Model reduction for vibrations

Karl Meerbergen

KU LEUVEN

August 9th, 2017

Joint work with Yao Yue and Friends from Engineering and Industry

Outline

Design optimization

Dynamical system:

$$
\begin{aligned}
A(\omega,\gamma)x &= fu(s) \\
y &= d^T x \\
g &= \int_{\omega_{\min}}^{\omega_{\max}} |y|^2 d\omega \quad \text{(energy)}
\end{aligned}
$$

- y/u: frequency response function (usually $u \equiv 1$
- Objective: minimize $g(\gamma)$
- Reduce the cost of computing y and ∇y:
  - Model reduction should be at least as fast the Lanczos eigenvalue solver
  - Allows for cheap computation of g and $\nabla_\gamma g$
- Assume uni-modal objective function

Design optimization

Dynamical system:

$$
\begin{aligned}
A(\omega,\gamma)x &= fu(s) \\
y &= d^T x \\
g &= \int_{\omega_{\min}}^{\omega_{\max}} |y|^2 d\omega \quad \text{(energy)}
\end{aligned}
$$

- y/u: frequency response function (usually $u \equiv 1$
- Objective: minimize $g(\gamma)$
- Reduce the cost of computing y and ∇y:
  - Model reduction should be at least as fast the Lanczos eigenvalue solver
  - Allows for cheap computation of g and $\nabla_\gamma g$
- Assume uni-modal objective function
- Note: more complex energy functions are possible (quadratic output): $y = x^* S x$

  [Saak, 2008] [Van Beeumen, Lombaert, Van Nimmen, M. 2012]

Model reduction for vibrations

- The reference is the Lanczos eigenvalues, used very frequently in simulation software.
- Currently used in industry:
  - Modal truncation
  - Krylov methods (With multiple interpolation)
  - Direct linear system solvers are used: reduce the number of sparse matrix factorization $\rightarrow$ no simple interpolation
- Pole selection:
  - Often one pole works well enough: e.g., 0
  - Approaches based on an error estimation [Bodendiek, Bollhöfer, 2012, 2014], [Feng, Antoulas, Benner, 2015]

Model reduction for vibrations

- The reference is the Lanczos eigenvalues, used very frequently in simulation software.
- Currently used in industry:
  - Modal truncation
  - Krylov methods (With multiple interpolation)
  - Direct linear system solvers are used: reduce the number of sparse matrix factorization $\rightarrow$ no simple interpolation
- Pole selection:
  - Often one pole works well enough: e.g., 0
  - Approaches based on an error estimation [Bodendiek, Bollhöfer, 2012, 2014], [Feng, Antoulas, Benner, 2015]
- Other methods:
  - Dominant pole algorithm (may be useful in a parametric context)
  - IRKA [van de Walle, Van Ophem, Desmet, 2017]
  - Optimal reduction is not required, minimal computation time for sufficient accuracy is the criterion
  - Often one-sided methods only (but not for optimization)
  - Balanced truncation for frequency range [Benner, Kürschner, Saak, 2016]

Left and right Krylov spaces

- Linear system (linear case)

$$\begin{aligned} (A_0 + \omega A_1)x &= f \\ y &= d^T x \end{aligned}$$

- Right Krylov space:

$$A_0^{-1}f, (A_0^{-1}A_1)A_0^{-1}f, \dots, (A_0^{-1}A_1)^{k-1}A_0^{-1}f$$
$$(A_0 + \sigma_1 A_1)^{-1}f, \dots, (A_0 + \sigma_k A_1)^{-1}f$$

  Basis: $V_k = [v_1, \dots, v_k]$

- Left Krylov space:

$$A_0^{-T}d, (A_0^{-T}A_1^T)A_0^{-T}f, \dots, (A_0^{-T}A_1^T)^{k-1}A_0^{-T}f$$
$$(A_0 + \sigma_1 A_1)^{-T}d, \dots, (A_0 + \sigma_k A_1)^{-T}d$$

  Basis: $W_k = [w_1, \dots, w_k]$

- Reduced model:

$$\begin{aligned} (\widehat{A}_0 - \omega \widehat{A}_1)\widehat{x} &= \widehat{f} \\ \widehat{y} &= \widehat{d}^T \widehat{x} \end{aligned}$$

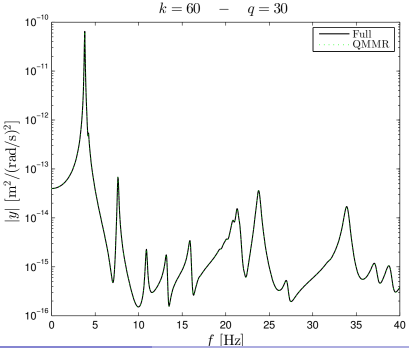
  with $\widehat{A}_i = W_k{}^T A_i V_k$, $\widehat{f} = W_k{}^T f$, $\widehat{d} = V_k{}^T d$.

