Salishan Workshop
April 28, 2015
Gleneden Beach, Oregon

On Numerical Resiliency in Numerical Linear Algebra Solvers

Luc Giraud

joint work with
E. Agullo (Inria), P. Salas (Sherbrooke Univ.),
F. Yetkin (Inria) and M. Zounon (Inria)

HiePACS - Inria Project
Inria Bordeaux Sud-Ouest

Outline

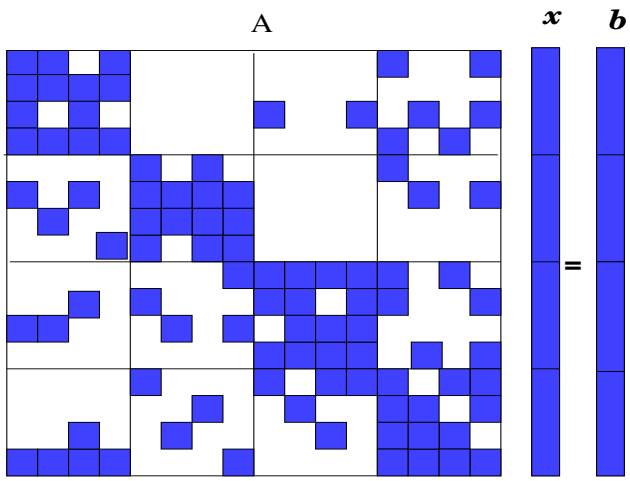
1. Hard fault: Interpolation-Restart strategies
   - In Krylov subspace linear solvers
   - In revisited eigensolvers

2. Soft errors in Conjugate Gradient (preliminary)

Outline

1. Hard fault: Interpolation-Restart strategies
   - In Krylov subspace linear solvers
   - In revisited eigensolvers

2. Soft errors in Conjugate Gradient (preliminary)



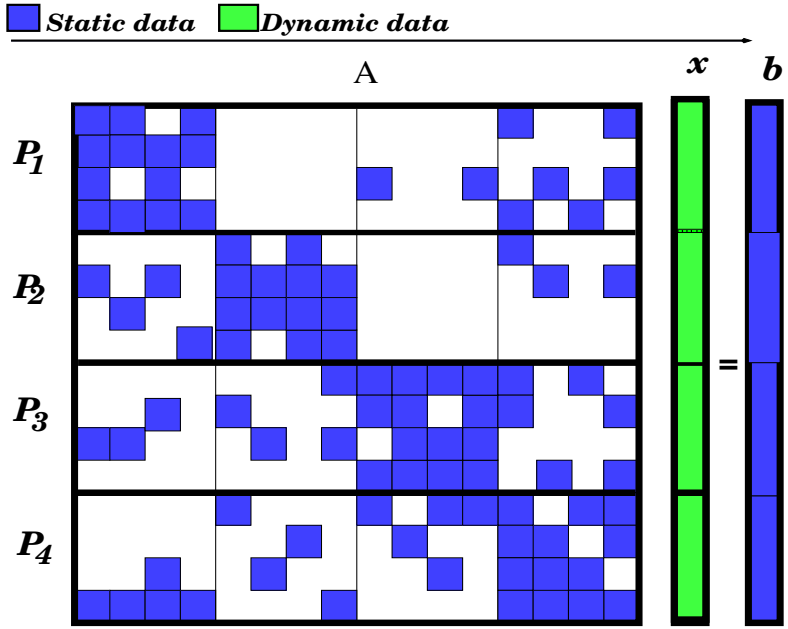
$$Ax = b$$

Objective: Design resilient algorithms for solving sparse linear systems

Two classes of iterative methods

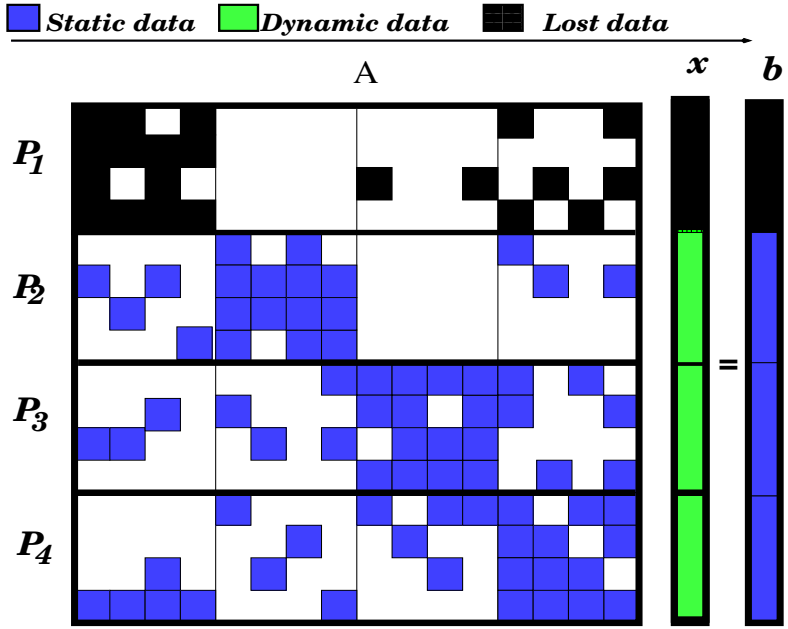
- Stationary methods (Jacobi, Gauss-Seidel, . . .)
- Krylov subspace methods (CG, GMRES, Bi-CGStab, . . .)

- Krylov methods are widely used but efforts are requied to go for extreme-scale



We distinguish two categories of data:

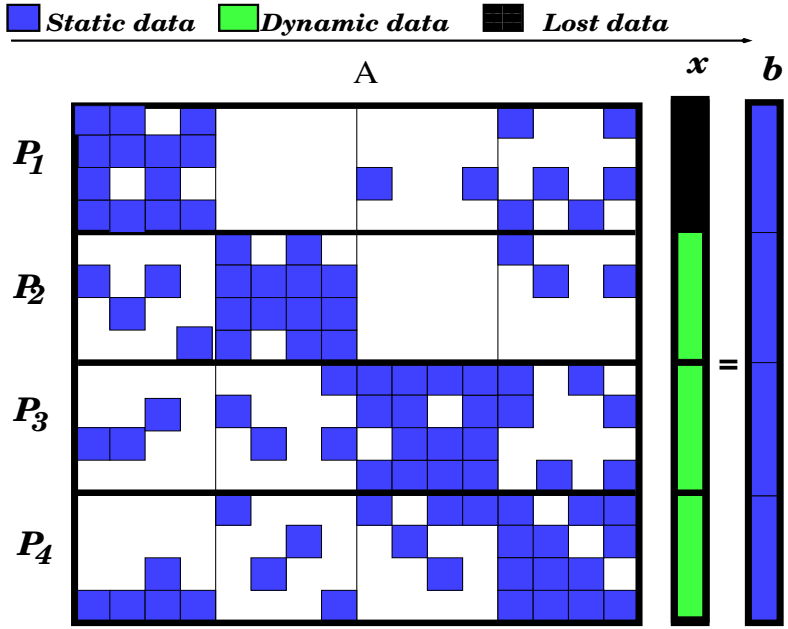
- Static data
- Dynamic data



We distinguish two categories of data:

- Static data
- Dynamic data

What will happen when P_1 crashes?



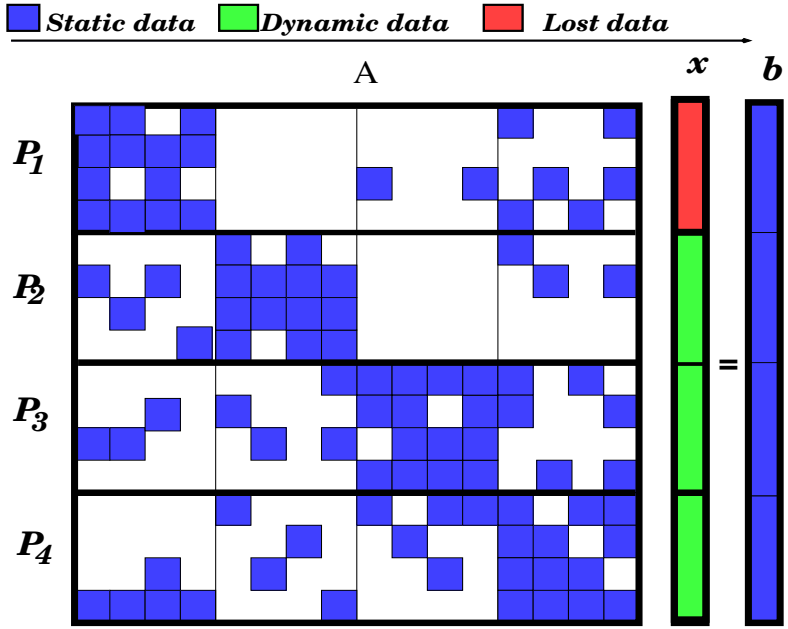
We distinguish two categories of data:

- Static data
- Dynamic data

What will happen when P_1 crashes?

Failed process is replaced

Static data are restored



We distinguish two categories of data:

- Static data
- Dynamic data

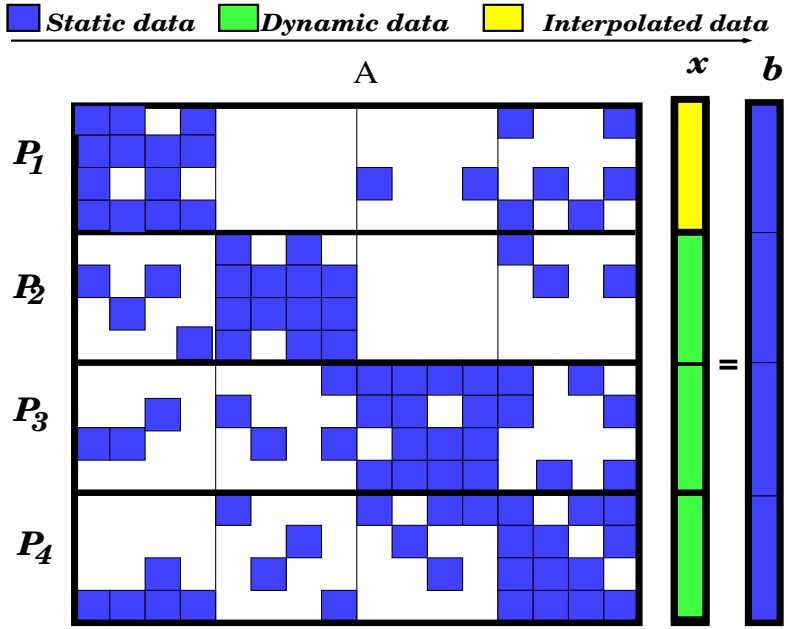
What will happen when P_1 crashes?

Failed process is replaced

Static data are restored

Reset: set x_1 to the initial value and restart

IR: interpolate x_1 and restart



We distinguish two categories of data:

- Static data
- Dynamic data

What will happen when P_1 crashes?

Failed process is replaced

Static data are restored

Reset: set x_1 to the initial value and restart

IR: interpolate x_1 and restart

General assumptions

1. Large dimensional vectors and matrices are distributed
2. Scalars and low dimensional vectors and matrices are replicated
3. There is a system mechanism to report faults
4. Faulty process is replaced
5. Static data are restored

Interpolation methods

Fault in linear system

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

Linear Interpolation (LI) [J. Langou et al., 2007]

Solve $A_{11}x_1 = b_1 - A_{12}x_2$ A_{11} must be non singular

Least Squares Interpolation (LSI)

Interpolation methods

Fault in linear system

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} ? \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$ How to interpolate x_1?

Linear Interpolation (LI) [J. Langou et al., 2007]

Solve $A_{11}x_1 = b_1 - A_{12}x_2$ A_{11} must be non singular

Least Squares Interpolation (LSI)

$$\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} x_1 + \begin{pmatrix} A_{21} \\ A_{22} \end{pmatrix} x_2 = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

$$x_1 = \underset{x}{argmin} \left\| \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} - \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} x_2 - \begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} x \right\|_2$$

Interpolation methods

Fault in linear system

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} ? \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$ How to interpolate x_1?

Linear Interpolation (LI) [J. Langou et al., 2007]

Solve $A_{11}x_1 = b_1 - A_{12}x_2$ A_{11} must be non singular

Least Squares Interpolation (LSI)

$$\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} x_1 + \begin{pmatrix} A_{21} \\ A_{22} \end{pmatrix} x_2 = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

$$x_1 = \underset{x}{argmin} \left\| \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} - \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} x_2 - \begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} x \right\|_2$$

Interpolation methods

Fault in linear system

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} ? \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$ How to interpolate x_1?

Linear Interpolation (LI) [J. Langou et al., 2007]

Solve $A_{11}x_1 = b_1 - A_{12}x_2$ A_{11} must be non singular

Least Squares Interpolation (LSI)

$$\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} x_1 + \begin{pmatrix} A_{21} \\ A_{22} \end{pmatrix} x_2 = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

$$x_1 = \underset{x}{argmin} \left\| \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} - \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} x_2 - \begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} x \right\|_2$$

Main properties of Interpolation-Restart strategies

LI property

The LI strategy preserves the monotone decrease of the A-norm of the forward error of CG or PCG.

LSI property

The LSI strategy preserves the monotone decrease of the residual norm of minimal residual Krylov subspace methods such as GMRES and MinRES.

Main properties of Interpolation-Restart strategies

LI property

The LI strategy preserves the monotone decrease of the A-norm of the forward error of CG or PCG.

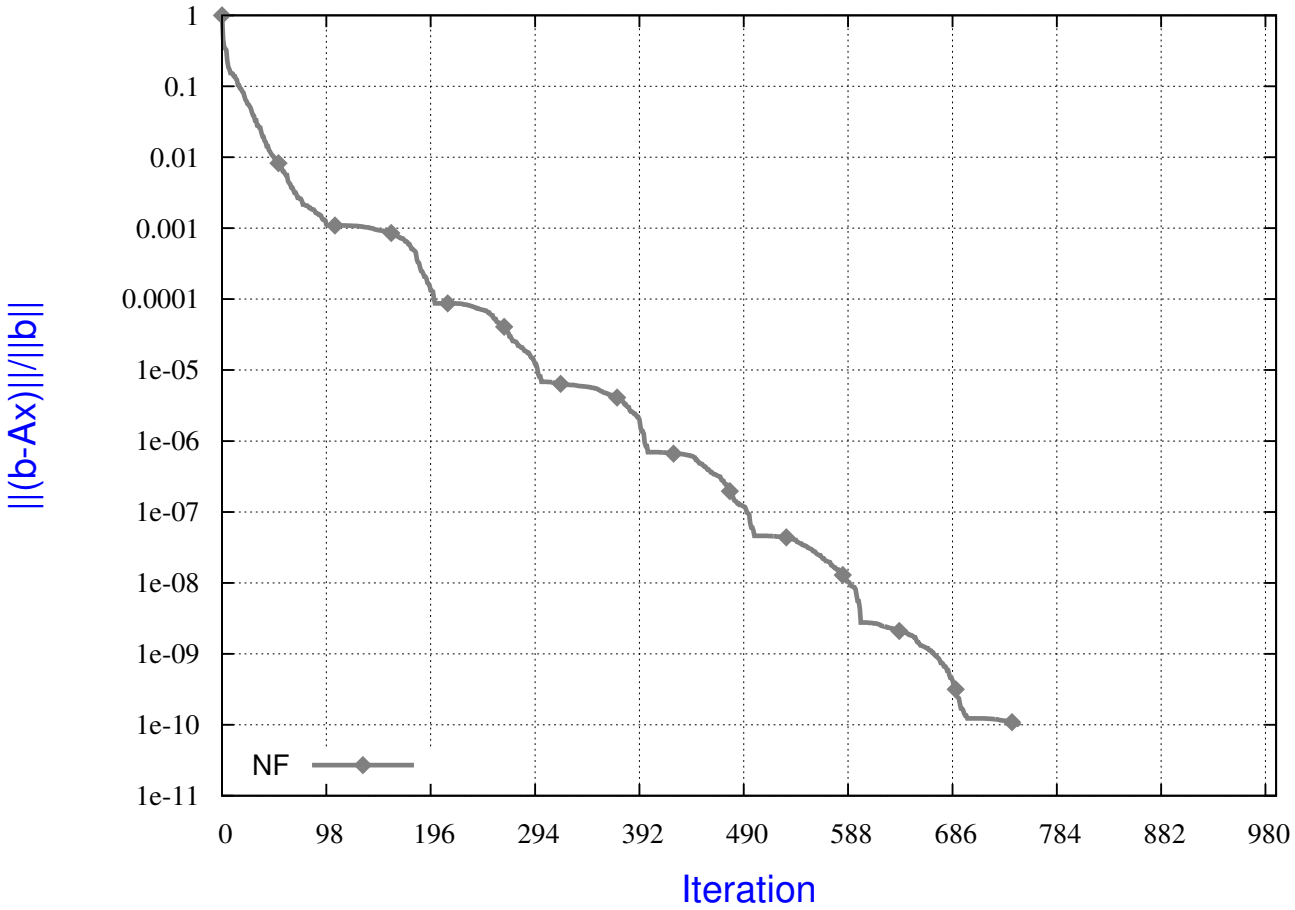
LSI property

The LSI strategy preserves the monotone decrease of the residual norm of minimal residual Krylov subspace methods such as GMRES and MinRES.

