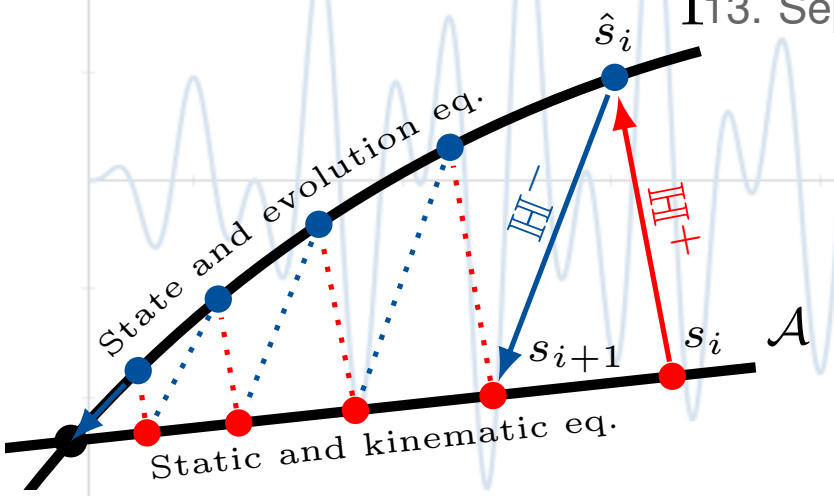
A Semi-incremental Scheme for Fatigue Damage Computations

S. Alameddin[†] , A. Fau[‡],
U. Nackenhorst[†], D. Néron[‡], P. Ladevèze[‡]
† IBNM, Leibniz Universität Hannover
‡ LMT, ENS Cachan, CNRS, Université Paris Saclay

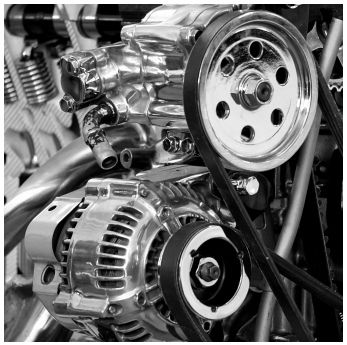
13. September 2019



DFG Deutsche Forschungsgemeinschaft
IRTG-1627

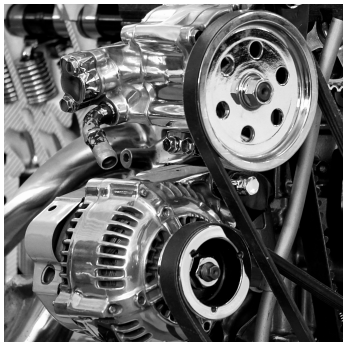
Fatigue damage

- Fluctuating loads


Fatigue damage

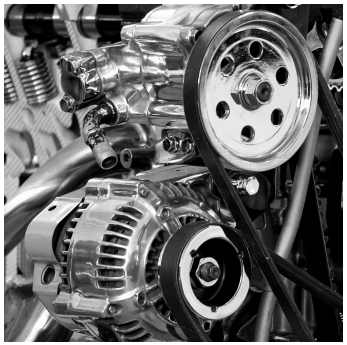
- Fluctuating loads


- Material degradation
  lower load carrying capacity
- Large number of load cycles
  thousands or millions
- Continuum damage model
  $\dot{D} = \mathrm{d}D/\mathrm{d}t$
- Computational expense
  memory and time

Fatigue damage

- Fluctuating loads


- Material degradation
  lower load carrying capacity
- Large number of load cycles
  thousands or millions
- Continuum damage model
  $\dot{D} = \mathrm{d}D/\mathrm{d}t$
- Computational expense
  memory and time

Model order reduction (MOR) techniques

Proper Generalised Decomposition

- Low-rank approximation

$$\boldsymbol{u}(\boldsymbol{x},t) = \sum_{j=1}^{N} \boldsymbol{v}_j(\boldsymbol{x}) \circ \boldsymbol{\lambda}_j(t)$$

- Enrichment to (μ) previously generated modes

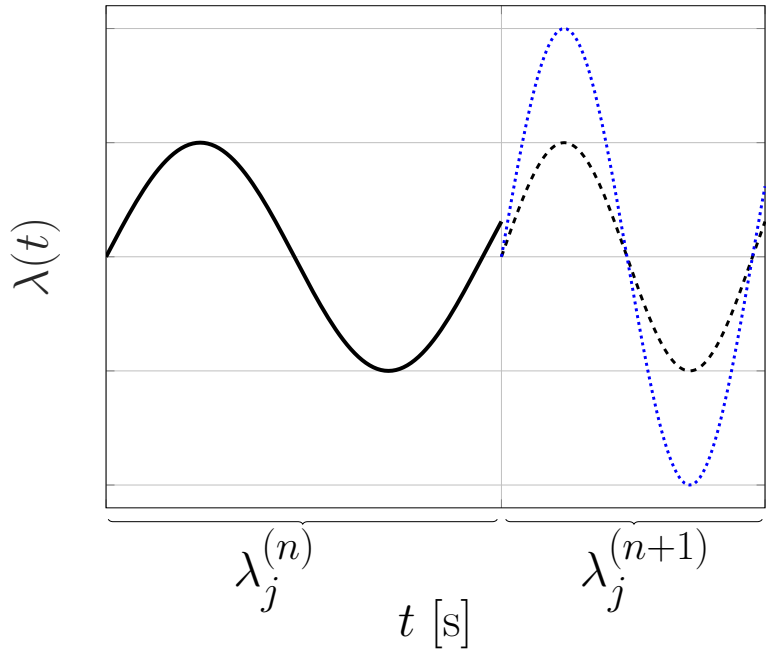
$$\Delta\boldsymbol{u}_{i+1}(\boldsymbol{x},t) = \boldsymbol{v}_{\mu+1}(\boldsymbol{x}) \circ \boldsymbol{\lambda}_{\mu+1}(t)$$

- POD-like update of (μ) previously generated modes

$$\Delta\boldsymbol{u}_{i+1}(\boldsymbol{x},t) = \sum_{j=1}^{\mu} \underbrace{\boldsymbol{v}_j(\boldsymbol{x})}_{\text{known}} \circ \Delta\boldsymbol{\lambda}_j(t)$$

Semi-incremental extension

- The temporal domain is divided into cycles
- Cycles are simulated consecutively

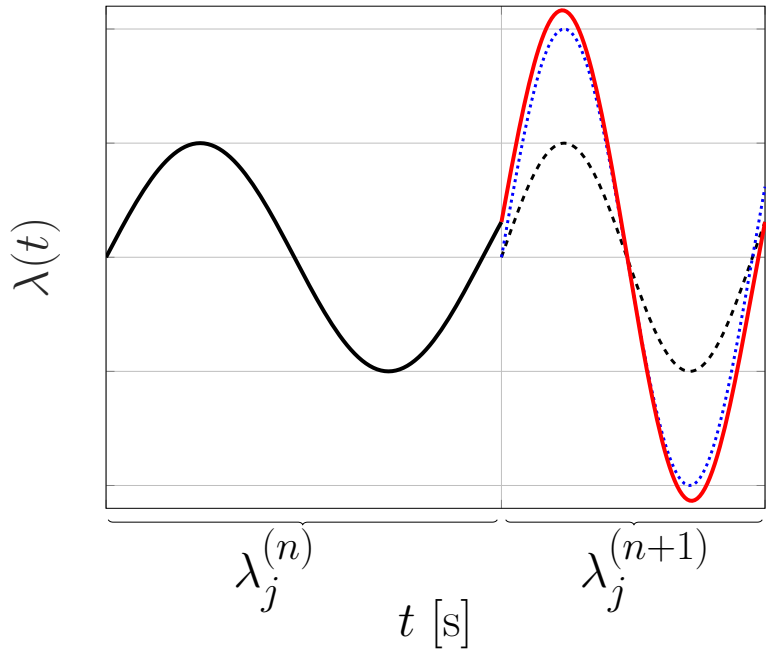


- Already generated $\{\lambda_j(t)\}_{j=1}^{\mu}$ are scaled to $\tilde{t} \in [0, 1]$
- Scaled back to the temporal coordinate of the current cycle

