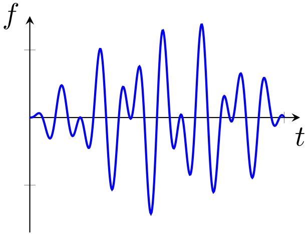
A Semi-incremental Scheme for Fatigue Damage Computations

S. Alameddin[†] , A. Fau[†],
U. Nackenhorst[†], D. Néron[‡], P. Ladevèze[‡]
† IBNM, Leibniz Universität Hannover
‡ LMT, ENS Cachan, CNRS, Université Paris Saclay
18. July 2019

IRTG-1627

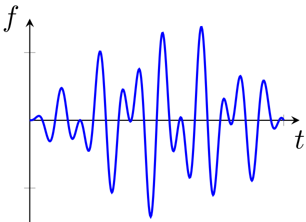
Fatigue damage

- Fluctuating loads



Fatigue damage

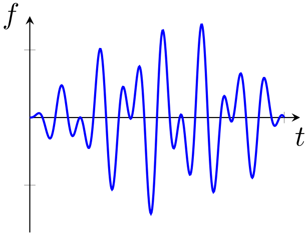
- Fluctuating loads



- Material degradation

Fatigue damage

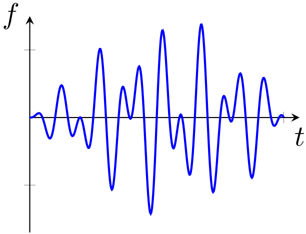
- Fluctuating loads



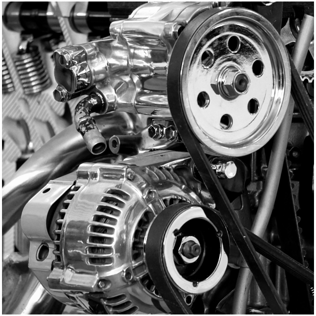
- Material degradation

Fatigue damage

- Fluctuating loads

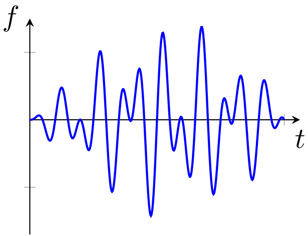


- Material degradation



Fatigue damage

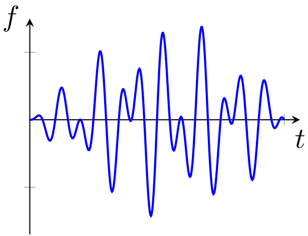
- Fluctuating loads



- Material degradation
- Continuum damage model
- Large number of cycles
- Macro crack initiation
- Computationally expensive

Fatigue damage

- Fluctuating loads



- Material degradation
- Continuum damage model
- Large number of cycles
- Macro crack initiation
- Computationally expensive

Model order reduction (MOR) techniques

Outline

1. State of art
2. Reduced order model for cyclic loading
3. Numerical examples
4. Conclusions and future research

Outline

ROM for nonlinear problems

- Short-term plastic/damage computations
  - Proper orthogonal decomposition (POD)
    [Kerfriden et al, 2011-2012; Ryckelynck, 2005-2011].
  - Proper generalised decomposition (PGD)
    [Vitse, 2016 ; Bhattacharyya et al, 2017].

- Long-term fatigue computations
  - (Modified) jump cycle approach
    [Desmorat 2005; Bhattacharyya et al, 2018]
  - Temporal homogenisation
    [Fish and Yu, 2002; Devulder et al, 2010]
  - Space-time finite element method
    [Bhamare, 2014; Fritzen et al, 2018]

Outline

Mechanical problem

Admissibility equations (AE)

- Static admissibility

$$\nabla\cdot\boldsymbol{\sigma} + \boldsymbol{b} = \boldsymbol{0} \quad \text{in } \Omega \times \mathcal{I}$$

$$\boldsymbol{\sigma}\cdot\boldsymbol{n} = \bar{\boldsymbol{t}} \quad \text{on } \partial\Omega_{\mathrm{N}} \times \mathcal{I}$$

- Kinematic admissibility

$$\boldsymbol{\varepsilon} = \nabla^{\mathrm{s}}\boldsymbol{u} \quad \text{in } \Omega \times \mathcal{I}$$

$$\boldsymbol{u} = \bar{\boldsymbol{u}} \quad \text{on } \partial\Omega_{\mathrm{D}} \times \mathcal{I}$$

Nonlinear material model (CM)