Experimental results

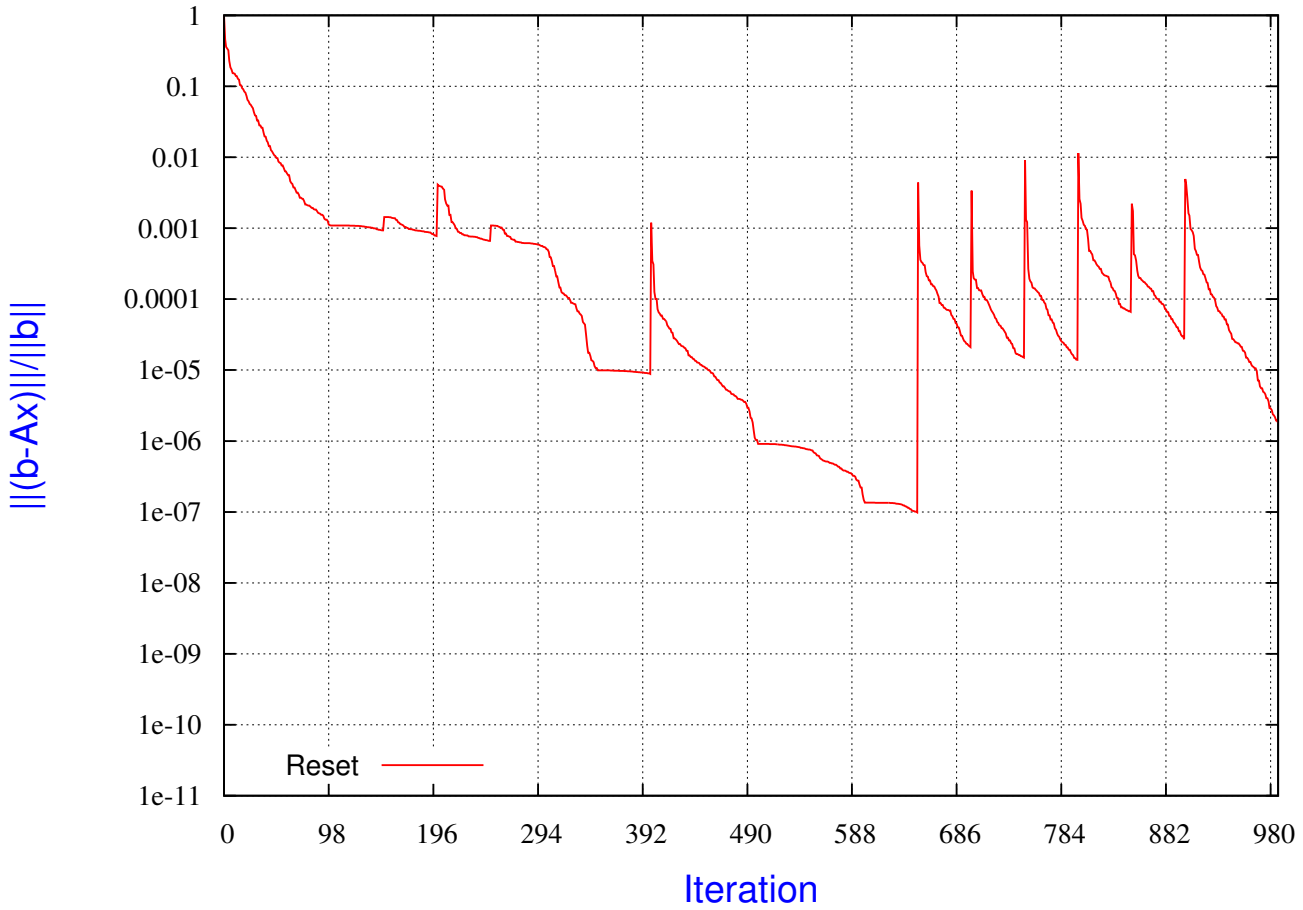
Impact of lost data volume

Impact of lost data volume



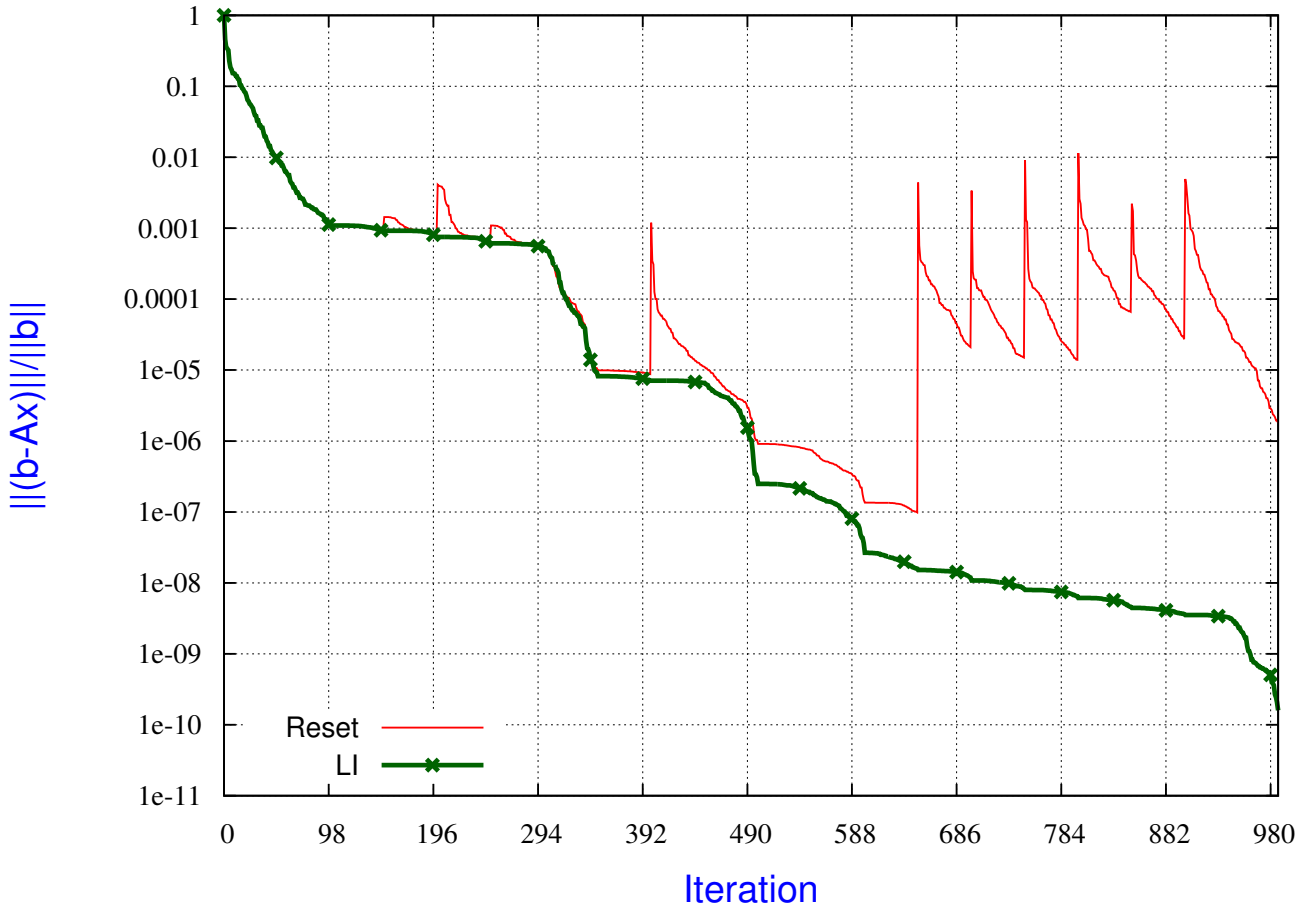
GMRES(100) - Averous/epb3

Impact of lost data volume



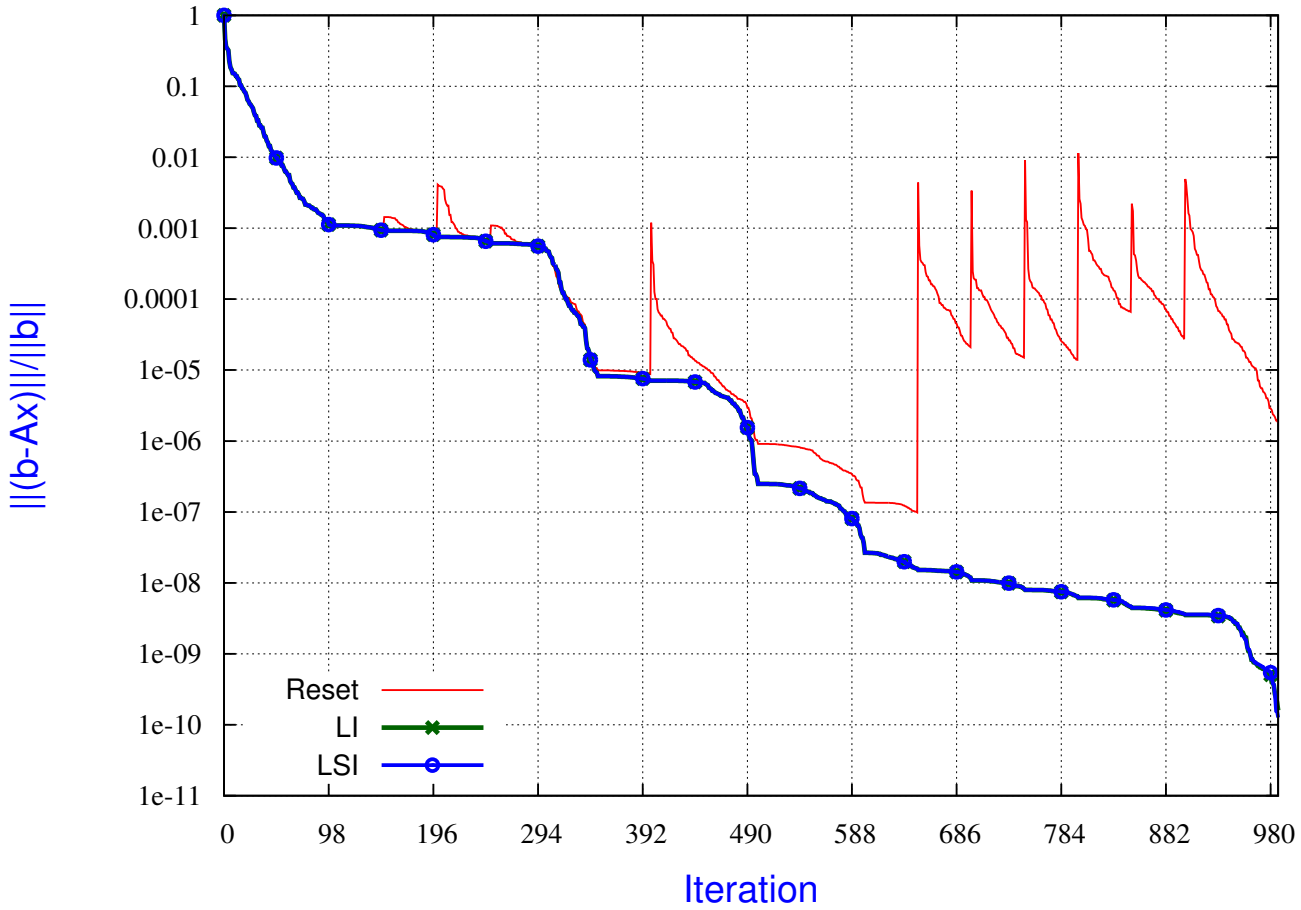
GMRES(100) - Averous/epb3 - 10 faults - **0.001%** data loss

Impact of lost data volume



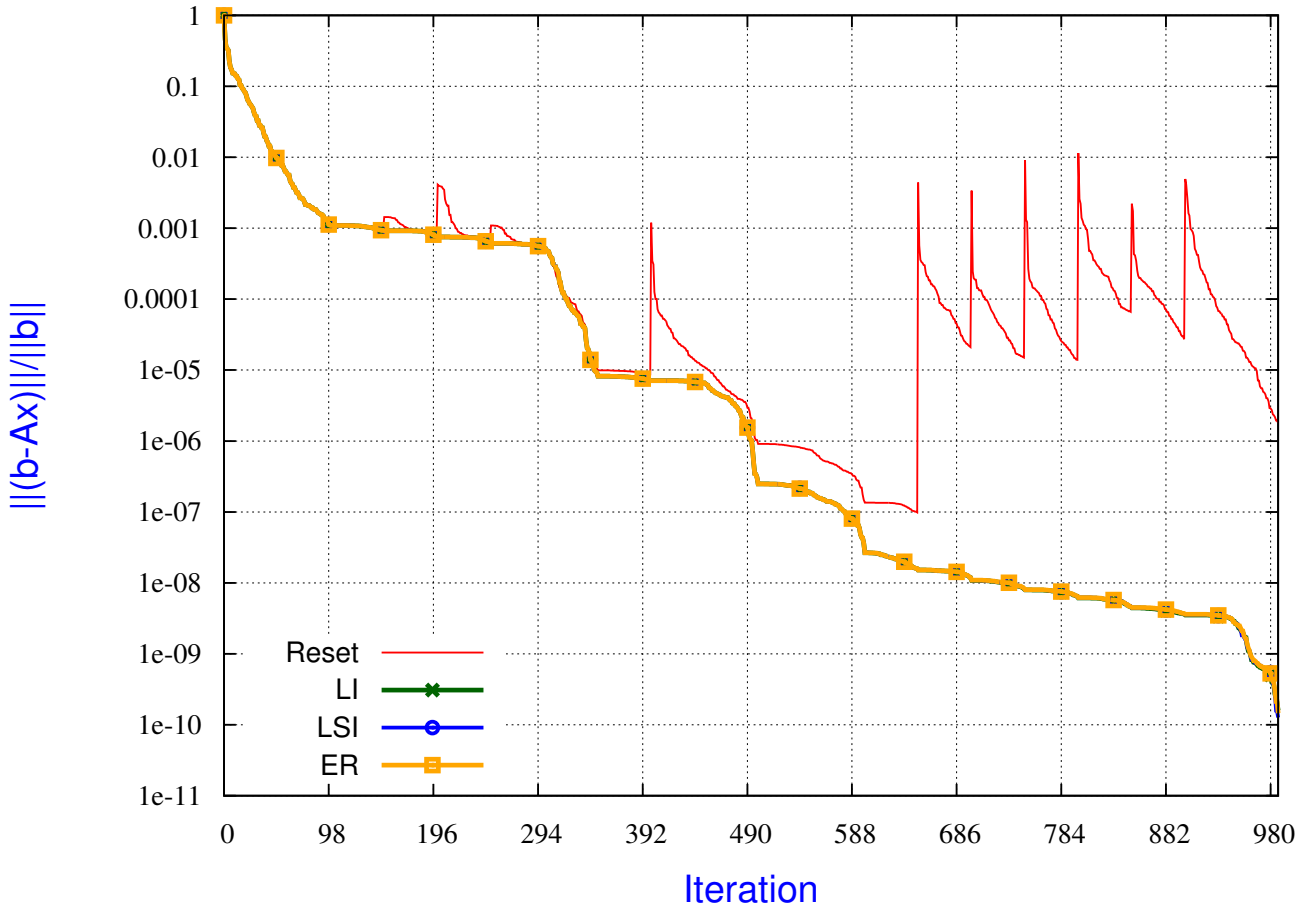
GMRES(100) - Averous/epb3 - 10 faults - **0.001%** data loss

Impact of lost data volume



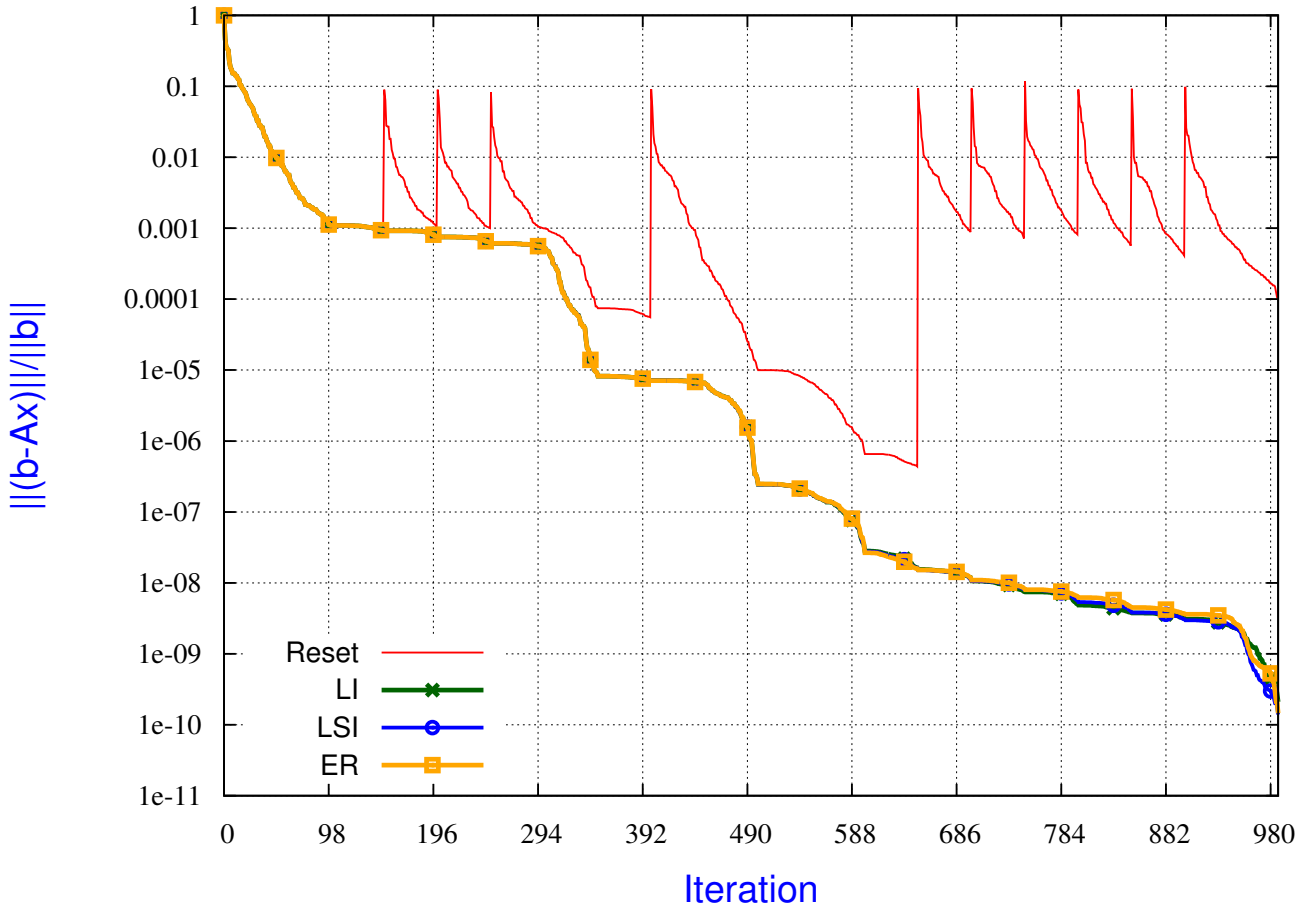
GMRES(100) - Averous/epb3 - 10 faults - **0.001%** data loss

Impact of lost data volume



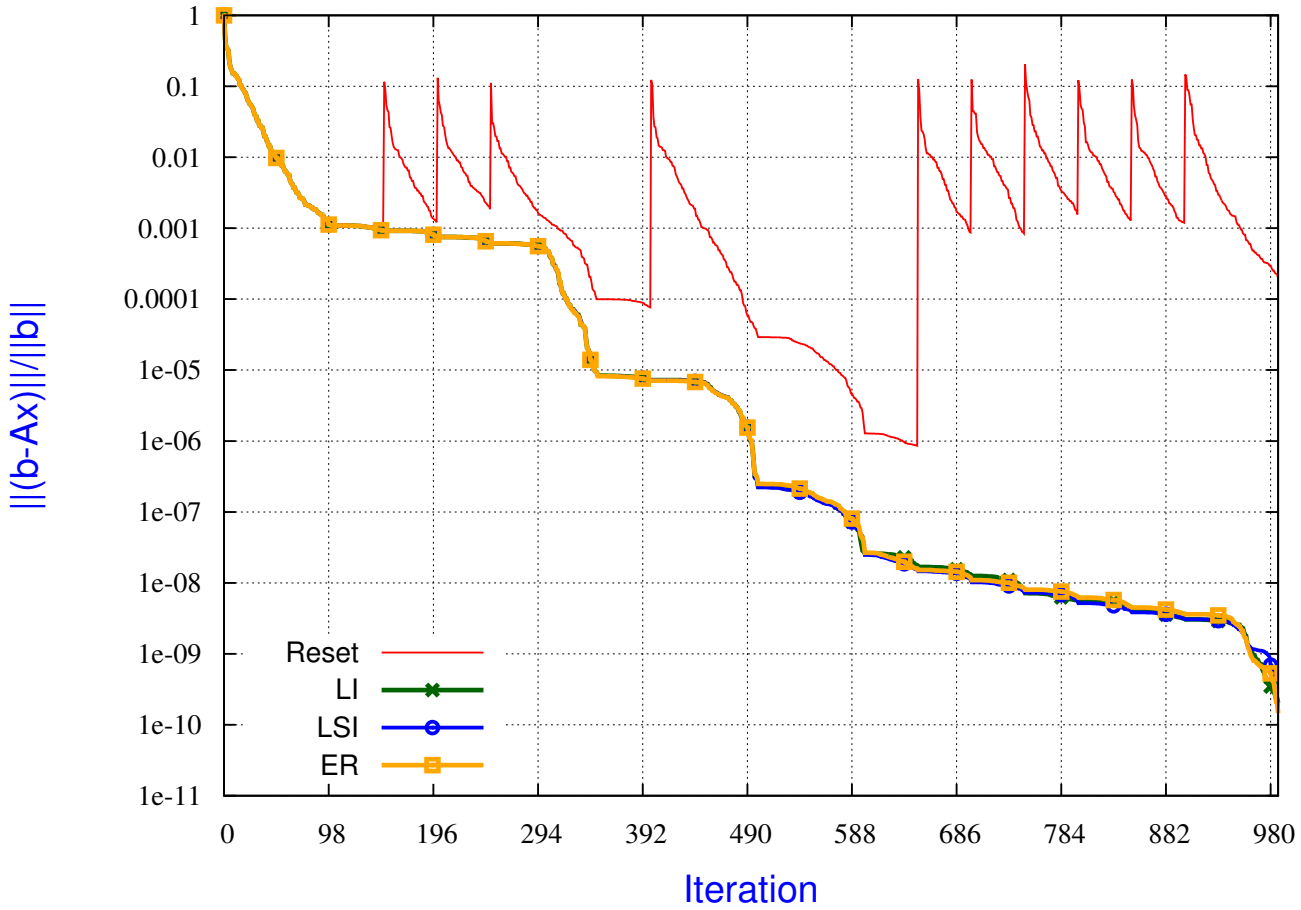
GMRES(100) - Averous/epb3 - 10 faults - **0.001%** data loss

Impact of lost data volume



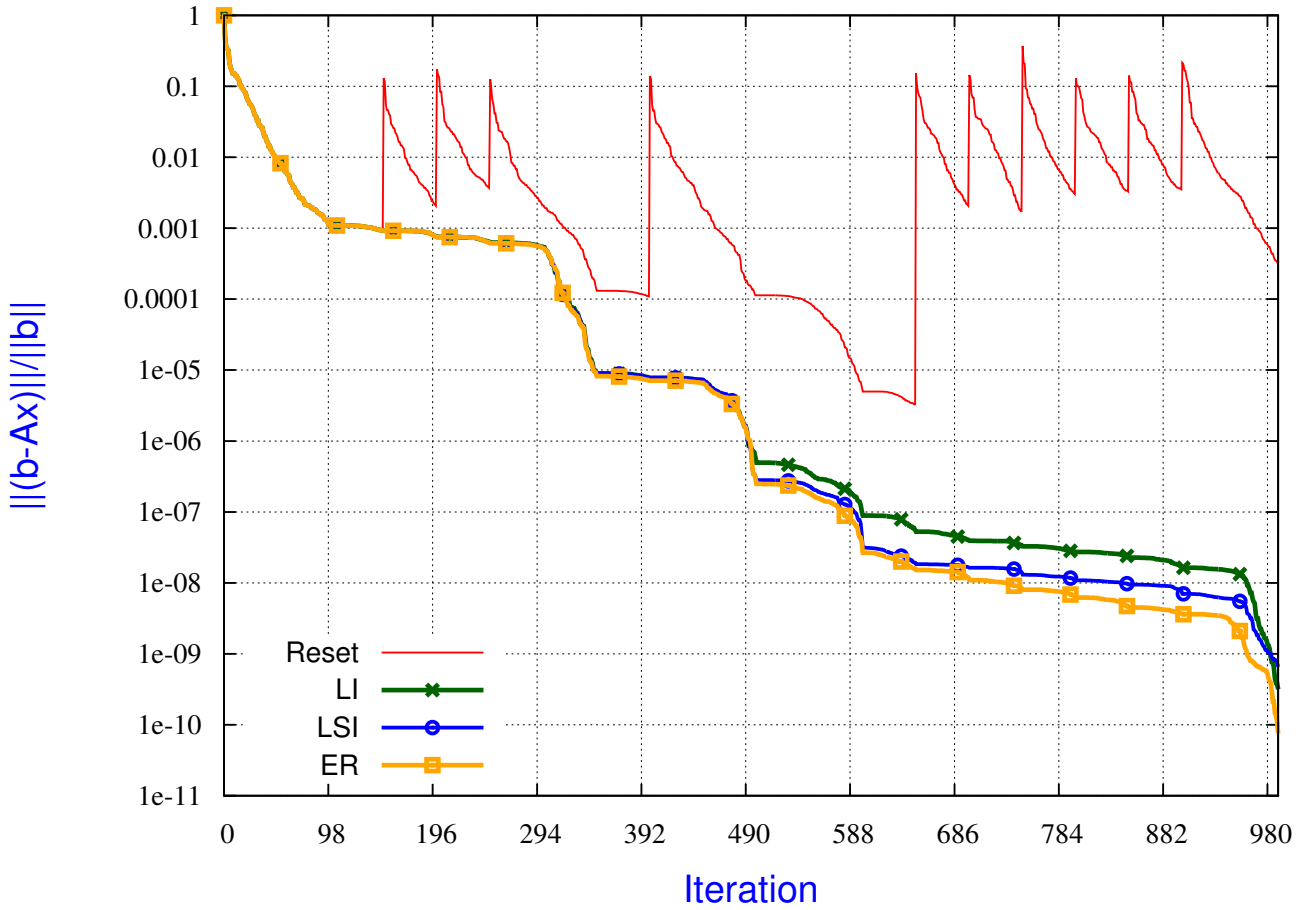
GMRES(100) - Averous/epb3 - 10 faults - **0.2%** data loss