Semi-incremental extension

- Continuity of the temporal modes is an issue
- Temporal modes are vertically scaled and shifted via

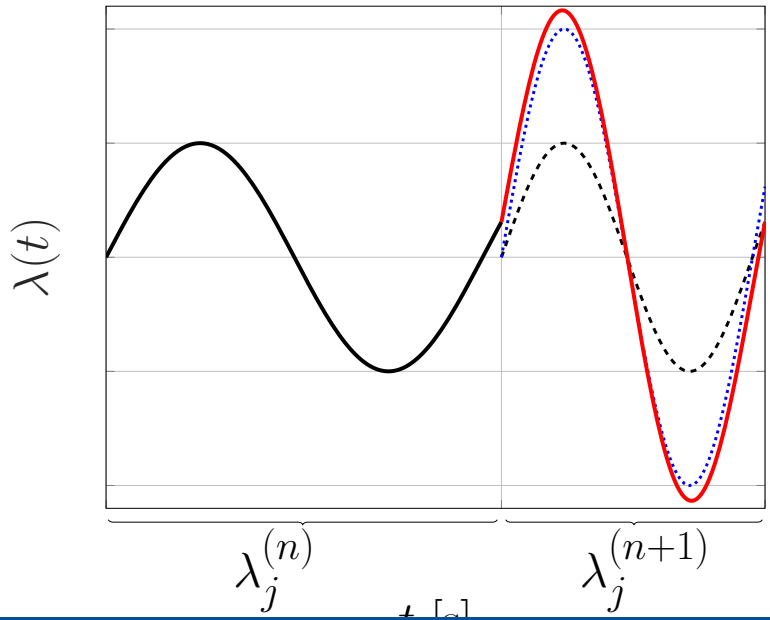
$$\tilde{\lambda}_j^{(n+1)}(\tilde{t}) = m\,\tilde{\lambda}_j^{(n)}(\tilde{t}) + g\,\tilde{t} + h \quad \text{with I.C. and B.C.}$$



Semi-incremental extension

- Continuity of the temporal modes is an issue
- Temporal modes are vertically scaled and shifted via

$$\tilde{\lambda}_j^{(n+1)}(\tilde{t}) = m\,\tilde{\lambda}_j^{(n)}(\tilde{t}) + g\,\tilde{t} + h \quad \text{with I.C. and B.C.}$$



Variable loading / integration over confined domains

SVD Compression of PGD

- Large number of modes
- Slow temporal update step

$$\underline{\underline{\tilde{A}}}\,\underline{\underline{\tilde{\Lambda}}}^{\mathsf{T}} = \underline{\underline{\tilde{B}}}$$

$$\underline{\underline{\tilde{A}}} \in \mathbb{R}^{\mu\times\mu} \qquad \underline{\underline{\tilde{B}}} \in \mathbb{R}^{\mu\times n_t} \qquad \underline{\underline{\tilde{\Lambda}}} = [\Delta\underline{\lambda}_1, \cdots, \Delta\underline{\lambda}_\mu]$$

- Enrichment with a Gram-Schmidt scheme is not optimal
- SVD is optimal but expensive to compute

SVD Compression of PGD

- Given a solution $\underline{\underline{U}} = \underline{\underline{V}}\ \underline{\underline{\Lambda}}^{\mathsf{T}} \in \mathbb{R}^{n \times n_t}$ with

$$\underline{\underline{V}} = [\underline{v}_1, \cdots, \underline{v}_\mu] \in \mathbb{R}^{n \times \mu} \qquad \underline{\underline{\Lambda}} = [\underline{\lambda}_1, \cdots, \underline{\lambda}_\mu] \in \mathbb{R}^{n_t \times \mu}$$

- Exploit the outer product format of PGD

$$\underline{\underline{V}} = \underline{\underline{Q}}_v\ \underline{\underline{R}}_v \qquad \underline{\underline{\Lambda}} = \underline{\underline{Q}}_\lambda\ \underline{\underline{R}}_\lambda$$

- Compute SVD of a small matrix

$$\underline{\underline{T}} = \underline{\underline{R}}_v\ \underline{\underline{R}}_\lambda^{\mathsf{T}} \in \mathbb{R}^{\mu \times \mu} \qquad \underline{\underline{T}} \approx \underline{\underline{\hat{V}}}\ \underline{\underline{\hat{S}}}\ \underline{\underline{\hat{\Lambda}}}^{\mathsf{T}}$$

- Solution with a compressed basis

$$\underline{\underline{U}} \approx \underline{\underline{Q}}_v\ \underline{\underline{\hat{V}}}\quad \underline{\underline{\hat{S}}}\quad \underline{\underline{\hat{\Lambda}}}^{\mathsf{T}}\ \underline{\underline{Q}}_\lambda^{\mathsf{T}}$$

SVD Compression of PGD

- Given a solution $\underline{\underline{U}} = \underline{\underline{V}}\ \underline{\underline{\Lambda}}^{\mathsf{T}} \in \mathbb{R}^{n \times n_t}$ with

$$\underline{\underline{V}} = [\underline{v}_1, \cdots, \underline{v}_\mu] \in \mathbb{R}^{n \times \mu} \qquad \underline{\underline{\Lambda}} = [\underline{\lambda}_1, \cdots, \underline{\lambda}_\mu] \in \mathbb{R}^{n_t \times \mu}$$

- Exploit the outer product format of PGD

$$\underline{\underline{V}} = \underline{\underline{Q}}_v\ \underline{\underline{R}}_v \qquad \underline{\underline{\Lambda}} = \underline{\underline{Q}}_\lambda\ \underline{\underline{R}}_\lambda$$

- Compute SVD of a small matrix

$$\underline{\underline{T}} = \underline{\underline{R}}_v\ \underline{\underline{R}}_\lambda^{\mathsf{T}} \in \mathbb{R}^{\mu \times \mu} \qquad \underline{\underline{T}} \approx \underline{\underline{\hat{V}}}\ \underline{\underline{\hat{S}}}\ \underline{\underline{\hat{\Lambda}}}^{\mathsf{T}}$$

- Solution with a compressed basis

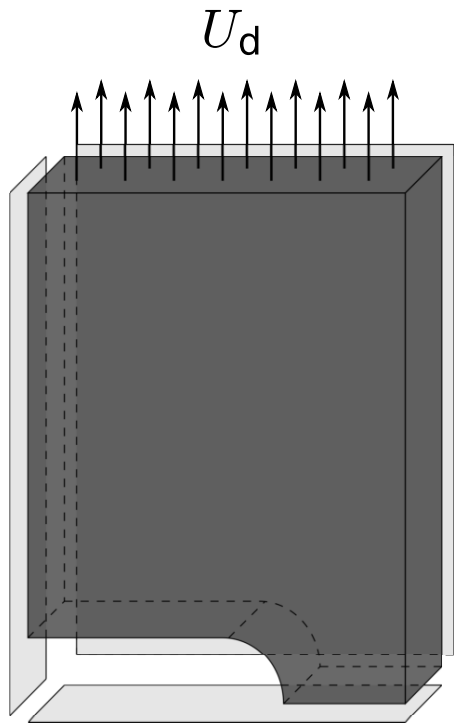
$$\underline{\underline{U}} \simeq \underline{\underline{Q}}_v\ \underline{\underline{\hat{V}}}\ \underline{\underline{\hat{S}}}\ \underline{\underline{\hat{\Lambda}}}^{\mathsf{T}}\ \underline{\underline{Q}}_\lambda^{\mathsf{T}}$$