(See talk by Serkan Gugercin aka Chris Beattie aka Thanos Antoulas)

Example

$$(K + i\omega C - \omega^2 M)x = f$$
$$y = x^T S x$$

Matrix size: $n = 25,962$



- Computation of y with full model: $3,534.5s$ (4001 points)
- Construction of reduced model with $k = 40$ and computation of $\widehat{y}$ require $5.3s$

(Van Beeumen, Van Nimmen, Lombaert, M. 2011)

Gradient

- Dynamical system:

$$\begin{aligned} A(\omega, \gamma)x &= f \\ y &= d^T x \end{aligned}$$

- Compute Right Krylov space V_k

Gradient

- Dynamical system:

$$\begin{aligned} A(\omega, \gamma)x &= f \\ y &= d^T x \end{aligned}$$

- Compute Right Krylov space V_k
- Adjoint system:

$$\begin{aligned} A(\omega, \gamma)^T z &= d \\ y &= f^T z \end{aligned}$$

- Compute Left Krylov space W_k

Gradient

- Build two-sided reduced model for fixed value of γ for
- $y = d^T A(\omega, \gamma_0)^{-1} f$
- Gradient:

$$\frac{dy}{d\gamma} = \left(A(\omega,\gamma_0)^{-*} d\right)^* \left(\frac{dA(\omega,\gamma_0)}{d\gamma}\right) \left(A(\omega,\gamma_0)^{-1} f\right)$$

- Blue part can be computed from the right subspace:

$$A(\omega,\gamma_0)^{-1} f \approx V_k(\widehat{A}^{-1}(\omega,\gamma_0)\widehat{f})$$

- Red part (adjoint) can be computed from the left subspace:

$$A(\omega,\gamma_0)^{-T} d \approx W_k(\widehat{A}^{-T}(\omega,\gamma_0)\widehat{d})$$

- (Parametric) reduced model:

$$\begin{aligned} \widehat{A}(\omega,\gamma)\widehat{x} &= \widehat{f} \\ y &= \widehat{d}^T \widehat{x} \\ \widehat{A} &= {W_k}^T A V_k \\ \widehat{f} = {W_k}^T f \qquad & \widehat{d} = {V_k}^T d \end{aligned}$$

Gradient

- Full gradient

$$\frac{dy}{d\gamma} = \left(A(\omega,\gamma_0)^{-T} d\right)^T \left(\frac{dA(\omega,\gamma_0)}{d\gamma}\right) \left(A(\omega,\gamma_0)^{-1} f\right)$$

- Reduced gradient

$$\begin{aligned} \frac{d\widehat{y}}{d\gamma} &= \left(W_k \widehat{A}(\omega,\gamma_0)^{-T} \widehat{d}\right)^T \left(\frac{dA(\omega,\gamma_0)}{d\gamma}\right) \left(V_k \widehat{A}(\omega,\gamma_0)^{-1} \widehat{f}\right) \\ &= \left(\widehat{A}(\omega,\gamma_0)^{-T} \widehat{d}\right)^T \left(\frac{d\widehat{A}(\omega,\gamma_0)}{d\gamma}\right) \left(\widehat{A}(\omega,\gamma_0)^{-1} \widehat{f}\right) \end{aligned}$$

- Accuracy of the gradient depends on the accuracy for both systems [Antoulas, Beattie, Gugercin 2010] [Yue, M. 2011]
- Almost free computatation of the gradient, regardless the number of parameters!

Gradient

Approximation properties of the frequency response function / transfer function:

- For Rational Krylov: $y(\sigma_i) = \hat{y}(\sigma_i)$ and $y'(\sigma_i) = \hat{y}'(\sigma_i)$
- For Krylov: $y^{(j)}(\sigma_i) = \hat{y}^{(j)}(\sigma_i)$ for $j = 0, \ldots, 2k-1$.