- State equations

$$\boldsymbol{\sigma} = f(\psi(\boldsymbol{\varepsilon}, \boldsymbol{\varepsilon}^{\mathrm{p}}, \boldsymbol{q}))$$

$$\boldsymbol{Q} = g(\psi(\boldsymbol{\varepsilon}, \boldsymbol{\varepsilon}^{\mathrm{p}}, \boldsymbol{q}))$$

- Evolution equations

$$\dot{\boldsymbol{\varepsilon}}^{\mathrm{p}} = \hat{f}(\phi(\boldsymbol{\sigma}, \boldsymbol{Q}))$$

$$\dot{\boldsymbol{q}} = \hat{g}(\phi(\boldsymbol{\sigma}, \boldsymbol{Q}))$$

LATIN solution algorithm over a time interval

- Start with an elastic initialisation
- Local stage, given initial conditions

  Solve the state and evolution equations (CM) to get $\hat{\square}$
- Global stage

  Solve the admissibility equations (AE) to get $\square_{i+1}$
- Data flow between these stages

$$
\begin{aligned}
(\boldsymbol{\sigma}_{i+1} - \hat{\boldsymbol{\sigma}}) - \mathbb{H}^{-}(\boldsymbol{\varepsilon}_{i+1} - \hat{\boldsymbol{\varepsilon}}) &= \mathbf{0} \\
(\hat{\boldsymbol{\sigma}} - \boldsymbol{\sigma}_{i}) + \mathbb{H}^{+}(\hat{\boldsymbol{\varepsilon}} - \boldsymbol{\varepsilon}_{i}) &= \mathbf{0}
\end{aligned}
$$

- Iterate until convergence with an energy error indicator

$$
\xi = \frac{\left\| s_{i+1} - \hat{s} \right\|}{\left\| \left(s_{i+1} + \hat{s} \right) / 2 \right\|}, \quad \|\square\|^2 = \frac{1}{2T} \int_{\Omega \times \mathcal{I}} \left(\boldsymbol{\sigma} : \mathbb{C}^{-1} : \boldsymbol{\sigma} + \boldsymbol{\varepsilon} : \mathbb{C} : \boldsymbol{\varepsilon} \right) \, \mathrm{d}\Omega \, \mathrm{d}t.
$$

LATIN solution algorithm over a time interval

- Start with an elastic initialisation
- Local stage, given initial conditions

  Solve the state and evolution equations (CM) to get $\hat{\square}$
- Global stage

  Solve the admissibility equations (AE) to get $\square_{i+1}$
- Data flow between these stages

$$
\begin{aligned}
(\boldsymbol{\sigma}_{i+1} - \hat{\boldsymbol{\sigma}}) - \mathbb{H}^{-}(\boldsymbol{\varepsilon}_{i+1} - \hat{\boldsymbol{\varepsilon}}) &= \mathbf{0} \\
(\hat{\boldsymbol{\sigma}} - \boldsymbol{\sigma}_{i}) + \mathbb{H}^{+}(\hat{\boldsymbol{\varepsilon}} - \boldsymbol{\varepsilon}_{i}) &= \mathbf{0}
\end{aligned}
$$

- Iterate until convergence with an energy error indicator

$$
\xi = \frac{\left\| s_{i+1} - \hat{s} \right\|}{\left\| \left(s_{i+1} + \hat{s} \right) / 2 \right\|}, \quad \left\| \square \right\|^2 = \frac{1}{2T} \int_{\Omega \times \mathcal{I}} \left(\boldsymbol{\sigma} : \mathbb{C}^{-1} : \boldsymbol{\sigma} + \boldsymbol{\varepsilon} : \mathbb{C} : \boldsymbol{\varepsilon} \right) \, \mathrm{d}\Omega \, \mathrm{d}t.
$$

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

Choosing the search directions as

$$\mathbb{H}^+ = 0, \qquad \mathbb{H}^- = \alpha\,\mathbb{C} \qquad \alpha \in \,]0,1]$$

Algorithmic point of view

- Enrichment

$$\gamma \, \underline{\underline{K}} \, \underline{v} = \underline{f} \qquad \gamma \in \mathbb{R} \qquad \underline{\underline{K}} \in \mathbb{R}^{n \times n} \qquad \underline{v}, \underline{f} \in \mathbb{R}^{n} \qquad \text{with B.C.}$$

$$a \, \underline{\lambda} = \underline{b} \qquad a \in \mathbb{R} \qquad \underline{\lambda}, \underline{b} \in \mathbb{R}^{n_t} \qquad \text{with I.C.}$$

