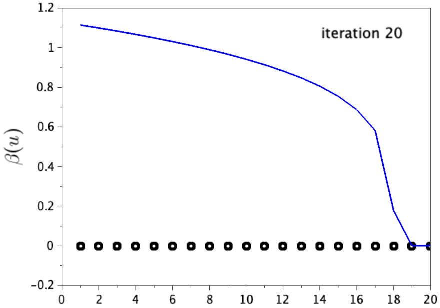
Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{diag}(A)\boldsymbol{u}$$



#unk = 20

baseline

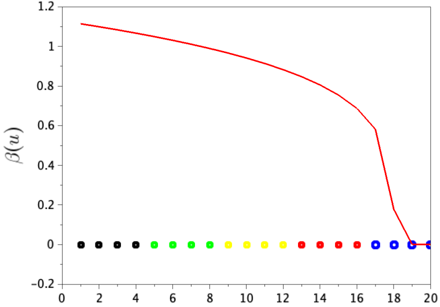
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



$\#$blocks $= 5$, $\#$unk $= 20$

baseline jacobi

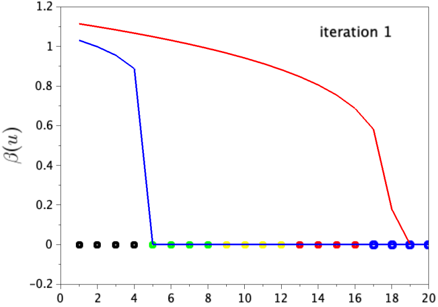
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 20

baseline jacobi

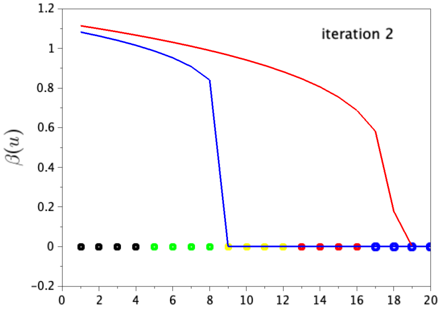
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 20

baseline jacobi

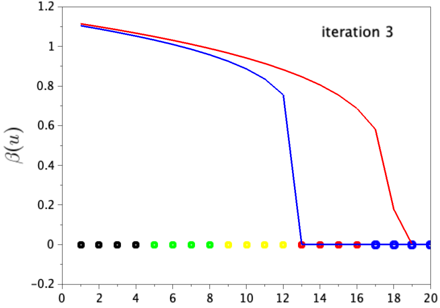
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 20

baseline jacobi

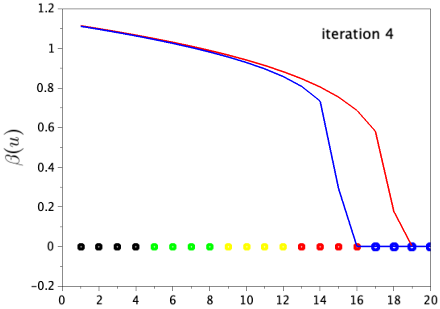
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



$\#$blocks $= 5$, $\#$unk $= 20$

baseline jacobi

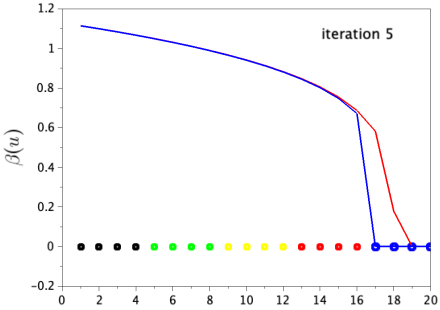
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 20

baseline jacobi

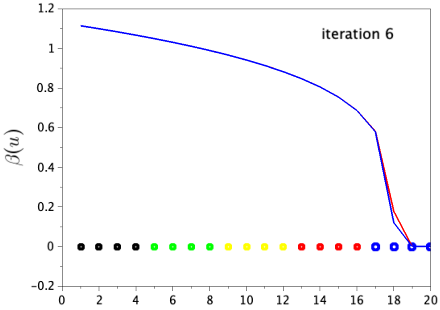
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 20

baseline jacobi

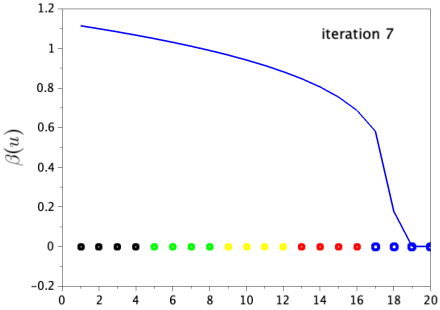
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag}(\mathrm{A})\boldsymbol{u}$$



#blocks = 5, #unk = 20

baseline jacobi

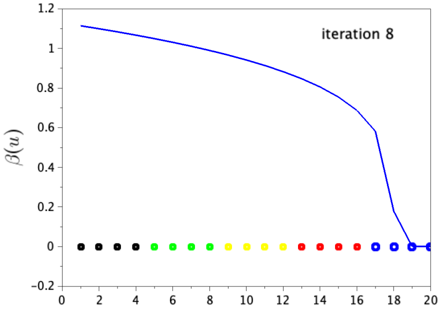
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 20

baseline jacobi

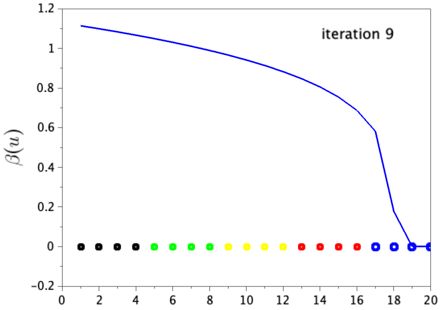
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks $=5$, #unk $=20$

baseline jacobi

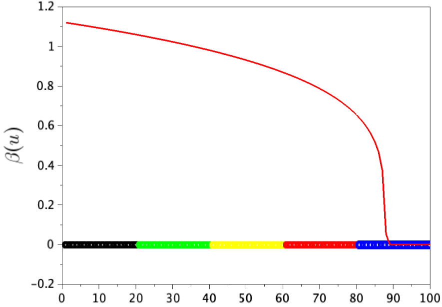
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

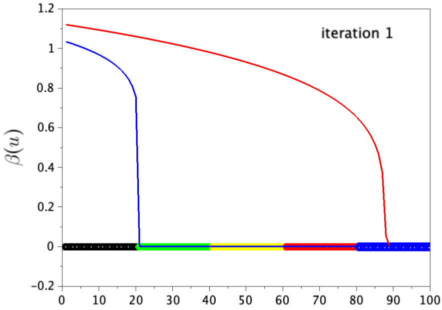
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



$\#$blocks $= 5$, $\#$unk $= 100$

baseline jacobi

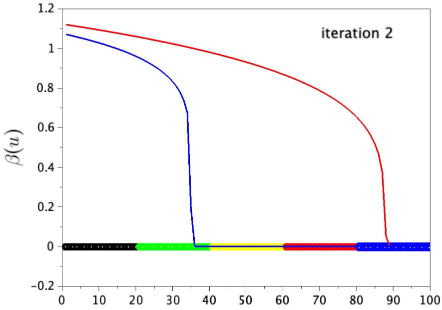
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

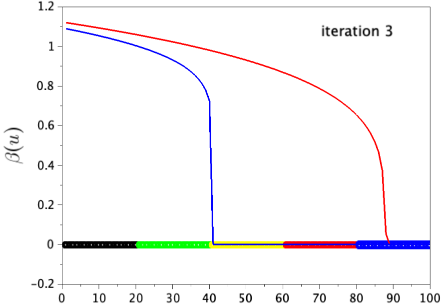
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

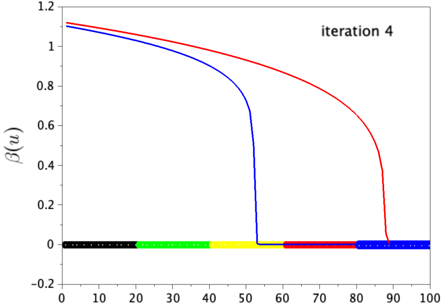
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

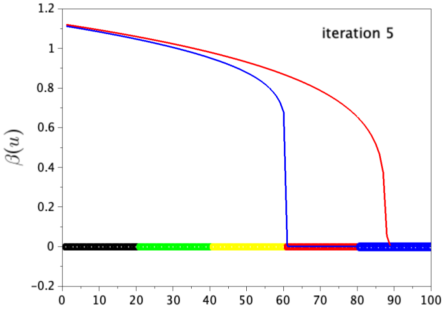
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



$\#$blocks $= 5$, $\#$unk $= 100$

baseline jacobi

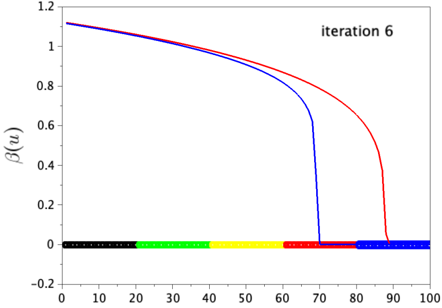
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

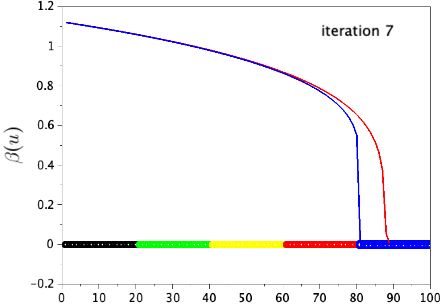
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

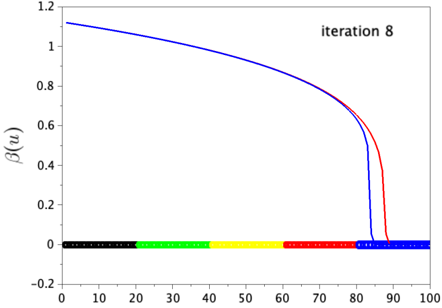
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag}(\mathrm{A})\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