Non-demanding optimal decomposition 

Numerical results

Viscoplastic viscodamage material

- A plate subjected to cyclic loading (Cr-Mo steel at $580^{\circ}\mathrm{C}$)

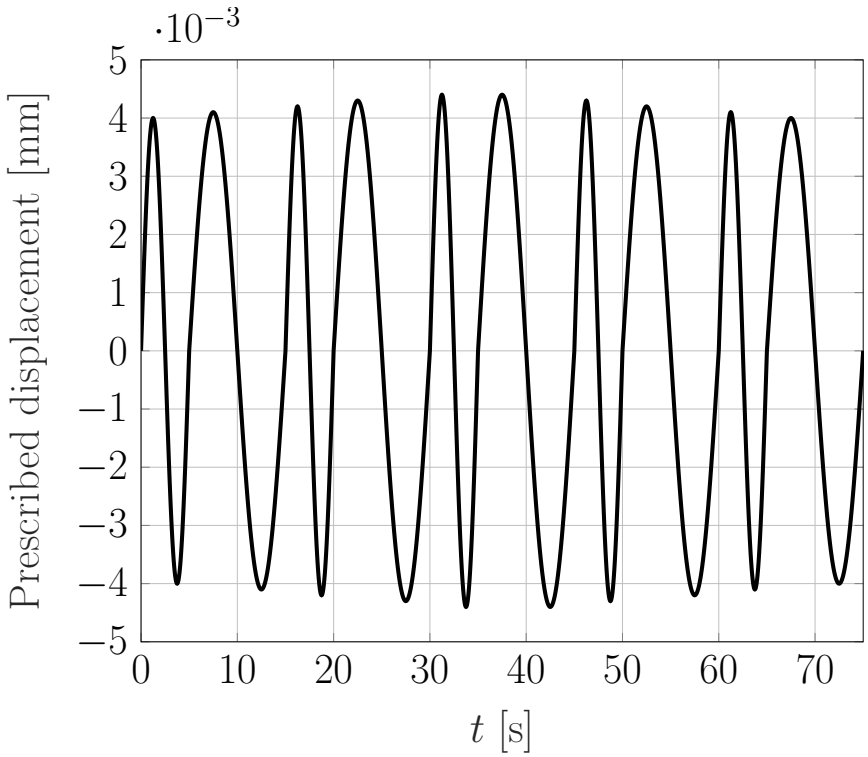


- Plasticity, kinematic and isotropic hardening, damage

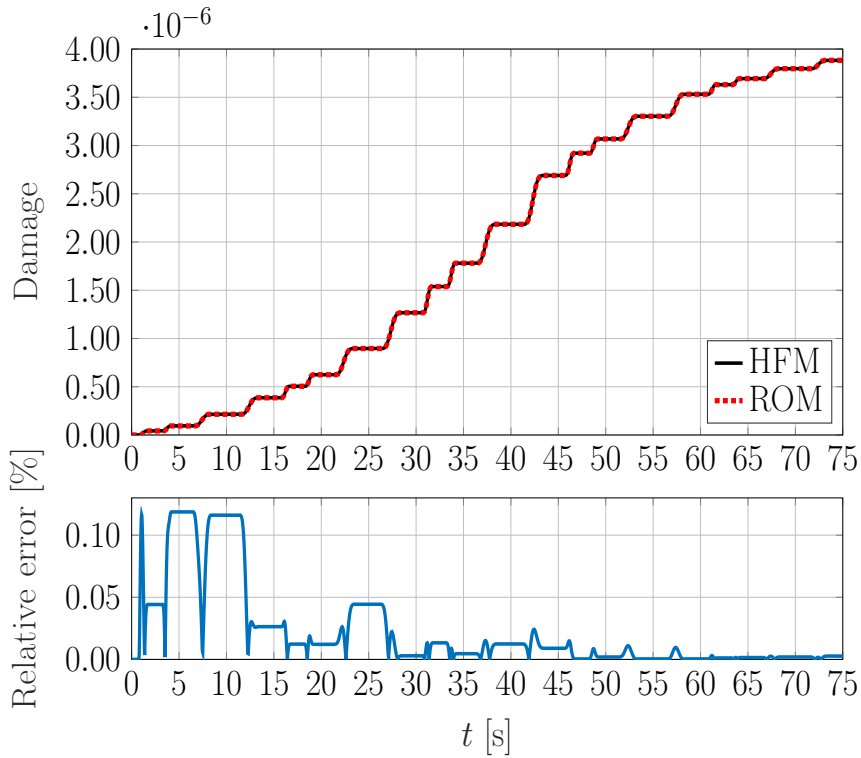
Model verification

$1884 \cdot 41 \cdot 10$ DOF

With respect to a modified Newton-Raphson scheme



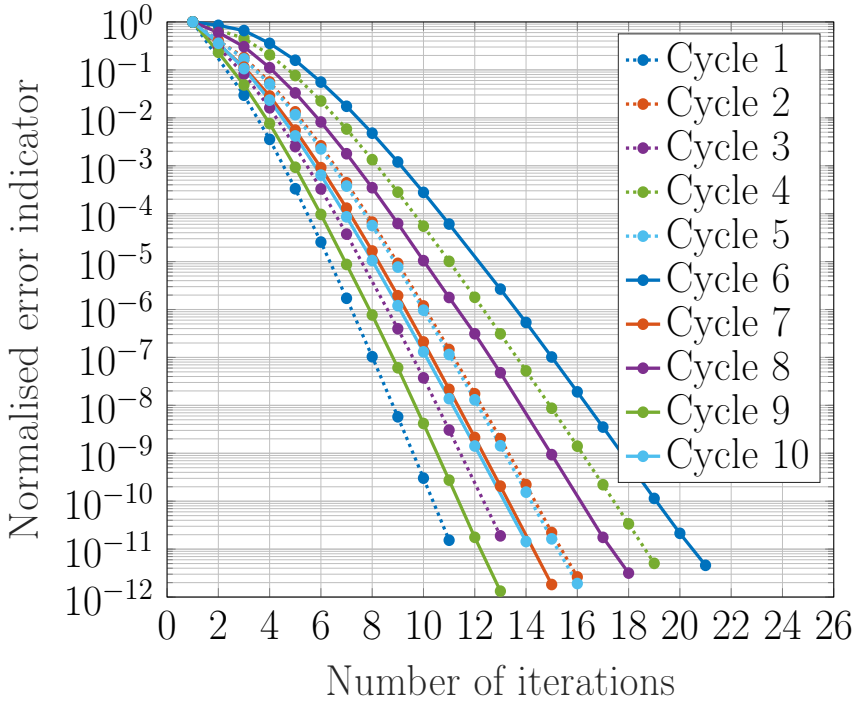
Prescribed displacement



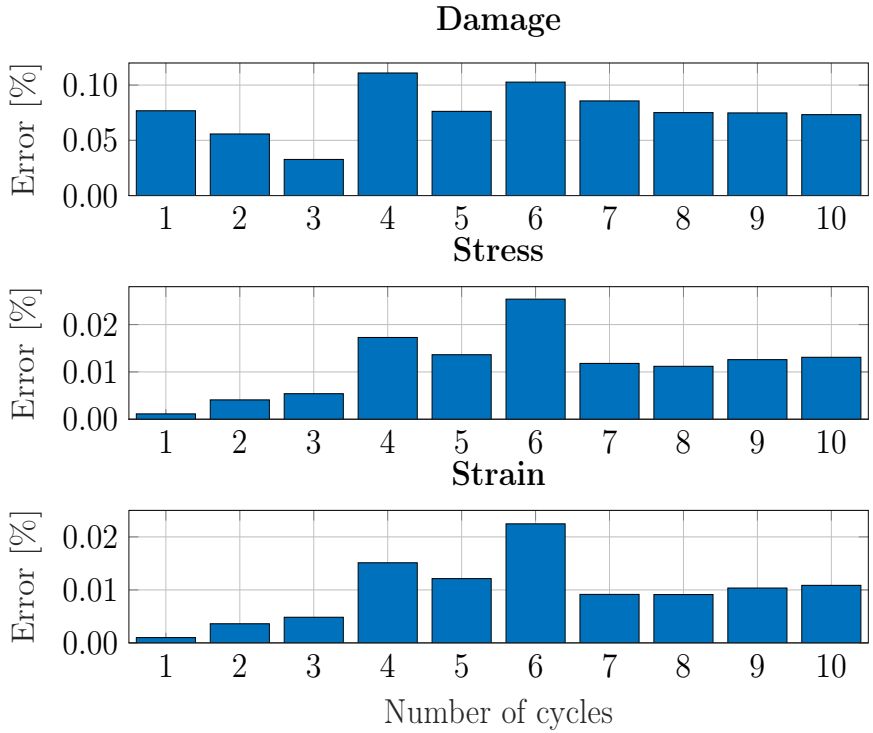
Damage evolution

Model verification

$$\|\boldsymbol{e}\|^2_{\Omega\times\mathcal{I}} = \frac{1}{T\,|\Omega|}\int_{\mathcal{I}}\int_{\Omega} \boldsymbol{e}:\boldsymbol{e}\;\mathrm{d}\Omega\,\mathrm{d}t$$