- Update

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

Algorithmic point of view

- Enrichment

$$\gamma\, \underline{\underline{K}}\, \underline{v} = \underline{f} \qquad \gamma \in \mathbb{R} \qquad \underline{\underline{K}} \in \mathbb{R}^{n\times n} \qquad \underline{v}, \underline{f} \in \mathbb{R}^{n} \qquad \text{with B.C.}$$

$$a\, \underline{\lambda} = \underline{b} \qquad a \in \mathbb{R} \qquad \underline{\lambda}, \underline{b} \in \mathbb{R}^{n_t} \qquad \text{with I.C.}$$

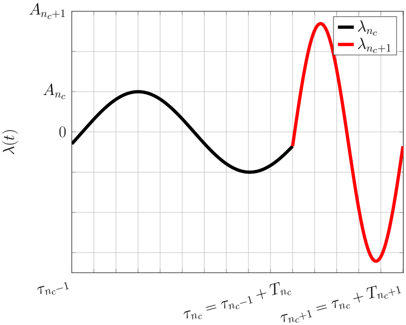
- Update

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

Semi-incremental scheme

- The temporal domain is divided into intervals/cycles
- Temporal modes are defined over $\tilde{t} \in [0, 1]$

$$\tilde{\lambda}_{n_c+1}(\tilde{t}) = m\,\lambda_{n_c}(\tilde{t}) + g\,\tilde{t} + h \quad \text{with}$$
$$\tilde{\lambda}_{n_c+1}(0) = \lambda_{n_c}(1), \quad \tilde{\lambda}_{n_c+1}(1) = \tilde{\lambda}_{n_c+1}(0).$$



SVD Compression of PGD

- Large number of modes
- Slow temporal update step

SVD Compression of PGD

- Large number of modes
- Slow temporal update step
- Randomised SVD (RSVD) orthonormalisation step

$$\underline{\underline{\tilde{U}}} = \underline{\underline{V}}\,\underline{\underline{\Lambda}}^{\mathsf{T}} \in \mathbb{R}^{n \times n_t}$$

$$\underline{\underline{V}} = \underline{\underline{Q}}_v\,\underline{\underline{R}}_v, \qquad \underline{\underline{\Lambda}} = \underline{\underline{Q}}_\lambda\,\underline{\underline{R}}_\lambda \qquad \sim \mathcal{O}(\mu^2\,(n + n_t))$$

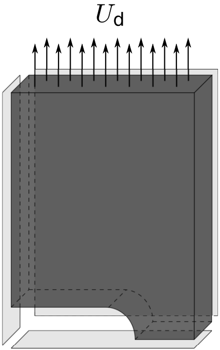
$$\underline{\underline{T}} = \underline{\underline{R}}_v\,\underline{\underline{R}}_\lambda^{\mathsf{T}} \in \mathbb{R}^{\mu \times \mu}, \quad \underline{\underline{T}} \approx \underline{\underline{\tilde{V}}}\,\underline{\underline{\tilde{S}}}\,\underline{\underline{\tilde{\Lambda}}}^{\mathsf{T}}$$

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

Outline

Numerical results

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



Viscoplastic viscodamage material model

- State equations

$$\boldsymbol{\sigma} = (1 - D)\, \mathbb{C} : \boldsymbol{\varepsilon}^{\mathrm{e}}$$

$$\boldsymbol{\beta} = \frac{2}{3}\, c\, \boldsymbol{\alpha}$$

$$R = R_\infty\, (1 - \mathrm{e}^{-b\, r})$$

$$Y = \frac{1}{2}\, \boldsymbol{\varepsilon}^{\mathrm{e}} : \mathbb{C} : \boldsymbol{\varepsilon}^{\mathrm{e}}$$