Impact of lost data volume



GMRES(100) - Averous/epb3 - 10 faults - **0.8%** data loss

Impact of lost data volume

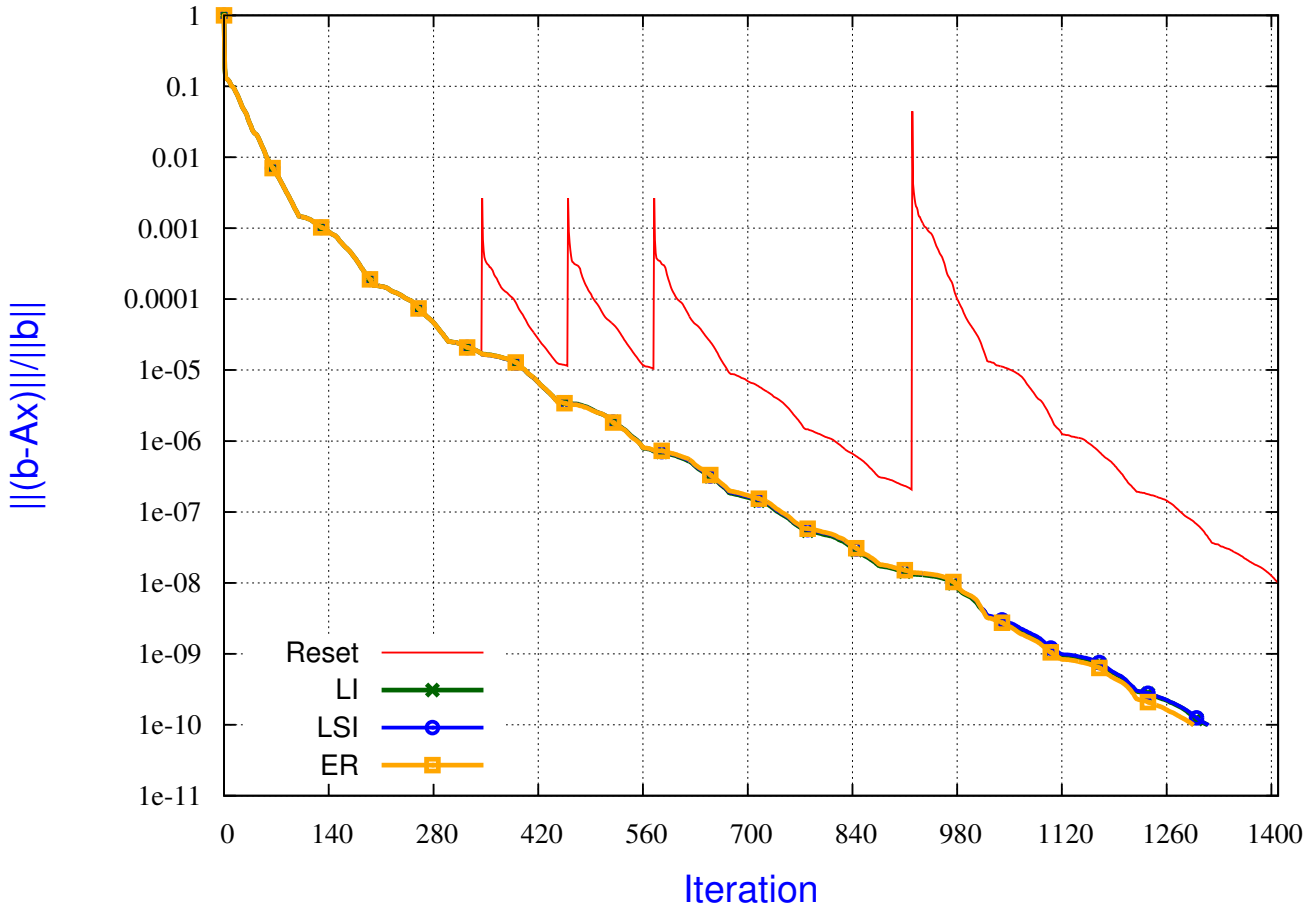


GMRES(100) - Averous/epb3 - 10 faults - **3%** data loss

Experimental results

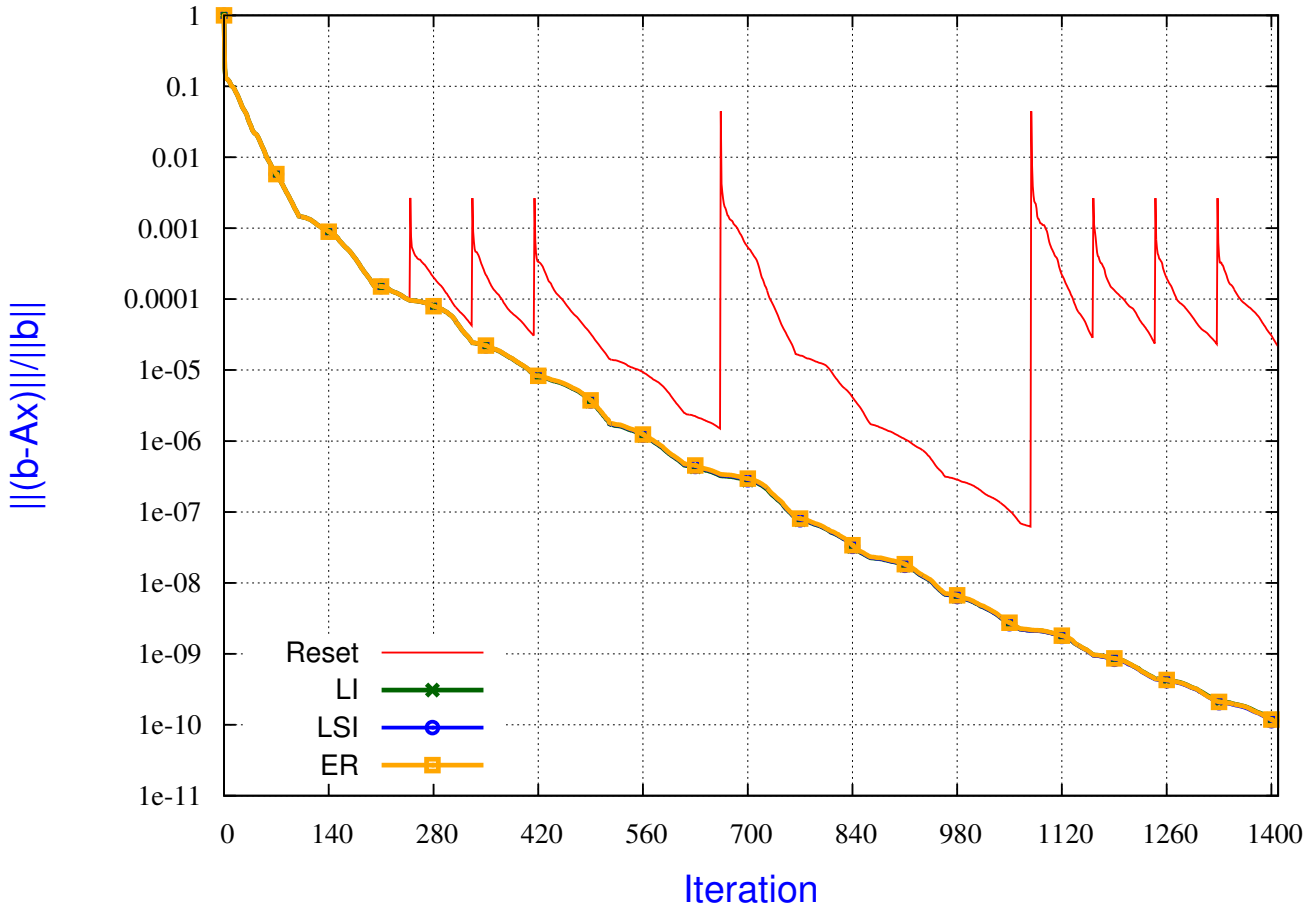
Fault rate impact

Impact of fault rate



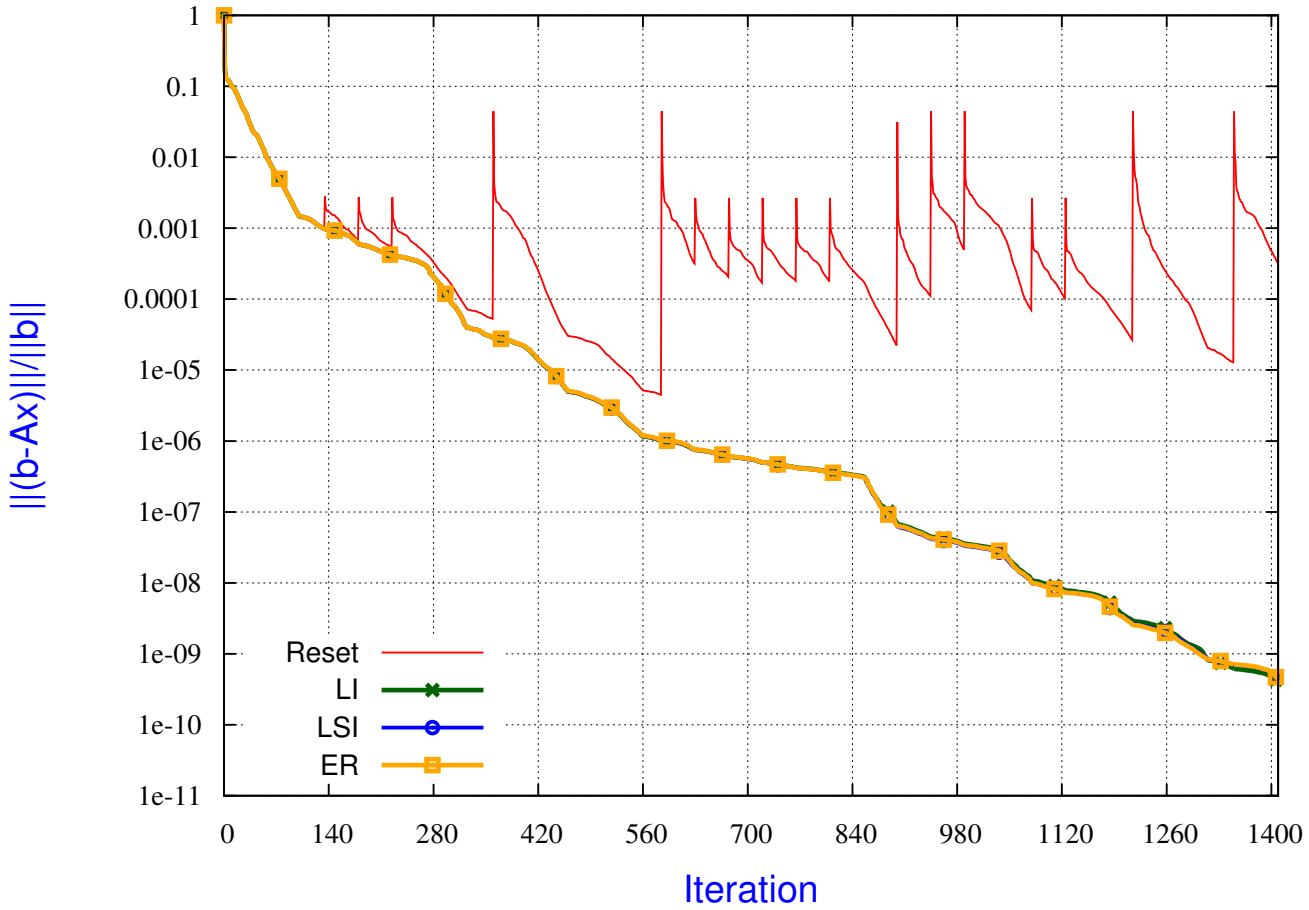
GMRES(100) - Kim - **4 faults** - 0.2% data loss

Impact of fault rate



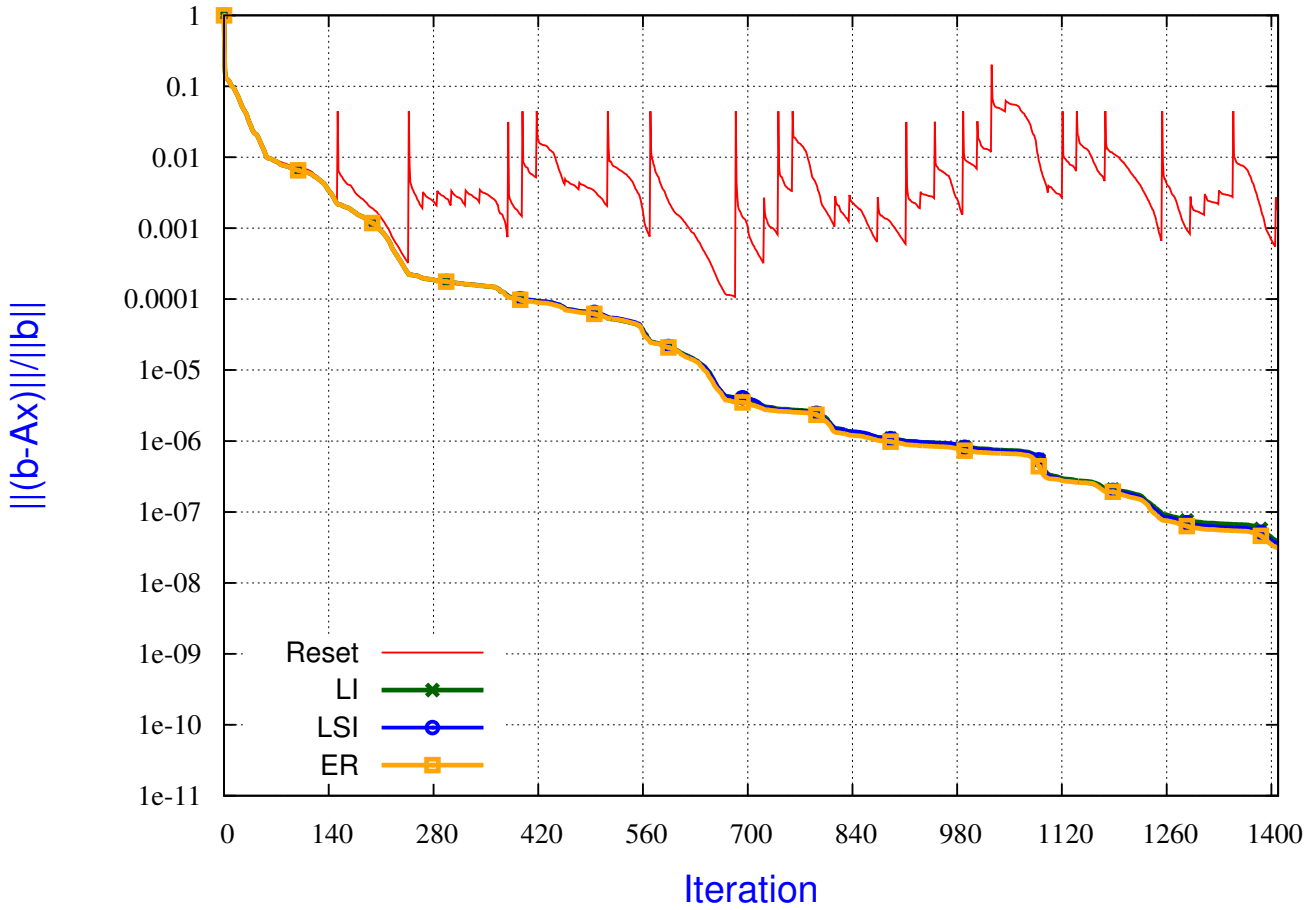
GMRES(100) - Kim - **8 faults** - 0.2% data loss

Impact of fault rate



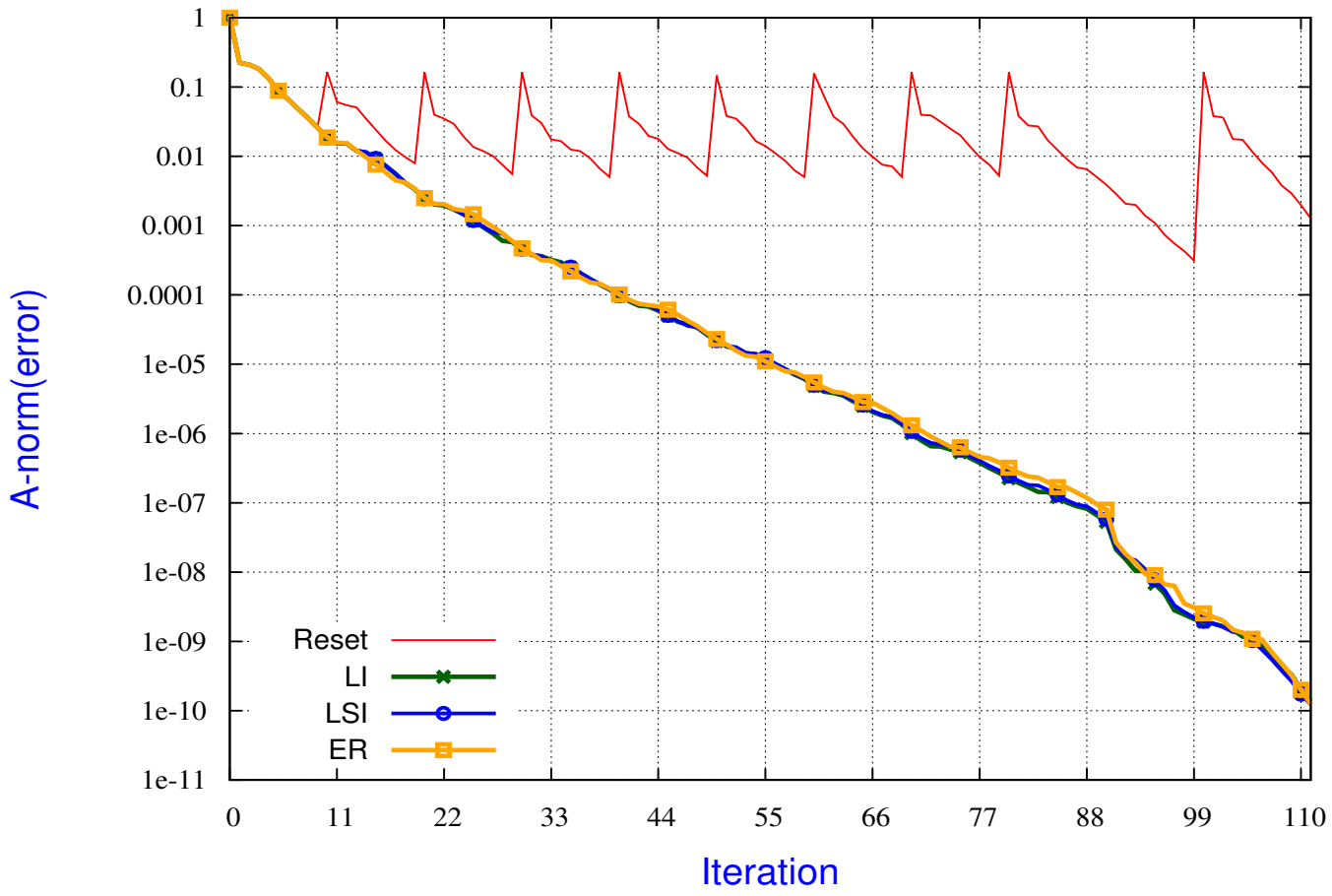
GMRES(100) - Kim - **17 faults** - 0.2% data loss

Impact of fault rate



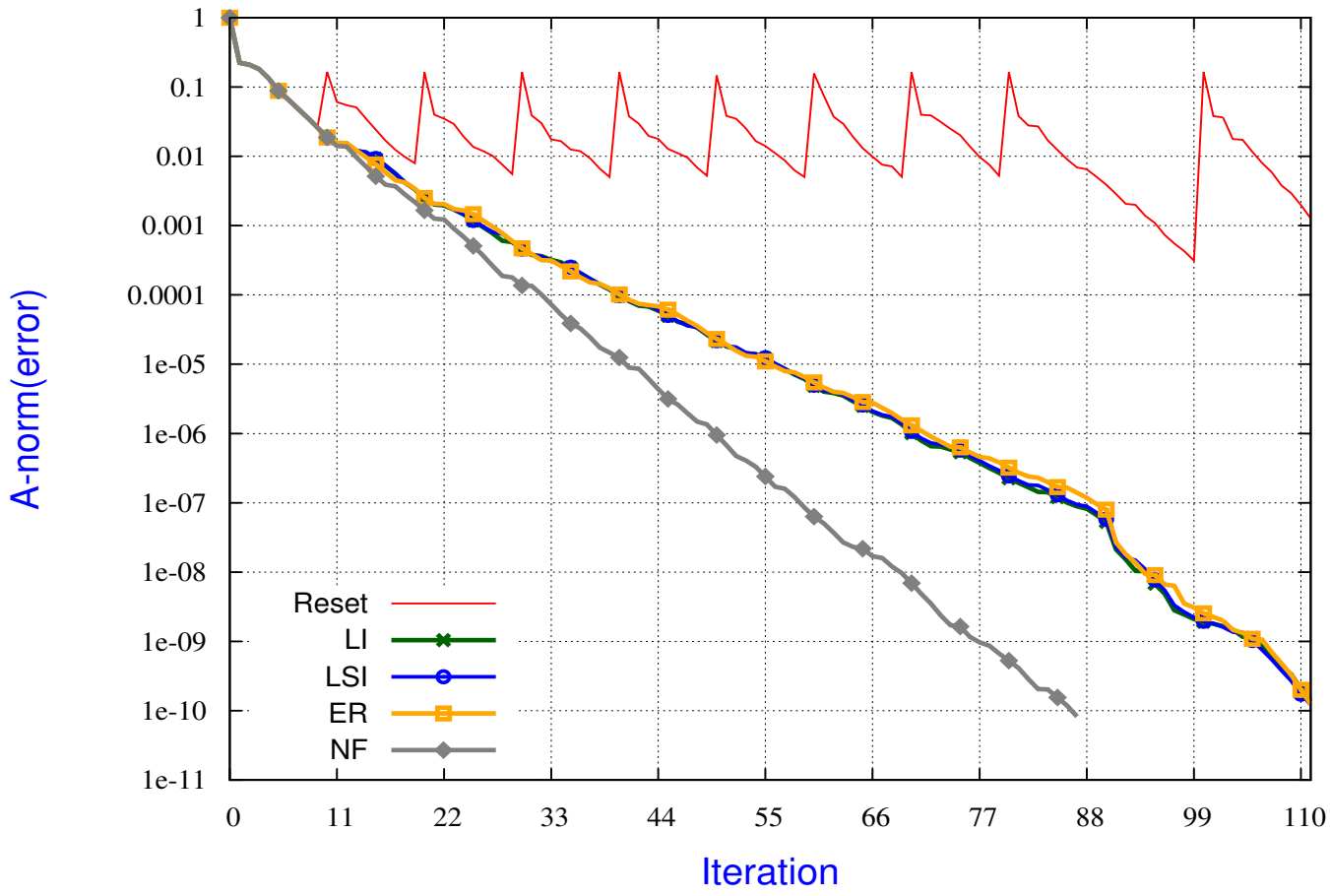
GMRES(100) - Kim - **40 faults** - 0.2% data loss