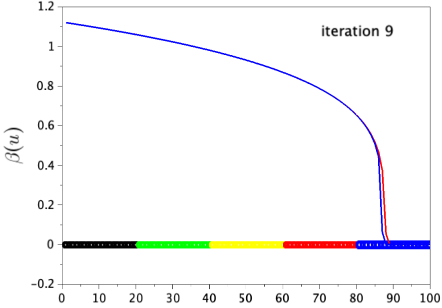
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

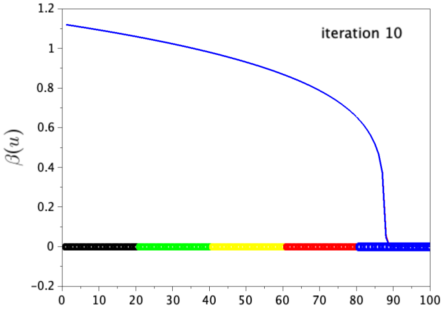
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

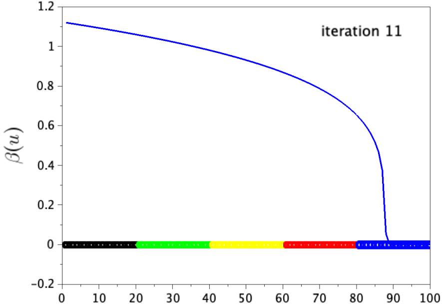
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

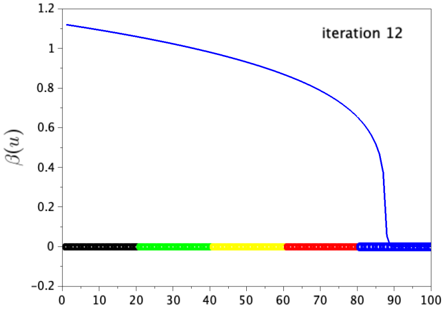
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

baseline jacobi

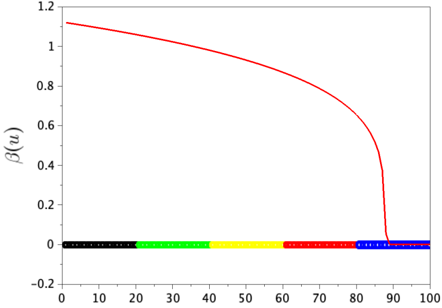
Block-Jacobi-Newton's iterates

Preconditioned Newton's method

$$\boldsymbol{u} - M^{-1}(M(\boldsymbol{u}) - F(\boldsymbol{u})) = 0$$

with

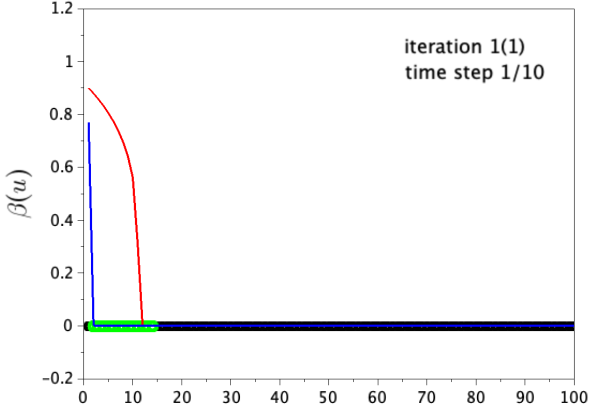
$$M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + \mathrm{BlockDiag(A)}\boldsymbol{u}$$



#blocks = 5, #unk = 100

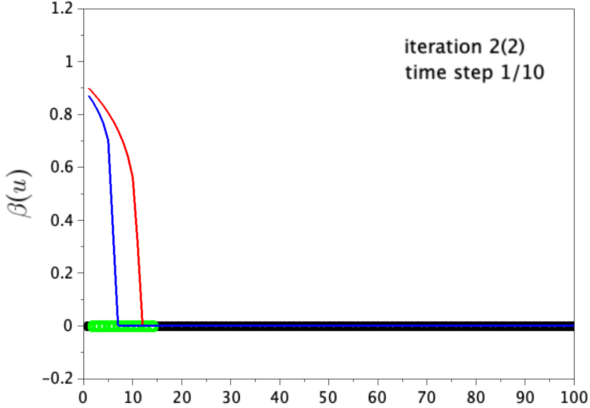
baseline jacobi

Adaptive Block-Jacobi-Newton method



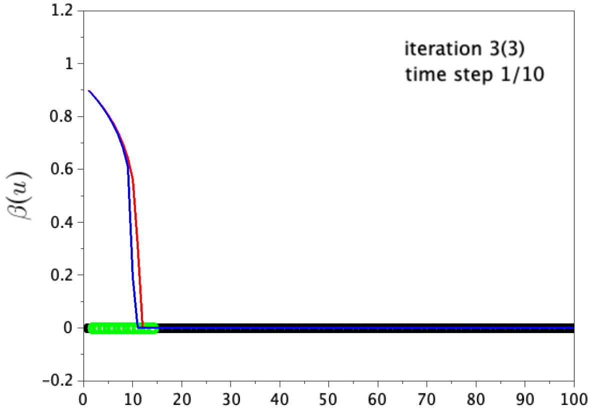
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



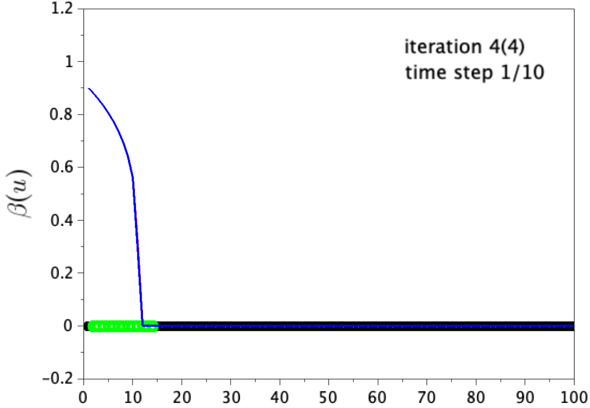
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



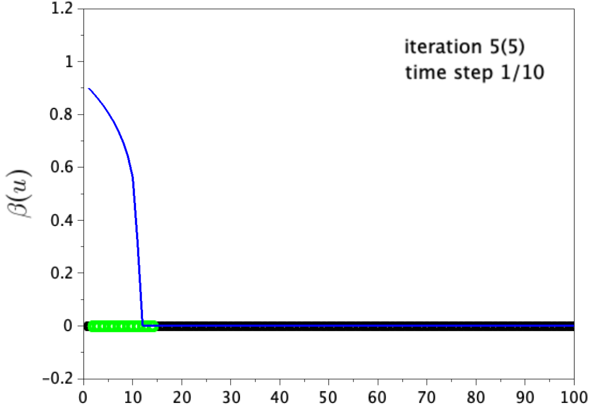
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



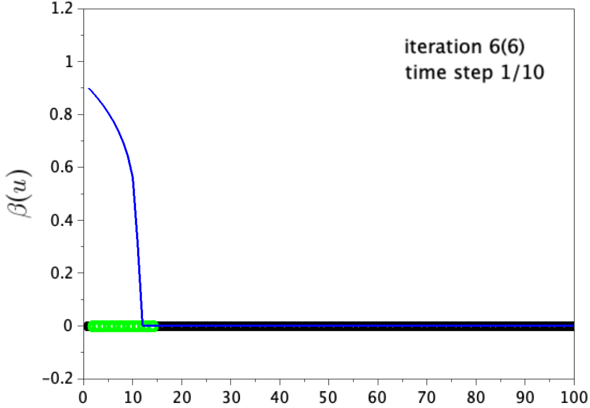
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



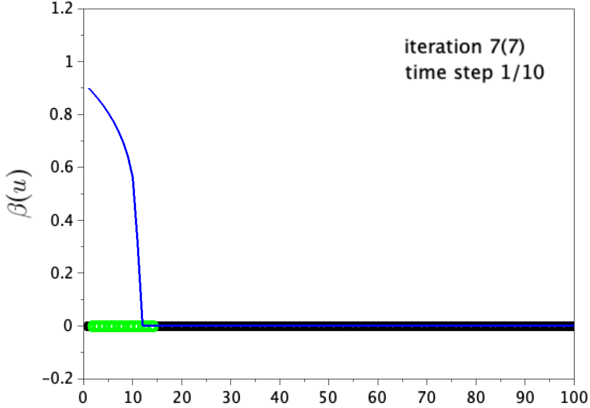
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



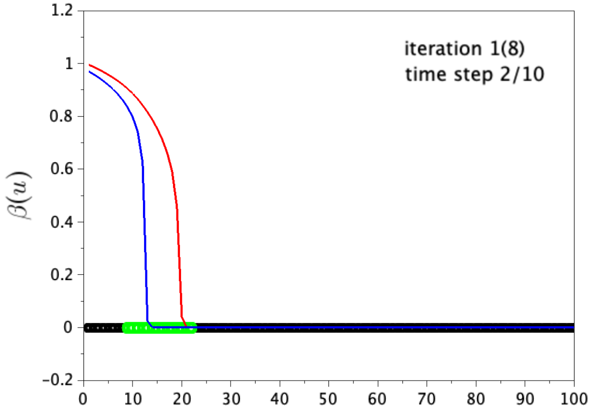
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