- Evolution equations

$$\dot{\boldsymbol{\varepsilon}}^{\mathrm{p}} = \dot{\lambda}\, \boldsymbol{N}$$

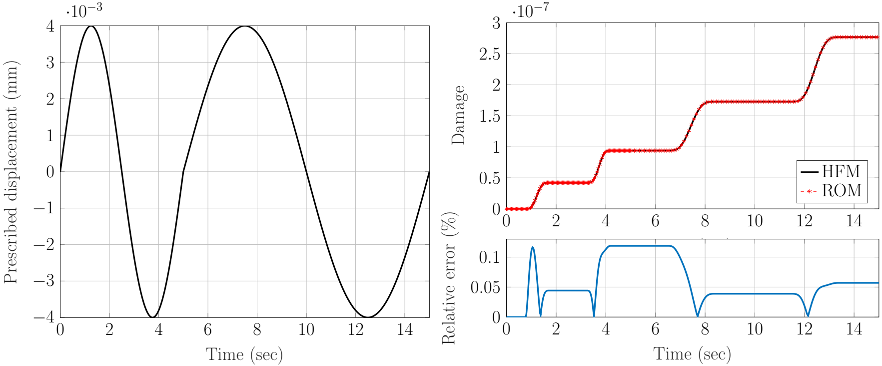
$$\dot{\boldsymbol{\alpha}} = \dot{\lambda}\, \left(\tilde{\boldsymbol{N}} - \frac{3\, a}{2\, c}\, \boldsymbol{\beta} \right)$$

$$\dot{r} = \dot{\lambda}$$

$$\dot{D} = \frac{\dot{\lambda}}{1 - D} \left(\frac{Y}{S} \right)^{s} \text{if } (r > p_{\mathrm{D}})$$

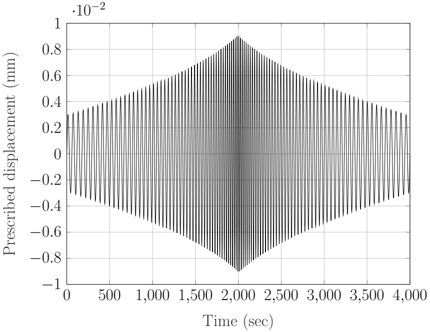
Model verification

- With respect to a classical Newton-Raphson scheme



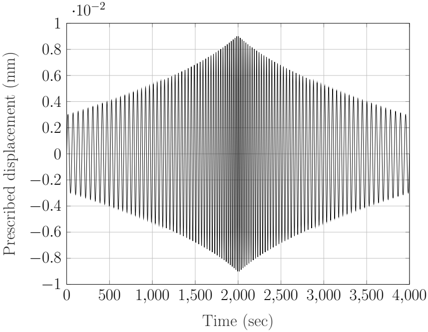
Variable amplitude and frequency loading

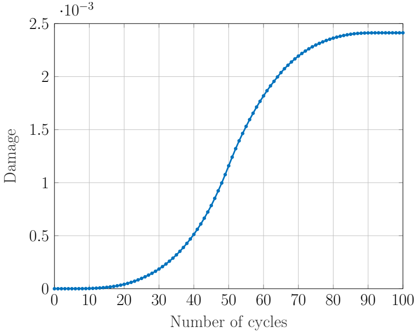
- Amplitudes: $[30, 90] \cdot 10^{-4}\,\mathrm{mm}$ Time periods: $[20, 60]\,\mathrm{sec}$



Variable amplitude and frequency loading

- Amplitudes: $[30, 90] \cdot 10^{-4}\,\mathrm{mm}$ Time periods: $[20, 60]\,\mathrm{sec}$