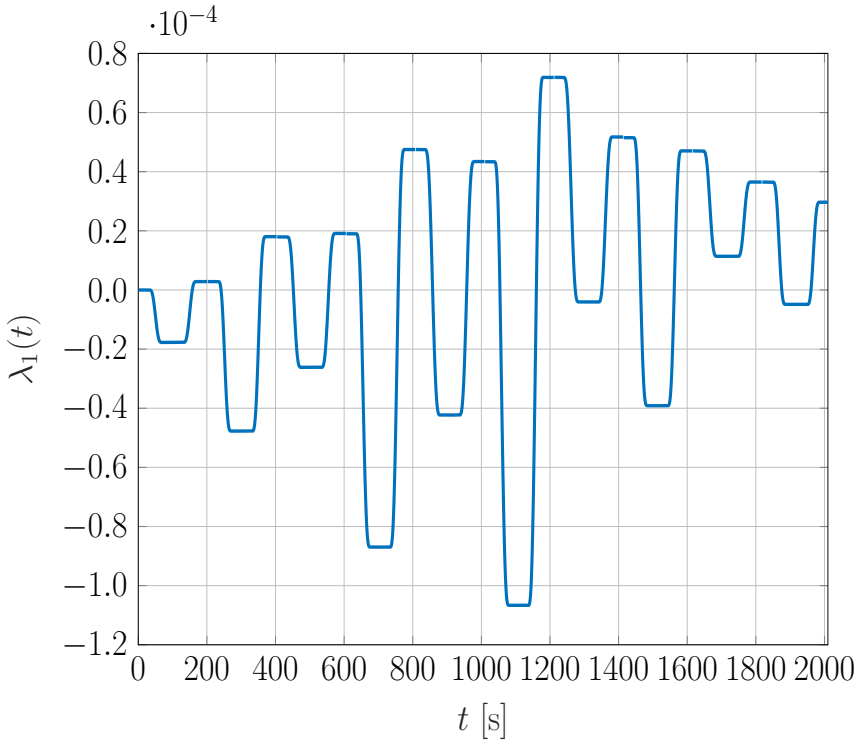
Error indicator

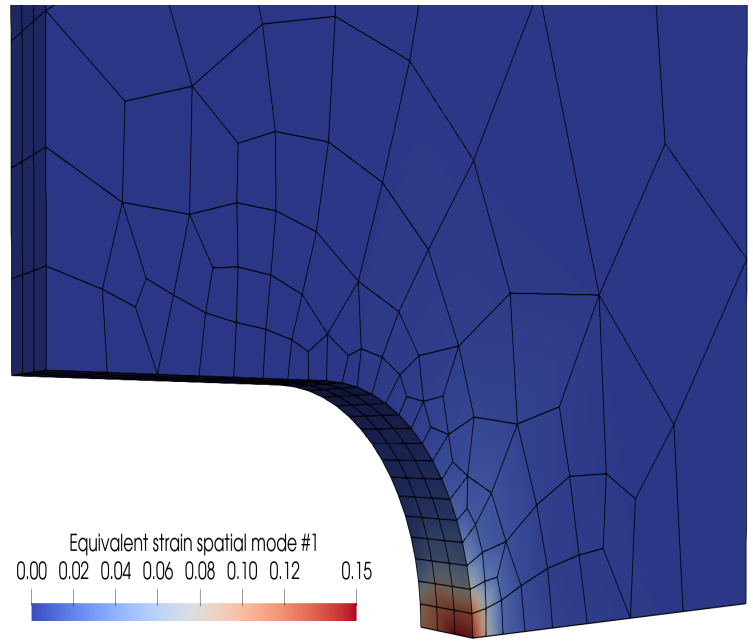


Space-time average relative error

Model verification

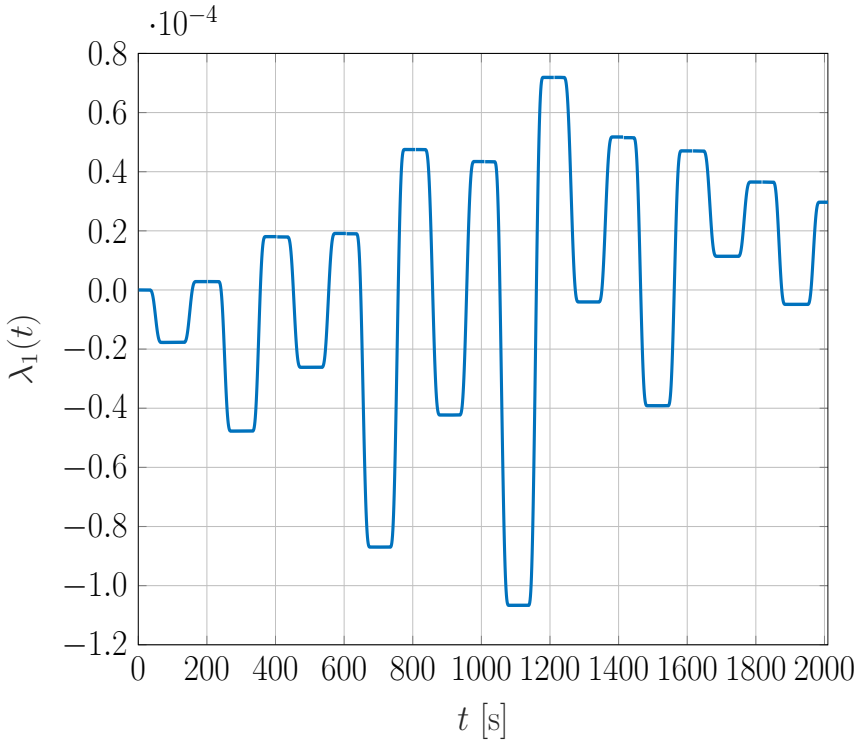
The first temporal and spatial modes

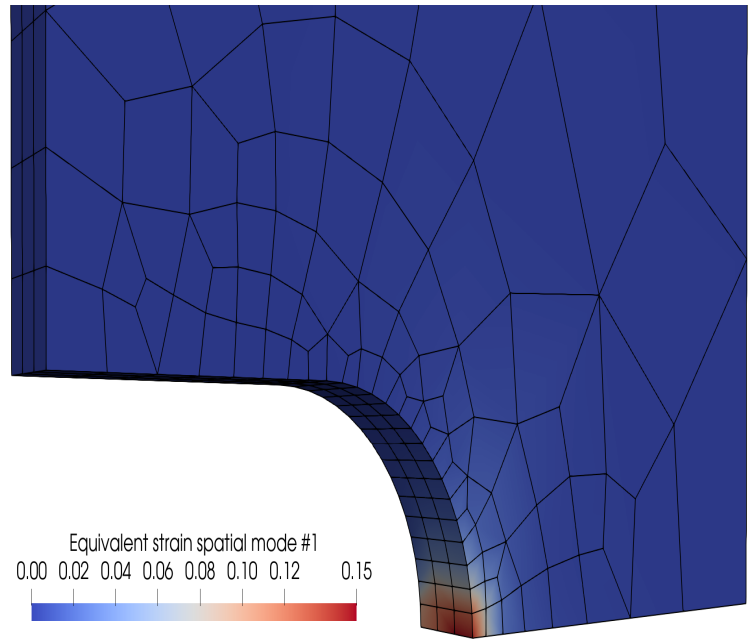




Model verification

The first temporal and spatial modes



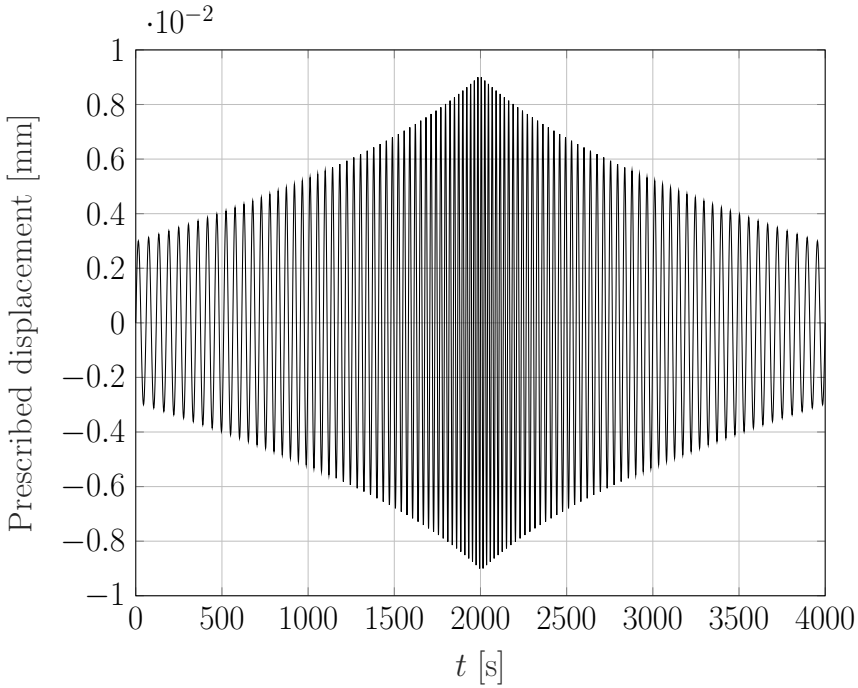


12 modes / Speedup factor ~ 25

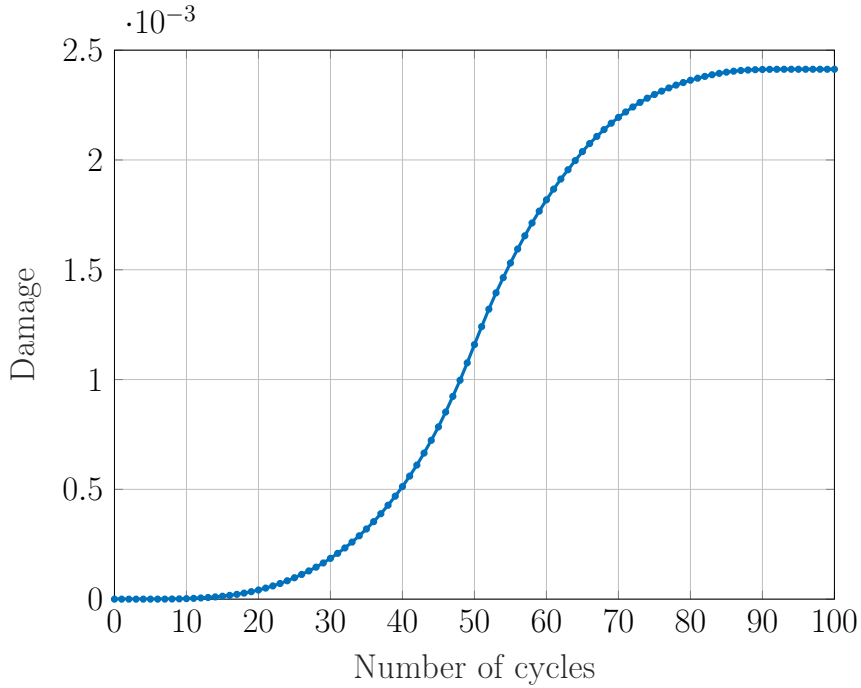
Variable amplitudes & frequencies

Amplitudes: $[30, 90] \cdot 10^{-4}\,\mathrm{mm}$

Time periods: $[20, 60]\,\mathrm{sec}$