Penalty of restart strategy on PCG



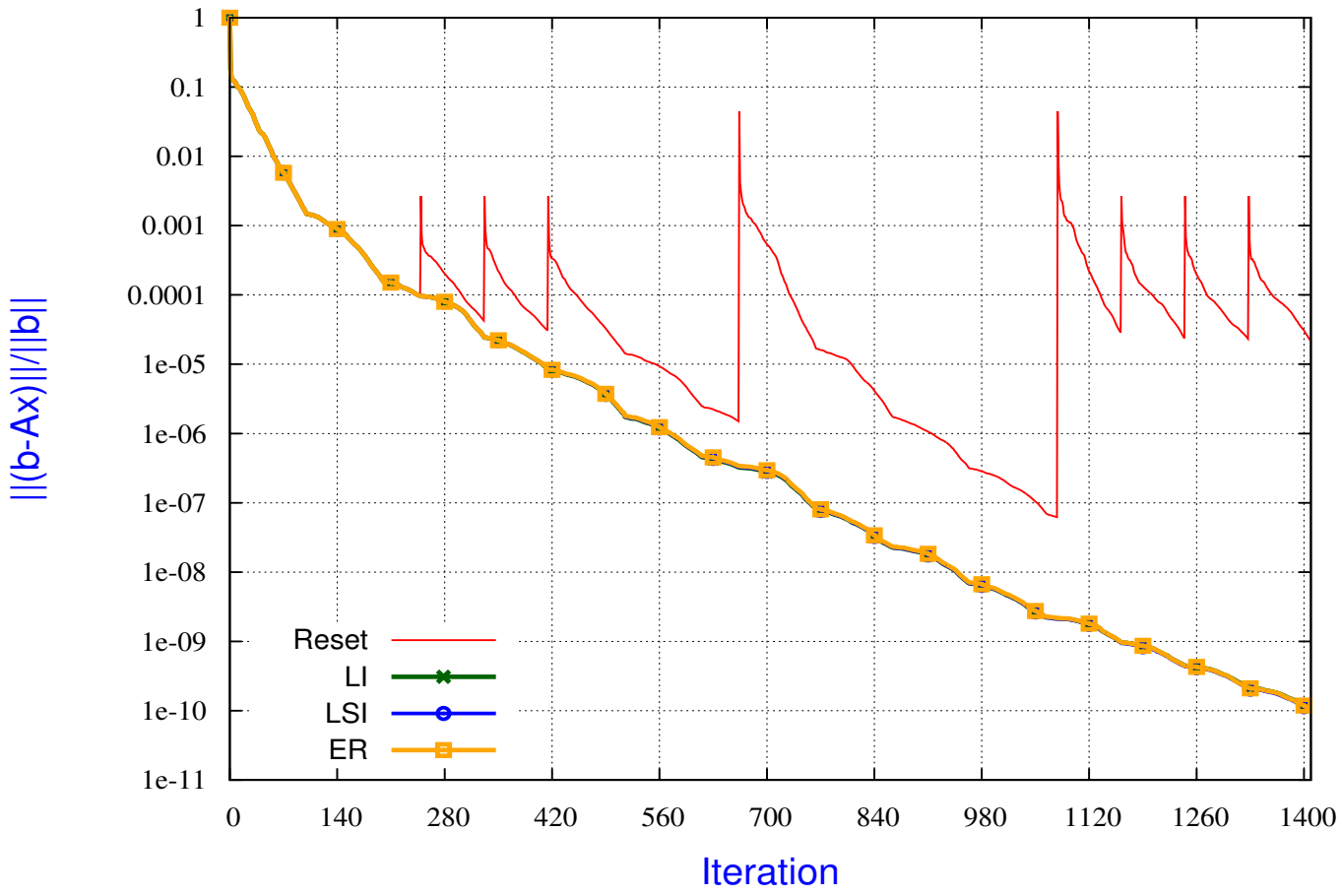
PCG (Cunningham/qa8fm - 9 faults)

Penalty of restart strategy on PCG



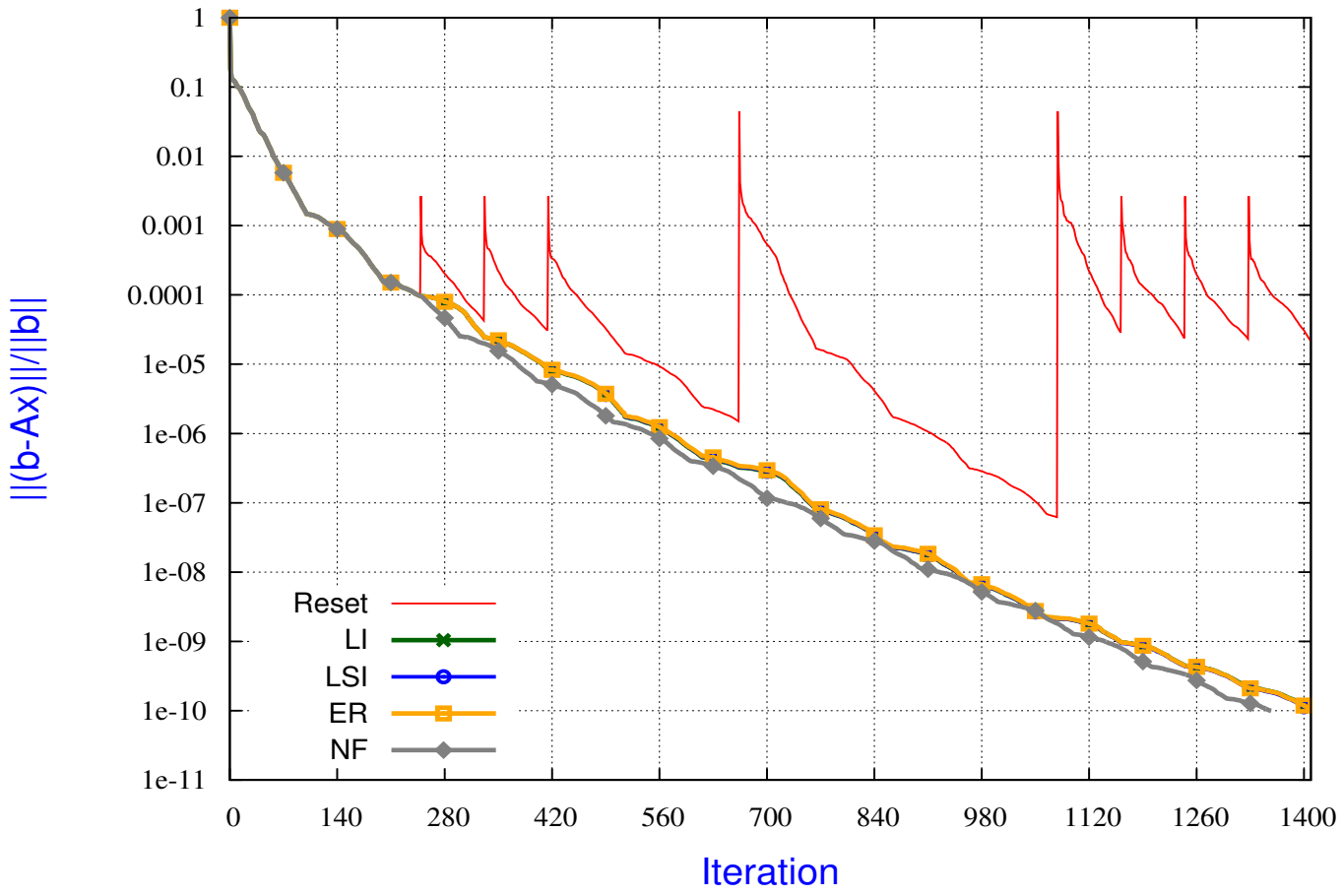
PCG (Cunningham/qa8fm - 9 faults)

Penalty of restart strategy on GMRES



Preconditioned GMRES(100) (Kim1 - 8 faults)

Penalty of restart strategy on GMRES

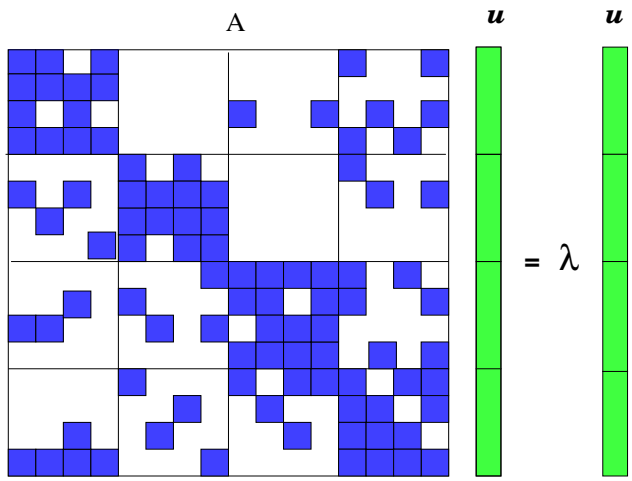


Preconditioned GMRES(100) (Kim1 - 8 faults)

Summary on resilient Krylov linear solvers

Interpolation-Restart strategies key features:

- Make sense: key properties of CG and GMRES are preserved
- The restarting effect remains reasonable within the restarted GMRES context
- Robust even with a high fault rate and a high volume of lost data
- No fault implies no overhead
- Extended to multiple faults



$\mathcal{A}u = \lambda u$ with $u \neq 0$, where $\mathcal{A} \in \mathbb{C}^{n \times n}$, $u \in \mathbb{C}^n$, and $\lambda \in \mathbb{C}$

- λ : eigenvalue
- u : eigenvector
- (λ, u) : eigenpair

Two classes of methods

- Fixed Point Methods (Power Method, Subspace iteration)
- Subpace Methods (Jacobi-Davidson, Arnoldi, IRA/Krylov Schur)

Interpolation strategies

Fault in eigenproblem

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \lambda \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

Linear Interpolation (LI)

$$\left(A_{11} - \lambda I_1\right) u_1 = -A_{12} u_2$$

Least Squares Interpolation (LSI)

$$\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} x_1 + \begin{pmatrix} A_{21} \\ A_{22} \end{pmatrix} u_2 = \lambda \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

$$u_1 = \underset{u}{argmin} \left\| \begin{pmatrix} A_{11} - \lambda I_1 \\ A_{21} \end{pmatrix} u + \begin{pmatrix} A_{12} \\ A_{22} - \lambda I_2 \end{pmatrix} u_2 \right\|_2$$

Interpolation strategies

Fault in eigenproblem

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} ? \\ u_2 \end{pmatrix} = \lambda \begin{pmatrix} ? \\ u_2 \end{pmatrix}$$

u_1 entries are lost

λ is replicated

Linear Interpolation (LI)

$$(A_{11} - \lambda I_1)\, u_1 = -A_{12} u_2$$

Least Squares Interpolation (LSI)

$$\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} x_1 + \begin{pmatrix} A_{21} \\ A_{22} \end{pmatrix} u_2 = \lambda \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

$$u_1 = \underset{u}{argmin} \left\| \begin{pmatrix} A_{11} - \lambda I_1 \\ A_{21} \end{pmatrix} u + \begin{pmatrix} A_{12} \\ A_{22} - \lambda I_2 \end{pmatrix} u_2 \right\|_2$$

Eigensolvers revisited

- Basic subspace iteration method
- Subspace iteration with polynomial acceleration
- Arnoldi method
- Implicitly Restarted Arnoldi method
- Jacobi-Davidson method (JDQR)

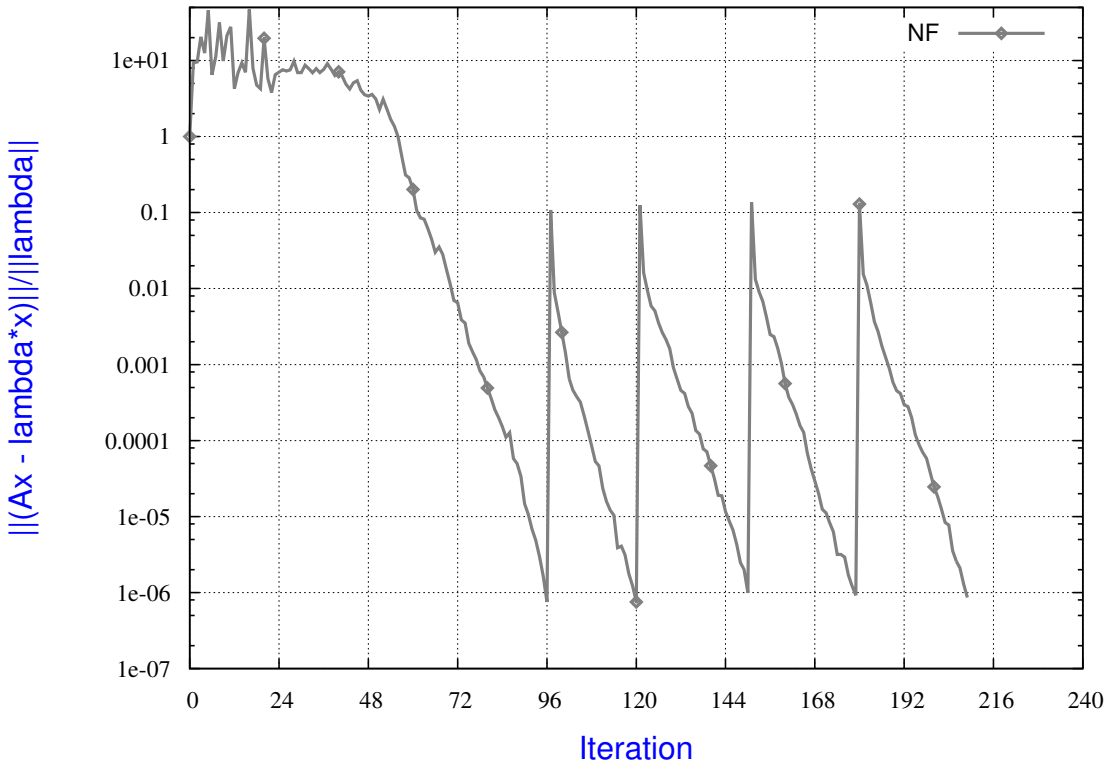
Eigensolvers revisited

- Basic subspace iteration method
- Subspace iteration with polynomial acceleration
- Arnoldi method
- Implicitly Restarted Arnoldi method
- Jacobi-Davidson method (JDQR)

Jacobi-Davidson at a glance

JDQR algorithm

- Compute a few **nev** eigenpairs associated with eigenvalues closed to a target τ
- Perform a partial Schur decomposition $AZ_{nconv} = Z_{nconv}T_{nconv}$
- The next Schur vectors are searched in the V_k search space



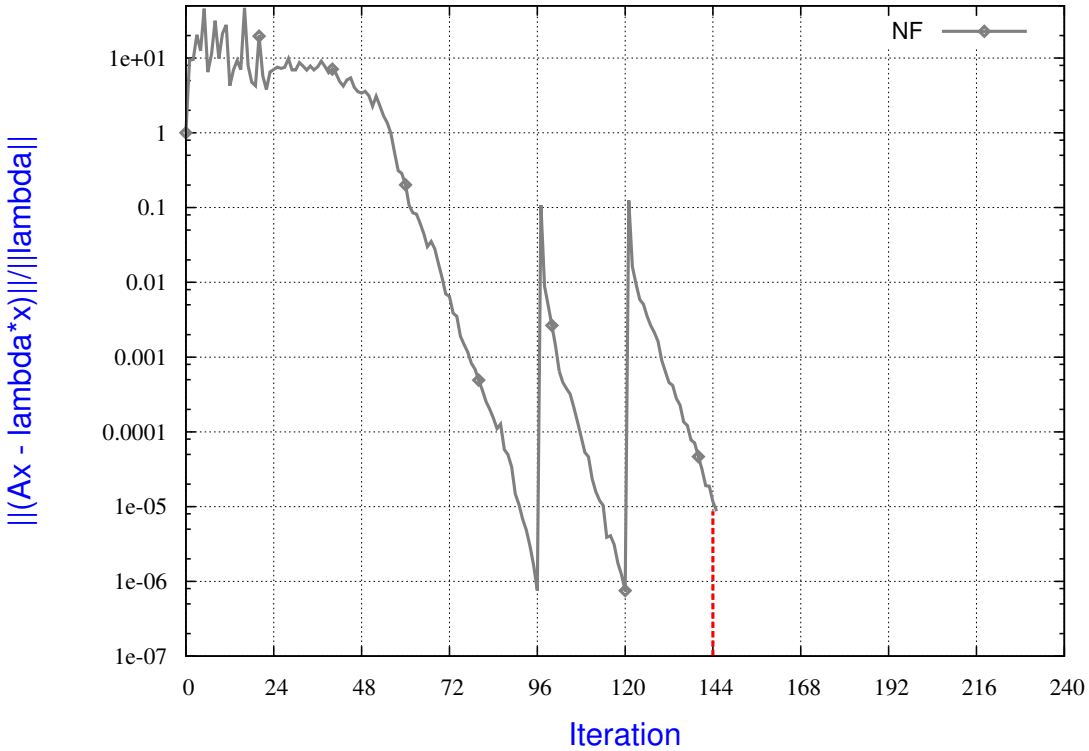
Resilient Jacobi-Davidson algorithm

Converged Schur vectors

- $AZ_{nconv} = Z_{nconv} T_{nconv}$
- Interpolate the **nconv** converged vectors

Search space

- $C_k = V_k^H \mathcal{A} V_k$
- Interpolate a few **s** best candidates



- JD is restarted with a search space of **nconv + s** vectors
- Flexibility to select the dimension of the search space for restarting

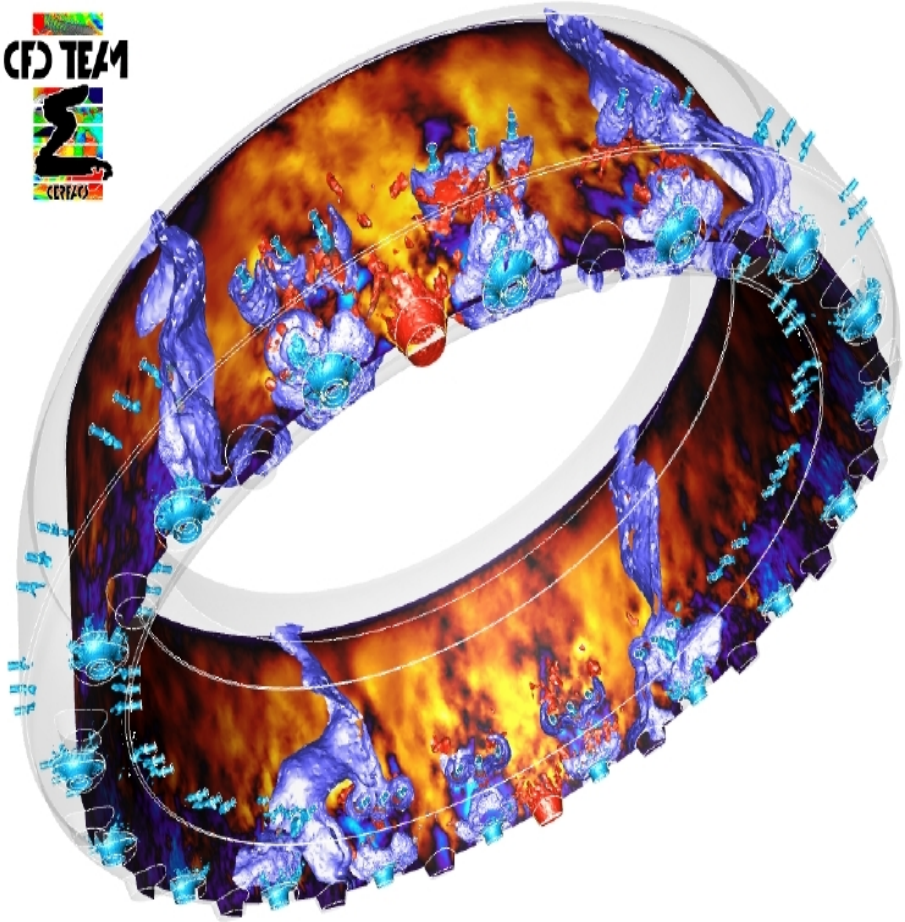
Experimental framework

Thermoacoustic instabilities in combustion chambers [Image courtesy of CERFACS]

