Semismooth and smoothing Newton methods for nonlinear systems with complementarity constraints: adaptivity and inexact resolution

presented by **Joëlle Ferzly**

IFPEN-Inria meeting

December 1st, 2020

under the direction of:

Martin Vohralík	Director	Inria, Paris & CERMICS, Marne-la-Vallée
Ibtihel Ben Gharbia	Promoter	IFP Energies nouvelles, Rueil-Malmaison
Soleiman Yousef	Promoter	IFP Energies nouvelles, Rueil-Malmaison

Outline

1. Introduction
2. Classical methods
3. Adaptive inexact smoothing Newton method
4. Numerical tests
5. Conclusion

Model Problem

- A system of algebraic inequalities of the form: Find $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F},$$
$$\underbrace{\boldsymbol{K}(\boldsymbol{X}) \geq \mathbf{0}, \quad \boldsymbol{G}(\boldsymbol{X}) \geq \mathbf{0}, \quad \boldsymbol{K}(\boldsymbol{X}) \cdot \boldsymbol{G}(\boldsymbol{X}) = 0.}_{\textit{complementarity constraints}}$$

$\rightarrow$ $n > 1$ and $0 < m < n$ are two integers.
$\rightarrow$ $\mathbb{E} \in \mathbb{R}^{n-m,n}$, $\boldsymbol{F} \in \mathbb{R}^{n-m}$.
$\rightarrow$ $\boldsymbol{K} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and $\boldsymbol{G} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ are linear operators.

- The category of PDEs containing complementarity constraints leads to systems of nonlinear algebraic inequalities.

Model Problem

- A system of algebraic inequalities of the form: Find $\boldsymbol{X} \in \mathbb{R}^n$ such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F},$$
$$\underbrace{\boldsymbol{K}(\boldsymbol{X}) \geq \mathbf{0}, \quad \boldsymbol{G}(\boldsymbol{X}) \geq \mathbf{0}, \quad \boldsymbol{K}(\boldsymbol{X}) \cdot \boldsymbol{G}(\boldsymbol{X}) = 0.}_{complementarity\ constraints}$$

$\rightarrow$ $n > 1$ and $0 < m < n$ are two integers.
$\rightarrow$ $\mathbb{E} \in \mathbb{R}^{n-m,n}$, $\boldsymbol{F} \in \mathbb{R}^{n-m}$.
$\rightarrow$ $\boldsymbol{K} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and $\boldsymbol{G} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ are linear operators.

- The category of PDEs containing complementarity constraints leads to systems of nonlinear algebraic inequalities.

- Using a complementarity function (C-function), such system can be equivalently reformulated as a system of algebraic equalities.
- $\tilde{\mathbf{C}} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ $(m \geq 1)$ is a C-function if

$$\tilde{\mathbf{C}}(\boldsymbol{x}, \boldsymbol{y}) = 0 \iff \boldsymbol{x} \geq 0,\ \boldsymbol{y} \geq 0,\ \boldsymbol{x} \cdot \boldsymbol{y} = 0,\ \forall (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^m \times \mathbb{R}^m.$$

- **Well known C-functions:**

$$\left(\tilde{\mathbf{C}}_{\mathrm{FB}}(\boldsymbol{x}, \boldsymbol{y})\right)_l = \sqrt{\boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - \boldsymbol{x}_l - \boldsymbol{y}_l, \quad l = 1, \ldots, m.$$

$$\left(\tilde{\mathbf{C}}_{\min}(\boldsymbol{x}, \boldsymbol{y})\right)_l = \frac{\boldsymbol{x}_l + \boldsymbol{y}_l}{2} - \frac{|\boldsymbol{x}_l - \boldsymbol{y}_l|}{2}, \quad l = 1, \ldots, m.$$

- By introducing $\mathbf{C} : \mathbb{R}^n \to \mathbb{R}^m$ defined by $\mathbf{C}(\boldsymbol{X}) := \tilde{\mathbf{C}}\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right)$, the problem will be equivalent to a nonlinear nonsmooth (not of class $\mathcal{C}^1$) system of equalities:

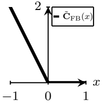
$$\begin{cases} \mathbb{E}\boldsymbol{X} &= \boldsymbol{F}, \\ \mathbf{C}(\boldsymbol{X}) &= \mathbf{0}. \end{cases}$$

- Using a complementarity function (C-function), such system can be equivalently reformulated as a system of algebraic equalities.
- $\tilde{\mathbf{C}} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ $(m \geq 1)$ is a C-function if

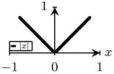
$$\tilde{\mathbf{C}}(\boldsymbol{x}, \boldsymbol{y}) = 0 \iff \boldsymbol{x} \geq 0,\ \boldsymbol{y} \geq 0,\ \boldsymbol{x} \cdot \boldsymbol{y} = 0,\ \forall (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^m \times \mathbb{R}^m.$$

- **Well known C-functions**:

$$\left(\tilde{\mathbf{C}}_{\mathrm{FB}}(\boldsymbol{x}, \boldsymbol{y})\right)_l = \sqrt{\boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - \boldsymbol{x}_l - \boldsymbol{y}_l, \quad l = 1, ..., m.$$



$$\left(\tilde{\mathbf{C}}_{\min}(\boldsymbol{x}, \boldsymbol{y})\right)_l = \frac{\boldsymbol{x}_l + \boldsymbol{y}_l}{2} - \frac{|\boldsymbol{x}_l - \boldsymbol{y}_l|}{2}, \quad l = 1, ..., m.$$



- By introducing $\mathbf{C} : \mathbb{R}^n \to \mathbb{R}^m$ defined by $\mathbf{C}(\boldsymbol{X}) := \tilde{\mathbf{C}}\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right)$, the problem will be equivalent to a nonlinear nonsmooth (not of class $\mathcal{C}^1$) system of equalities:

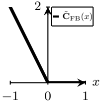
$$\begin{cases} \mathbb{E}\boldsymbol{X} = \boldsymbol{F}, \\ \mathbf{C}(\boldsymbol{X}) = \mathbf{0}. \end{cases}$$

- Using a complementarity function (C-function), such system can be equivalently reformulated as a system of algebraic equalities.
- $\tilde{\mathbf{C}} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ $(m \geq 1)$ is a C-function if

