Model order reduction for PDE constrained optimization in vibrations

Karl Meerbergen[1] and Yao Yue[2]

KU Leuven, Department of Computer Science, Karl.Meerbergen@cs.kuleuven.be, Yao.Yue@cs.kuleuven.be

Summary. In PDE constrained optimization, physical parameters need to be determined so that some objective function is minimized. We assume here an objective function that depends on the output of a dynamical system, modeled by a discretized PDE. Krylov-Padé model reduction for computing the output can significantly decrease the computation time. In addition, gradients are well approximated, which allows using gradient based optimization on the reduced model. We show numerical results for different methods embedded in line search and trust region methods for benchmark problems from structural engineering.

1 Introduction

Given a dynamical system described by the following system of second order ordinary differential equations:

$$\begin{aligned}(K+sC+s^2M)x &= fu(s) \\ y(s) &= d^T x\end{aligned} \tag{1}$$

where K, C, and M are respectively the stiffness, damping and mass matrices, are sparse, and have size $n \times n$ with n large. The vector f is input and d the output vector. Let the matrices depend on parameters $\gamma \in \mathbb{R}^p$, then we wish to determine the value of γ that minimizes one of the following functions

$$g_2(\gamma) = \int_{\omega_{\min}}^{\omega_{\max}} |y(i\omega)|^2 d\omega \tag{2}$$

$$g_\infty(\gamma) = \max_{\omega_{\min}}^{\omega_{\max}} |y(i\omega)| \tag{3}$$

The evaluation of g is expensive. We will therefore define a reduced model of (1) that approximates y well and is, because of its size, much cheaper to evaluate. The results of this paper are a summary of the full papers [4] [3]. In the remainder of the paper we use $A(s) = K + sC + s^2M$.

2 Krylov-Padé model reduction

One of the most popular model reduction techniques for vibrations are Krylov methods. For second order problems as (1), the SOAR method [1] is preferred. This method builds the matrices $V_k, W_k \in \mathbb{C}^{n\times k}$ with $k \ll n$. The columns of V_k span the derivatives of orders $0,\ldots,k-1$ of $A^{-1}(s)f$ around $s = s_0$. Similarly, the columns of W_k span the derivatives of orders $0,\ldots,k-1$ of $A^{-*}(s)d$ around $s = s_0$. The reduced model is defined as

$$\begin{aligned}(\widehat{K}+s\widehat{C}+s^2\widehat{M})\widehat{x} &= \widehat{f}u(s) \\ \widehat{y}(s) &= \widehat{d}^T\widehat{x}\end{aligned} \tag{4}$$

with $\widehat{K} = W_k^* K V_k$, $\widehat{C} = W_k^* C V_k$, and $\widehat{M} = W_k^* M V_k$, and the vectors $\widehat{f} = W_k^* f$ and $\widehat{d} = V_k^* d$. It can be proven that the first $2k$ derivatives of y and $\widehat{y}$ evaluated at $s = s_0$ match. This property is also known as moment matching.

In order to evaluate g for a given $\gamma = \gamma_\star$, we first build the corresponding reduced model $\widehat{y}$ using the two-sided SOAR method for the given value $\gamma_\star$. The matrices $\widehat{K}, \widehat{C}$, and $\widehat{M}$ then also depend on γ. We hence have an interpolatory reduced model around the interpolation point $\gamma_\star$. It can be shown [4] that $\nabla_\gamma y(\gamma_\star)$ and $\nabla_\gamma \widehat{y}(\gamma_\star)$ match the first k derivatives around s_0. We may thus conclude that two-sided SOAR models compute accurate approximations to y and its gradient s around s_0 and for fixed γ.

Then, g is computed using $\widehat{y}$. For g_2, we use a quadrature rule, and for $g_{\max}$ we use a global optimization method consisting of a coarse grid search and local improvement by a quasi Newton method [4]. The gradient is computed accordingly.

Let us assume $p = 1$, i.e., γ is a single parameter. Another method that we will use for optimization is the PIMTAP method [2]. This is also a moment matching method for s, where in addition, also moments are matched for γ, as well as cross moments, i.e. the derivatives $\partial^{j+i} y / \partial s^j \partial \gamma^i$. The reduced model can then be used to efficiently evaluate y for all s near s_0 and γ near $\gamma_\star$.