Damaged rocket chamber from combustion instabilities

Experimental framework

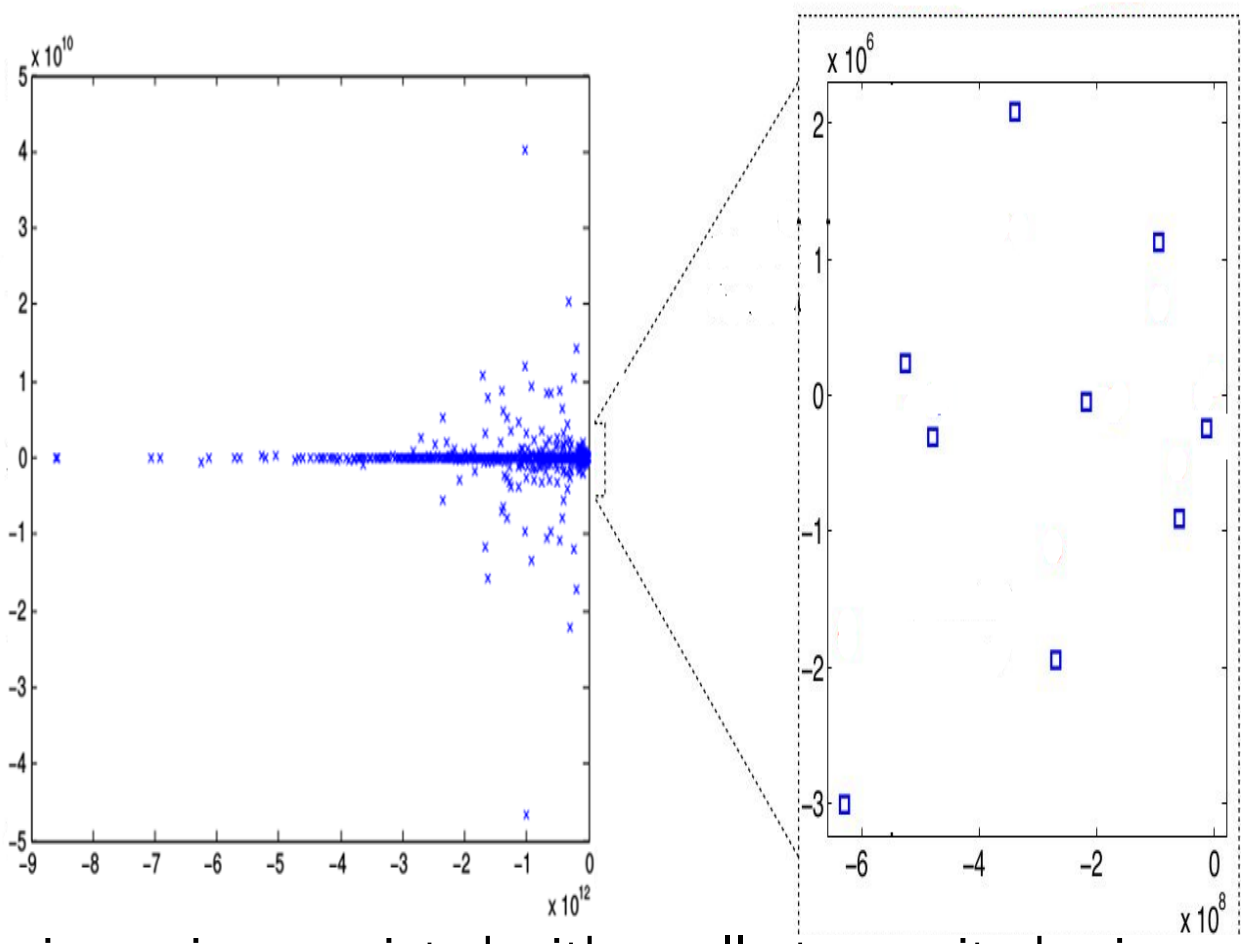
Thermoacoustic instabilities in combustion chambers [Image courtesy of CERFACS]



Simulation of combustion instabilities in annular chambers

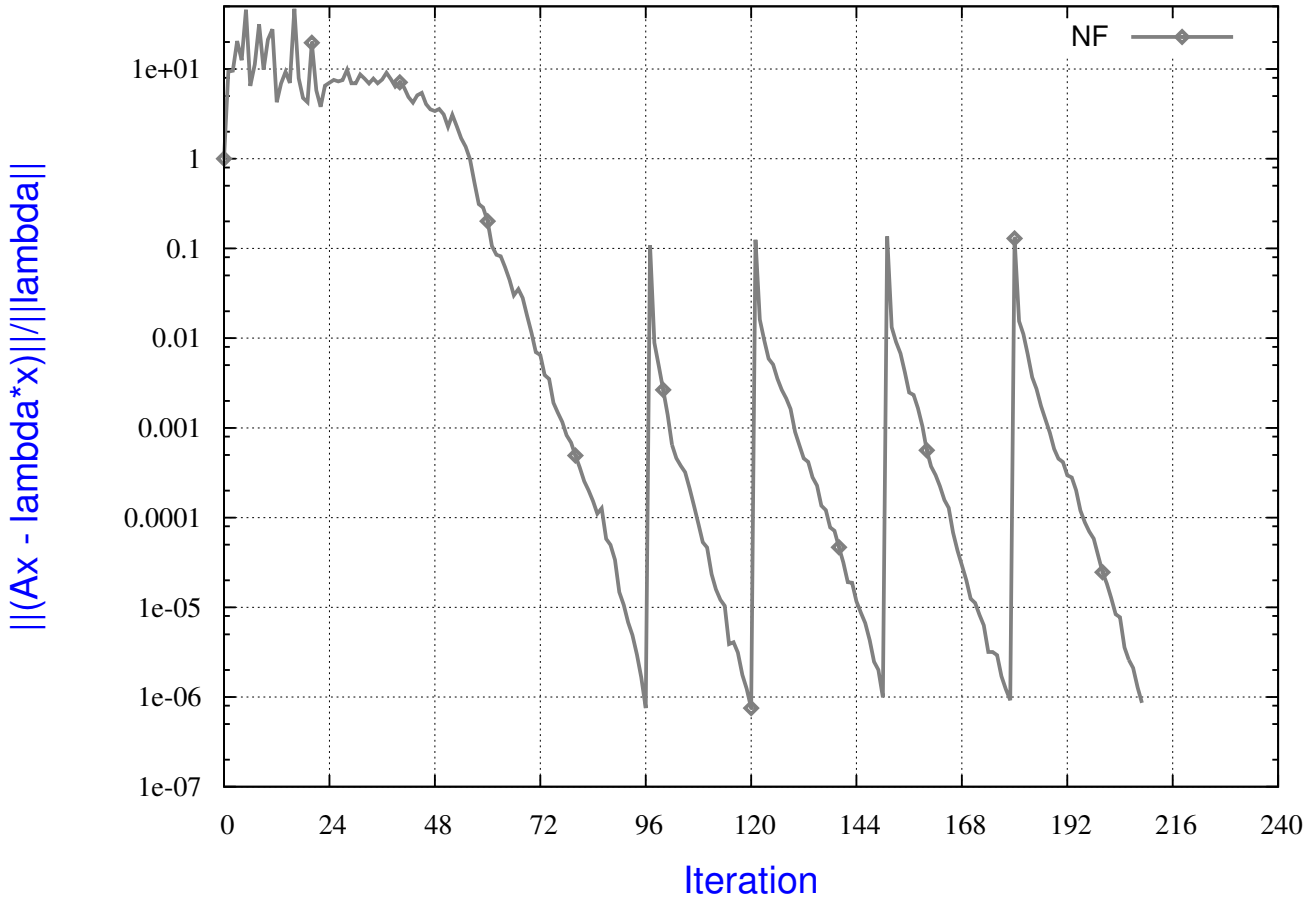
Experimental framework

Thermoacoustic instabilities in combustion chambers



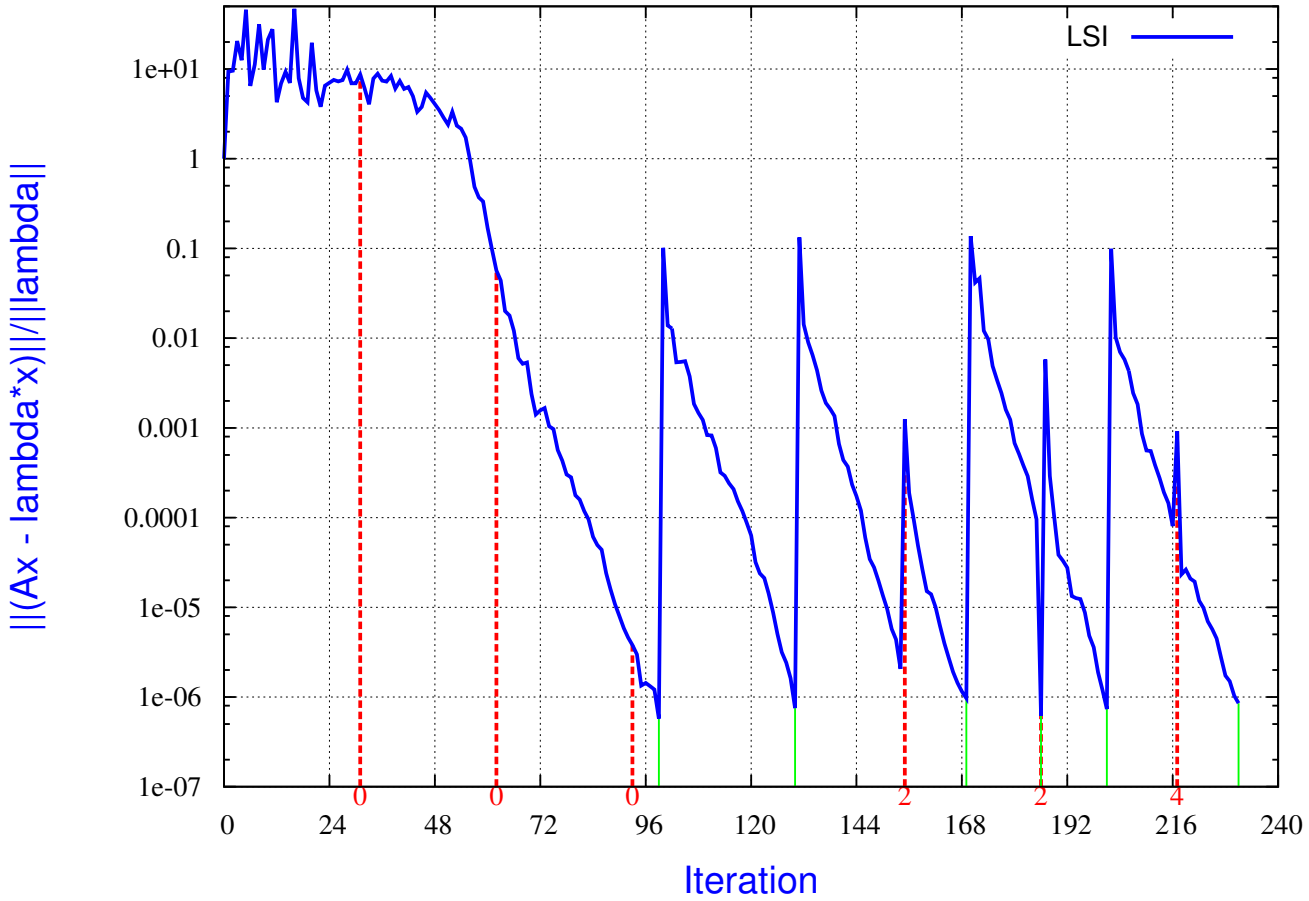
Compute eigenpairs associated with smallest magnitude eigenvalues laying in the periphery of the spectrum

Experimental results



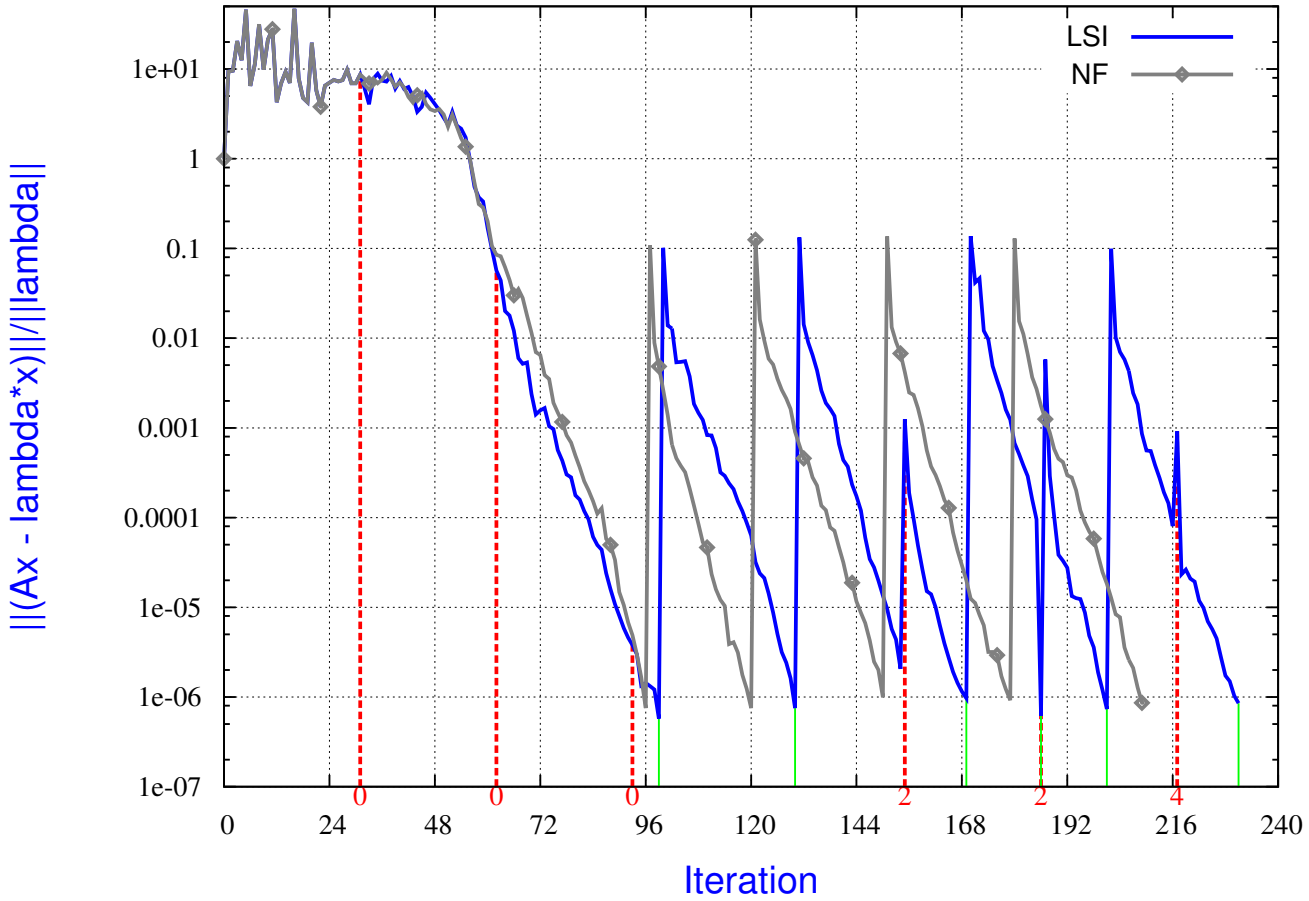
Convergence of 5 eigenpairs associated with smallest magnitude eigenvalues

Experimental results



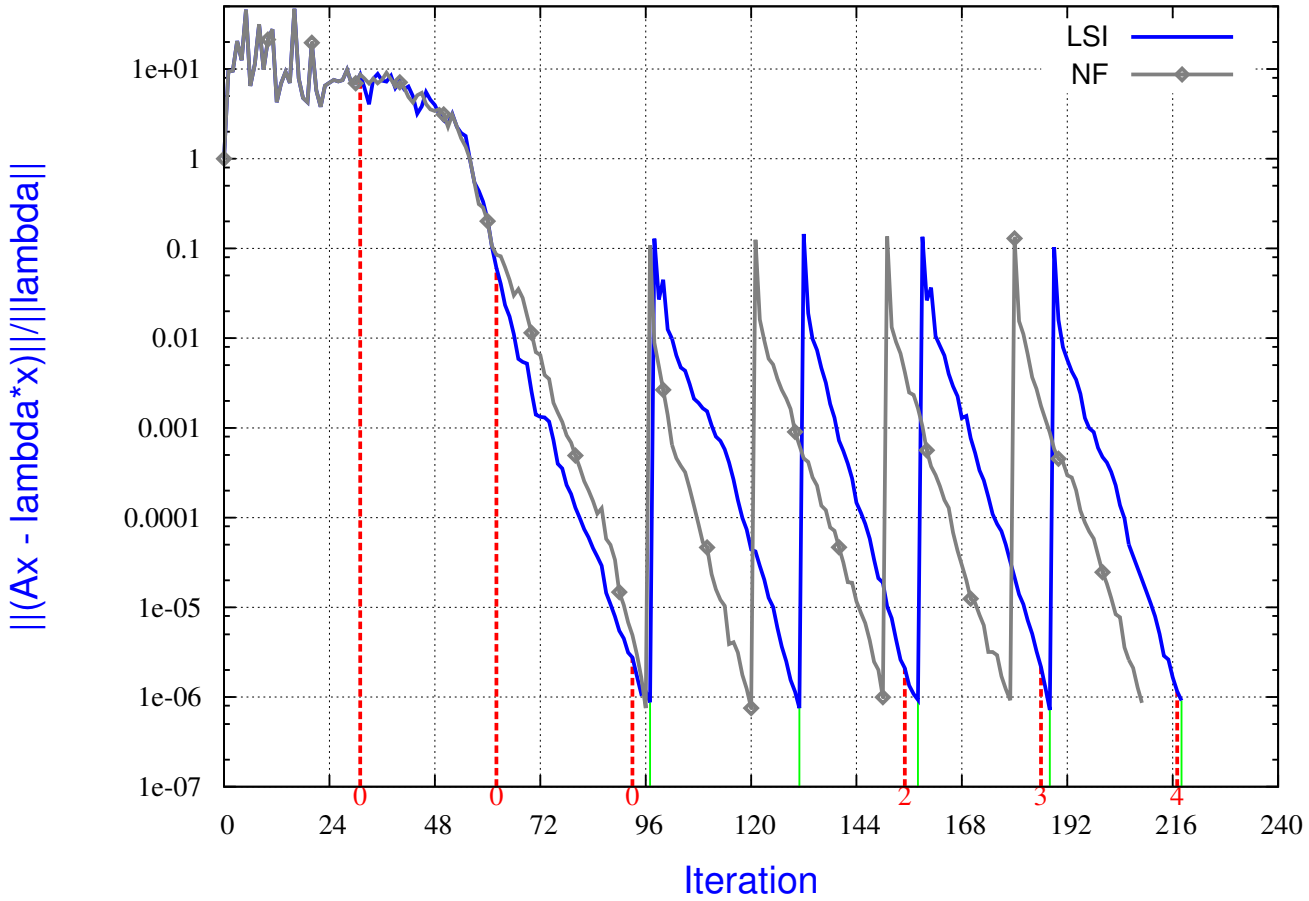
Restart with **nconv vectors** from converged Schur vectors combined with **5 best candidates** from the search space (6 faults)

Experimental results



Restart with **nconv vectors** from converged Schur vectors combined with **5 best candidates** from the search space (6 faults)

Experimental results



Impact of **keeping the best Schur vector candidate** in the search space after a fault (6 faults)

Summary on resilient eigensolvers

- Specific Interpolation-Restart strategies for each eigensolver
  - Basic subspace iteration method
  - Subspace iteration with polynomial acceleration
  - Arnoldi method
  - IRAM
  - Jacobi-Davidson
- More flexibility in eignesolver than in Krylov linear solvers

Outline

1. Hard fault: Interpolation-Restart strategies
   - In Krylov subspace linear solvers
   - In revisited eigensolvers

2. Soft errors in Conjugate Gradient (preliminary)

Aim of this study

Context

Soft Errors in Conjugate Gradient (CG)

Question 1

Impact of soft errors on convergence of CG

Question 2

Reliability of numerical detection mechanisms ?

Question 3

Robust numerical recovery schemes ?

Method to address question 1

Experimental study with basic statistics

Aim of this study

Context

Soft Errors in Conjugate Gradient (CG)

Question 1

Impact of soft errors on convergence of CG

Question 2

Reliability of numerical detection mechanisms ?

Question 3

Robust numerical recovery schemes ?

Method to address question 1

Experimental study with basic statistics

Aim of this study

Context

Soft Errors in Conjugate Gradient (CG)

Question 1

Impact of soft errors on convergence of CG

Question 2

Reliability of numerical detection mechanisms ?

Question 3

Robust numerical recovery schemes ?

Method to address question 1

Experimental study with basic statistics

Aim of this study

Context

Soft Errors in Conjugate Gradient (CG)

Question 1

Impact of soft errors on convergence of CG

Question 2

Reliability of numerical detection mechanisms ?

Question 3

Robust numerical recovery schemes ?

Method to address question 1

Experimental study with basic statistics

Aim of this study

Context

Soft Errors in Conjugate Gradient (CG)

Question 1

Impact of soft errors on convergence of CG

Question 2

Reliability of numerical detection mechanisms ?

Question 3

Robust numerical recovery schemes ?