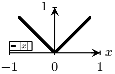
$$\tilde{\mathbf{C}}(\boldsymbol{x}, \boldsymbol{y}) = 0 \iff \boldsymbol{x} \geq 0,\ \boldsymbol{y} \geq 0,\ \boldsymbol{x} \cdot \boldsymbol{y} = 0,\ \forall (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^m \times \mathbb{R}^m.$$

- **Well known C-functions**:

$$\left(\tilde{\mathbf{C}}_{\mathrm{FB}}(\boldsymbol{x}, \boldsymbol{y})\right)_l = \sqrt{\boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - \boldsymbol{x}_l - \boldsymbol{y}_l, \quad l = 1, \ldots, m.$$



$$\left(\tilde{\mathbf{C}}_{\min}(\boldsymbol{x}, \boldsymbol{y})\right)_l = \frac{\boldsymbol{x}_l + \boldsymbol{y}_l}{2} - \frac{|\boldsymbol{x}_l - \boldsymbol{y}_l|}{2}, \quad l = 1, \ldots, m.$$



- By introducing $\mathbf{C} : \mathbb{R}^n \to \mathbb{R}^m$ defined by $\mathbf{C}(\boldsymbol{X}) := \tilde{\mathbf{C}}\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right)$, the problem will be equivalent to a nonlinear nonsmooth (not of class $\mathcal{C}^1$) system of equalities:

$$\begin{cases} \mathbb{E}\boldsymbol{X} & = & \boldsymbol{F}, \\ \mathbf{C}(\boldsymbol{X}) & = & \mathbf{0}. \end{cases}$$

Motivation

- Replace the semismooth (non-differentiable) C-function $\mathbf{C}(\cdot)$ by a smooth (differentiable) function $\mathbf{C}_\mu(\cdot)$, with μ a small parameter, such that

$$\|\mathbf{C}_\mu(\cdot) - \mathbf{C}(\cdot)\| \to \mathbf{0} \text{ as } \mu \to 0.$$

- Establish a posteriori error estimate that allows to:
  → Estimate the total error.
  → Distinguish the smoothing, linearization, and algebraic error components.
  → Formulate adaptive stopping criteria.

- Propose adaptive inexact algorithms for the smoothing Newton method and the interior point method.

Motivation

- Replace the semismooth (non-differentiable) C-function $\mathbf{C}(\cdot)$ by a smooth (differentiable) function $\mathbf{C}_\mu(\cdot)$, with μ a small parameter, such that

$$\|\mathbf{C}_\mu(\cdot) - \mathbf{C}(\cdot)\| \to \mathbf{0} \text{ as } \mu \to 0.$$

- Establish a posteriori error estimate that allows to:
  $\to$ Estimate the total error.
  $\to$ Distinguish the smoothing, linearization, and algebraic error components.
  $\to$ Formulate adaptive stopping criteria.

- Propose adaptive inexact algorithms for the smoothing Newton method and the interior point method.

Motivation

- Replace the semismooth (non-differentiable) C-function $\mathbf{C}(\cdot)$ by a smooth (differentiable) function $\mathbf{C}_\mu(\cdot)$, with μ a small parameter, such that
$$\|\mathbf{C}_\mu(\cdot) - \mathbf{C}(\cdot)\| \to \mathbf{0} \text{ as } \mu \to 0.$$

- Establish a posteriori error estimate that allows to:
  $\to$ Estimate the total error.
  $\to$ Distinguish the smoothing, linearization, and algebraic error components.
  $\to$ Formulate adaptive stopping criteria.

- Propose adaptive inexact algorithms for the smoothing Newton method and the interior point method.

1 Introduction

2 Classical methods

3 Adaptive inexact smoothing Newton method

4 Numerical tests

5 Conclusion

Classical semismooth Newton methods

- Iterative semismooth linearization method.
- For $\boldsymbol{X}^0 \in \mathbb{R}^n$, on step $k \geq 1$, one looks for $\boldsymbol{X}^k \in \mathbb{R}^n$ such that

$$\mathbb{A}^{k-1}\boldsymbol{X}^k = \boldsymbol{B}^{k-1},$$

where the Jacobian matrix and the right-hand side vector are given by

$$\mathbb{A}^{k-1} := \begin{bmatrix} \mathbb{E} \\ \mathbf{J_C}(\boldsymbol{X}^{k-1}) \end{bmatrix} \in \mathbb{R}^{n,n},$$

$$\boldsymbol{B}^{k-1} := \begin{bmatrix} \boldsymbol{F} \\ \mathbf{J_C}(\boldsymbol{X}^{k-1})\boldsymbol{X}^{k-1} - \mathbf{C}(\boldsymbol{X}^{k-1}) \end{bmatrix} \in \mathbb{R}^{n},$$

and $\mathbf{J_C}$ is the (generalized) Jacobian matrix in the sense of Clarke of the semismooth C-function $\mathbf{C}$.

Nonparametric interior-point method

- Introduce:
  $\rightarrow$ a smoothing parameter $\mu > 0$,
  $\rightarrow$ a vector $\boldsymbol{\mu} \in \mathbb{R}^m$, such that $\boldsymbol{\mu} = \mu \mathbf{1}$, $\mathbf{1} = [1, \ldots, 1] \in \mathbb{R}^m$.

- Replace the original nonsmooth problem by the smoothed problem: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F},$$
$$\boldsymbol{K}(\boldsymbol{X}) \geq \mathbf{0}, \;\; \boldsymbol{G}(\boldsymbol{X}) \geq \mathbf{0}, \;\; \boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}) - \boldsymbol{\mu} = \mathbf{0},$$

where $\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}) = \left[(\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}))_1, \ldots, (\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}))_m\right]^T$.

- Treat μ as an unknown.
- Introduce the following new equation into the system

$$\epsilon\mu + \mu^2 = 0.$$

- Rewrite the problem as enlarged nonlinear smooth system.

D. T. S. Vu, Numerical resolution of algebraic systems with complementarity conditions, Ph.D. thesis, Paris Saclay University, (2020).