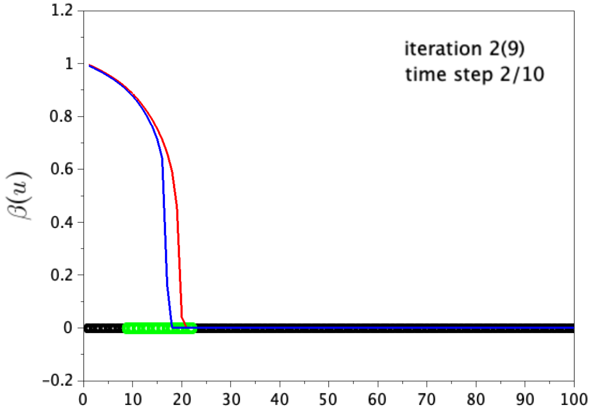
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



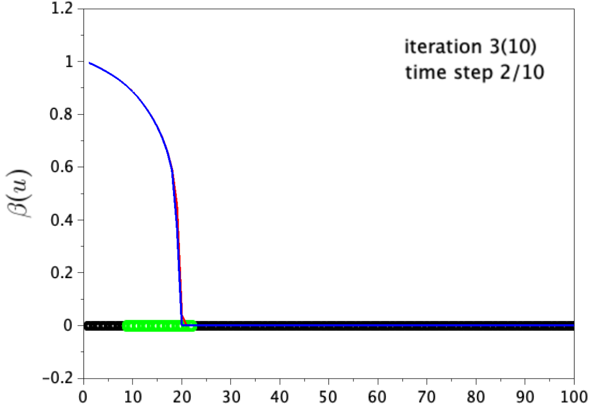
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



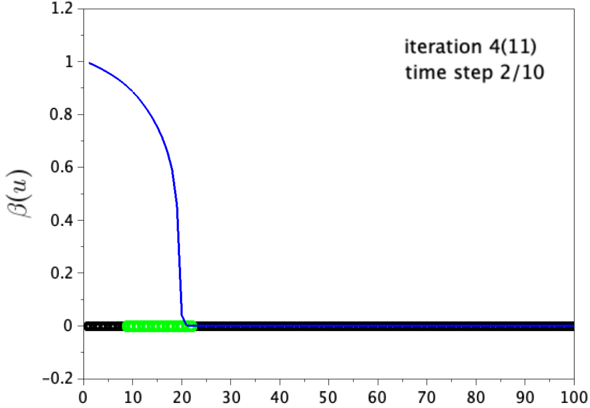
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



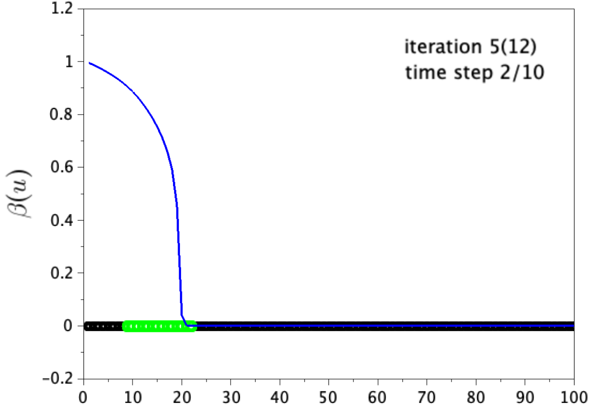
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



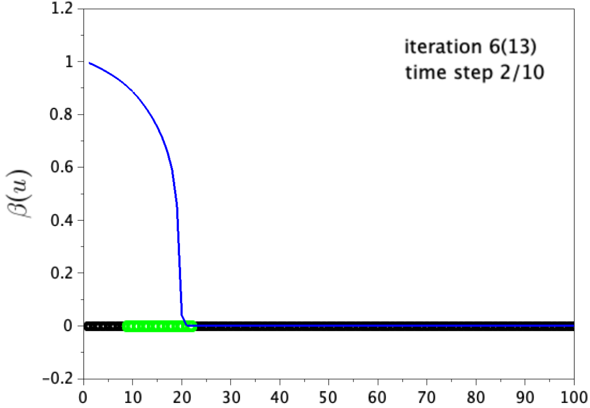
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



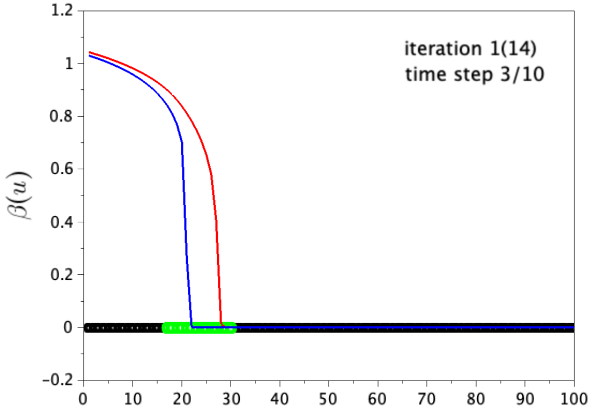
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



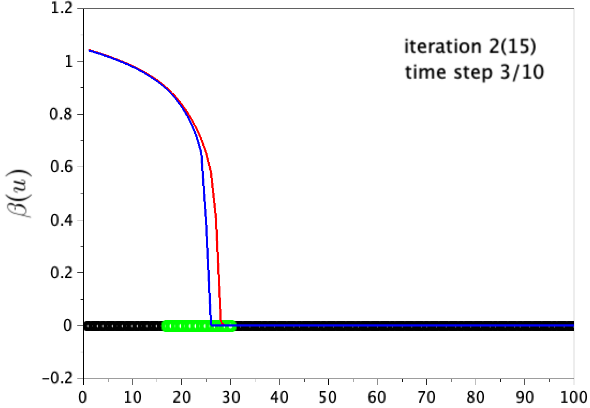
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



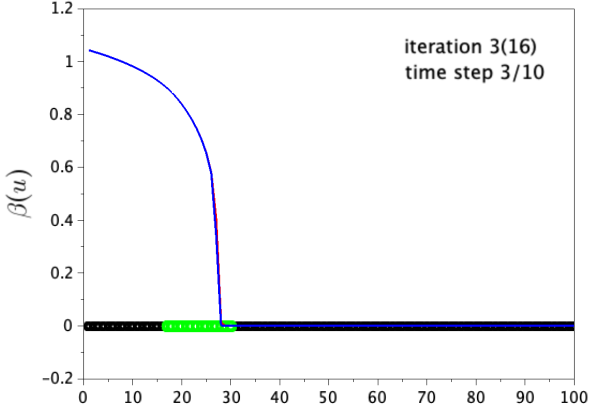
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



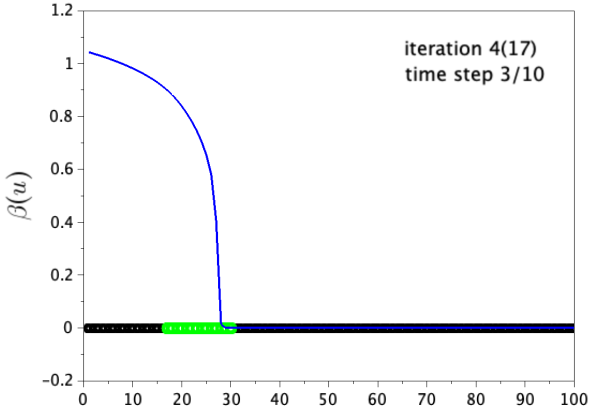
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



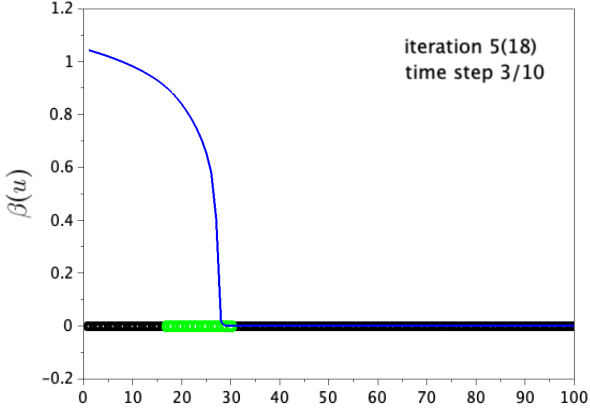
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



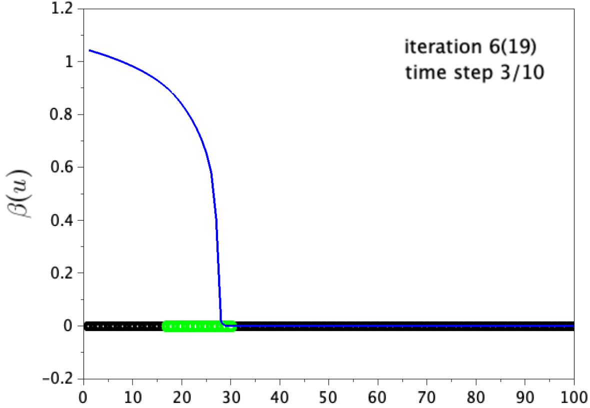
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



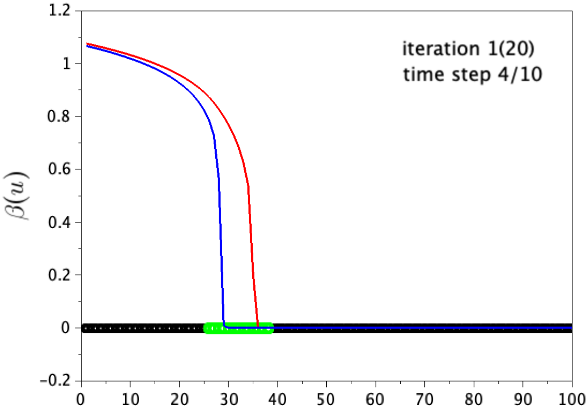
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



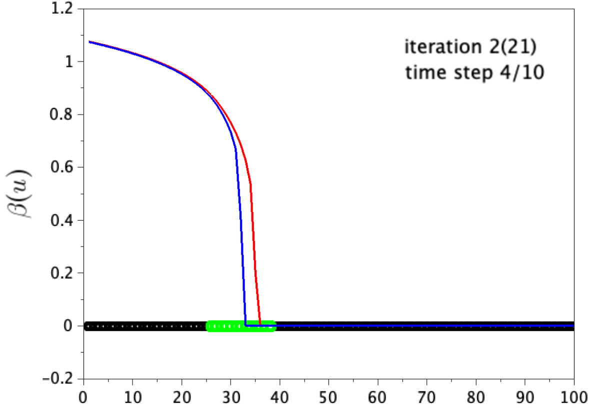
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