3 Line search optimization

In general, (2) is a nonconvex optimization problem. The default method for such problem is problably the damped BFGS method. In iteration j, the $j+1$st iterate is computed as

$$\gamma_{j+1} = \gamma_j + \alpha_j p_j \quad , \quad H_j p_j = -\nabla_\gamma g(\gamma_j)$$

where H_j is the BFGS approximation of the true Hessian of g. For a nonconvex function, we determine α_j

so that the Armijo condition is satisfied. This is a condition that forces sufficient decrease of the objective function in order to achieve convergence. In general, a number of γ_{j+1}'s have to be computed for a sequence of values of α_j's until the Armijo condition is satisfied. This is called backtracking. For each new value of γ, we build a new reduced model using the SOAR method.

Since γ_{j+1} lies on a line in the parameter space, $g(\gamma_{j+1})$ can be efficiently computed when we have a reduced model for both s and $\gamma_j + \alpha p_j$ for α in some interval. This can be achieved by a reduced model using the PIMTAP method.

We used damped BFGS accelerated by SOAR or PIMTAP for minimizing the vibrations in a concrete floor subjected to road noise by determining the best parameters of the floor damper [4]. In this case, there were two parameters: the stiffness and the damping of the floor damper. We see from Table 1 that for the optimization of g_2, which is usually smooth and differentiable, the two-sided SOAR method reaches the best performance, while the minimization of g_∞, which is not a smooth function and therefore requires many backtracking steps, is more efficient using PIMTAP. We also conclude that g_∞ optimization is harder than g_2 optimization, because g_∞ is a nonsmooth function.

	g_2	g_∞
Direct	7,626	41,069
SOAR	179	1,104
PIMTAP	360	417

Table 1. Timings in seconds for the damped BFGS method for the optimization of a concrete floor damper

4 Trust region based optimization

In [3], we discussed trust region approaches exploiting the effort done to build a reduced model. In the SOAR approach, we used the reduced model (4) for evaluating g for a fixed $\gamma = \gamma_j$. The idea here is simple: since $\nabla_\gamma g$ is well approximated at γ_j, the reduced model approximates g well for other values of γ, if g is Lipschitz continuous at γ_j. In contrast to the SOAR approach, we use (4) as a parametric model in s but also in γ, in a trust region setting. The difficulty is that this reduced model is an extrapolation and may therefore quickly lose its accuracy. Therefore, we developed a simple error estimation to control the quality of the reduced model. Since the Hessian is not necessarily well approximated, we rely again on a quasi-Newton method.

We then defined a trust region method, based on the error estimation of the reduced model. The trust region contains the set of parameters γ where the reduced model is accurate. An error-based trust region approach is then used, relying on Cauchy points to guarantee convergence of the method. The solution of the trust region subproblem is cheap, since it fully relies on the reduced model. However, in order to have a provably convergent method, we may sometimes require additional reduced models to refine the trust region [3].

Table 2 compares the SOAR approach and the trust region approach for a model of a footbridge with four dampers. The eight parameters that model these dampers have to be determined so that the vibration in some point on the bridge is minimized. Note that only two reduced models are required, while 70 are needed for the SOAR approach. This leads to an important reduction of the computation time. However, it should be noted that the error estimation of g using the reduced model is much more expensive than evaluating the reduced model. This explains why there is no speed-up of a factor 35.

	Direct	SOAR	Trust region
Time (s)	70×540	897	194
iter.	70	70	2

Table 2. Results for the footbridge problem

Acknowledgement. This paper presents research results supported in part by the Belgian Network DYSCO, funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State Science Policy, and supprted in part by the Research Council KU Leuven grants PFV/10/002 and OT/10/038.

References

1. Z. Bai and Y. Su. Dimension reduction of second-order dynamical systems via a second-order Arnoldi method. *SIAM J. Matrix Anal. and Applic.*, 26(5):1692–1709, 2005.
2. Y.-T. Li, Z. Bai, Y. Su, and X. Zeng. Model order reduction of parameterized interconnect networks via a Two-Directional Arnoldi process. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 27(9):1571–1582, 2008.
3. Y. Yue and K. Meerbergen. Accelerating pde-constrained optimization by model order reduction with error control. Technical report TW611, Department of Computer Science, K.U.Leuven, Celestijnenenlaan 200A, 3001 Heverlee (Leuven), Belgium, 2012.
4. Y. Yue and K. Meerbergen. Using model order reduction for the design optimization of structures and vibrations. *International Journal of Numerical Methods in Engineering*, 2012. In press.