Approximation properties of the gradient

- For Rational Krylov: $y_\gamma(\sigma_i) = \hat{y}_\gamma(\sigma_i)$
- For Krylov: $y_\gamma^{(j)}(\sigma_i) = \hat{y}_\gamma^{(j)}(\sigma_i)$ for $j = 0, \ldots, k-1$.

Optimization and model order reduction

Unimodal optimization

Combine optimization with model reduction: build a surrogate model that is accurate around the current iteration and exploit the model as much as possible.

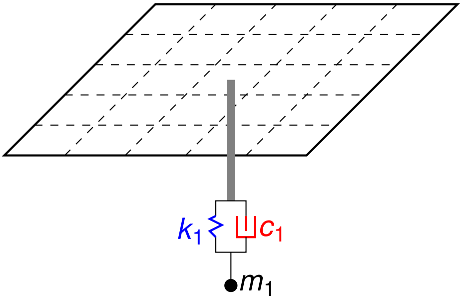
- Gradient is easy to approximate with model reduction by interpolation
- The Hessian is more difficult
- We will not satisfy the first order condition (match function value and gradient in all points)
- No need to first build a parametric reduced model for the entire parameter space: instead integrate MOR with optimization
- Simple interpolation method (in 1 point γ or a line)

Damped BFGS with backtracking

- We use the Damped BFGS optimization method
- On iteration i:
  - Build reduced model for y (and ∇y) at $\gamma = \gamma^{(i)}$
  - Compute g and ∇g from the reduced model
- Objective may not be smooth: use sufficient decrease condition and possibly backtracking after the BFGS step

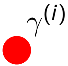
Numerical examples

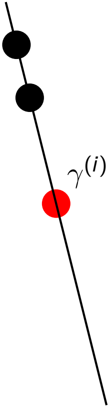
- Floor with damper ($n = 29800$)
- Reduced model $k = 7$
- Determine optimal parameters c_1 and k_1



	Direct method	MOR
Matrix size	29800	7
Optimizer computed	(12231609, 106031.18)	(12231614, 106031.22)
Function value	$1.316093349\,10^{10}$	$1.316093349\,10^{10}$
CPU time	7626s	179s

Overview





- Use MOR for function and gradient evaluation
- Use PMOR for line search optimization + backtracking

Parametric MOR

Dynamical system:

$$\begin{aligned} A(\omega,\gamma)x &= f \\ y &= d^T x \end{aligned}$$

Basis V_k is now built from interpolation points in ω and γ:

$$\begin{array}{ccc} x(\omega_1,\gamma_0) & \cdots & x(\omega_k,\gamma_0) \\ x(\omega_1,\gamma_1) & \cdots & x(\omega_k,\gamma_1) \end{array}$$

Lots of papers. A few names: Feng, Daniel, Bai, Su, Michielsen, ...

(See talk by Chris Beattie)

Parametric MOR

Dynamical system:

$$\begin{aligned} A(\omega,\gamma)x &= f \\ y &= d^T x \end{aligned}$$

Basis V_k is now built from interpolation points in ω and γ:

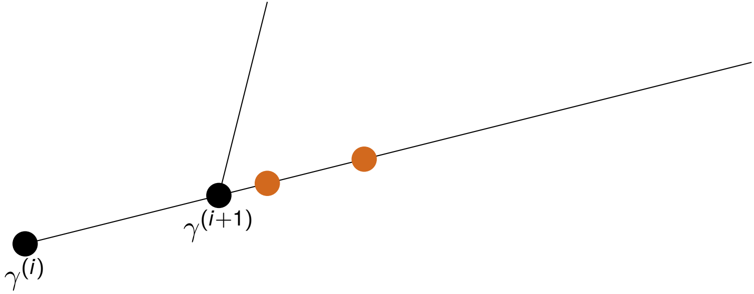
$$\begin{matrix} x(\omega_1,\gamma_0) & \cdots & x(\omega_k,\gamma_0) \\ x(\omega_1,\gamma_1) & \cdots & x(\omega_k,\gamma_1) \end{matrix}$$

Lots of papers. A few names: Feng, Daniel, Bai, Su, Michielsen, ...

(See talk by Chris Beattie (or was it Serkan?))