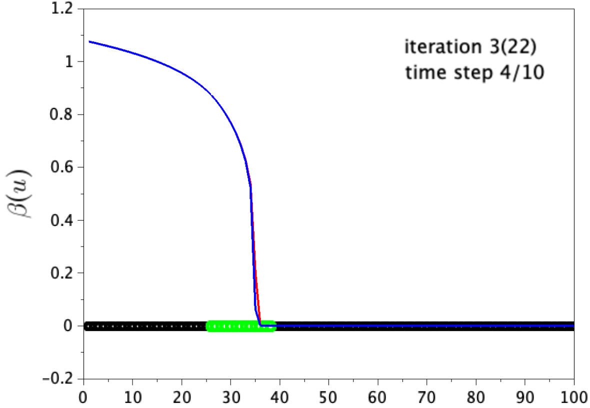
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



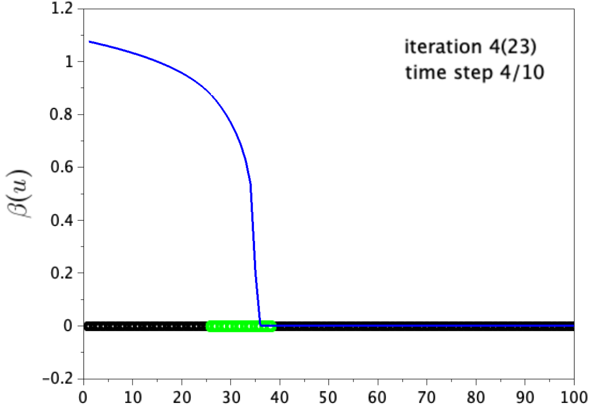
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



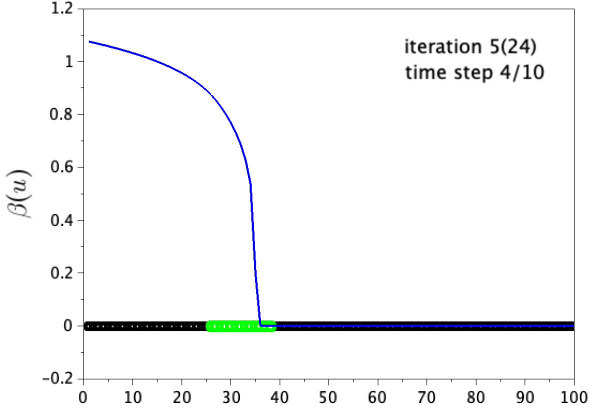
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



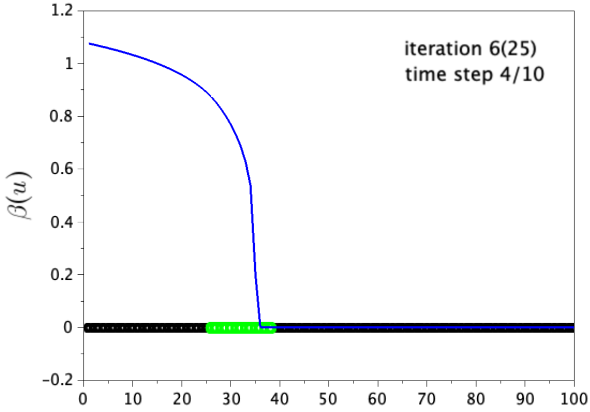
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



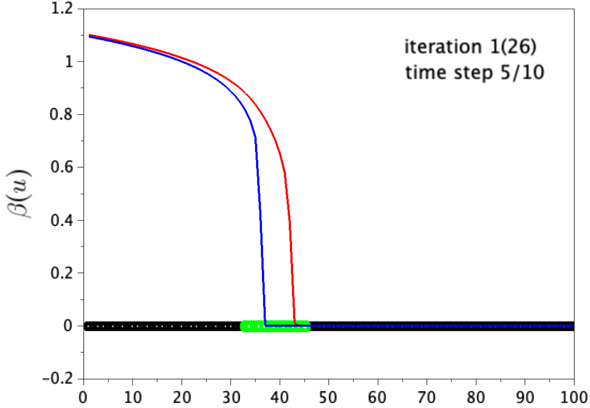
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



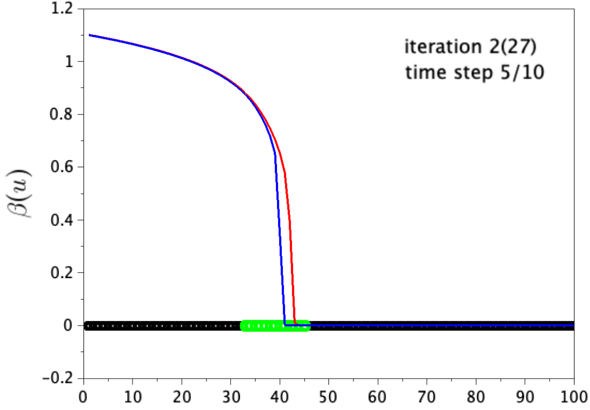
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



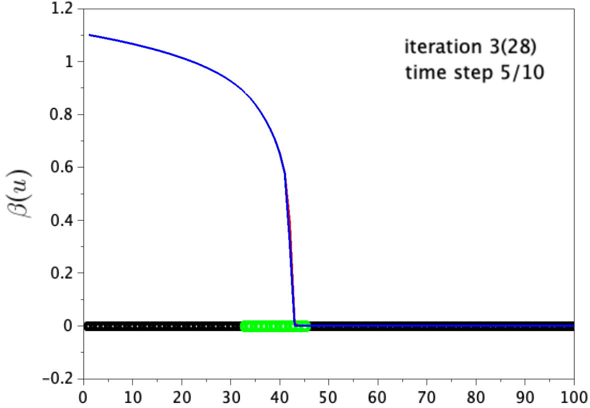
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



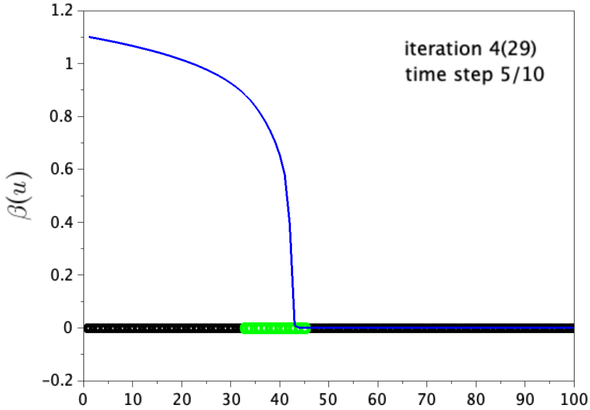
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



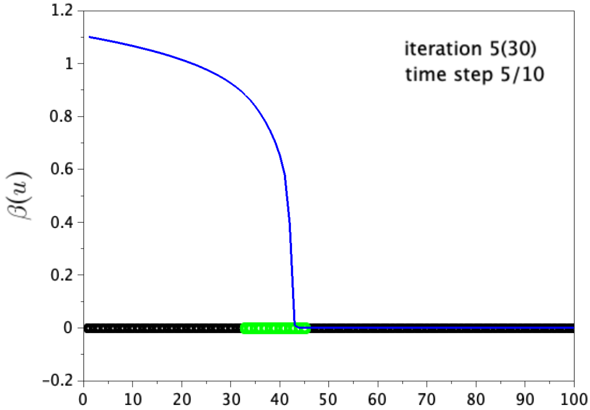
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



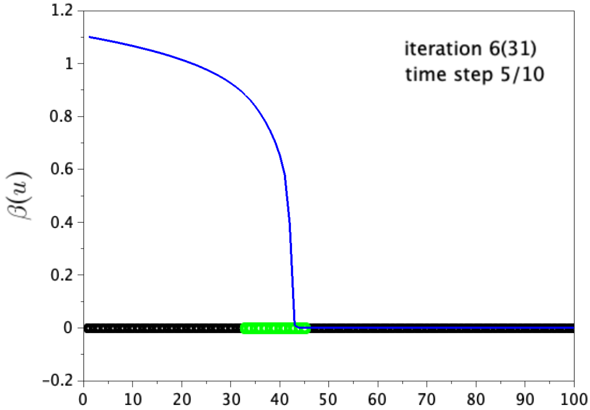
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



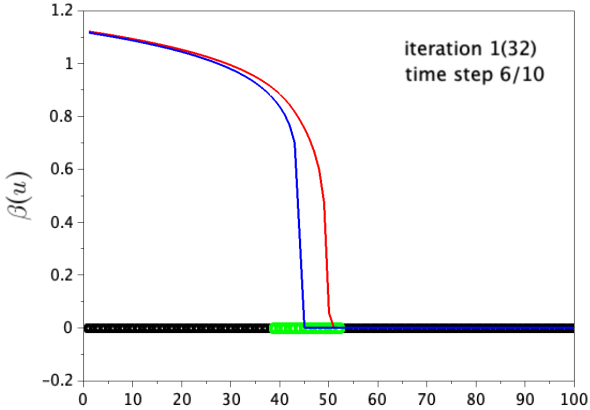
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



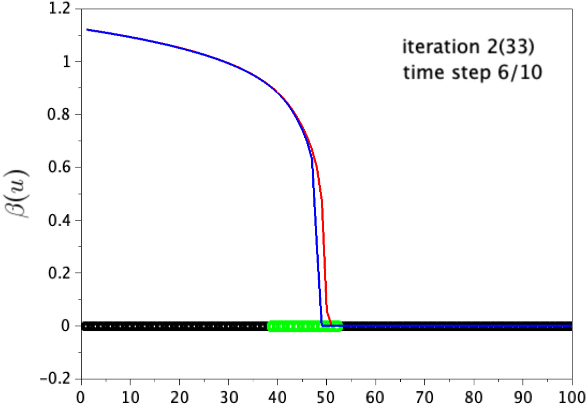
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