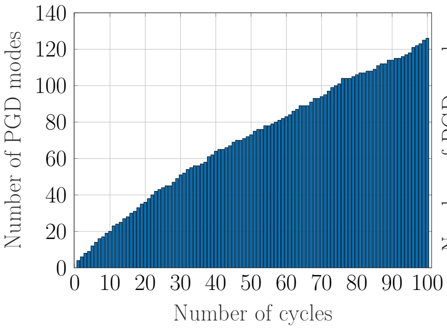




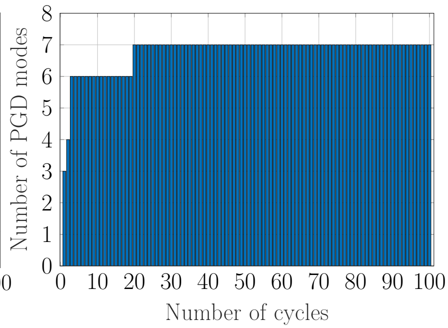
Number of modes is eight

Random amplitude loading

- 100 cycles with different orthonormalisation schemes



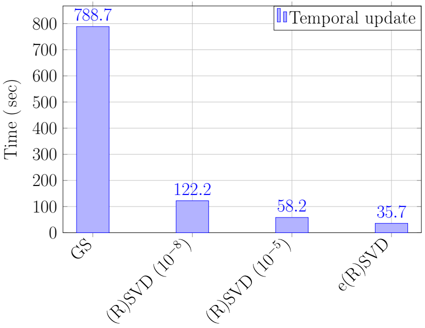
Gram-Schmidt



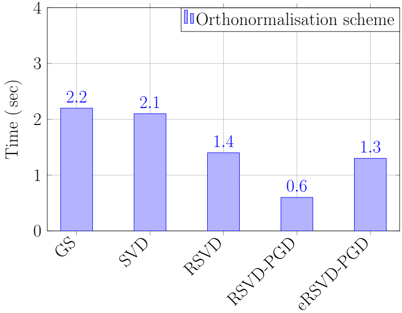
SVD scheme

Random amplitude loading

- Required time



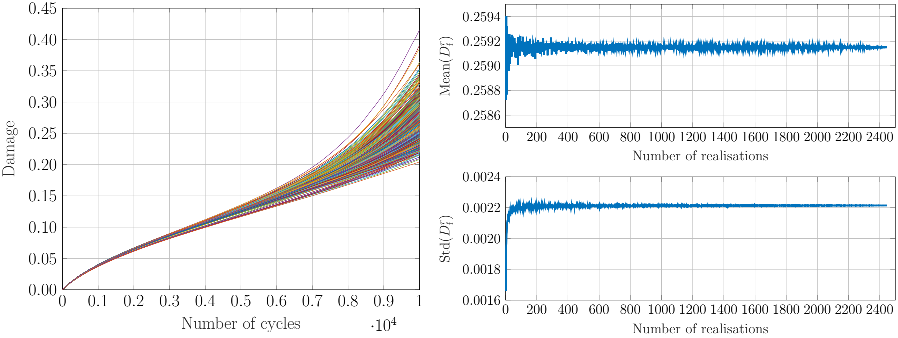
Temporal update



Orthonormalisation step

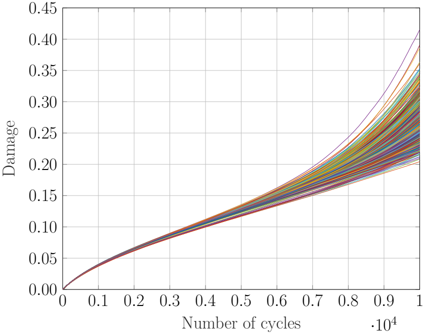
Random amplitude loading

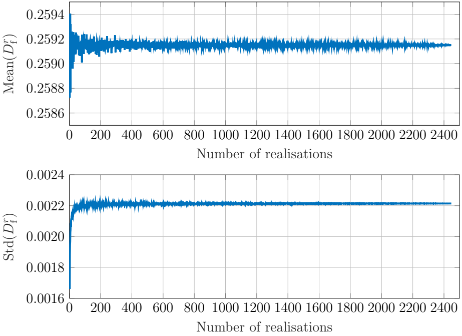
- 10000 cycles, uniform distribution in $[53, 56] \cdot 10^{-2}\,\mathrm{mm}$



Random amplitude loading

- 10000 cycles, uniform distribution in $[53, 56] \cdot 10^{-2}\,\mathrm{mm}$

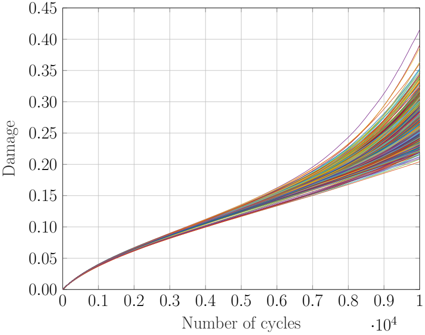


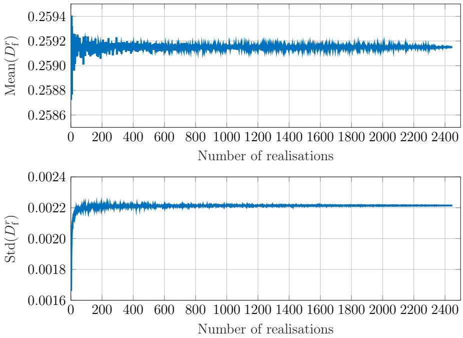


- $D_c = 0.3 \quad \rightarrow \quad P_f = 5.4\%$
- Requirements: $[15 - 35]\,\mathrm{min}$ and $[1 - 1.5]\,\mathrm{GB}$

Random amplitude loading

- 10000 cycles, uniform distribution in $[53, 56] \cdot 10^{-2}\,\mathrm{mm}$





Time-saving factors $50 \sim 100$

Outline

Conclusion and future research

- Efficient cycle by cycle simulation for damage problems
- Open-source: `gitlab.com/shadialameddin/romfem`