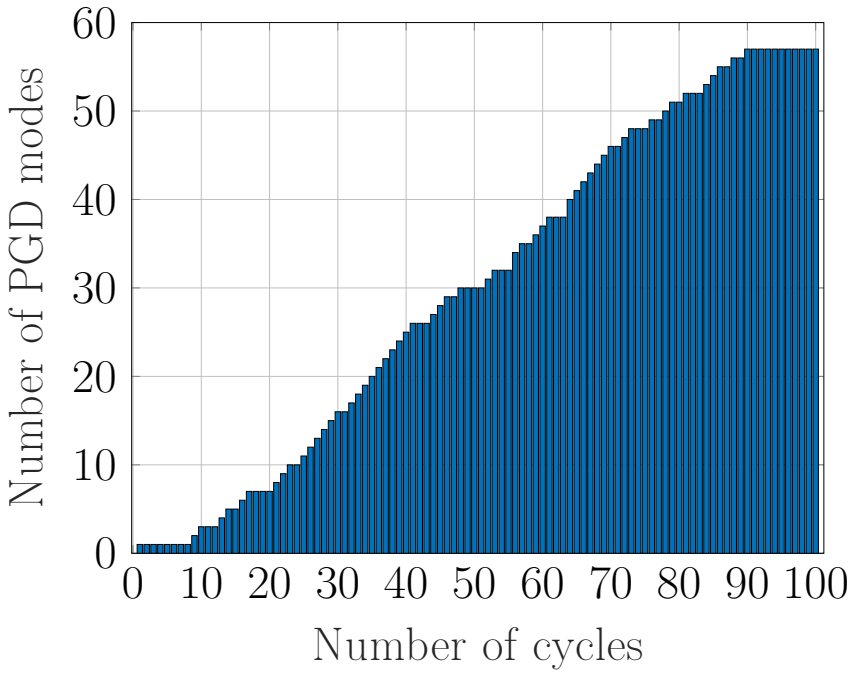
Prescribed displacement



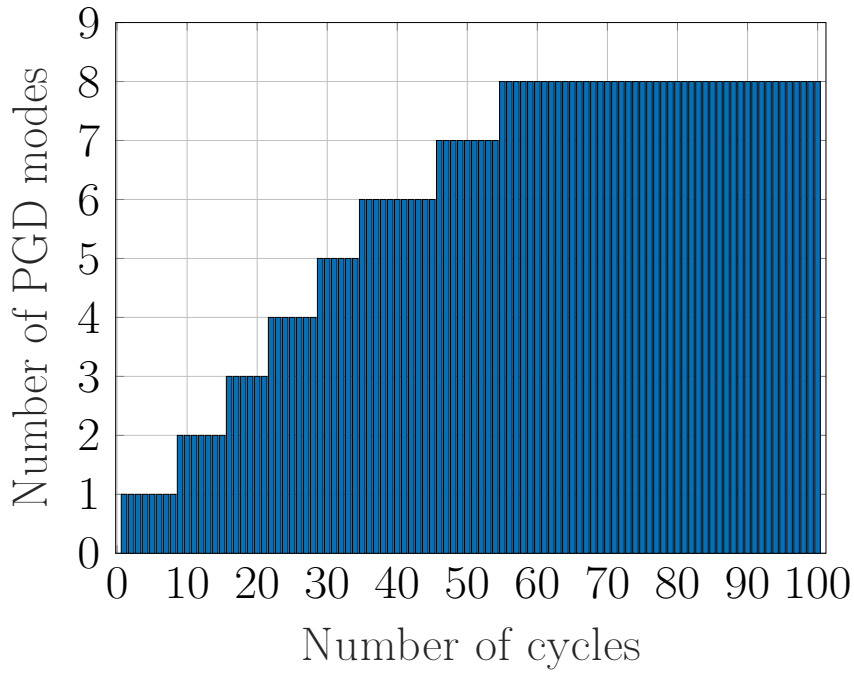
Damage evolution

Variable amplitudes & frequencies

The growth of the ROB using an SVD scheme



Gram-Schmidt

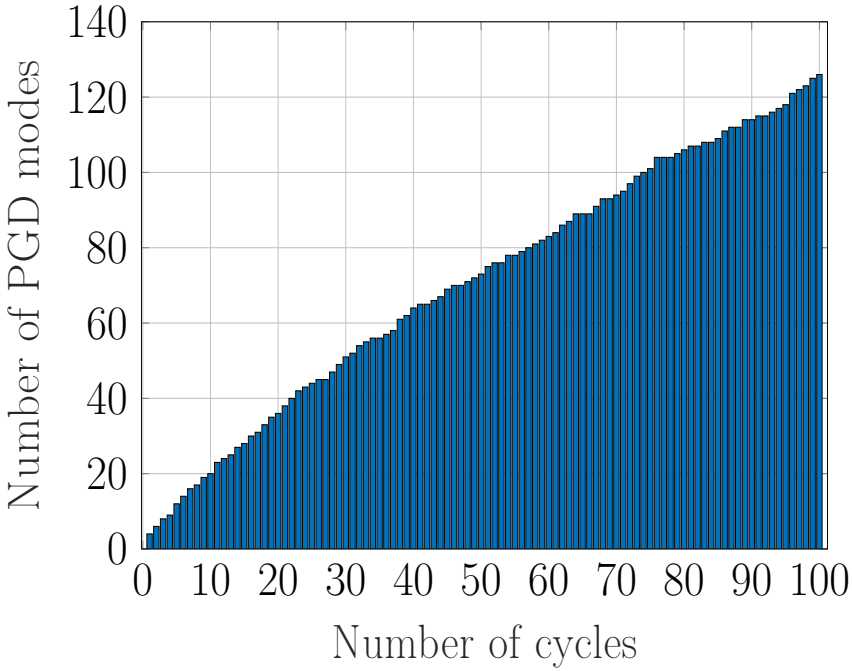


SVD scheme

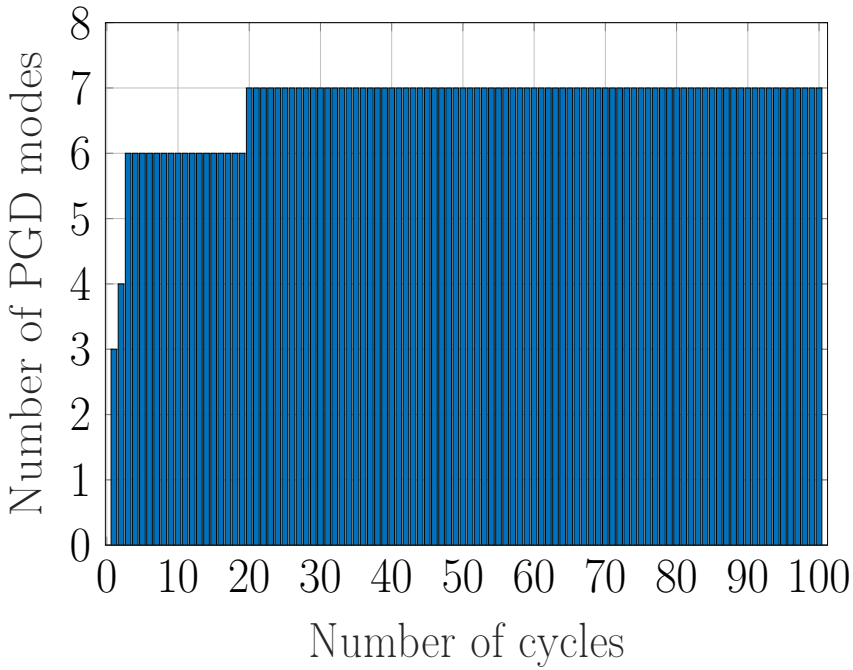
Different ortho. schemes

$50,547 \cdot 33 \cdot 100$ DOF

In case of random loading and fine discretisation



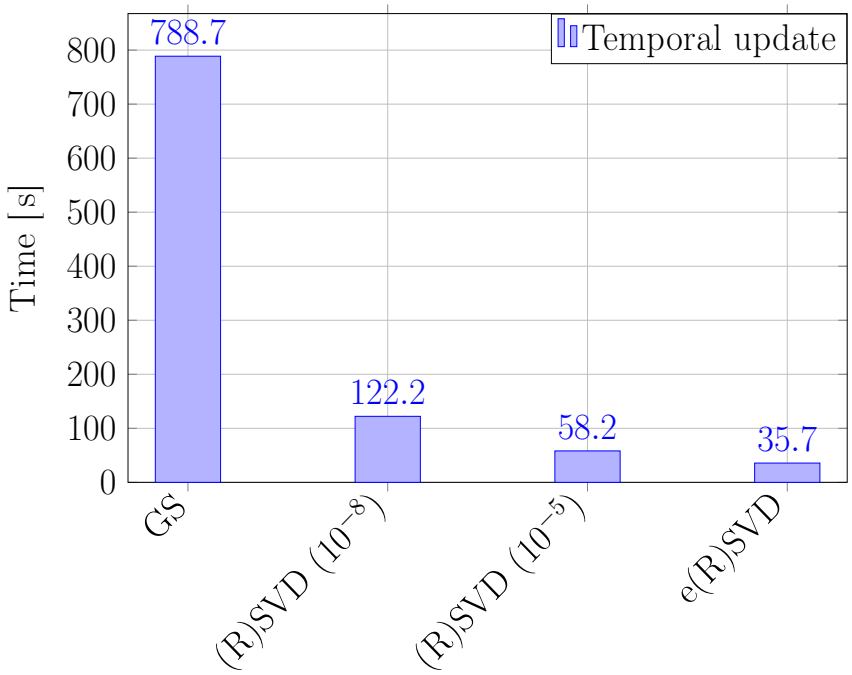
Gram-Schmidt



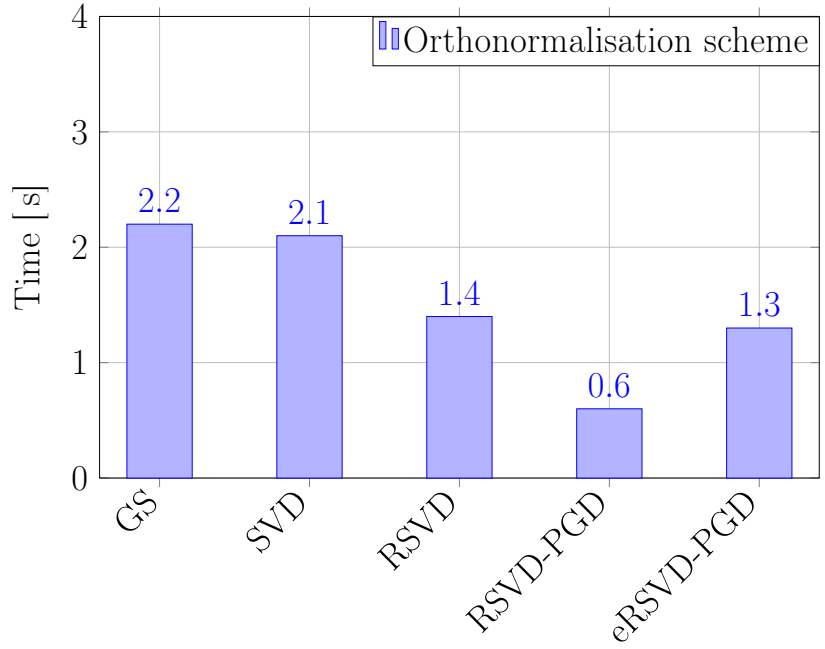
SVD scheme

Different ortho. schemes

The required time to update and orthonormalise the modes