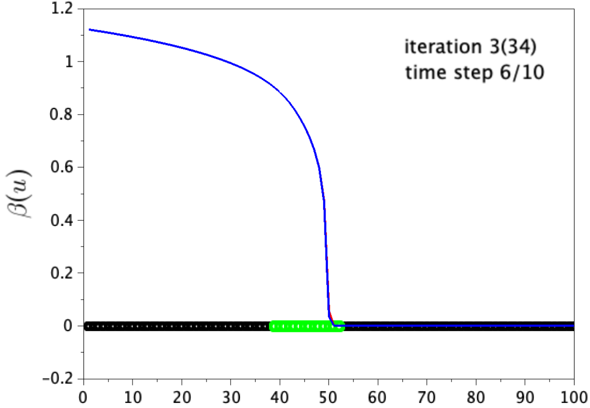
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



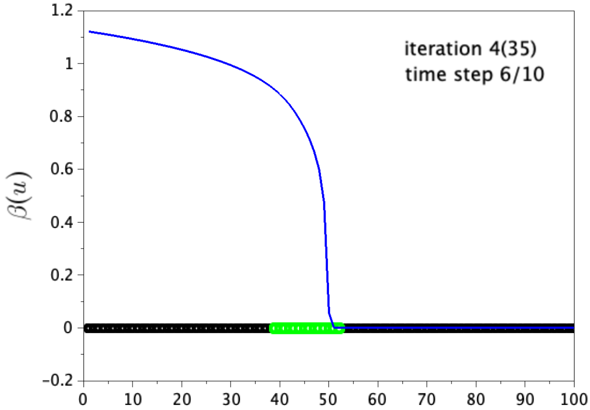
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



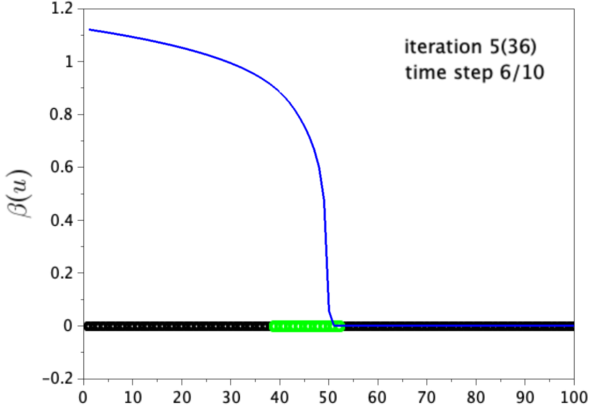
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



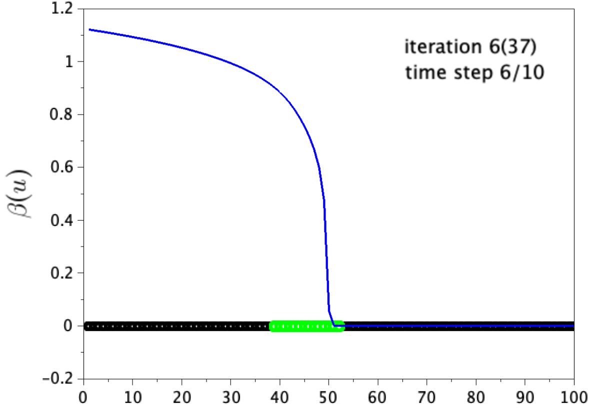
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



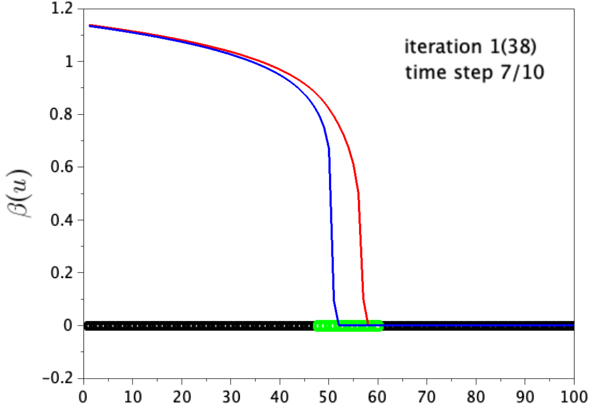
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



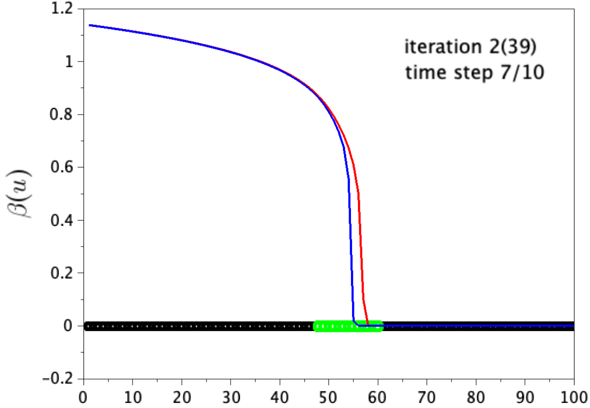
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



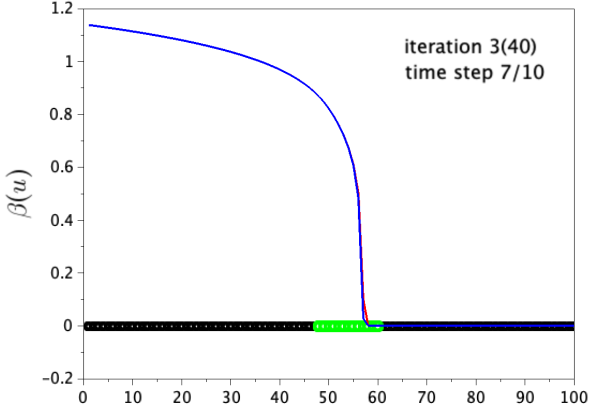
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



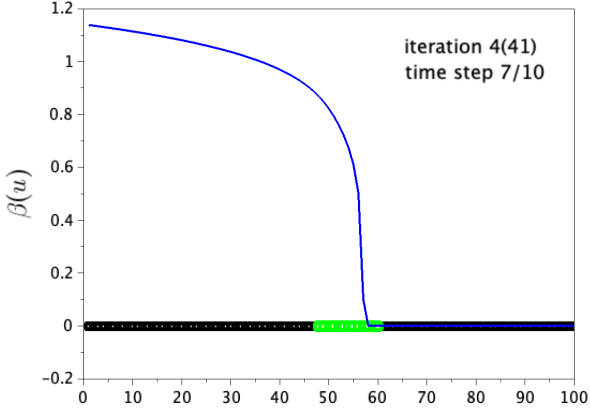
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



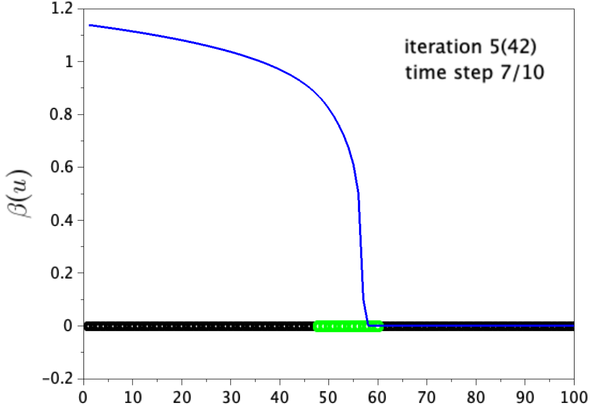
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



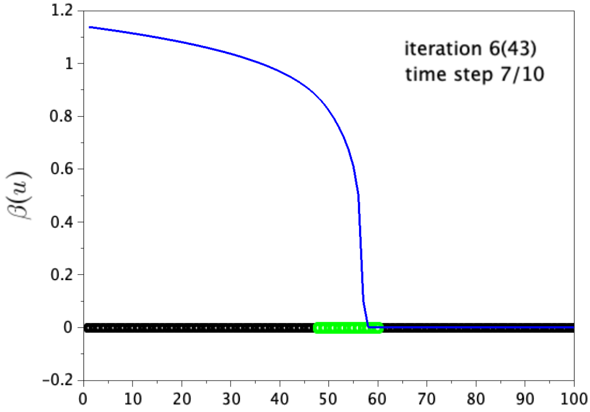
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



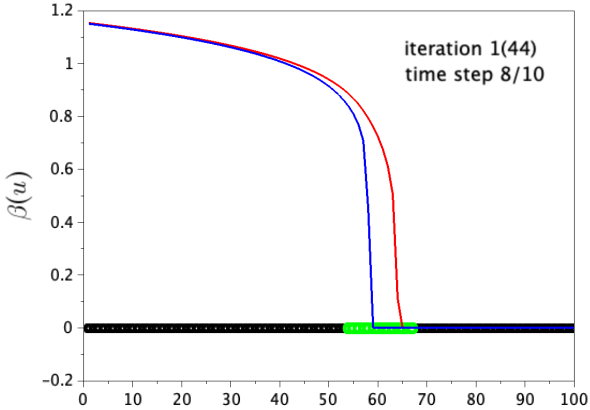
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



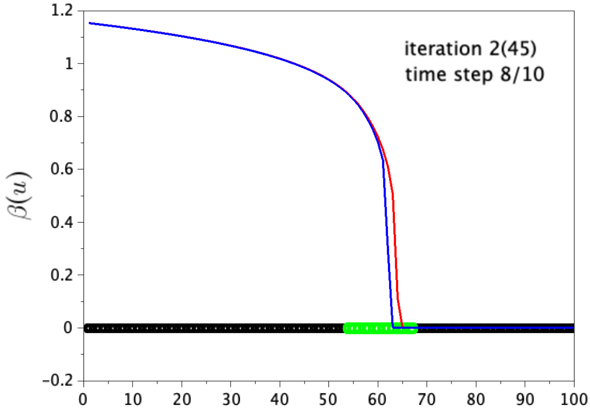
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