Nonparametric interior-point method

- Introduce:
  $\rightarrow$ a smoothing parameter $\mu > 0$,
  $\rightarrow$ a vector $\boldsymbol{\mu} \in \mathbb{R}^m$, such that $\boldsymbol{\mu} = \mu \mathbf{1}$, $\mathbf{1} = [1, \ldots, 1] \in \mathbb{R}^m$.

- Replace the original nonsmooth problem by the smoothed problem: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\mathbb{E}\boldsymbol{X} = \boldsymbol{F},$$
$$\boldsymbol{K}(\boldsymbol{X}) \geq \mathbf{0}, \;\; \boldsymbol{G}(\boldsymbol{X}) \geq \mathbf{0}, \;\; \boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}) - \boldsymbol{\mu} = \mathbf{0},$$

where $\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}) = \left[(\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}))_1, \ldots, (\boldsymbol{K}(\boldsymbol{X})\boldsymbol{G}(\boldsymbol{X}))_m\right]^T$.

- Treat μ as an unknown.
- Introduce the following new equation into the system

$$\epsilon\mu + \mu^2 = 0.$$

- Rewrite the problem as enlarged nonlinear smooth system.

D. T. S. Vu, Numerical resolution of algebraic systems with complementarity conditions, Ph.D. thesis, Paris Saclay University, (2020).

1 Introduction

2 Classical methods

3 Adaptive inexact smoothing Newton method

4 Numerical tests

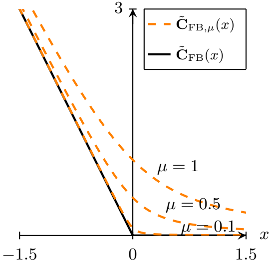
5 Conclusion

Smoothed C-functions

Smoothed F-B function:

$$(\tilde{\mathbf{C}}_{\mathrm{FB},\mu}(\boldsymbol{x},\boldsymbol{y}))_l = \sqrt{\mu^2 + \boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - (\boldsymbol{x}_l + \boldsymbol{y}_l)$$

$$l = 1, \ldots, m.$$



Smoothed min function:

$$(\tilde{\mathbf{C}}_{\min,\mu}(\boldsymbol{x},\boldsymbol{y}))_l = \frac{\boldsymbol{x}_l + \boldsymbol{y}_l}{2} - \frac{(|\boldsymbol{x} - \boldsymbol{y}|_\mu)_l}{2}$$

$$\text{where} \quad (|\boldsymbol{x}|_\mu)_l = \sqrt{\mu^2 + \boldsymbol{x}_l^2},$$

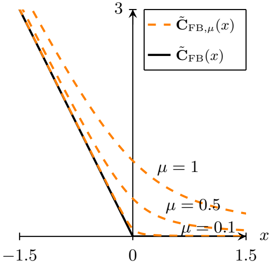
$$l = 1, \ldots, m.$$

Smoothed C-functions

Smoothed F-B function:

$$\left(\tilde{\mathbf{C}}_{\mathrm{FB},\mu}(\boldsymbol{x},\boldsymbol{y})\right)_l = \sqrt{\mu^2 + \boldsymbol{x}_l^2 + \boldsymbol{y}_l^2} - (\boldsymbol{x}_l + \boldsymbol{y}_l)$$

$$l = 1, \ldots, m.$$

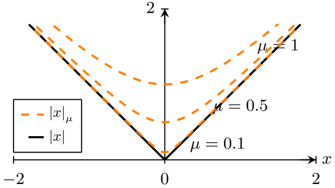


Smoothed min function:

$$\left(\tilde{\mathbf{C}}_{\min,\mu}(\boldsymbol{x},\boldsymbol{y})\right)_l = \frac{\boldsymbol{x}_l + \boldsymbol{y}_l}{2} - \frac{\left(|\boldsymbol{x}-\boldsymbol{y}|_\mu\right)_l}{2}$$

where $\left(|\boldsymbol{x}|_\mu\right)_l = \sqrt{\mu^2 + \boldsymbol{x}_l^2}$,

$$l = 1, \ldots, m.$$



Adaptive inexact smoothing Newton method

- Denote by $j \geq 0$ a smoothing iteration.
- Update of μ^j :
  $\rightarrow$ Actual work: a geometric sequence $\mu^{j+1} = 0.1\mu^j$.
  $\rightarrow$ Future work: an update based on the PDE discretization error.
- Define a function $\mathbf{C}_{\mu^j} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ as

$$\mathbf{C}_{\mu^j}(\boldsymbol{X}) := \tilde{\mathbf{C}}_{\mu^j}\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right),$$

  where $\tilde{\mathbf{C}}_{\mu^j} : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}^m$ is a smoothed C-function.
- The smoothed problem is a system of smooth (of class $\mathcal{C}^1$) nonlinear equations, written as: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\begin{cases} \mathbb{E}\boldsymbol{X}^j &= \boldsymbol{F}, \\ \mathbf{C}_{\mu^j}(\boldsymbol{X}^j) &= \mathbf{0}. \end{cases}$$

  $\rightarrow$ Apply the classical Newton method.
  $\rightarrow$ Solve the resulting linear system using GMRES.

Adaptive inexact smoothing Newton method

- Denote by $j \geq 0$ a smoothing iteration.
- Update of μ^j :
  → Actual work: a geometric sequence $\mu^{j+1} = 0.1\mu^j$.
  → Future work: an update based on the PDE discretization error.
- Define a function $\mathbf{C}_{\mu^j} : \mathbb{R}^n \to \mathbb{R}^m$ as

$$\mathbf{C}_{\mu^j}(\boldsymbol{X}) := \tilde{\mathbf{C}}_{\mu^j}\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right),$$

where $\tilde{\mathbf{C}}_{\mu^j} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ is a smoothed C-function.

- The smoothed problem is a system of smooth (of class $\mathcal{C}^1$) nonlinear equations, written as: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\begin{cases} \mathbb{E}\boldsymbol{X}^j & = & \boldsymbol{F}, \\ \mathbf{C}_{\mu^j}(\boldsymbol{X}^j) & = & \mathbf{0}. \end{cases}$$

→ Apply the classical Newton method.
→ Solve the resulting linear system using GMRES.