The MOR/PMOR Framework

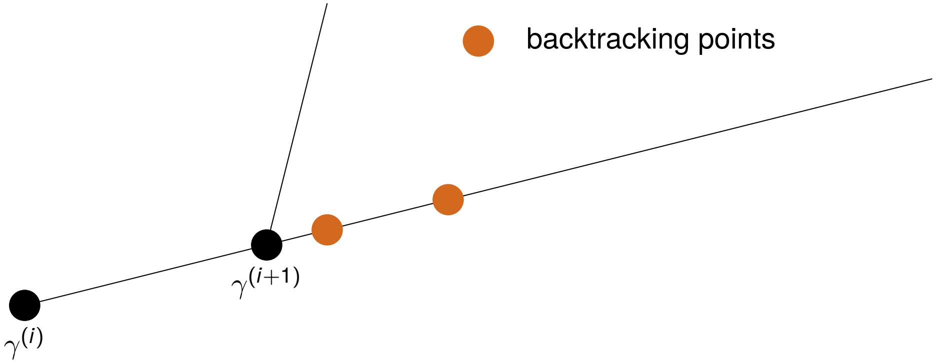
- We can make a reduced model for ω and all parameters γ
- This is usually expensive and overkill: only reduce on the important directions
- We use PIMTAP[1] for efficient line search:



  - The MOR Framework generates a reduced model for each γ accessed.
  - The PMOR Framework generates a reduced model for each line search iteration.

[1] [Li, Bai, Su & Zeng 2007] [Li, Bai, Su, Zeng 2008], [Li, Bai, Su 2009]

PMOR framework

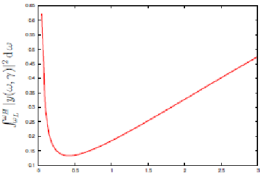


- For backtracking using Armijo line searcher:
  - the MOR Framework is better for smooth objective functions (The first trial point is often accepted for Quasi-Newton methods);
  - the PMOR Framework is better for non-smooth objective functions.
- For smooth objective functions, we can use exact line searches using the PMOR Framework.

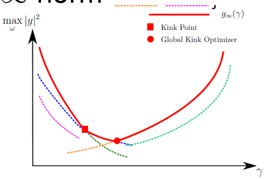
Example: min-max optimization

- Floor optimization problem (2 unknowns)
- Univariate objective function:

2 norm



∞ norm



- Comparison of MOR Framework and PMOR Framework

	2 norm			∞ norm		
	iter	k	Time	iter	k	Time
direct	11 (+6)	29,800	7626	15 (+108)	29,800	41,069
MOR	13 (+8)	7	179	15 (+120)	7	1,104
PMOR	12 (+4)	7+3	360	14 (+90)	7+3	417

[Yue & M. 2011]

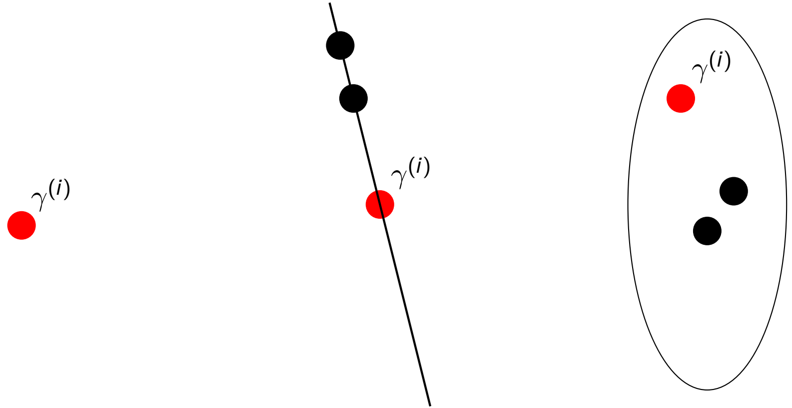
Example: exact line search optimization

- Lamot bridge finite element model ($n = 25,962$)
- The goal is to determine the optimal stiffness and damping coefficient of four bridge dampers (=8 parameters).