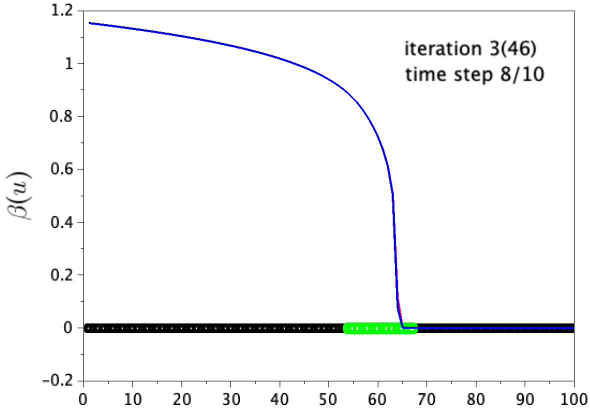
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



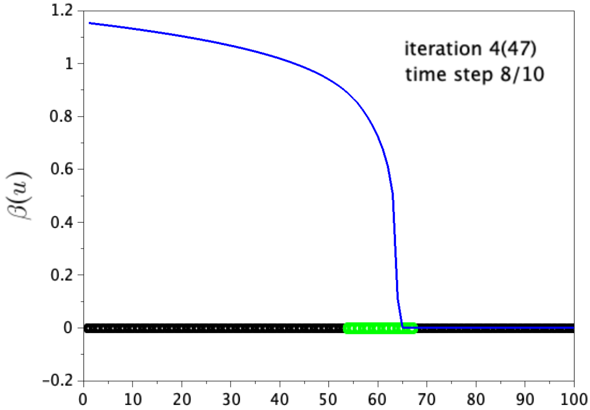
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



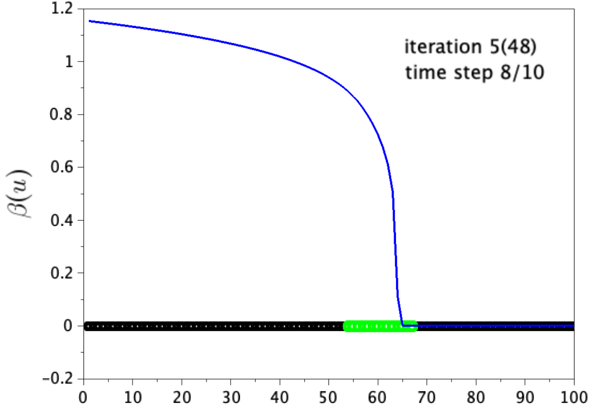
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



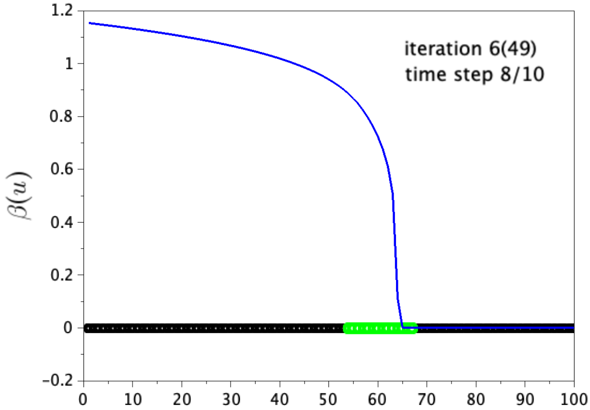
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



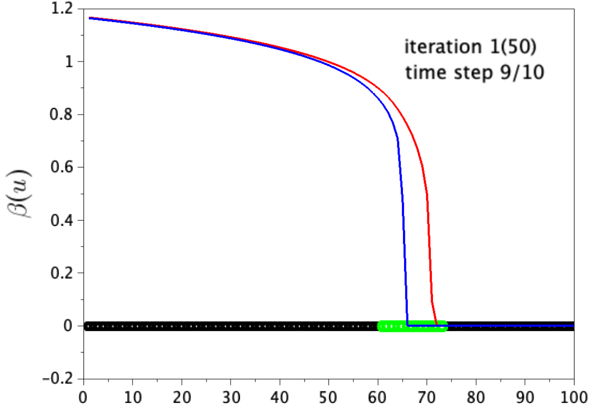
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



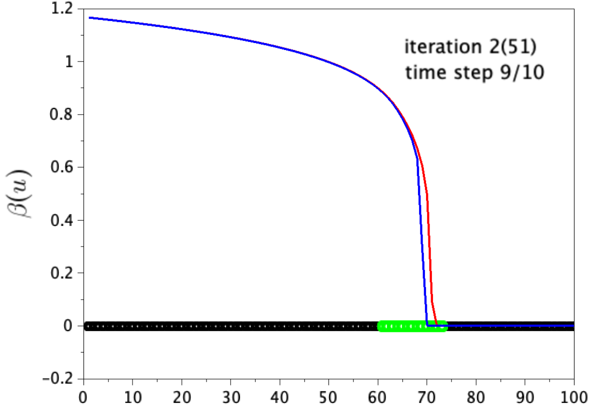
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



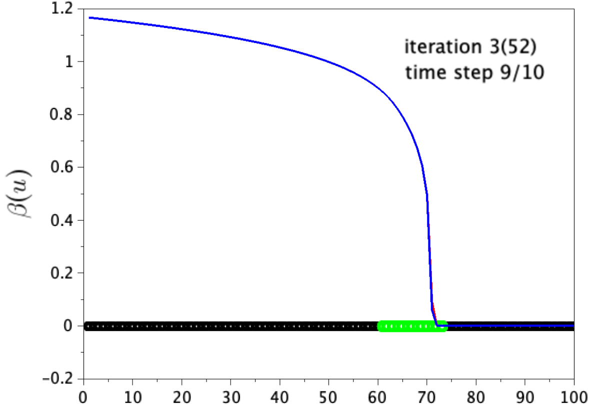
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



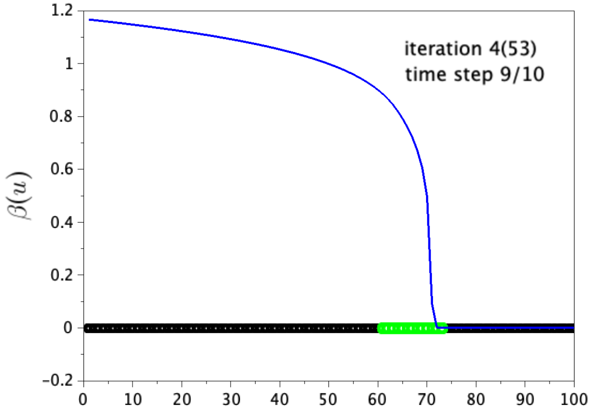
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



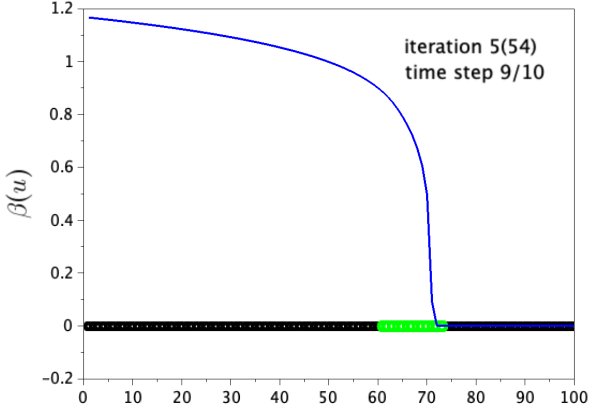
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



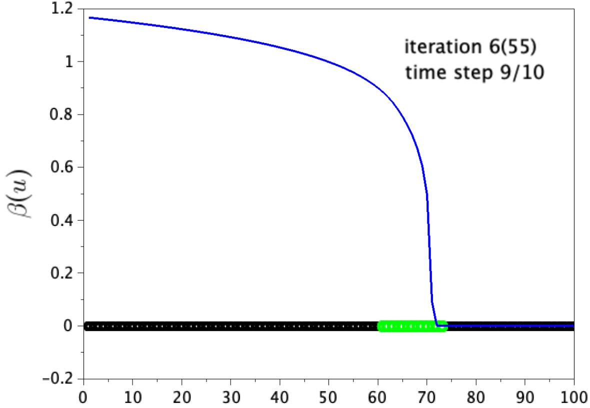
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



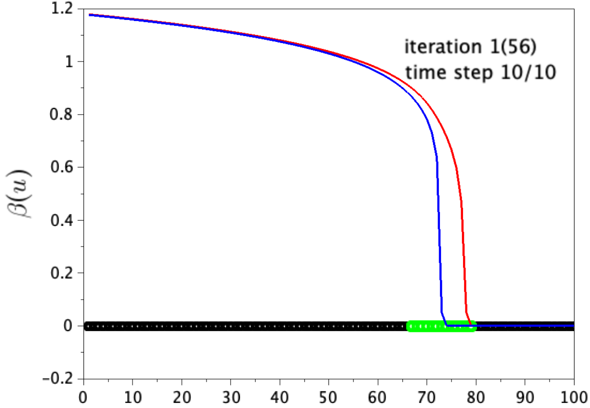
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



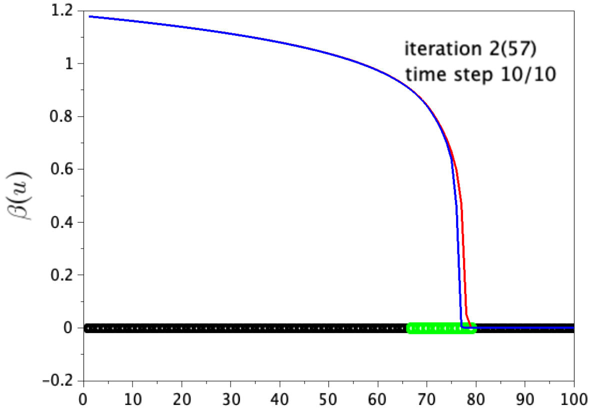
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