Conclusion and future research

- Efficient cycle by cycle simulation for damage problems
- Open-source: `gitlab.com/shadialameddin/romfem`

Challenges

- The computation of the local stage
- The integration of the error indicator

Conclusion and future research

- Efficient cycle by cycle simulation for damage problems
- Open-source: `gitlab.com/shadialameddin/romfem`

Challenges

- The computation of the local stage
- The integration of the error indicator

Future development

- PGD of all QoI in the local stage
- Machine learning to approximate the constitutive manifold

Conclusion and future research

- Efficient cycle by cycle simulation for damage problems
- Open-source: `gitlab.com/shadialameddin/romfem`

Challenges

- The computation of the local stage
- The integration of the error indicator

Future development

- PGD of all QoI in the local stage
- Machine learning to approximate the constitutive manifold

Thank you for your attention

The global stage

- Weak form at iteration $i+1$

$$\int\limits_{\Omega\times\mathcal{I}} \boldsymbol{\sigma}_{i+1} : \boldsymbol{\varepsilon}(u^*)\ \mathrm{d}\Omega\,\mathrm{d}t = \int\limits_{\Omega\times\mathcal{I}} \boldsymbol{b}\cdot u^*\,\mathrm{d}\Omega\,\mathrm{d}t + \int\limits_{[0,T]\times\partial\Omega_N} \bar{\boldsymbol{t}}\cdot u^*\,\mathrm{d}S\,\mathrm{d}t, \quad \forall u^* \in \mathcal{U}_0$$

- Correction $\Delta\bullet_{i+1} = \bullet_{i+1} - \bullet_i$

$$\Delta\boldsymbol{\sigma}_{i+1} - \mathbb{H}^{-}\Delta\boldsymbol{\varepsilon}_{i+1} - \hat{f} = \mathbf{0}, \quad \hat{f} = \underbrace{(\hat{\boldsymbol{\sigma}} - \boldsymbol{\sigma}_i) - \mathbb{H}^{-}(\hat{\boldsymbol{\varepsilon}} - \boldsymbol{\varepsilon}_i)}_{known}$$

$$\int\limits_{\Omega\times\mathcal{I}} \Delta\boldsymbol{\sigma}_{i+1} : \boldsymbol{\varepsilon}(\Delta u^*)\ \mathrm{d}\Omega\,\mathrm{d}t = 0$$

Proper Generalised Decomposition

- Low-rank approximation of the solution

$$u(\boldsymbol{x}, t) = \sum_{i=1}^{N} \lambda_i(t)\; v_i(\boldsymbol{x})$$

- Enriching with one mode

$$\Delta u = \lambda(t)\; v(\boldsymbol{x}) \qquad \Delta u^* = \lambda^*\; v + \lambda\; v^*$$

- Updating (μ) previously generated time modes

$$\Delta u = \sum_{i=1}^{\mu} \Delta\lambda_i(t)\; \underbrace{v_i(\boldsymbol{x})}_{known}$$

Static admissibility

- Static admissibility and global search direction

$$\int_{\Omega\times\mathcal{I}} \mathbb{H}^{-}\Delta\boldsymbol{\varepsilon}_{i+1} : \boldsymbol{\varepsilon}(\Delta u^{*})\ \mathrm{d}\Omega\,\mathrm{d}t = -\int_{\Omega\times\mathcal{I}} \hat{f} : \boldsymbol{\varepsilon}(\Delta u^{*})\ \mathrm{d}\Omega\,\mathrm{d}t$$

- Space problem: $\langle\bullet\rangle = \int_{[0,T]} \bullet\ \mathrm{d}t$, given λ_j

$$\langle\lambda_j\lambda_j\rangle \int_{\Omega} \nabla v^{*} : \mathbb{H}^{-}\nabla v_{j+1}\ \mathrm{d}\Omega = -\langle\lambda_j\rangle \int_{\Omega} \nabla v^{*} : \hat{f}\,\mathrm{d}\Omega$$

- Time problem, given v_{j+1}:

$$\int_{[0,T]} \lambda^{*} \left[\int_{\Omega} \nabla v_{j+1} : \mathbb{H}^{-}\ \nabla v_{j+1}\,\mathrm{d}\Omega\right] \lambda_{j+1}\ \mathrm{d}t = -\langle\lambda^{*}\rangle \int_{\Omega} \nabla v_{j+1} : \hat{f}\,\mathrm{d}\Omega$$