Method to address question 1

Experimental study with basic statistics

Fault injection

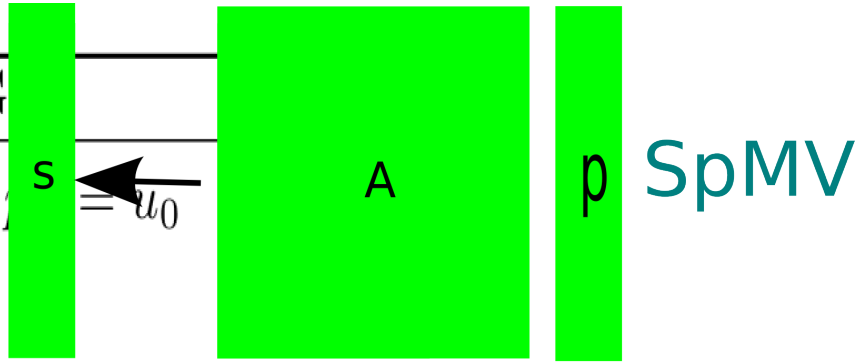
Algorithm 1 Preconditoned CG

1: $r_0 := b - Ax_0$; $u_0 := M^{-1}r_0$; $p_0 := u_0$
2: **for** $i = 0, \ldots$ **do**
3: $\quad s := Ap_i$
4: $\quad \alpha := (r_i, u_i)/(s, p_i)$
5: $\quad x_{i+1} := x_i + \alpha p_i$
6: $\quad r_{i+1} := r_i - \alpha s$
7: $\quad u_{i+1} := M^{-1}r_{i+1}$
8: $\quad \beta := (r_{i+1}, u_{i+1})/(r_i, u_i)$
9: $\quad p_{i+1} := u_{i+1} + \beta p_i$
10: **end for**

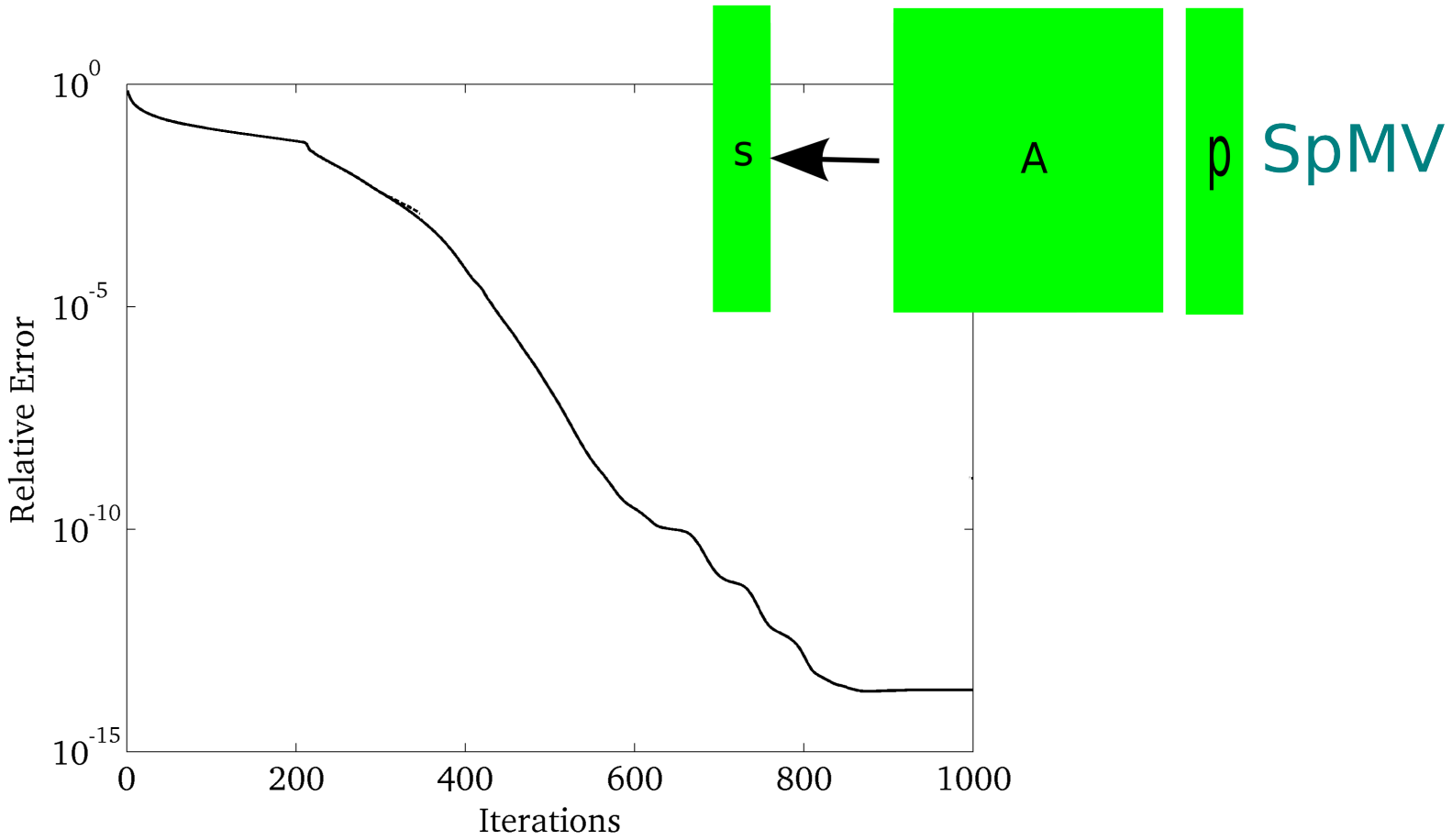
Fault injection

Algorithm 1 Preconditoned CG

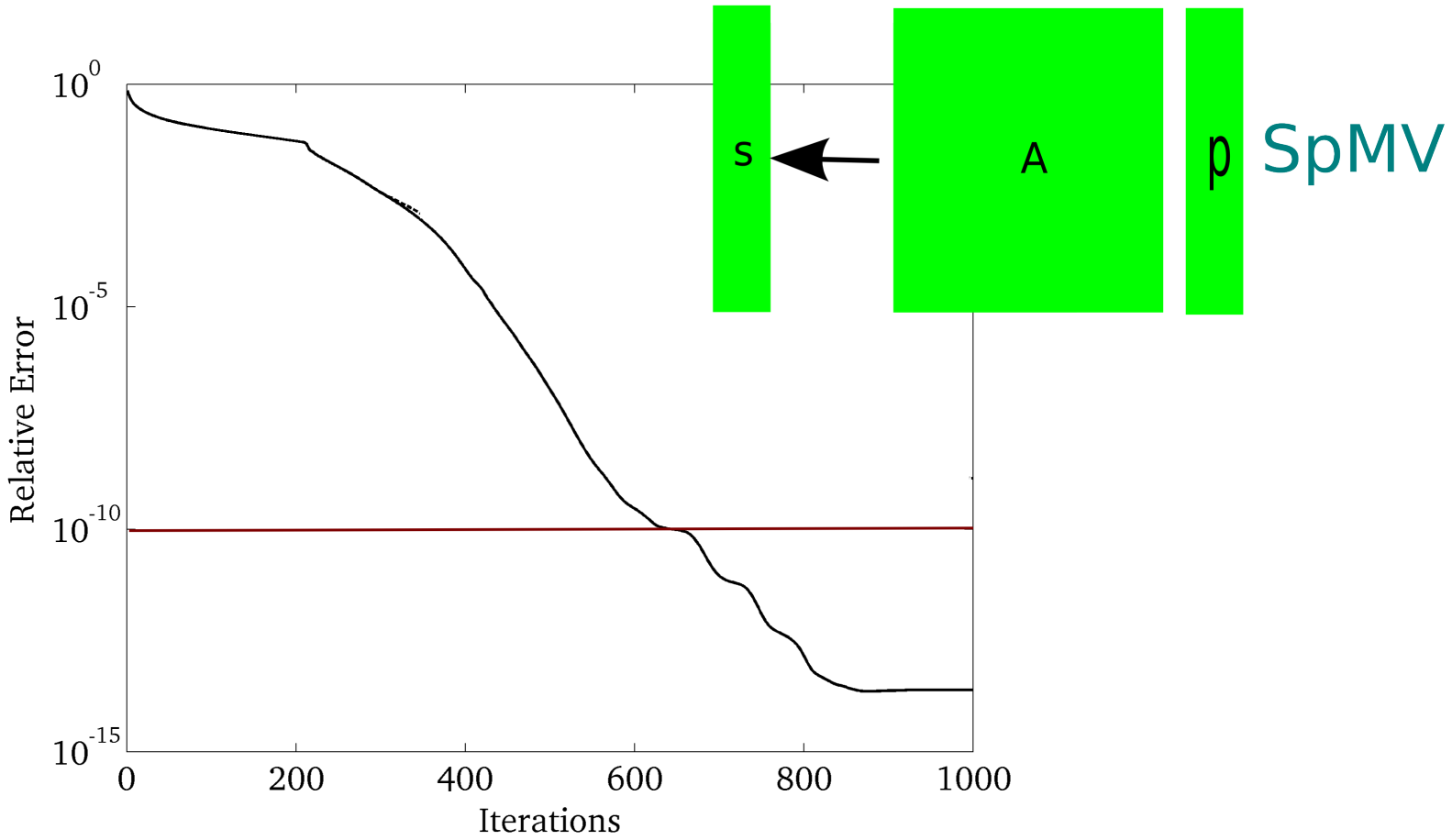
1: $r_0 := b - Ax_0$; $u_0 := M^{-1}r_0$; $p_0 := u_0$
2: **for** $i = 0, \ldots$ **do**
3: $\quad s := Ap_i$
4: $\quad \alpha := (r_i, u_i)/(s, p_i)$
5: $\quad x_{i+1} := x_i + \alpha p_i$
6: $\quad r_{i+1} := r_i - \alpha s$
7: $\quad u_{i+1} := M^{-1} r_{i+1}$
8: $\quad \beta := (r_{i+1}, u_{i+1})/(r_i, u_i)$
9: $\quad p_{i+1} := u_{i+1} + \beta p_i$
10: **end for**