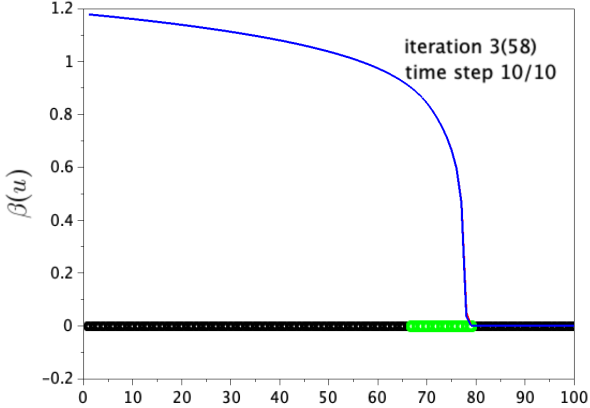
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



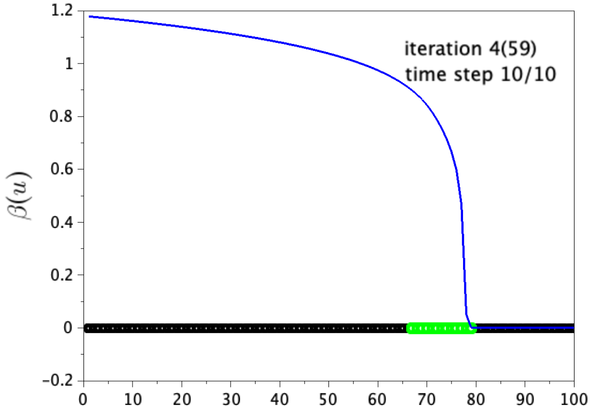
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



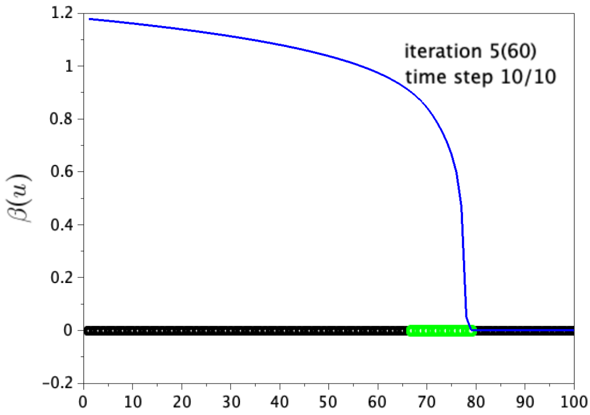
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



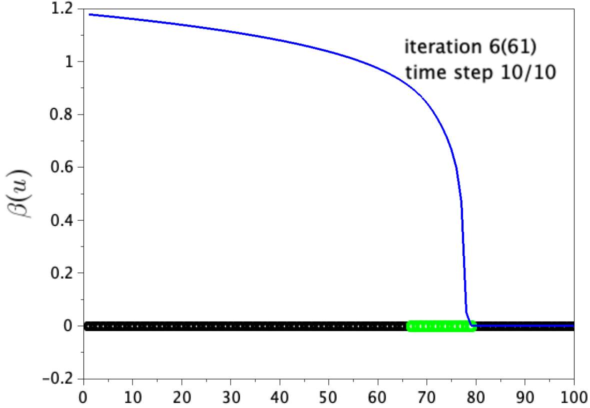
# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Adaptive Block-Jacobi-Newton method



# unk.	100	1000
Jacobi	11.4	80
Adaptive	6.1	6.7
Block-Jacobi (10)	6	7.1

Implementation and extensions

Notations: $F(\boldsymbol{u}) = \beta(\boldsymbol{u}) + A\boldsymbol{u} - b, \quad M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + P\boldsymbol{u}, \quad A = P - Q$

Implementation: Newton's method applied to the preconditioned system

$$\boldsymbol{u} = M^{-1}(Q\boldsymbol{u} + b)$$

is equivalent to the following two-step scheme[1]

$$\begin{array}{rcl} \widetilde{\boldsymbol{u}}_n & = & M^{-1}(Q\boldsymbol{u}_n + \boldsymbol{b}) \\ \widetilde{\boldsymbol{u}}_{n+1} & = & \widetilde{\boldsymbol{u}}_n - F'(\widetilde{\boldsymbol{u}}_n)^{-1}F(\widetilde{\boldsymbol{u}}_n) \end{array}$$

[1] Cai, X. C., & Li, X. Inexact Newton methods with restricted additive Schwarz based nonlinear elimination for problems with high local nonlinearity, 2011

Implementation and extensions

Notations: $F(\boldsymbol{u}) = \beta(\boldsymbol{u}) + A\boldsymbol{u} - b, \quad M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + P\boldsymbol{u}, \quad A = P - Q$

Implementation: Newton's method applied to the preconditioned system

$$\boldsymbol{u} = M^{-1}(Q\boldsymbol{u} + b)$$

is equivalent to the following two-step scheme[1]

$$\begin{array}{rcl} \widetilde{\boldsymbol{u}}_n & = & M^{-1}(Q\boldsymbol{u}_n + \boldsymbol{b}) \\ \widetilde{\boldsymbol{u}}_{n+1} & = & \widetilde{\boldsymbol{u}}_n - F'(\widetilde{\boldsymbol{u}}_n)^{-1}F(\widetilde{\boldsymbol{u}}_n) \end{array}$$

Extensions: Multi-splitting methods

- Let $A = P_i - Q_i$ be a sequence of splittings and

$$M_i(\boldsymbol{u}) = \beta(\boldsymbol{u}) + P_i\boldsymbol{u}$$

- Let $(E_i)_i$ be such that $E_i \geq 0$ and $\sum_i E_i = I$.

[1] Cai, X. C., & Li, X. Inexact Newton methods with restricted additive Schwarz based nonlinear elimination for problems with high local nonlinearity, 2011

Implementation and extensions

Notations: $F(\boldsymbol{u}) = \beta(\boldsymbol{u}) + A\boldsymbol{u} - b, \quad M(\boldsymbol{u}) = \beta(\boldsymbol{u}) + P\boldsymbol{u}, \quad A = P - Q$

Implementation: Newton's method applied to the preconditioned system

$$\boldsymbol{u} = M^{-1}(Q\boldsymbol{u} + b)$$

is equivalent to the following two-step scheme[1]

$$\begin{array}{rcl} \widetilde{\boldsymbol{u}}_n & = & M^{-1}(Q\boldsymbol{u}_n + \boldsymbol{b}) \\ \widetilde{\boldsymbol{u}}_{n+1} & = & \widetilde{\boldsymbol{u}}_n - F'(\widetilde{\boldsymbol{u}}_n)^{-1}F(\widetilde{\boldsymbol{u}}_n) \end{array}$$

Extensions: Multi-splitting methods

- Let $A = P_i - Q_i$ be a sequence of splittings and

$$M_i(\boldsymbol{u}) = \beta(\boldsymbol{u}) + P_i\boldsymbol{u}$$

- Let $(E_i)_i$ be such that $E_i \geq 0$ and $\sum_i E_i = I$.

The system

$$\boldsymbol{u} = \sum_i E_i M_i^{-1}(Q_i\boldsymbol{u} + b)$$

satisfies MNT if $F'(\boldsymbol{u}) = M_i'(\boldsymbol{u}) - Q_i$ is weak regular.

[1] Cai, X. C., & Li, X. Inexact Newton methods with restricted additive Schwarz based nonlinear elimination for problems with high local nonlinearity, 2011

Conclusions

Theory: Global convergence analysis

- Mildly nonlinear, monotone systems
- Splitting: (Block-)Jacobi, Gauss-Seidel
- Multi-splitting: Restricted Additive Schwarz (RASPEN)

Practice: promising results for degenerate evolutionary PDEs

- Jacobi prec. is robust w.r.t. the "physical data"
  by absorbing the stiffness of the nonlinear closure laws
- Block-Jacobi (and RAS) is robust w.r.t. discretization parameters
  and allows to focus the computational efforts

Conclusions

Theory: Global convergence analysis

- Mildly nonlinear, monotone systems
- Splitting: (Block-)Jacobi, Gauss-Seidel
- Multi-splitting: Restricted Additive Schwarz (RASPEN)

Practice: promising results for degenerate evolutionary PDEs

- Jacobi prec. is robust w.r.t. the "physical data"
  by absorbing the stiffness of the nonlinear closure laws
- Block-Jacobi (and RAS) is robust w.r.t. discretization parameters
  and allows to focus the computational efforts

Limitations and extensions of the analysis:

- Concavity assumption can be removed
- M-matrix assumption is crustal for analysis, but might be relaxed in practice

Conclusions

Theory: Global convergence analysis

- Mildly nonlinear, monotone systems
- Splitting: (Block-)Jacobi, Gauss-Seidel
- Multi-splitting: Restricted Additive Schwarz (RASPEN)

Practice: promising results for degenerate evolutionary PDEs

- Jacobi prec. is robust w.r.t. the “physical data”
  by absorbing the stiffness of the nonlinear closure laws
- Block-Jacobi (and RAS) is robust w.r.t. discretization parameters
  and allows to focus the computational efforts

Limitations and extensions of the analysis:

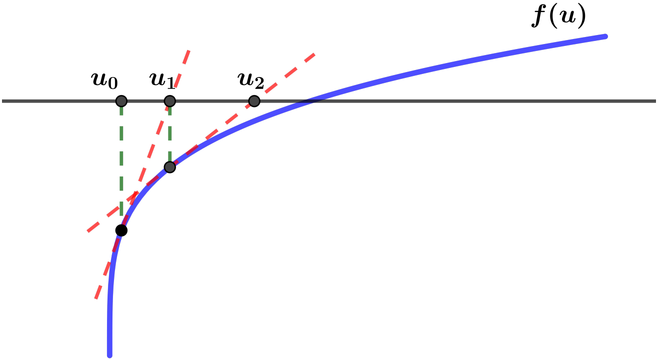
- Concavity assumption can be removed
- M-matrix assumption is crustal for analysis, but might be relaxed in practice
- Two-level preconditioning, probably depending on the coarse space...