Adaptive inexact smoothing Newton method

- Denote by $j \geq 0$ a smoothing iteration.
- Update of μ^j :
  → Actual work: a geometric sequence $\mu^{j+1} = 0.1\mu^j$.
  → Future work: an update based on the PDE discretization error.
- Define a function $\mathbf{C}_{\mu^j} : \mathbb{R}^n \to \mathbb{R}^m$ as

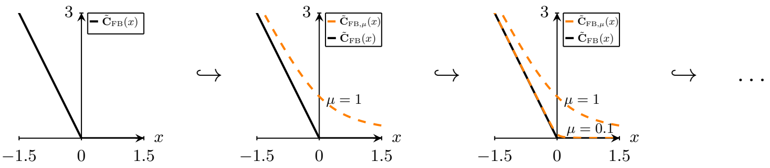
$$\mathbf{C}_{\mu^j}(\boldsymbol{X}) := \tilde{\mathbf{C}}_{\mu^j}\left(\boldsymbol{K}(\boldsymbol{X}), \boldsymbol{G}(\boldsymbol{X})\right),$$

  where $\tilde{\mathbf{C}}_{\mu^j} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ is a smoothed C-function.
- The smoothed problem is a system of smooth (of class $\mathcal{C}^1$) nonlinear equations, written as: Find $\boldsymbol{X}^j \in \mathbb{R}^n$ such that

$$\left\{\begin{array}{rcl} \mathbb{E}\boldsymbol{X}^j & = & \boldsymbol{F}, \\ \mathbf{C}_{\mu^j}(\boldsymbol{X}^j) & = & \mathbf{0}. \end{array}\right.$$

  → Apply the classical Newton method.
  → Solve the resulting linear system using GMRES.

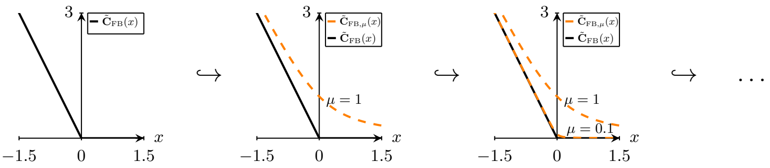
$j = 1$	$j = 2$	$j = j + 1$
$\mu^1 = 1$	$\mu^2 = 0.1$	$\mu^{j+1} = 0.1\mu^j$

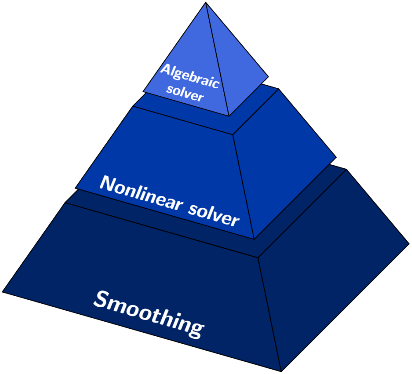


$j = 1$ $\mu^1 = 1$

$j = 2$ $\mu^2 = 0.1$

$j = j + 1$ $\mu^{j+1} = 0.1\mu^j$



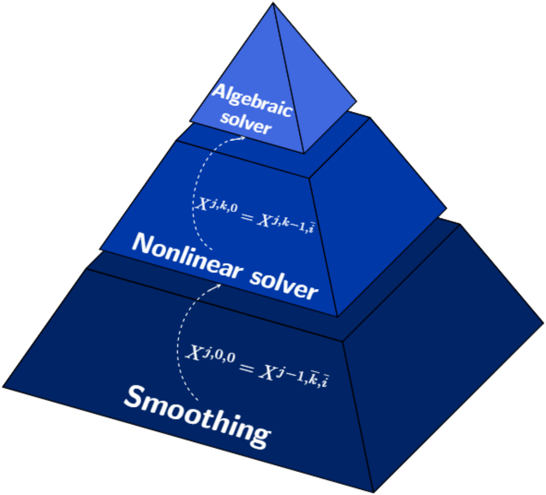


Notations:

→ j : smoothing iteration

→ k : linearization iteration

→ i : algebraic iteration

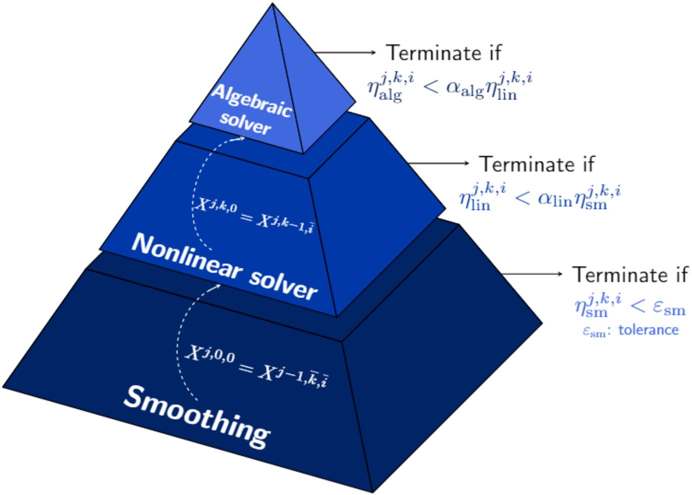


Notations:

→ j : smoothing iteration

→ k : linearization iteration

→ i : algebraic iteration



A posteriori error estimate distinguishing the error components

- Recall that the initial problem to solve is

$$\begin{cases} \mathbb{E}\boldsymbol{X} &= \boldsymbol{F}, \\ \mathbf{C}(\boldsymbol{X}) &= \mathbf{0}. \end{cases}$$

- The total residual vector of the system is given by

$$\boldsymbol{R}(\boldsymbol{X}^{j,k,i}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\mathbf{C}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}.$$

- Introduce $\mathbf{C}_{\mu^j}^{j,k-1} : \mathbb{R}^n \to \mathbb{R}^m$, the linearization of $\mathbf{C}_{\mu^j}$:

$$\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{V}) := \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k-1}) + \mathbf{J}_{\mathbf{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})(\boldsymbol{V} - \boldsymbol{X}^{j,k-1}), \quad \boldsymbol{V} \in \mathbb{R}^n.$$

- Add and substract $\mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i})$ and its linearization $\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i})$

$$\mathbf{C}(\boldsymbol{X}^{j,k,i}) = \mathbf{C}(\boldsymbol{X}^{j,k,i}) \pm \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \pm \mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}).$$