Temporal update

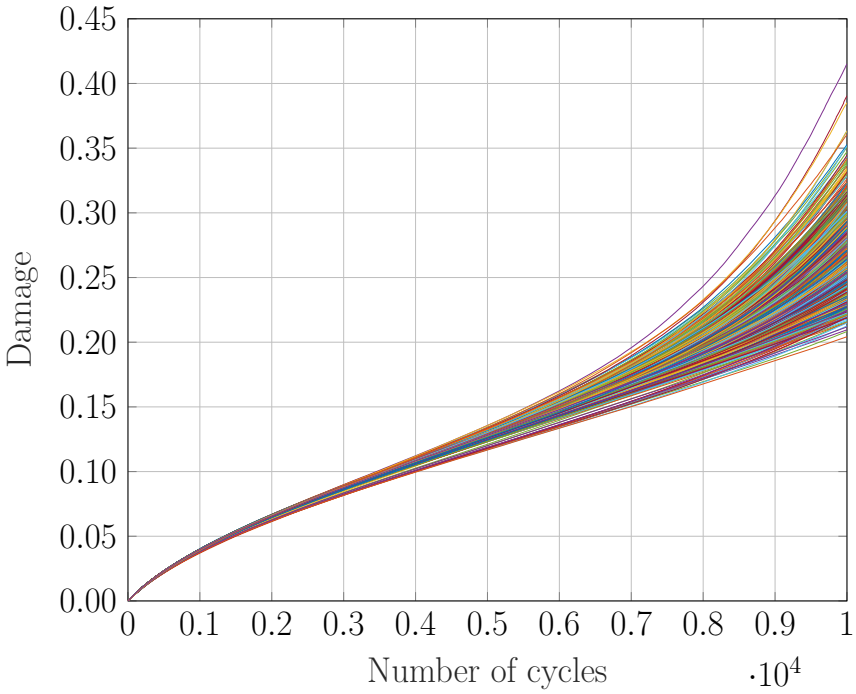


Orthonormalisation step

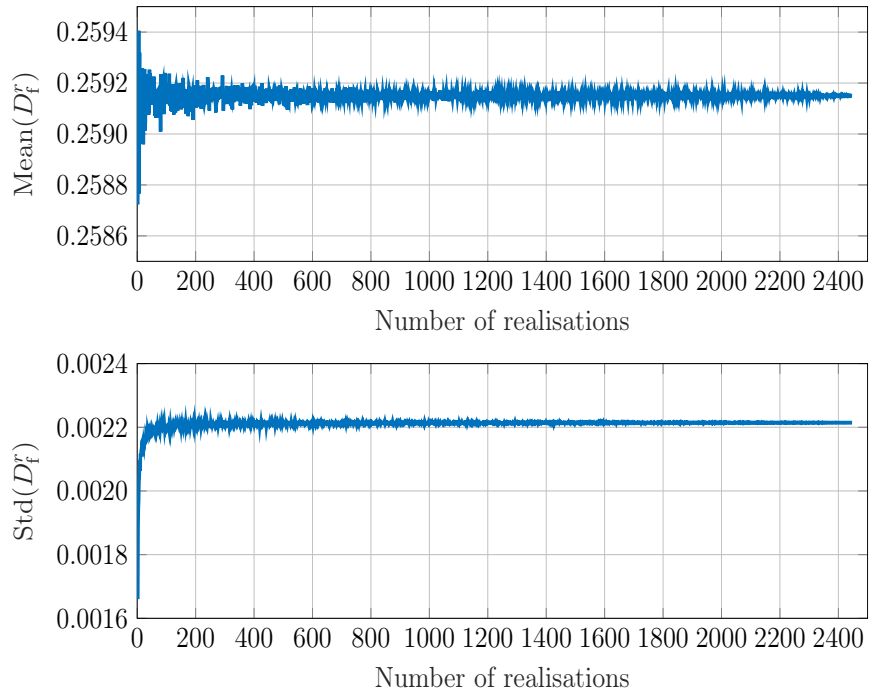
Random amplitude loading

$1884 \cdot 41 \cdot 10^4$ DOF

- 10^4 cycles, uniform distribution in $[53, 56] \cdot 10^{-2}\,\mathrm{mm}$



Different damage realisations

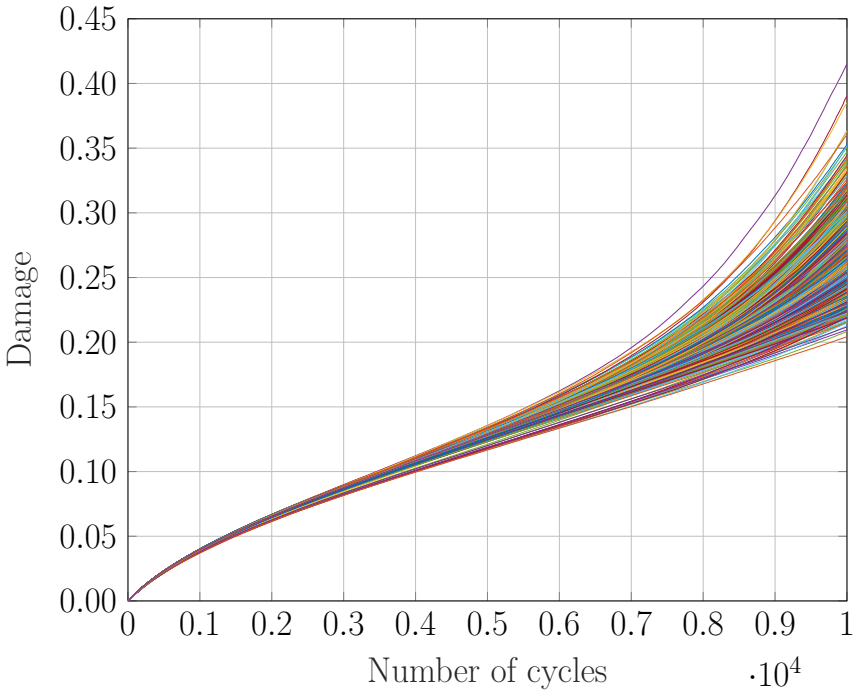


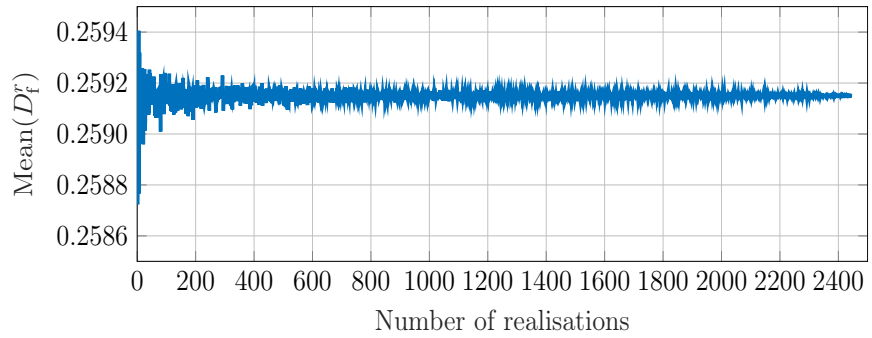
The mean and STD of D_f

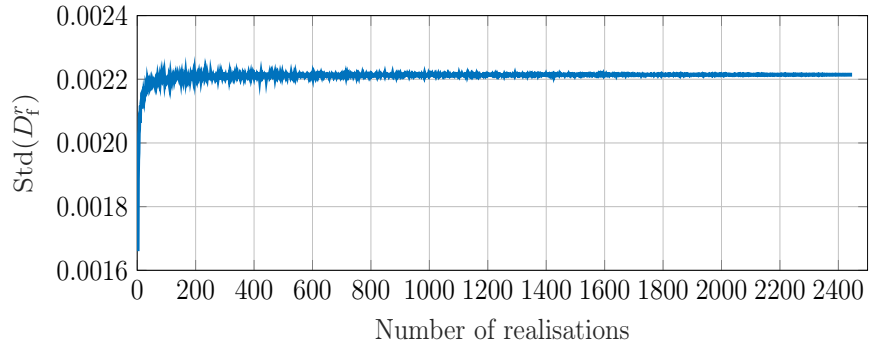
Random amplitude loading

$1884 \cdot 41 \cdot 10^4$ DOF

- 10^4 cycles, uniform distribution in $[53, 56] \cdot 10^{-2}\,\mathrm{mm}$