Appendix: Newton's method for a scalar concave problem

Newton's method for

$$f(p) = 0, \qquad p \in \mathbb{R}$$

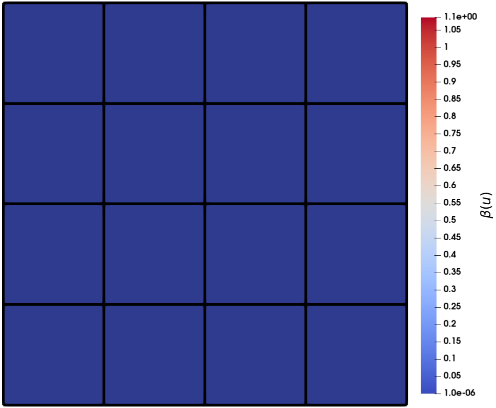
- f concave and increasing



Go back

Appendix: Porous medium equation in 2D case

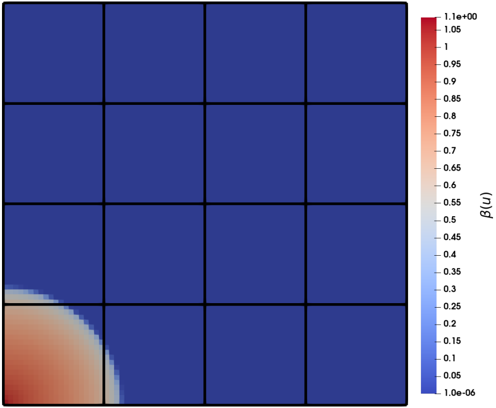
$$\partial_t \beta(u) - \Delta u = \delta_{\mathbf{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62

Appendix: Porous medium equation in 2D case

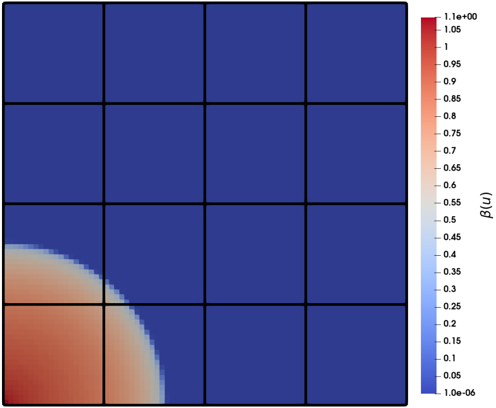
$$\partial_t \beta(u) - \Delta u = \delta_{\mathbf{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62

Appendix: Porous medium equation in 2D case

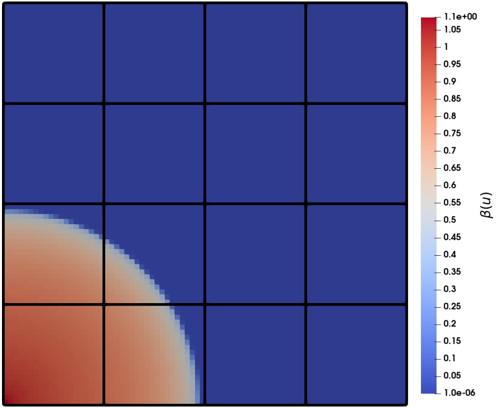
$$\partial_t \beta(u) - \Delta u = \delta_{\mathrm{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62

Appendix: Porous medium equation in 2D case

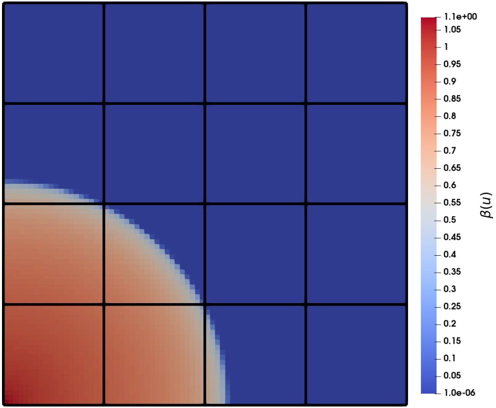
$$\partial_t \beta(u) - \Delta u = \delta_{\mathbf{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62

Appendix: Porous medium equation in 2D case

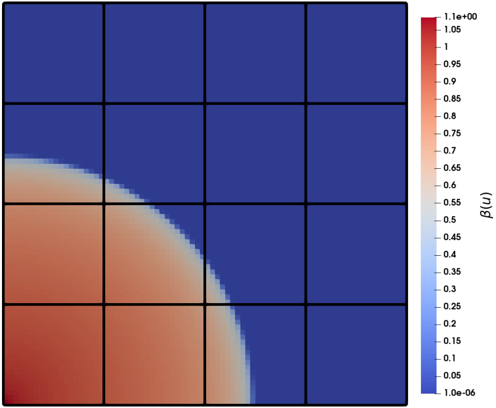
$$\partial_t \beta(u) - \Delta u = \delta_{\mathsf{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62

Appendix: Porous medium equation in 2D case

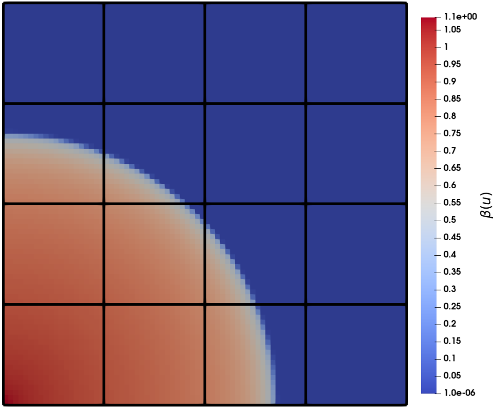
$$\partial_t \beta(u) - \Delta u = \delta_{\mathsf{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62

Appendix: Porous medium equation in 2D case

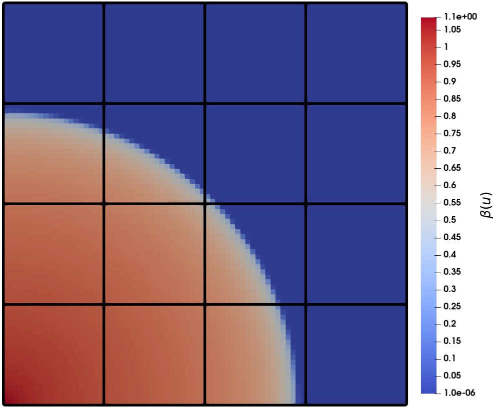
$$\partial_t \beta(u) - \Delta u = \delta_{\mathbf{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62

Appendix: Porous medium equation in 2D case

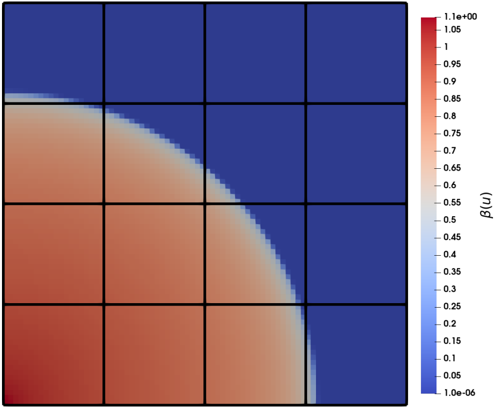
$$\partial_t \beta(u) - \Delta u = \delta_{\mathbf{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62

Appendix: Porous medium equation in 2D case

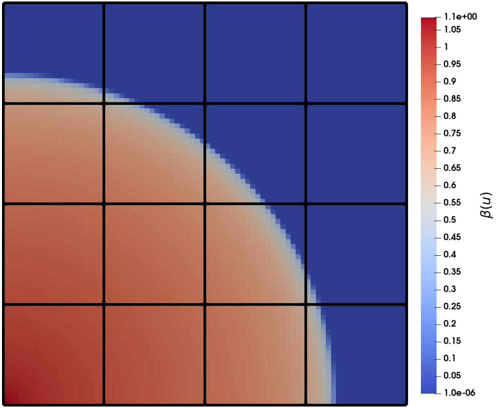
$$\partial_t \beta(u) - \Delta u = \delta_{\mathbf{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62

Appendix: Porous medium equation in 2D case

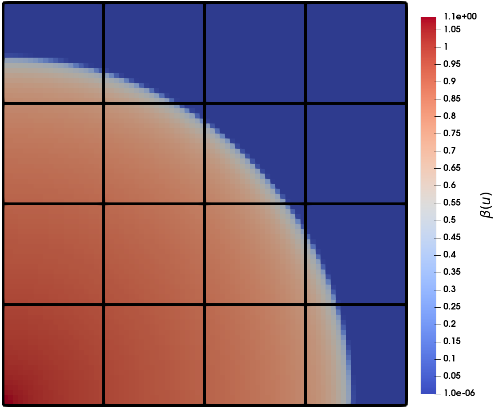
$$\partial_t \beta(u) - \Delta u = \delta_{\mathbf{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62

Appendix: Porous medium equation in 2D case

$$\partial_t \beta(u) - \Delta u = \delta_{\mathbf{x}=0}, \qquad \beta(u) = u^{1/10}$$



	$Nt = 1$			$Nt = 5$			$Nt = 10$		
$\sqrt{\#\text{unk}}$	20	40	80	20	40	80	20	40	80
Jac.	28	54	106	43	66	114	66	86	136
B.Jac.	10	11	13	30	33	38	50	58	62