A posteriori error estimate distinguishing the error components

- Recall that the initial problem to solve is

$$\left\{ \begin{array}{rcl} \mathbb{E}\boldsymbol{X} & = & \boldsymbol{F}, \\ \mathbf{C}(\boldsymbol{X}) & = & \mathbf{0}. \end{array} \right.$$

- The total residual vector of the system is given by

$$\boldsymbol{R}(\boldsymbol{X}^{j,k,i}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\mathbf{C}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}.$$

- Introduce $\mathbf{C}_{\mu^j}^{j,k-1} : \mathbb{R}^n \to \mathbb{R}^m$, the linearization of $\mathbf{C}_{\mu^j}$:

$$\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{V}) := \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k-1}) + \mathbf{J}_{\mathbf{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})(\boldsymbol{V} - \boldsymbol{X}^{j,k-1}), \quad \boldsymbol{V} \in \mathbb{R}^n.$$

- Add and substract $\mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i})$ and its linearization $\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i})$

$$\mathbf{C}(\boldsymbol{X}^{j,k,i}) = \mathbf{C}(\boldsymbol{X}^{j,k,i}) \pm \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \pm \mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}).$$

A posteriori error estimate distinguishing the error components

- Recall that the initial problem to solve is

$$\begin{cases} \mathbb{E}\boldsymbol{X} &= \boldsymbol{F}, \\ \mathbf{C}(\boldsymbol{X}) &= \mathbf{0}. \end{cases}$$

- The total residual vector of the system is given by

$$\boldsymbol{R}(\boldsymbol{X}^{j,k,i}) := \begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\mathbf{C}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}.$$

- Introduce $\mathbf{C}_{\mu^j}^{j,k-1} : \mathbb{R}^n \to \mathbb{R}^m$, the linearization of $\mathbf{C}_{\mu^j}$:

$$\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{V}) := \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k-1}) + \mathbf{J}_{\mathbf{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})(\boldsymbol{V} - \boldsymbol{X}^{j,k-1}), \quad \boldsymbol{V} \in \mathbb{R}^n.$$

- Add and substract $\mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i})$ and its linearization $\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i})$

$$\mathbf{C}(\boldsymbol{X}^{j,k,i}) = \mathbf{C}(\boldsymbol{X}^{j,k,i}) \pm \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \pm \mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}).$$

A posteriori error estimate distinguishing the error components

- Recall that the initial problem to solve is

$$\left\{ \begin{array}{rcl} \mathbb{E}\boldsymbol{X} & = & \boldsymbol{F}, \\ \mathbf{C}(\boldsymbol{X}) & = & \mathbf{0}. \end{array} \right.$$

- The total residual vector of the system is given by

$$\boldsymbol{R}(\boldsymbol{X}^{j,k,i}) := \left[\begin{array}{c} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\mathbf{C}(\boldsymbol{X}^{j,k,i}) \end{array} \right].$$

- Introduce $\mathbf{C}_{\mu^j}^{j,k-1} : \mathbb{R}^n \to \mathbb{R}^m$, the linearization of $\mathbf{C}_{\mu^j}$:

$$\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{V}) := \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k-1}) + \mathbf{J}_{\mathbf{C}_{\mu^j}}(\boldsymbol{X}^{j,k-1})(\boldsymbol{V} - \boldsymbol{X}^{j,k-1}), \quad \boldsymbol{V} \in \mathbb{R}^n.$$

- Add and substract $\mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i})$ and its linearization $\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i})$

$$\mathbf{C}(\boldsymbol{X}^{j,k,i}) = \mathbf{C}(\boldsymbol{X}^{j,k,i}) \pm \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \pm \mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}).$$

- The total residual vector can be decomposed as follows:

$$\boldsymbol{R}(\boldsymbol{X}^{j,k,i}) = \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \mathbf{C}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{smoothness}} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) - \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{linearization}}$$

$$+ \underbrace{\begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{algebraic}}$$

- The relative L_2-norm of $\boldsymbol{R}(\boldsymbol{X}^{j,k,i})$ is bounded by

$$\left\|\boldsymbol{R}(\boldsymbol{X}^{j,k,i})\right\|_{\mathrm{r}} \leq \eta_{\mathrm{sm}}^{j,k,i} + \eta_{\mathrm{lin}}^{j,k,i} + \eta_{\mathrm{alg}}^{j,k,i},$$

with

$$\eta_{\mathrm{sm}}^{j,k,i} := \left\|\mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \mathbf{C}(\boldsymbol{X}^{j,k,i})\right\|_{\mathrm{r}},$$

$$\eta_{\mathrm{lin}}^{j,k,i} := \left\|\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) - \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i})\right\|_{\mathrm{r}},$$

$$\eta_{\mathrm{alg}}^{j,k,i} := \left(\left\|\boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i}\right\|_{\mathrm{r}}^{2} + \left\|\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i})\right\|_{\mathrm{r}}^{2}\right)^{\frac{1}{2}}.$$

- The total residual vector can be decomposed as follows:

$$\boldsymbol{R}(\boldsymbol{X}^{j,k,i}) = \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \mathbf{C}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{smoothness}} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) - \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{linearization}}$$

$$+ \underbrace{\begin{bmatrix} \boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i} \\ -\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) \end{bmatrix}}_{\text{algebraic}}$$

- The relative L_2-norm of $\boldsymbol{R}(\boldsymbol{X}^{j,k,i})$ is bounded by

$$\left\|\boldsymbol{R}(\boldsymbol{X}^{j,k,i})\right\|_{\mathrm{r}} \leq \eta_{\mathrm{sm}}^{j,k,i} + \eta_{\mathrm{lin}}^{j,k,i} + \eta_{\mathrm{alg}}^{j,k,i},$$

with

$$\eta_{\mathrm{sm}}^{j,k,i} := \left\|\mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i}) - \mathbf{C}(\boldsymbol{X}^{j,k,i})\right\|_{\mathrm{r}},$$

$$\eta_{\mathrm{lin}}^{j,k,i} := \left\|\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i}) - \mathbf{C}_{\mu^j}(\boldsymbol{X}^{j,k,i})\right\|_{\mathrm{r}},$$

$$\eta_{\mathrm{alg}}^{j,k,i} := \left(\left\|\boldsymbol{F} - \mathbb{E}\boldsymbol{X}^{j,k,i}\right\|_{\mathrm{r}}^{2} + \left\|\mathbf{C}_{\mu^j}^{j,k-1}(\boldsymbol{X}^{j,k,i})\right\|_{\mathrm{r}}^{2}\right)^{\frac{1}{2}}.$$

Adaptive inexact smoothing Newton algorithm

- **Initialization:**
  Choose a tolerance $\varepsilon_{\mathrm{sm}} > 0$, $\alpha \in]0,1[$ and $\alpha_{\mathrm{lin}}, \alpha_{\mathrm{alg}} \in]0,1]$.
  Set $j := 1$ and $\boldsymbol{X}^{j,0,0} := \boldsymbol{X}^0 \in \mathbb{R}^n$. Choose $\mu^j > 0$.