- Numerical results

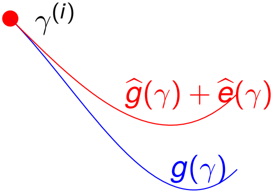
	MOR	(backtracking) PMOR	Exact line search PMOR
order	12	18	18
iterations	70	73	27
time (s)	879	1830	703

Overview



- Use MOR for function and gradient evaluation
- Use PMOR for line search optimization + backtracking
- Use interpolatory MOR for trust region optimization

Trust region approach



- The reduced model interpolates the exact function and the gradient in one point,
- so, we expect the model to be useful in a region around the interpolation point.
- As long as the reduced model + upper bound of the error lead to a decreasing function value, we do not have to recompute a reduced model = penalty.
- On the *i*th iteration:
  1. Build Krylov space for $\gamma = \gamma^{(i)}$
  2. Use the subspace to make a reduced model in ω and γ:

$$\begin{aligned} \widehat{A}(\omega, \gamma)\widehat{x} &= \widehat{f} \\ \widehat{y} &= \widehat{d}^T\widehat{x} \end{aligned}$$

with $\widehat{A} = W_k{}^*AV_k$, $\widehat{f} = W_k{}^*f$, $\widehat{d} = V_k{}^*d$.

Convergence

Provable convergence for unconstrained optimization problem under the '*Relaxed First Order Condition*':

1. MOR method has an error bound on the entire parameter space

$$|g(\gamma) - \hat{g}^{(i)}(\gamma)| \leq e^{(i)}(\gamma) \qquad \|\nabla g(\gamma_i) - \nabla \hat{g}(\gamma_i)\| \leq e_g^{(i)}$$

2. the error can infinitely be reduced

$$e^{(i)}(\gamma_i) \leq \tau_g g(\gamma_i) \qquad e_g^{(i)} \leq \tau_{\nabla g} \|\nabla \hat{g}(\gamma_i)\|$$

3. surrogate model (= reduced model) $\hat{g}(\gamma)$ is smooth with finite gradient everywhere

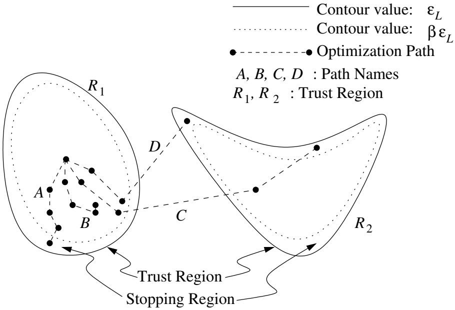
Methods:

- Trust region: $e^{(i)}(\gamma) < \epsilon_L \hat{g}^{(i)}(\gamma)$
- penalty function: $\hat{g}^{(i)} + w\left(\frac{e^{(i)}(\gamma)}{\hat{g}^{(i)}(\gamma)}\right) e^{(i)}(\gamma)$

[Yue & M., 2013]

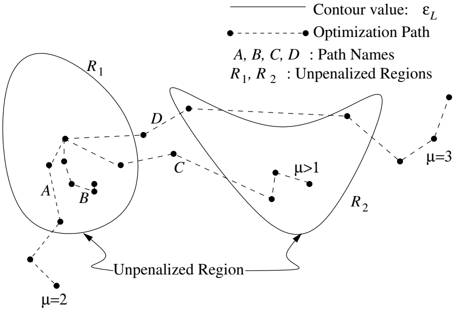
Subproblems

ETR



Terminate if close to boundary.

EP



Terminate if w is active for μ successive steps.

Example of the Lamot footbridge

- Lamot bridge finite element model ($n = 25,962$)
- The goal is to determine the optimal stiffness and damping coefficient of four bridge dampers (=8 parameters).

- Computation times:
  - without reduced modeling: 545 for one function evaluation, 70 function evaluations needed.
  - MOR Framework: 879 sec.
  - ETR: 200 sec.
  - EP: 189 sec.

MIMO formulation for low rank parametric dependency

- Parametric system

$$
\begin{aligned}
(K - \omega^2 M + BC(\omega,\gamma)B^T)x &= f \\
y &= \varphi(x)
\end{aligned}
$$

where B is low rank r

- Reduced model:

$$
\begin{aligned}
(\widehat{K} - \omega^2 \widehat{M} + \widehat{B}C(\omega,\gamma)\widehat{B}^T)\widehat{x} &= \widehat{f} \\
y &= \varphi(V\widehat{x})
\end{aligned}
$$

$$
\widehat{K} = V^T K V \qquad \widehat{M} = V^T M V
$$

$$
\widehat{B} = V^T B \qquad \widehat{f} = V^T f
$$

Block Krylov method for low rank parameters

- Reduced model:

$$\begin{aligned}(\widehat{K} - \omega^2 \widehat{M} + \widehat{B} C(\omega,\gamma) \widehat{B}^T)\widehat{x} &= \widehat{f} \\ y &= \varphi(V\widehat{x}) \\ \widehat{K} = V^T K V \qquad & \widehat{M} = V^T M V \\ \widehat{B} = V^T B \qquad & \widehat{f} = V^T f\end{aligned}$$

  with V Krylov basis for

$$\begin{aligned}(K - \omega^2 M)x &= f - B(C(\omega,\gamma)Bx) \\ &= \begin{bmatrix} f & B \end{bmatrix} \begin{bmatrix} 1 \\ C(\omega,\gamma)Bx \end{bmatrix}\end{aligned}$$

- Approximation properties:
  - For any γ, y is interpolated in the shifts of rational Krylov (derivatives for multiple shifts)
  - γ dependence is fully held by C and therefore represented exactly.
- We need only one reduced model for all optimization steps.

Example

2-norm Optimization of the Footbridge Damper Optimization Problem

	Nr Models	Order	Optimum	CPU Tim
The MOR Framework	70	12	24.77751651	879s
The PMOR Framework	27	18	24.77751661	703s
ETR	3	20	24.78594112	205s
EP	3	20	24.7775166	189s
Block Arnoldi	1	30	24.77751815	20.4s

- Objective at the initial point: 142.34188.
- Damped-BFGS converges in 71 iterations.
- A single evaluation of g costs 540s.

Conclusions

- We proposed three types of methods to accelerate design optimization with (P)MOR.
- Performance:

  Block Arnoldi $>$ *ETR/EP* $>$ *The (P)MOR Framework*

- Applicability:

  The MOR Framework, ETR/EP $>$ *The PMOR Framework, Block Arnoldi*

- Reliability:

  Depends on error estimation for the reduced model

Bibliography

1. K. Meerbergen. The solution of parametrized symmetric linear systems. *SIAM J. Matrix Anal. Appl.*, 24(4):1038–1059, 2003.
2. K. Meerbergen. The Quadratic Arnoldi method for the solution of the quadratic eigenvalue problem. *SIAM Journal on Matrix Analysis and Applications*, 30(4):1463–1482, 2008.
3. K. Meerbergen and Z. Bai. The Lanczos method for parameterized symmetric linear systems with multiple right-hand sides. *SIAM Journal on Matrix Analysis and Applications*, 31(4):1642–1662, 2010.
4. K. Meerbergen, P. Lietaert. Tensor Krylov methods for model reduction of the stochastic mean of a parametric dynamical system, European Control Conference, Linz, 15-17 July 2015.
5. W. Michiels, E. Jarlebring, and K. Meerbergen. Krylov based model order reduction of time-delay systems. *SIAM Journal on Matrix Analysis and Applications*, 32(4):1399–1421, 2011.
6. M. Saadvandi, K. Meerbergen, and W. Desmet, Parametric Dominant Pole Algorithm for Parametric Model Order Reduction, *Journal of Computational and Applied Mathematics*, 295:259–280, 2014.
7. M. Saadvandi, K. Meerbergen, and E. Jarlebring, On dominant poles and model reduction of second order time-delay systems, *Applied Numerical Mathematics*, 62(1):21–34, 2012.
8. R. Van Beeumen and K. Meerbergen. Model reduction by balanced truncation of second order systems with a quadratic output. In T. Simos, editor, *Proceedings of the ICNAAM10 Conference*, 2010.
9. R. Van Beeumen, K. Van Nimmen, G. Lombaert, and K. Meerbergen. Model reduction for dynamical systems with quadratic output. *International Journal of Numerical Methods in Engineering*, 91(3):229–248, 2012.
10. Y. Yue and K. Meerbergen. Using model order reduction for the design optimization of structures and vibrations. *International Journal of Numerical Methods in Engineering*, 90(10):1207–1232, 2012.
11. Y. Yue and K. Meerbergen. Accelerating optimization of parametric linear systems by model order reduction. *SIAM Journal on Optimization*, 23(2):687–1370, 2013.
12. Y. Yue and K. Meerbergen. Parametric model order reduction of damped mechanical systems via the block Arnoldi process. *Applied Mathematics Letters*, 26:643–648, 2013.