Critical damage value $D_\mathrm{c} = 0.3$

Probability of failure $P_\mathrm{f} = 5.4\%$

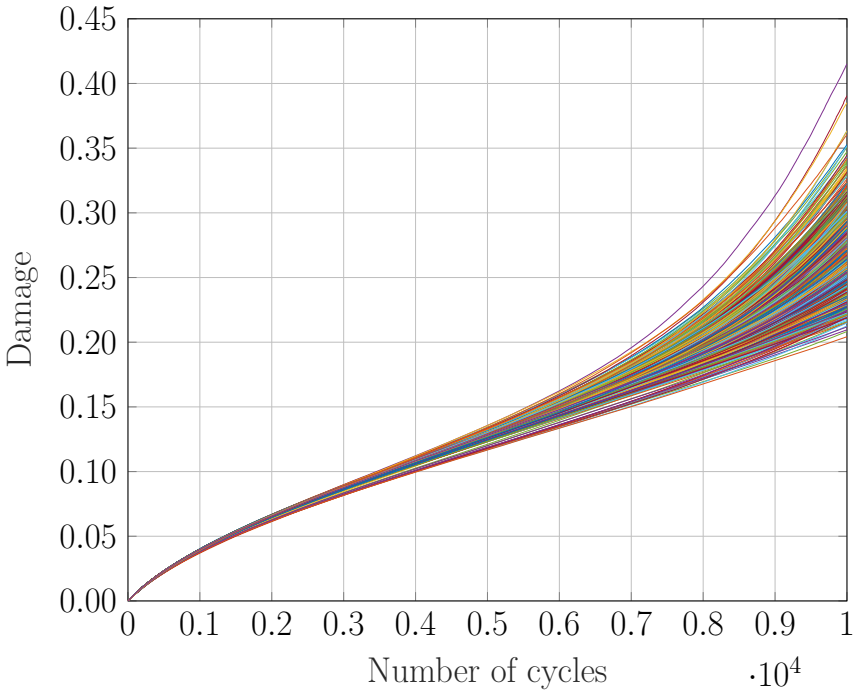
Requirements:

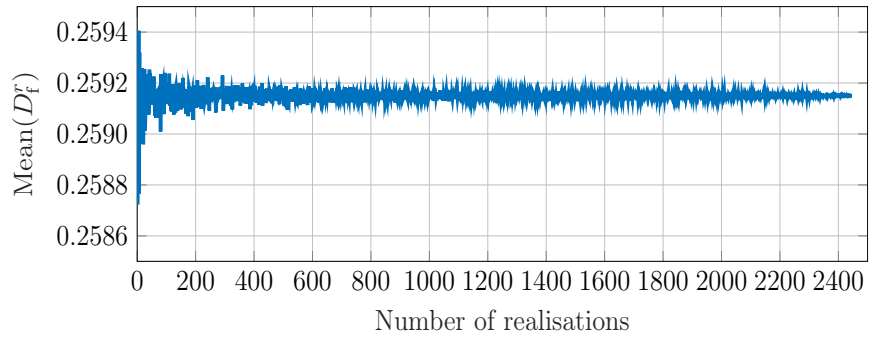
$[15 - 35]\,\mathrm{min}$ and $[1 - 1.5]\,\mathrm{GB}$

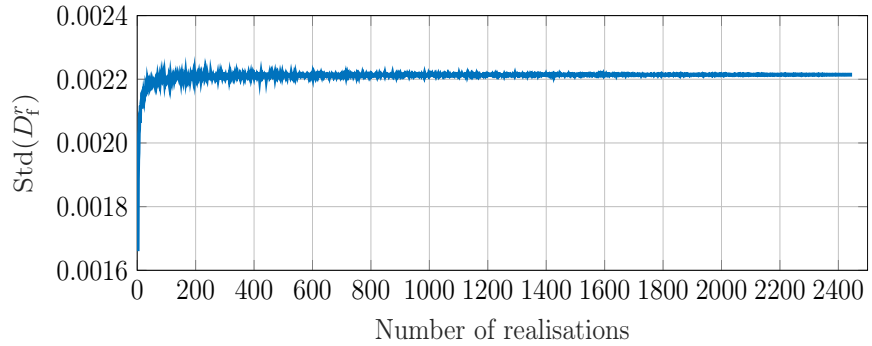
Random amplitude loading

$1884 \cdot 41 \cdot 10^4$ DOF

- 10^4 cycles, uniform distribution in $[53, 56] \cdot 10^{-2}\,\mathrm{mm}$







Critical damage value $D_\mathrm{c} = 0.3$

Probability of failure $P_\mathrm{f} = 5.4\%$

Requirements:

$[15 - 35]\,\mathrm{min}$ and $[1 - 1.5]\,\mathrm{GB}$

Approx. 10 modes / Time-saving factors $50 \sim 100$

Conclusions

- Efficient semi-incremental scheme for damage problems
- Can handle variable amplitude and frequency loadings
- Provides a minimal expansion of PGD
- Most operations are done over decomposed QoI
- Open-source code: `gitlab.com/shadialameddin/romfem`

Conclusions

- Efficient semi-incremental scheme for damage problems
- Can handle variable amplitude and frequency loadings
- Provides a minimal expansion of PGD
- Most operations are done over decomposed QoI
- Open-source code: `gitlab.com/shadialameddin/romfem`

Thank you for your attention