- **Smoothing loop:**
  - **Newton linearization loop:**
    0. Set $k := 1$.
    1. Consider the problem of finding a solution $\boldsymbol{X}^{j,k}$ to
    $$\mathbb{A}_{\mu^j}^{j,k-1,\bar{i}} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1,\bar{i}}.$$
    2. Algebraic solver loop
    a) Set $i := 1$ and $\boldsymbol{X}^{j,k,i} := \boldsymbol{X}^{j,k-1,\bar{i}}$ as initial guess.
    b) Perform one step of the iterative algebraic solver to obtain $\boldsymbol{X}^{j,k,i}$
    $$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\mathrm{alg}}^{j,k,i}.$$
    c) If $\eta_{\mathrm{alg}}^{j,k,i} < \alpha_{\mathrm{alg}} \eta_{\mathrm{lin}}^{j,k,i}$, stop. If not, set $i := i+1$ and go to **2b)**.
    3. If $\eta_{\mathrm{lin}}^{j,k,i} < \alpha_{\mathrm{lin}} \eta_{\mathrm{sm}}^{j,k,i}$, stop. If not, set $k := k+1$, go to **1**.
  - If $\|\boldsymbol{R}(\boldsymbol{X}^{j,k,i})\|_{\mathrm{r}} < \varepsilon_{\mathrm{sm}}$, stop. If not, set $j := j+1$ and $\mu^j := \alpha\mu^{j-1}$.
    Then set $\boldsymbol{X}^{j,0} := \boldsymbol{X}^{j-1,\bar{k},\bar{i}}$ and $k := 1$, and go to **1**.

Adaptive inexact smoothing Newton algorithm

- **Initialization:**
  Choose a tolerance $\varepsilon_{\mathrm{sm}} > 0$, $\alpha \in]0,1[$ and $\alpha_{\mathrm{lin}}, \alpha_{\mathrm{alg}} \in]0,1]$.
  Set $j := 1$ and $\boldsymbol{X}^{j,0,0} := \boldsymbol{X}^0 \in \mathbb{R}^n$. Choose $\mu^j > 0$.
- **Smoothing loop:**
  - **Newton linearization loop:**
    0. Set $k := 1$.
    1. Consider the problem of finding a solution $\boldsymbol{X}^{j,k}$ to

$$\mathbb{A}_{\mu^j}^{j,k-1,\bar{i}} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1,\bar{i}}.$$

    2. Algebraic solver loop
    a) Set $i := 1$ and $\boldsymbol{X}^{j,k,i} := \boldsymbol{X}^{j,k-1,\bar{i}}$ as initial guess.
    b) Perform one step of the iterative algebraic solver to obtain $\boldsymbol{X}^{j,k,i}$

$$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\mathrm{alg}}^{j,k,i}.$$

    c) If $\eta_{\mathrm{alg}}^{j,k,i} < \alpha_{\mathrm{alg}} \eta_{\mathrm{lin}}^{j,k,i}$, stop. If not, set $i := i+1$ and go to **2b)**.
    3. If $\eta_{\mathrm{lin}}^{j,k,i} < \alpha_{\mathrm{lin}} \eta_{\mathrm{sm}}^{j,k,i}$, stop. If not, set $k := k+1$, go to **1**.
  - If $\|\boldsymbol{R}(\boldsymbol{X}^{j,k,i})\|_{\mathrm{r}} < \varepsilon_{\mathrm{sm}}$, stop. If not, set $j := j+1$ and $\mu^j := \alpha\mu^{j-1}$.
    Then set $\boldsymbol{X}^{j,0} := \boldsymbol{X}^{j-1,\bar{k},\bar{i}}$ and $k := 1$, and go to **1**.

Adaptive inexact smoothing Newton algorithm

- **Initialization:**
  Choose a tolerance $\varepsilon_{\mathrm{sm}} > 0$, $\alpha \in]0,1[$ and $\alpha_{\mathrm{lin}}, \alpha_{\mathrm{alg}} \in]0,1]$.
  Set $j := 1$ and $\boldsymbol{X}^{j,0,0} := \boldsymbol{X}^0 \in \mathbb{R}^n$. Choose $\mu^j > 0$.
- **Smoothing loop:**
  - **Newton linearization loop:**
    0. Set $k := 1$.
    1. Consider the problem of finding a solution $\boldsymbol{X}^{j,k}$ to

    $$\mathbb{A}_{\mu^j}^{j,k-1,\bar{i}} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1,\bar{i}}.$$

    2. Algebraic solver loop
    a) Set $i := 1$ and $\boldsymbol{X}^{j,k,i} := \boldsymbol{X}^{j,k-1,\bar{i}}$ as initial guess.
    b) Perform one step of the iterative algebraic solver to obtain $\boldsymbol{X}^{j,k,i}$

    $$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\mathrm{alg}}^{j,k,i}.$$

    c) If $\eta_{\mathrm{alg}}^{j,k,i} < \alpha_{\mathrm{alg}} \eta_{\mathrm{lin}}^{j,k,i}$, stop. If not, set $i := i + 1$ and go to **2b)**.
    3. If $\eta_{\mathrm{lin}}^{j,k,i} < \alpha_{\mathrm{lin}} \eta_{\mathrm{sm}}^{j,k,i}$, stop. If not, set $k := k + 1$, go to **1**.
  - If $\|\boldsymbol{R}(\boldsymbol{X}^{j,k,i})\|_{\mathrm{r}} < \varepsilon_{\mathrm{sm}}$, stop. If not, set $j := j + 1$ and $\mu^j := \alpha\mu^{j-1}$.
    Then set $\boldsymbol{X}^{j,0} := \boldsymbol{X}^{j-1,\bar{k},\bar{i}}$ and $k := 1$, and go to **1**.