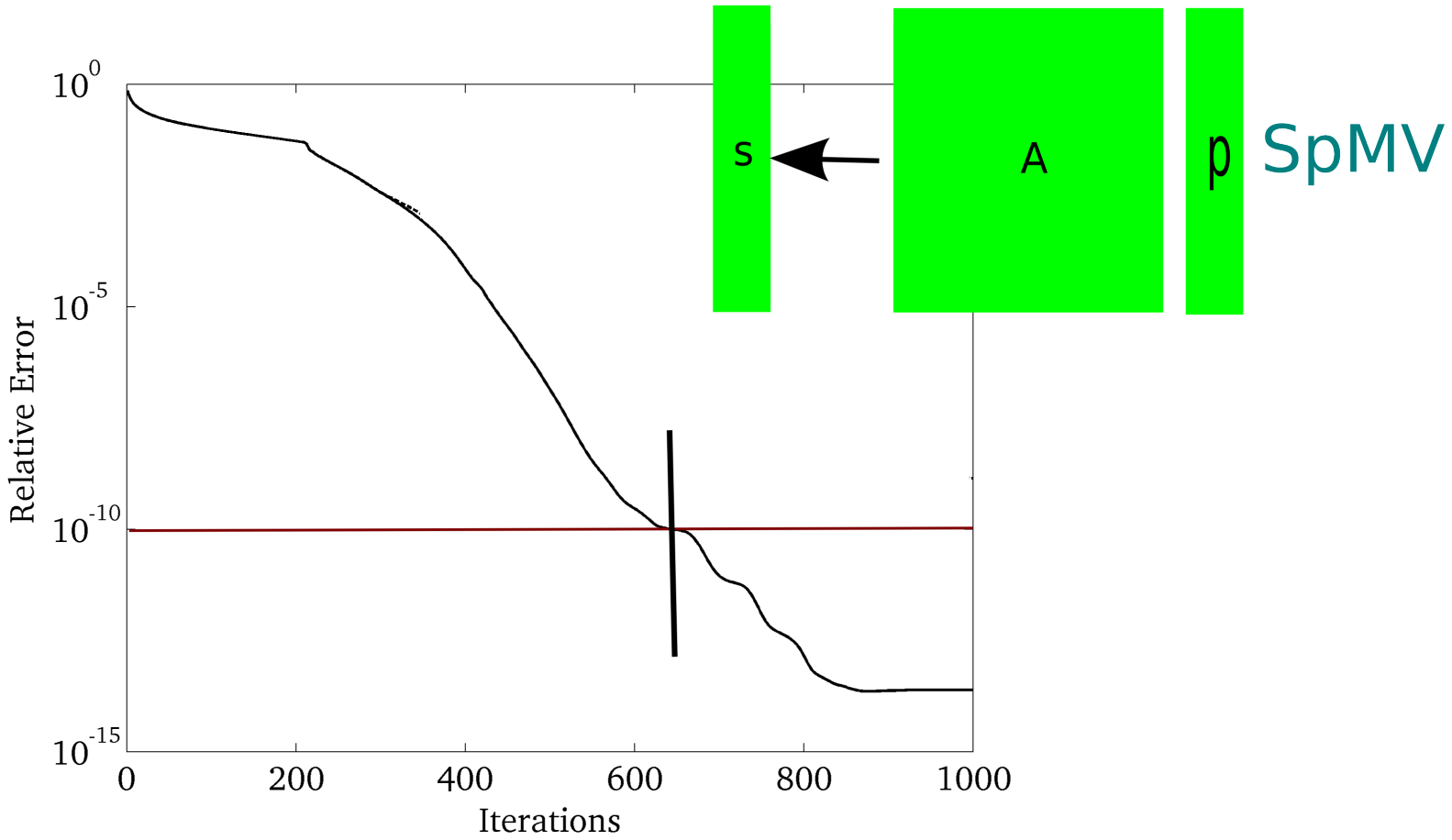
Fault injection



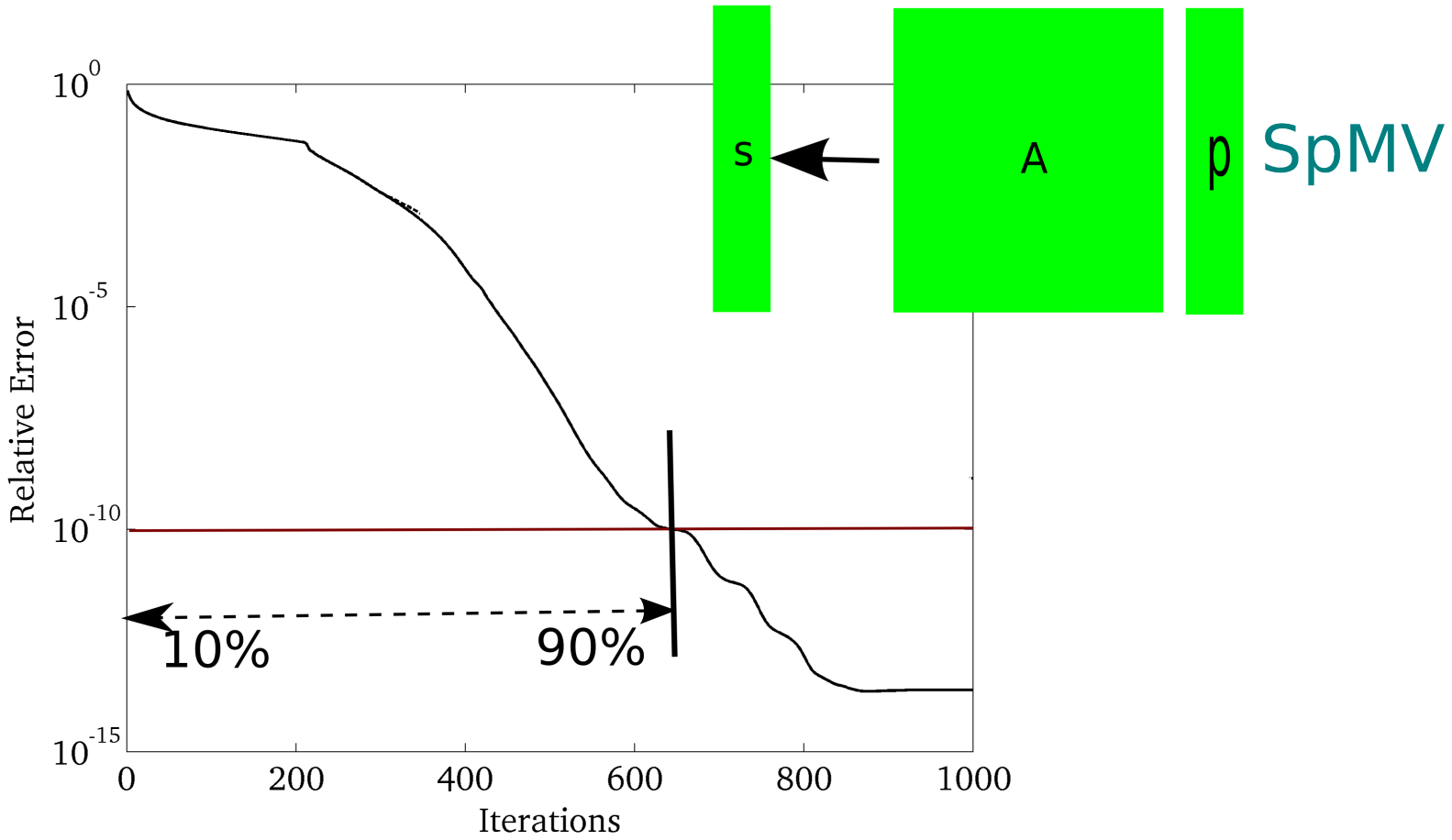
Fault injection



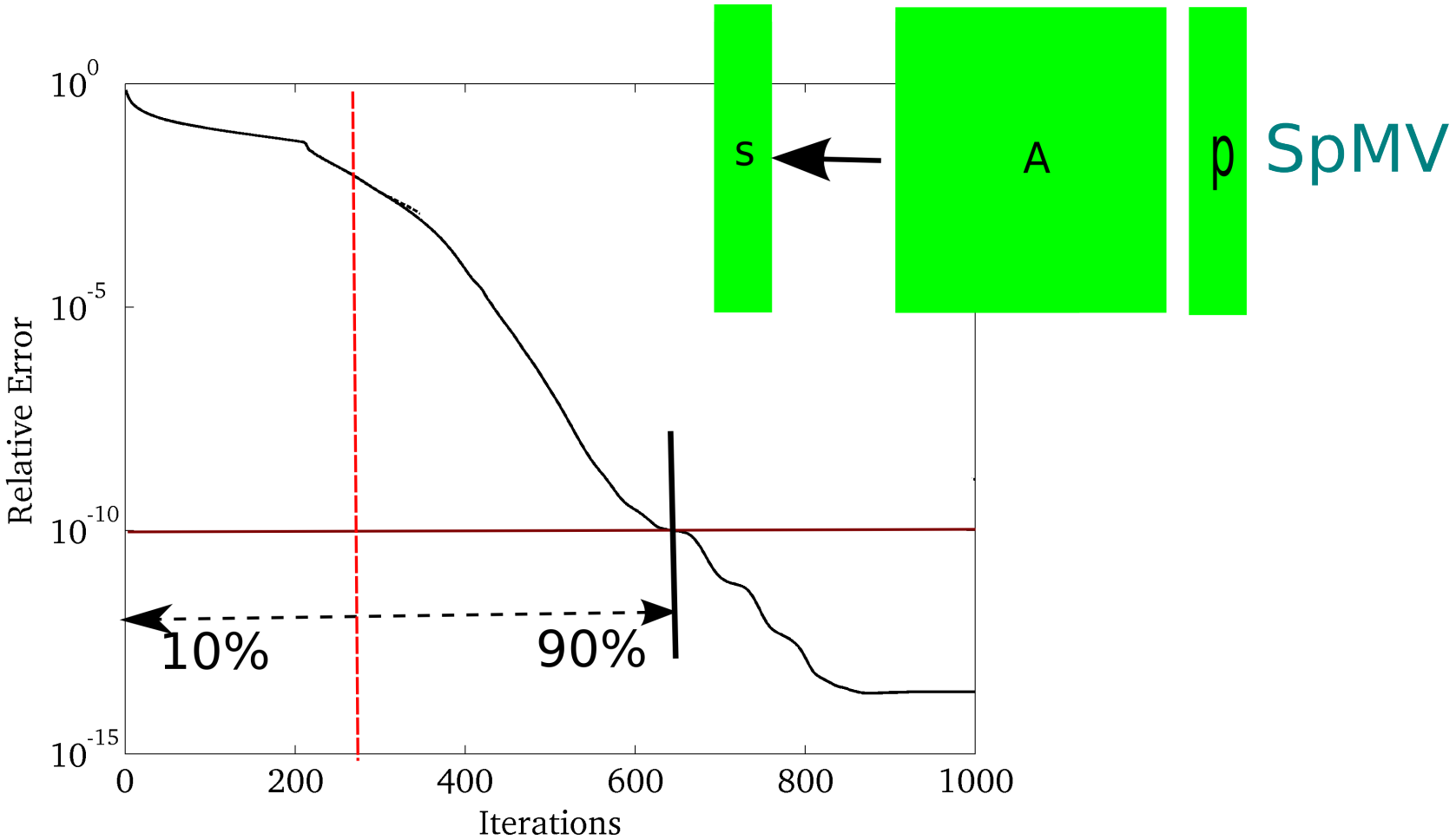
Fault injection



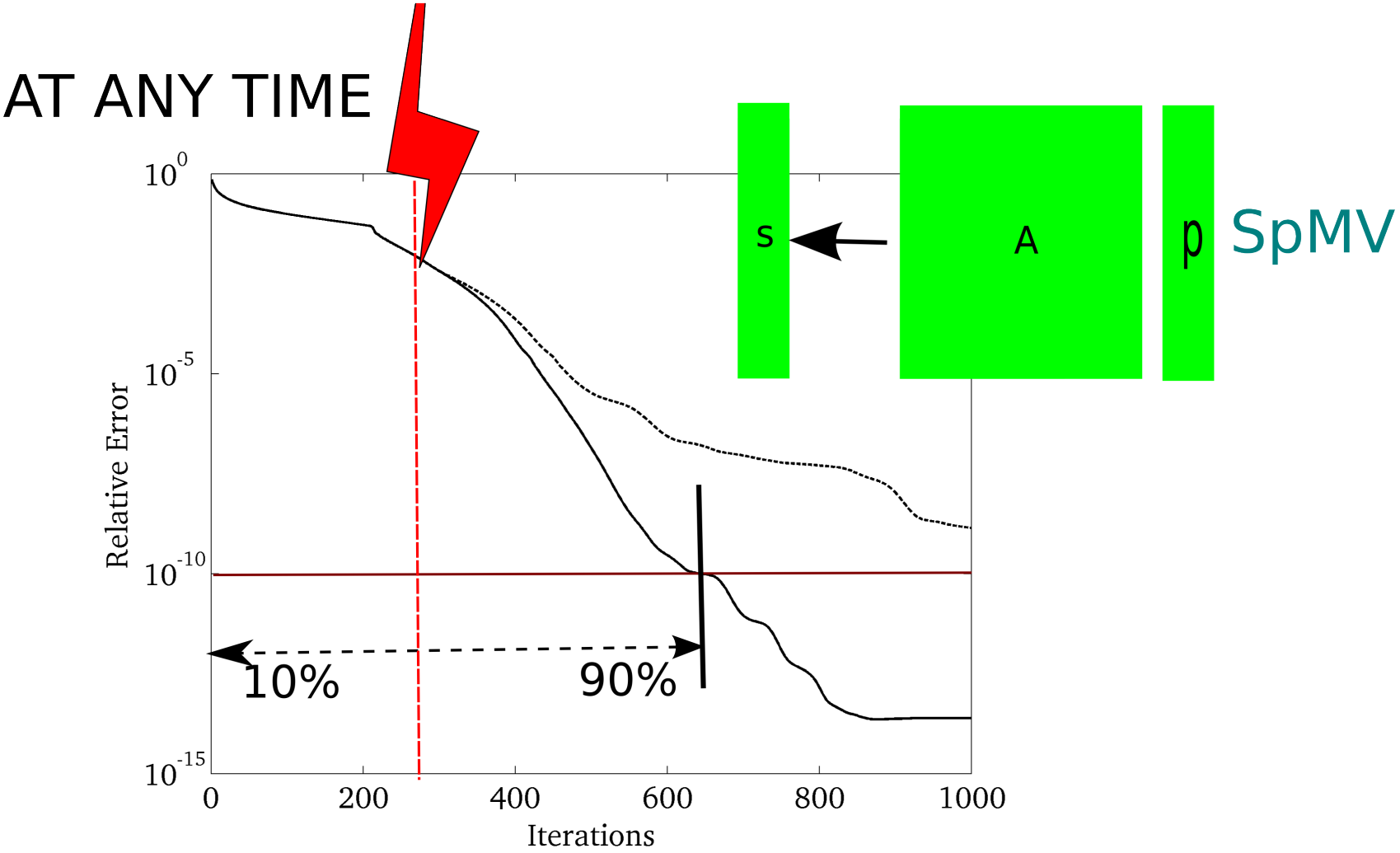
Fault injection



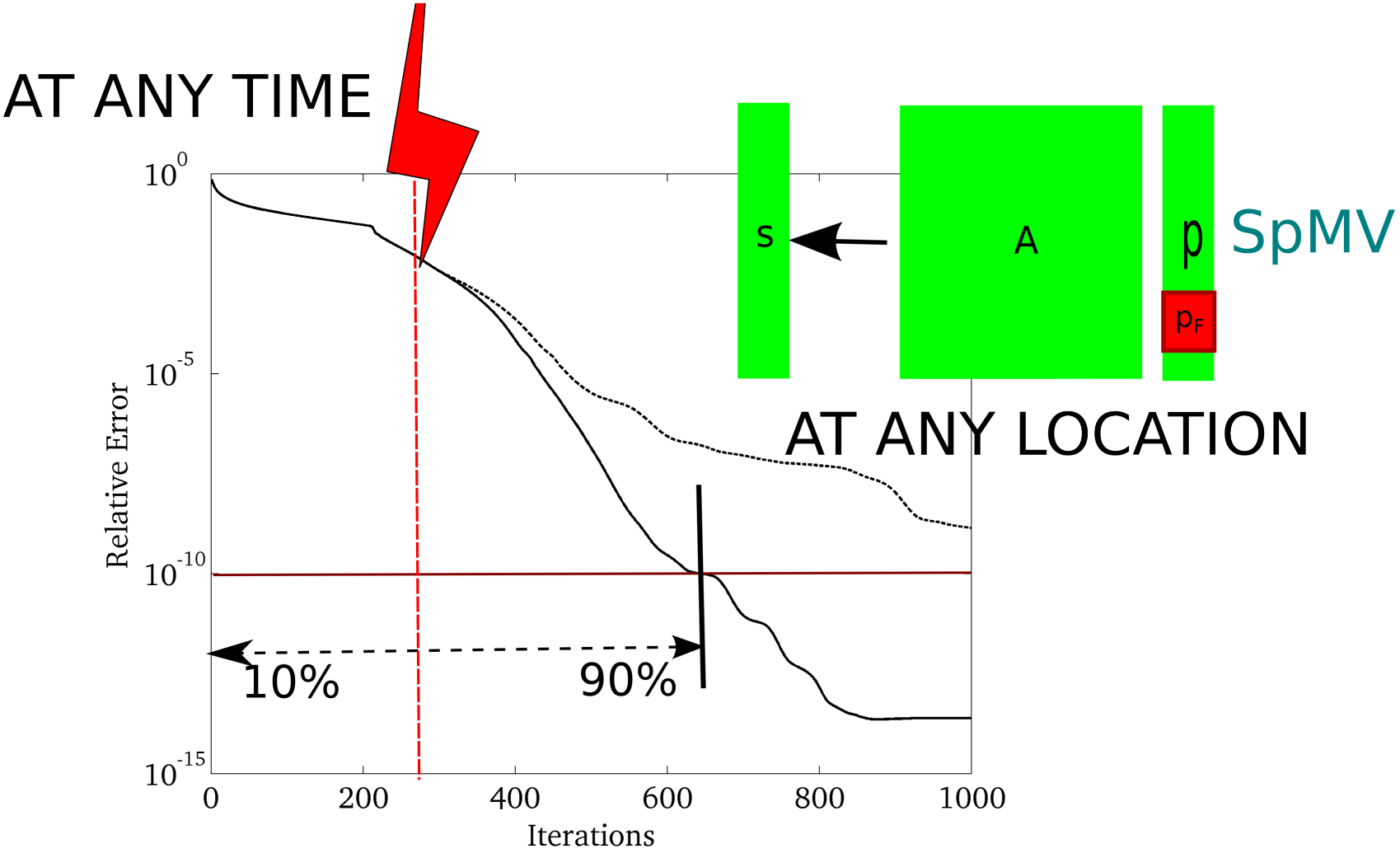
Fault injection



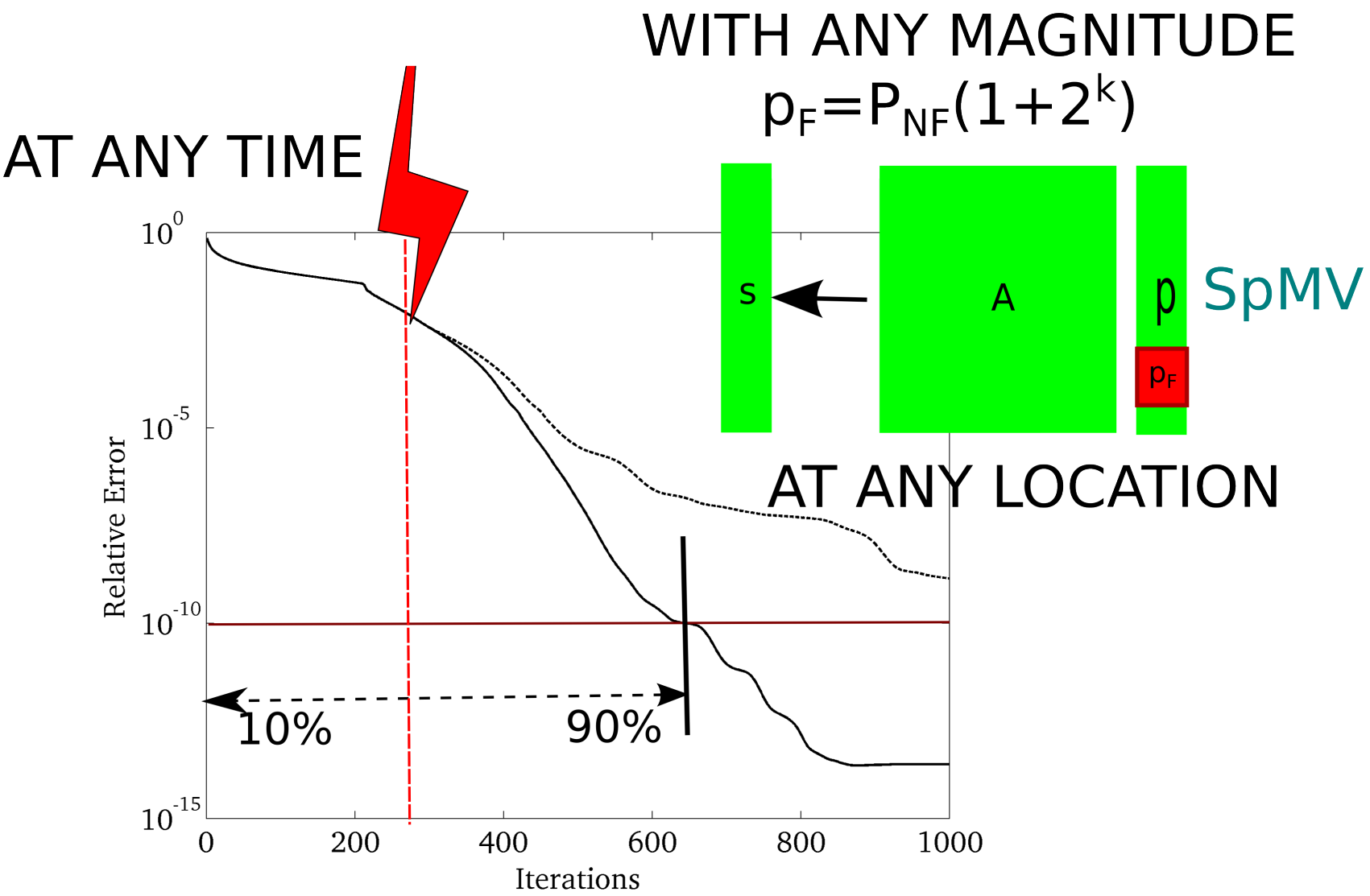
Fault injection



Transient fault injection



Transient fault injection



Parameters for Fault Injection

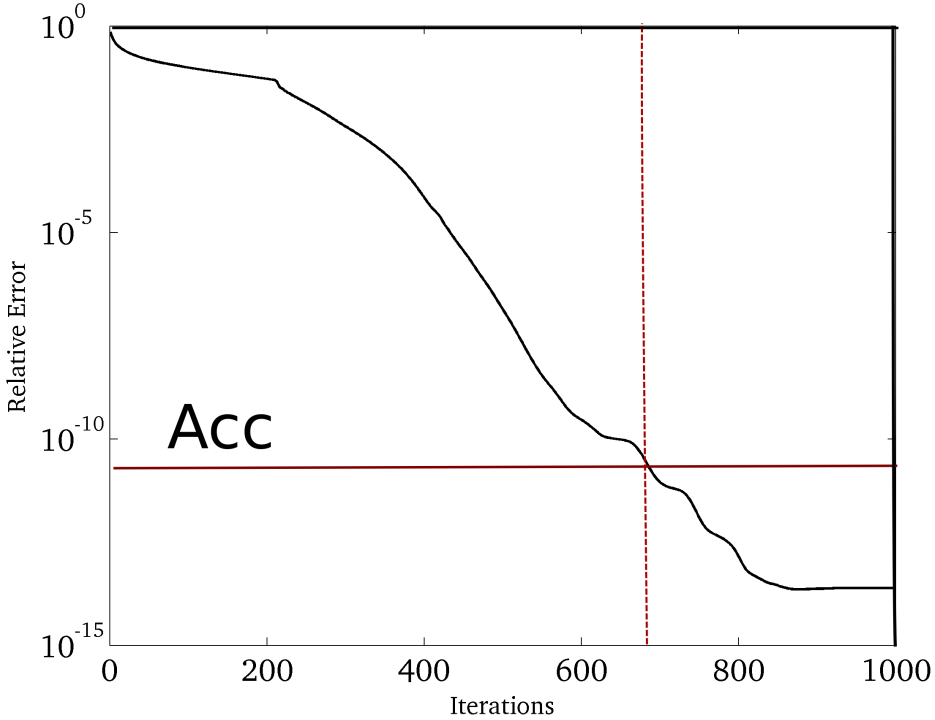
Parameter	Explanation	
Type	Type of the Conjugate Gradient Method	[Classical, Chrono, Pipelined]
ID	Matrix ID	[1 : 30]
Time	Fault injection iteration as percentage	[10%,. . . ,90%]
k	Magnitude of fault injection	[-16:2:16]
Location	Location of fault injection	50 random

[A. T. Chronopoulos, C. W. Gear - JCAM, 1989]

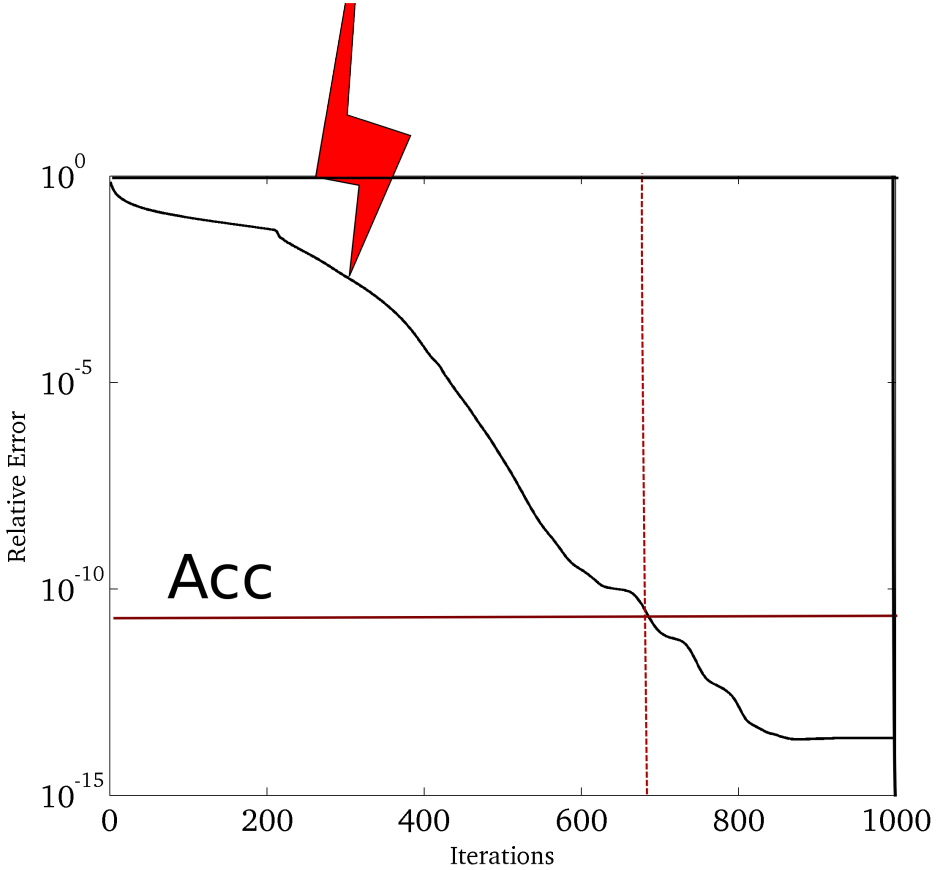
[P. Ghysels, W. Vanroose - ParCo, 2014]

Protocol: correct convergence

	Effect on the convergence
Converged	
Not Converged	

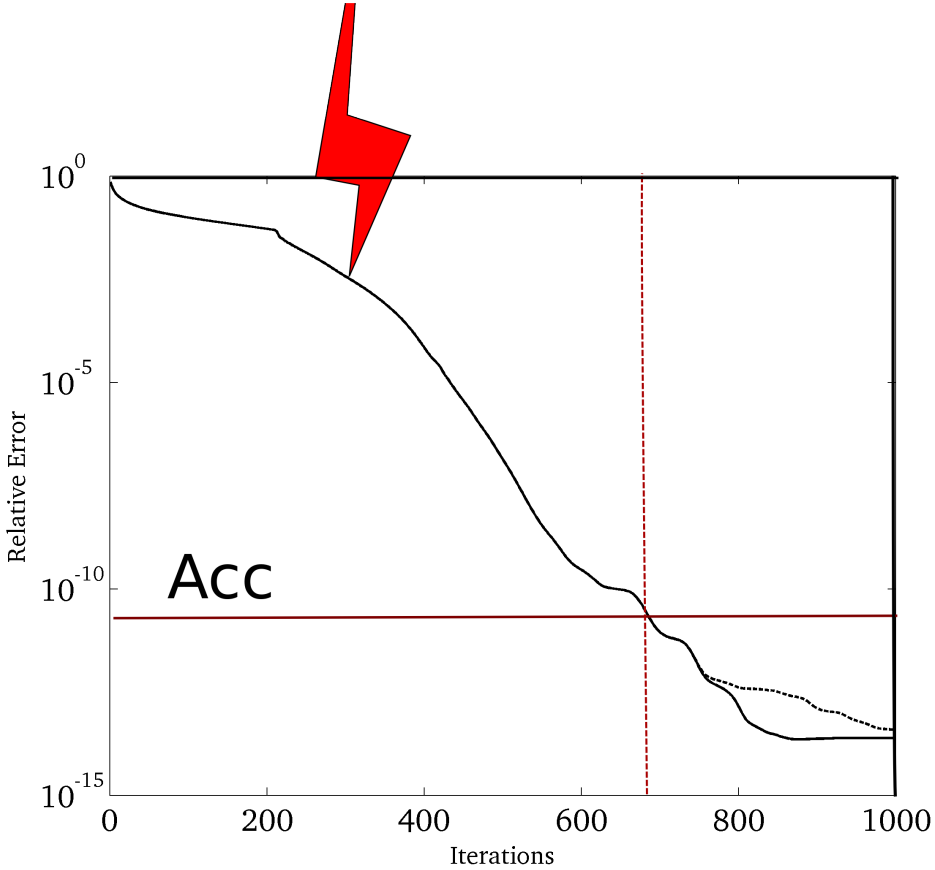


Protocol: correct convergence



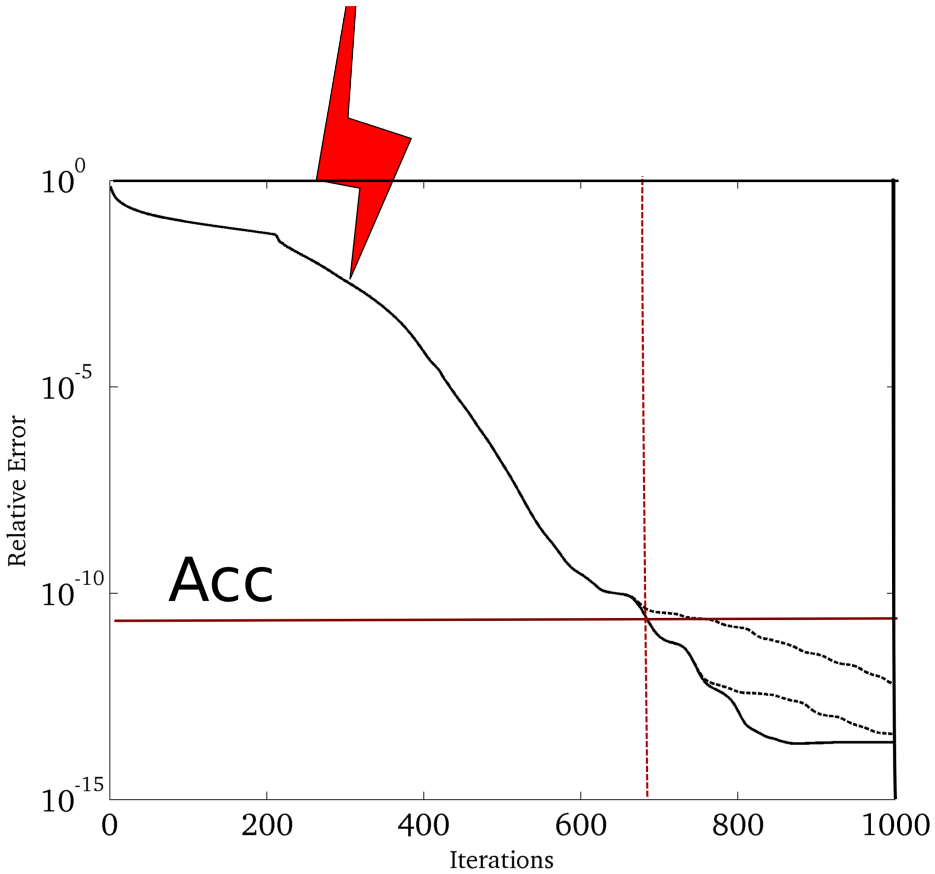
	Effect on the convergence
Converged	
Not Converged	