Adaptive inexact smoothing Newton algorithm

- **Initialization:**
  Choose a tolerance $\varepsilon_{\mathrm{sm}} > 0$, $\alpha \in]0,1[$ and $\alpha_{\mathrm{lin}}, \alpha_{\mathrm{alg}} \in]0,1]$.
  Set $j := 1$ and $\boldsymbol{X}^{j,0,0} := \boldsymbol{X}^0 \in \mathbb{R}^n$. Choose $\mu^j > 0$.
- **Smoothing loop:**
  - **Newton linearization loop:**
    0. Set $k := 1$.
    1. Consider the problem of finding a solution $\boldsymbol{X}^{j,k}$ to
    $$\mathbb{A}_{\mu^j}^{j,k-1,\bar{i}} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1,\bar{i}}.$$
    2. Algebraic solver loop
    a) Set $i := 1$ and $\boldsymbol{X}^{j,k,i} := \boldsymbol{X}^{j,k-1,\bar{i}}$ as initial guess.
    b) Perform one step of the iterative algebraic solver to obtain $\boldsymbol{X}^{j,k,i}$
    $$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\mathrm{alg}}^{j,k,i}.$$
    c) If $\eta_{\mathrm{alg}}^{j,k,i} < \alpha_{\mathrm{alg}} \eta_{\mathrm{lin}}^{j,k,i}$, stop. If not, set $i := i + 1$ and go to **2b)**.
    3. If $\eta_{\mathrm{lin}}^{j,k,i} < \alpha_{\mathrm{lin}} \eta_{\mathrm{sm}}^{j,k,i}$, stop. If not, set $k := k + 1$, go to **1**.
  - If $\|\boldsymbol{R}(\boldsymbol{X}^{j,k,i})\|_{\mathrm{r}} < \varepsilon_{\mathrm{sm}}$, stop. If not, set $j := j + 1$ and $\mu^j := \alpha \mu^{j-1}$.
    Then set $\boldsymbol{X}^{j,0} := \boldsymbol{X}^{j-1,\bar{k},\bar{i}}$ and $k := 1$, and go to **1**.

Adaptive inexact smoothing Newton algorithm

- **Initialization:**
  Choose a tolerance $\varepsilon_{\mathrm{sm}} > 0$, $\alpha \in]0,1[$ and $\alpha_{\mathrm{lin}}, \alpha_{\mathrm{alg}} \in]0,1]$.
  Set $j := 1$ and $\boldsymbol{X}^{j,0,0} := \boldsymbol{X}^0 \in \mathbb{R}^n$. Choose $\mu^j > 0$.
- **Smoothing loop:**
  - **Newton linearization loop:**
    0. Set $k := 1$.
    1. Consider the problem of finding a solution $\boldsymbol{X}^{j,k}$ to
    $$\mathbb{A}_{\mu^j}^{j,k-1,\bar{i}} \boldsymbol{X}^{j,k} = \boldsymbol{B}_{\mu^j}^{j,k-1,\bar{i}}.$$
    2. Algebraic solver loop
      a) Set $i := 1$ and $\boldsymbol{X}^{j,k,i} := \boldsymbol{X}^{j,k-1,\bar{i}}$ as initial guess.
      b) Perform one step of the iterative algebraic solver to obtain $\boldsymbol{X}^{j,k,i}$
      $$\mathbb{A}_{\mu^j}^{j,k-1} \boldsymbol{X}^{j,k,i} = \boldsymbol{B}_{\mu^j}^{j,k-1} - \boldsymbol{R}_{\mathrm{alg}}^{j,k,i}.$$
      c) If $\eta_{\mathrm{alg}}^{j,k,i} < \alpha_{\mathrm{alg}} \eta_{\mathrm{lin}}^{j,k,i}$, stop. If not, set $i := i+1$ and go to **2b)**.
    3. If $\eta_{\mathrm{lin}}^{j,k,i} < \alpha_{\mathrm{lin}} \eta_{\mathrm{sm}}^{j,k,i}$, stop. If not, set $k := k+1$, go to **1**.
  - If $\|\boldsymbol{R}(\boldsymbol{X}^{j,k,i})\|_{\mathrm{r}} < \varepsilon_{\mathrm{sm}}$, stop. If not, set $j := j+1$ and $\mu^j := \alpha \mu^{j-1}$.
    Then set $\boldsymbol{X}^{j,0} := \boldsymbol{X}^{j-1,\bar{k},\bar{i}}$ and $k := 1$, and go to **1**.

1 Introduction

2 Classical methods

3 Adaptive inexact smoothing Newton method

4 Numerical tests

5 Conclusion

Adaptive inexact smoothing Newton method

Settings: $n = 75000,\ \varepsilon_{\rm sm} = 10^{-5},\ \mu^1 = 1,\ \alpha = 0.1,\ \alpha_{\rm lin} = 1,\ \alpha_{\rm alg} = 10^{-3}.$

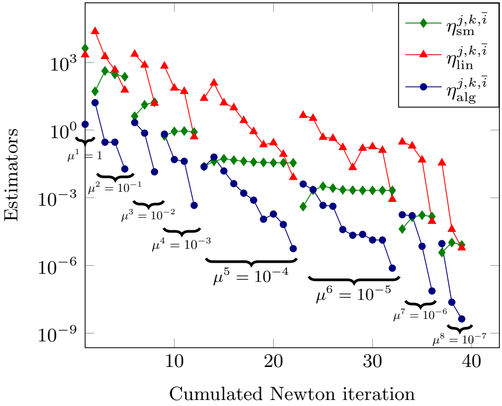


Figure: Estimators as a function of cumulated Newton iterations k, at convergence of the linear solver.

GMRES stopping criterion:

$\rightarrow$ Classical: $\boldsymbol{R}_{\text{alg}}^{j,k,i} := \dfrac{\|\mathbb{M}_2 \backslash (\mathbb{M}_1 \backslash (\boldsymbol{B} - \mathbb{A}\boldsymbol{X}^{j,k,i}))\|}{\|\mathbb{M}_2 \backslash (\mathbb{M}_1 \backslash \boldsymbol{B} - \mathbb{A}\boldsymbol{X}^{j,k-1})\|} \leq \tau$,
($\mathbb{M}_1, \mathbb{M}_2$:preconditioner matrices, τ : tolerance).

$\rightarrow$ Adaptive: $\eta_{\text{alg}}^{j,k,i} < \alpha_{\text{alg}} \eta_{\text{lin}}^{j,k,i}$.

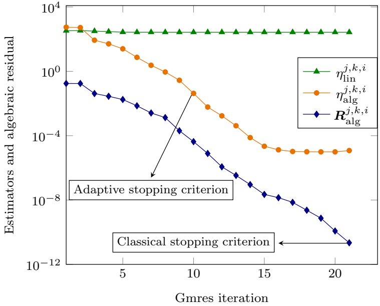


Figure: Algebraic and linearization estimators and GMRES algebraic residual as a function of GMRES iterations, for $j = 2,\ k = 2,\ i$ varies.

Figure: Estimators as a function of smoothing iterations j, at convergence of the linear and nonlinear solvers.

GMRES stopping criterion:

$\rightarrow$ Classical: $\boldsymbol{R}_{\text{alg}}^{j,k,i} := \frac{\|\mathbb{M}_2 \backslash (\mathbb{M}_1 \backslash (\boldsymbol{B} - \mathbb{A}\boldsymbol{X}^{j,k,i}))\|}{\|\mathbb{M}_2 \backslash (\mathbb{M}_1 \backslash \boldsymbol{B} - \mathbb{A}\boldsymbol{X}^{j,k-1})\|} \leq \tau$,
($\mathbb{M}_1, \mathbb{M}_2$:preconditioner matrices, τ : tolerance).

$\rightarrow$ Adaptive: $\eta_{\text{alg}}^{j,k,i} < \alpha_{\text{alg}} \eta_{\text{lin}}^{j,k,i}$.

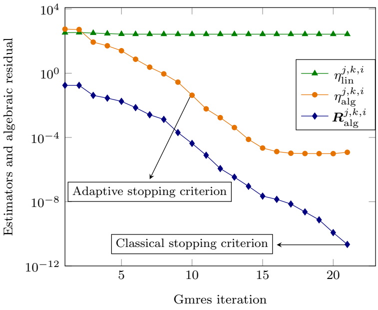


Figure: Algebraic and linearization estimators and GMRES algebraic residual as a function of GMRES iterations, for $j = 2,\ k = 2,\ i$ varies.

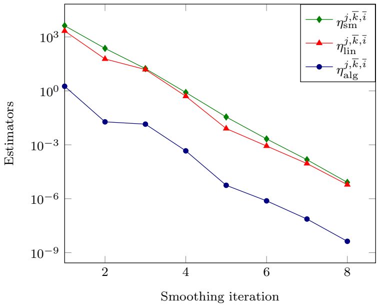


Figure: Estimators as a function of smoothing iterations j, at convergence of the linear and nonlinear solvers.

Comparison of the methods

- Comparison of the:
  - → Semismooth Newton method with F–B function (SSN-FB),
  - → Adaptive smoothing Newton method with smoothed F–B (ASN-FB),
  - → Nonparametric interior-point method (IP),
  - → Adaptive interior-point method (AIP).

- We introduce a unified linearization residual given for $\boldsymbol{V} \in \mathbb{R}^n$ by

$$\boldsymbol{R}(\boldsymbol{V}) = ||\boldsymbol{F} - \mathbb{E}\boldsymbol{V}|| + \left|\left|\boldsymbol{K}(\boldsymbol{V})^-\right|\right| + \left|\left|\boldsymbol{G}(\boldsymbol{V})^-\right|\right| + |\boldsymbol{K}(\boldsymbol{V}) \cdot \boldsymbol{G}(\boldsymbol{V})|,$$

where

$$\boldsymbol{K}(\boldsymbol{V})^- := \min[\boldsymbol{0}, \boldsymbol{K}(\boldsymbol{V})] \text{ and } \boldsymbol{G}(\boldsymbol{V})^- := \min[\boldsymbol{0}, \boldsymbol{G}(\boldsymbol{V})].$$

Settings: $n = 75000, \ \varepsilon_{\text{sm}} = 10^{-8}, \ \mu^1 = 1, \ \alpha = 0.1, \ \alpha_{\text{lin}} = 1.$

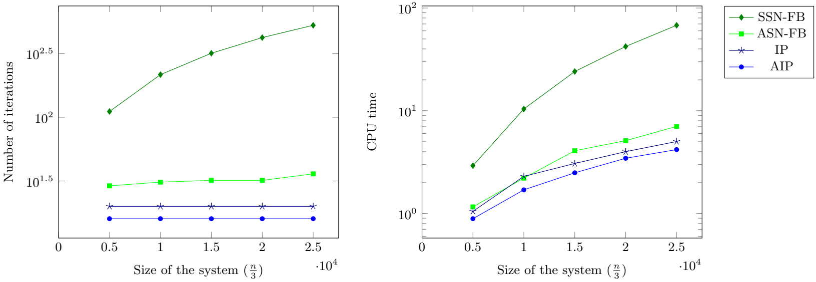


Figure: Left: Number of cumulated Newton iterations, right: CPU time, as a function of the size of the system, using a stopping criterion on the unified relative residual.

1 Introduction

2 Classical methods

3 Adaptive inexact smoothing Newton method

4 Numerical tests

5 Conclusion

Conclusions and outlook

Conclusions

→ The adaptive inexact smoothing Newton method provides an interesting reduction of the number of iterations.

→ The nonparametric interior-point method and the adaptive interior-point method behave almost similarly.

Outlook

→ Adaptively choose the smoothing parameter by defining an estimator related to the discretization error.

→ Apply the method to more involved problems.

Conclusions and outlook

- **Conclusions**
  - → The adaptive inexact smoothing Newton method provides an interesting reduction of the number of iterations.
  - → The nonparametric interior-point method and the adaptive interior-point method behave almost similarly.

- **Outlook**
  - → Adaptively choose the smoothing parameter by defining an estimator related to the discretization error.
  - → Apply the method to more involved problems.

Thank you
for your attention.