Protocol: correct convergence



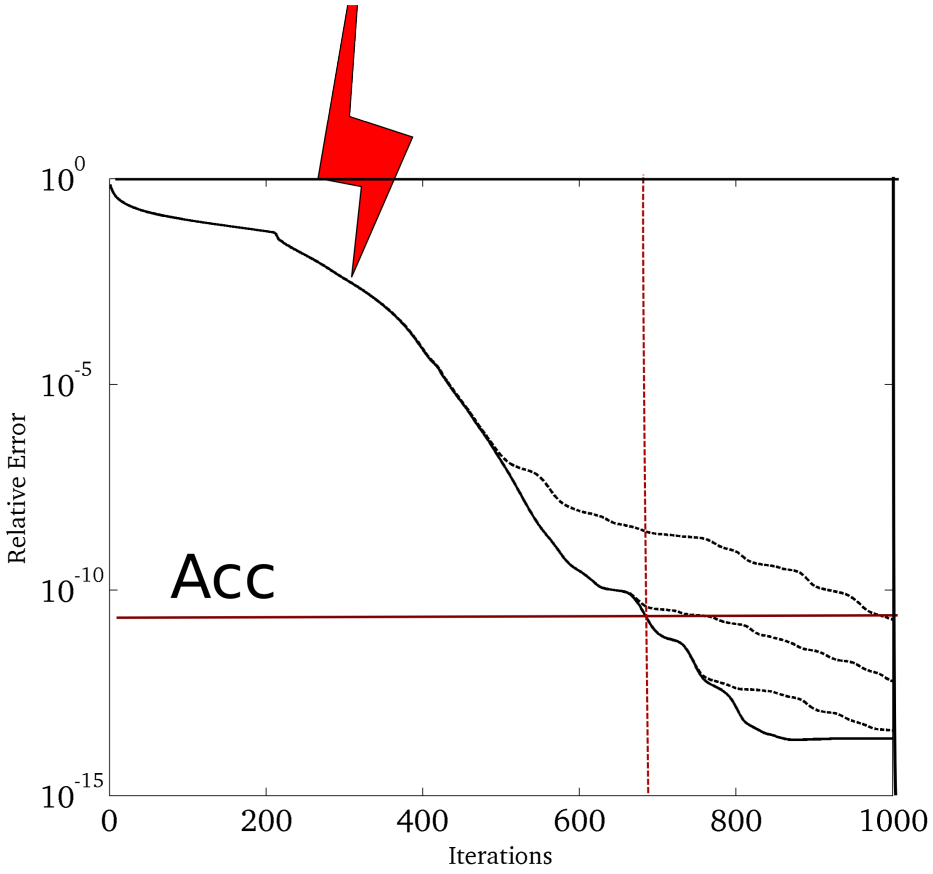
	Effect on the convergence
Converged	
Not Converged	

Protocol: correct convergence



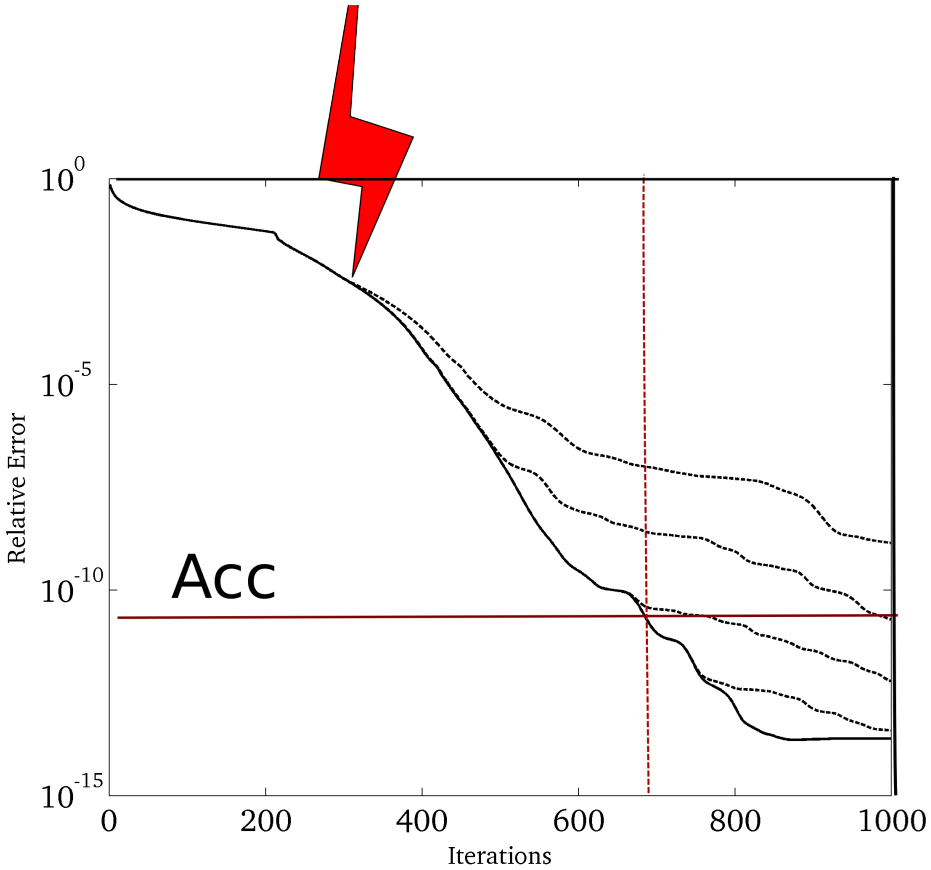
	Effect on the convergence
Converged	
Not Converged	

Protocol: correct convergence



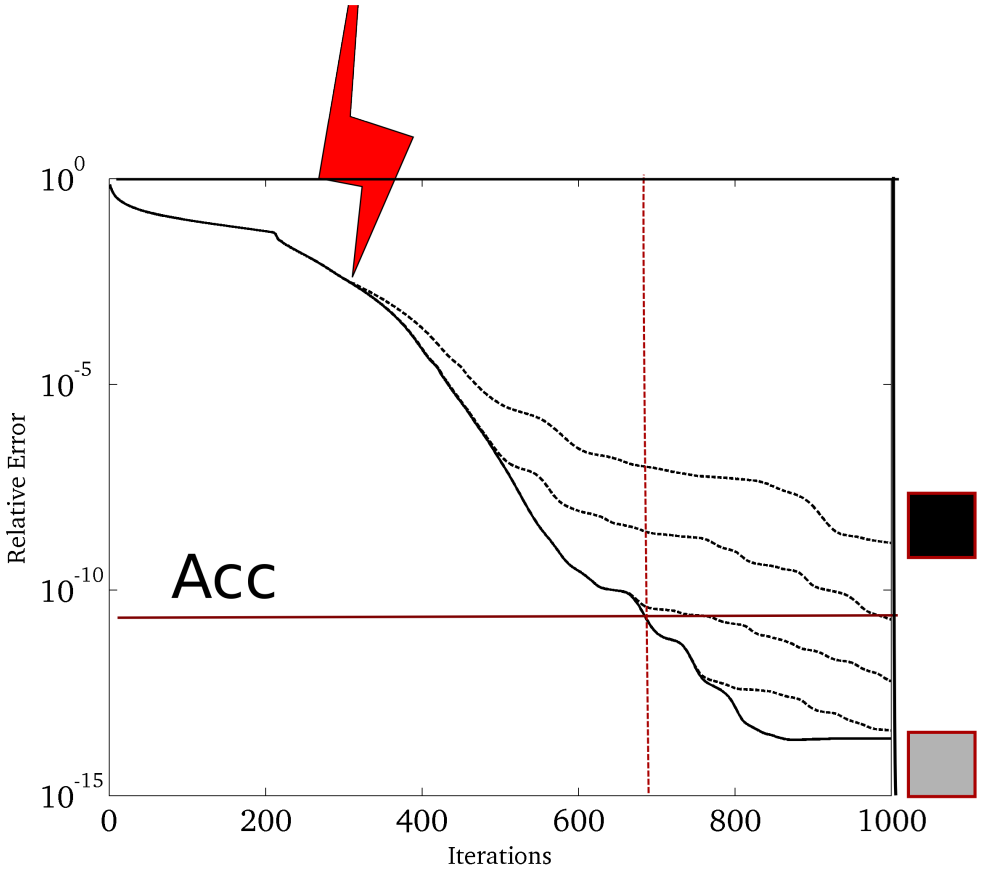
	Effect on the convergence
Converged	
Not Converged	

Protocol: correct convergence



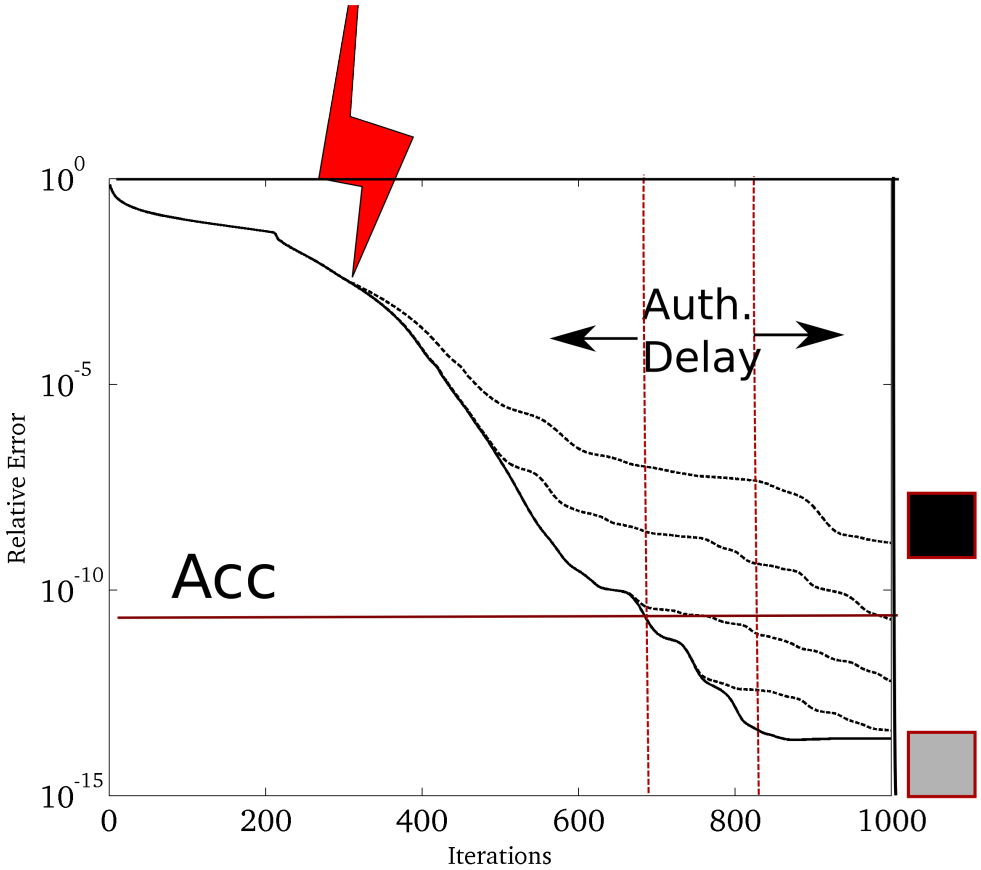
	Effect on the convergence
Converged	
Not Converged	

Protocol: correct convergence



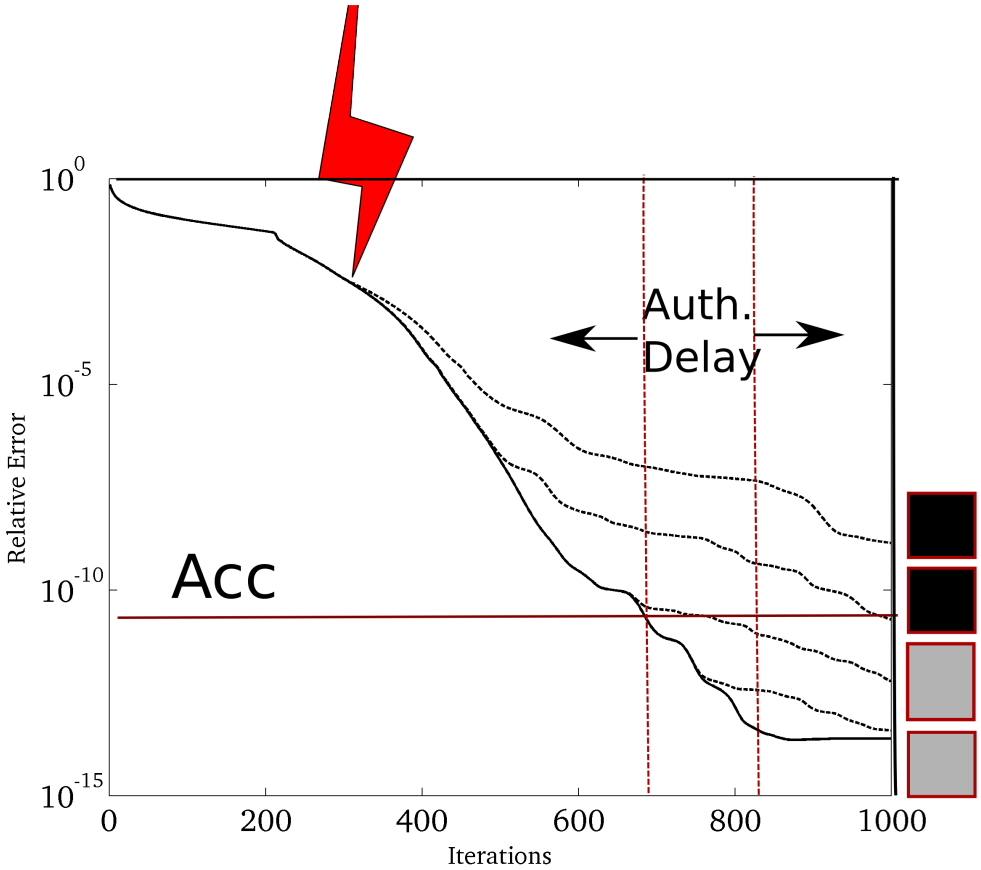
	Effect on the convergence
Converged	
Not Converged	

Protocol: correct convergence



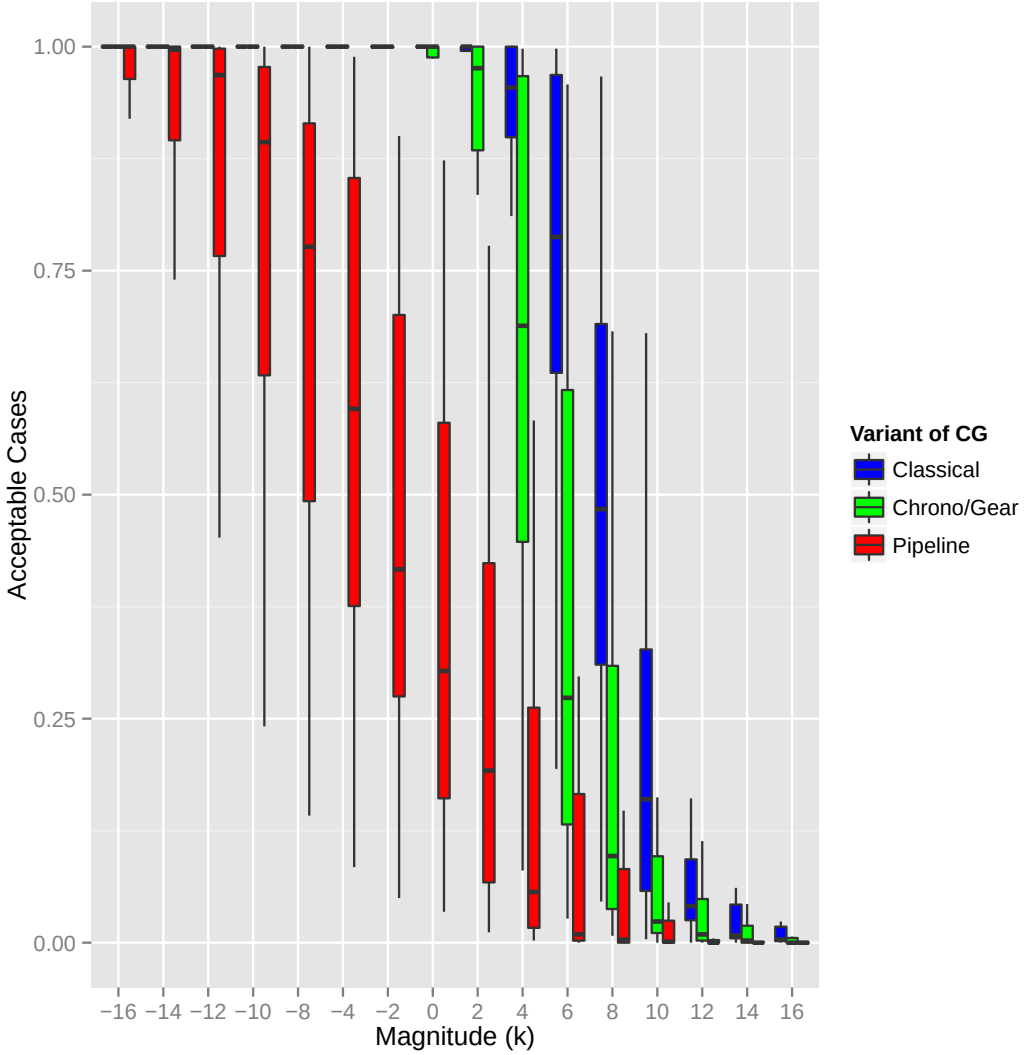
	Effect on the convergence
Converged	
Not Converged	

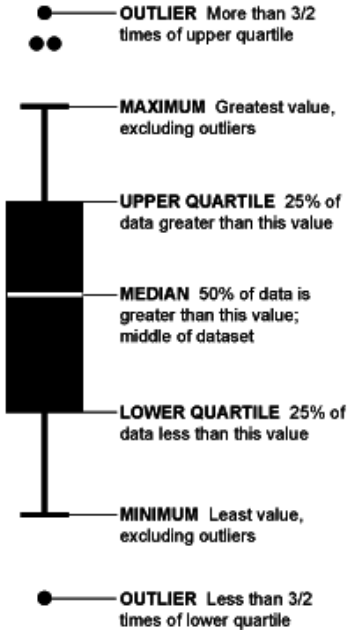
Protocol: correct convergence



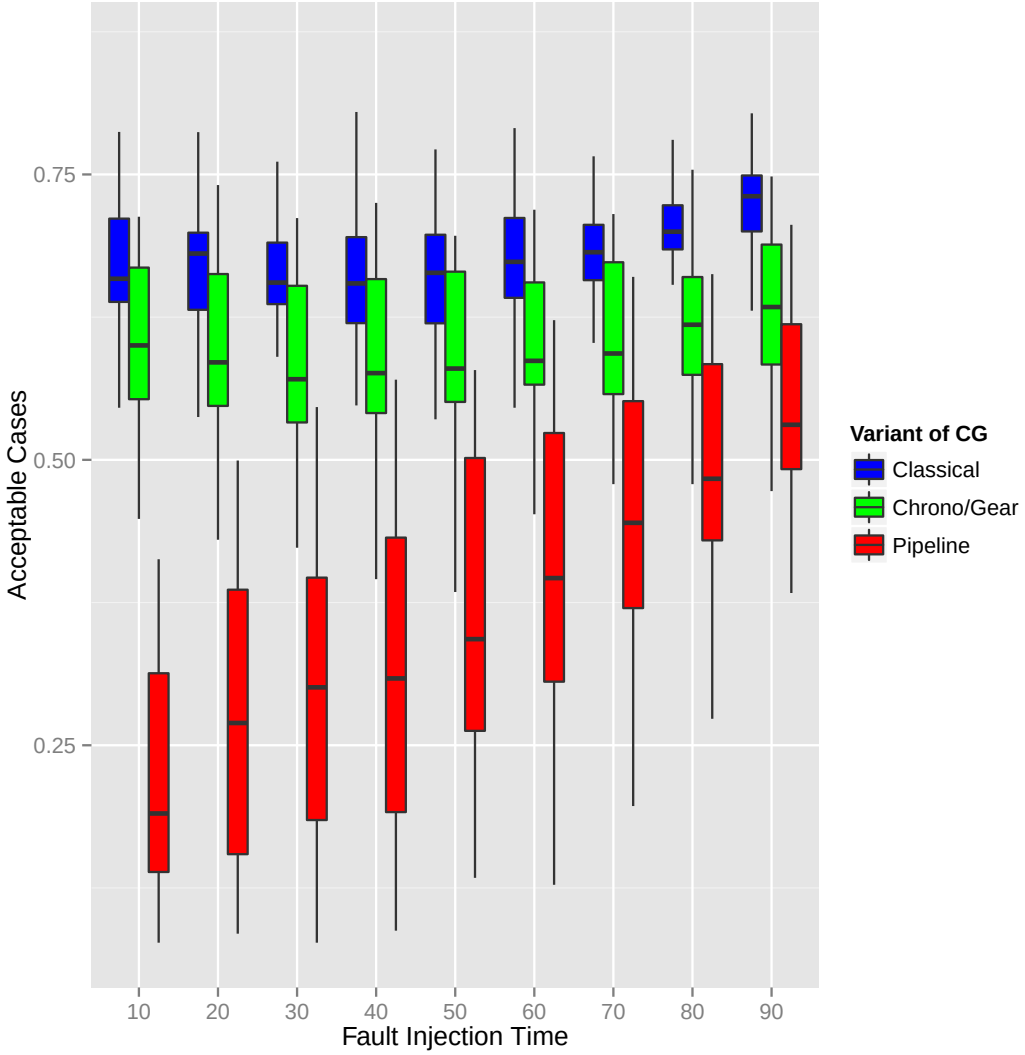
	Effect on the convergence
Converged	
Not Converged	

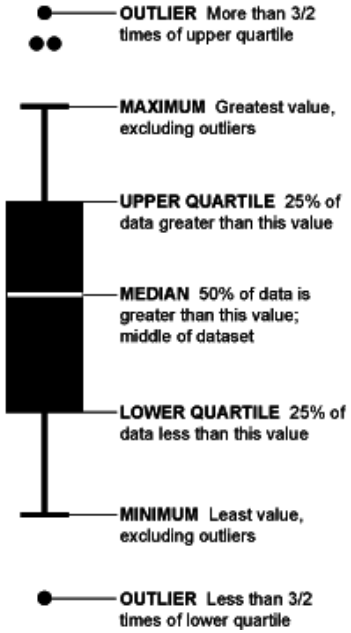
Characterize effect of magnitude (k)





Characterize effect of fault injection time





Summary

Qualitive observations

- Classical CG and Chronopolus/Gear CG are less sensitive for soft errors than Pipelined CG.

Future Work

- Extend the study to GMRES
- Detection based on checksum mechanisms to protect the sparse marix-vector product
- Detection mechanisms based on round-off error analysis (additional benefit : insight on mixed precision calculation)

Acknowlegement for financial support:

- French ANR: RESCUE project
- European FP7 : Exa2CT project
- G8 : ECS project

http://hiepacs.bordeaux.inria.fr/

Merci for your attention

Questions ?

Acknowlegement for financial support:

- French ANR: RESCUE project
- European FP7 : Exa2CT project
- G8 : ECS project

http://hiepacs.bordeaux.inria.fr/