ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Seminar for Applied Mathematics SAM

Adaptive Approximation of Shapes

A. Buffa and R. Hiptmair and P. Panchal

Research Report No. 2020-60
September 2020

Seminar für Angewandte Mathematik
Eidgenössische Technische Hochschule
CH-8092 Zürich
Switzerland

ADAPTIVE APPROXIMATION OF SHAPES

A. BUFFA*, R. HIPTMAIR†, AND P. PANCHAL‡

Abstract. We consider scalar-valued shape functionals on sets of shapes which are small perturbations of a reference shape. The shapes are described by parameterizations and their closeness is induced by a Hilbert space structure on the parameter domain. We justify a heuristic for finding the best low-dimensional parameter subspace with respect to uniformly approximating a given shape functional. We also propose an adaptive algorithm for achieving a prescribed accuracy when representing the shape functional with a small number of shape parameters.

Key words. Model reduction, shape calculus, shape gradient, shape Hessian, low-rank approximation, power iteration

AMS subject classifications. 49Q10, 65F55

1. Shape Model Reduction for Shape Functionals.

1.1. Shape functionals.

Write $\mathcal{A}$ for a set of admissible domains, for instance,

$$\mathcal{A} := \left\{ \overline{\Omega} \subset D : \partial\Omega \text{ is of class } C^{1,1} \right\} ,$$

where $D \subset \mathbb{R}^d$, $d \in \mathbb{N}$, is a simple bounded domain, the so-called hold-all domain. A *shape functional* J is a real-valued mapping $J : \mathcal{A} \to \mathbb{R}$ [9, Section 2.5]. In this work it represents the principal quantity of interest. A simple example is the domain integral functional

$$J_f(\Omega) := \int_\Omega f(\boldsymbol{x}) \,\mathrm{d}\boldsymbol{x} \quad \text{for fixed} \quad f \in L^1(D) . \tag{1.1}$$

More complicated examples involve quantities depending on the solution of a boundary value problem on Ω and those play a central role in PDE-constrained shape optimization [9, Section 2.6], but in this note we keep an exclusive focus on the simple choice (1.1).

1.2. Small shape perturbations.

We consider the shape functional J on "slight perturbations" of a reference domain $\Omega_0 \in \mathcal{A}$. To quantify the notion of a "slight perturbation", we assume that $\mathcal{A}$ is endowed with a *parameterization*, that is, there is a mapping $\Pi : U \subset X \to \mathcal{A}$ from a subset U of a vector space X onto $\mathcal{A}$: $\mathcal{A} = \Pi(U)$. We endow X with a norm $\|\cdot\|$, choose the parameterization Π such that $\Pi(0) = \Omega_0$ and, for "small" $\epsilon > 0$, define the set of ϵ-perturbations of Ω_0 as

$$\mathcal{A}_\epsilon := \{\Omega = \Pi(v) : \ \|v\| < \epsilon\} \subset \mathcal{A} . \tag{1.2}$$

We give two examples of commonly used parameterizations assuming sufficient smoothness of $\partial\Omega_0$:

(i) Parameterization by displacement vectorfield[1]: For $X = \left(C_0^1(D)\right)^d$ we can define

$$\Pi(\boldsymbol{\xi}) := \{\boldsymbol{x} \in D : \boldsymbol{x} = \boldsymbol{x}_0 + \boldsymbol{\xi}(\boldsymbol{x}_0), \boldsymbol{x}_0 \in \Omega_0)\} , \quad \boldsymbol{\xi} \in X . \tag{1.3}$$

If $\|\boldsymbol{\xi}\|_{C^1(D)}$ is small, this will yield a valid $\Pi(\boldsymbol{\xi}) \in \mathcal{A}$.

*MNS, EPF Lausanne, Switzerland, annalisa.buffa@epfl.ch

†SAM, ETH Zurich, hiptmair@sam.math.ethz.ch

‡SAM, ETH Zurich, panchal@sam.math.ethz.ch

[1]We write C_0^k for spaces of k-times continuously differentiable functions with compact support. A subscript 'per' indicates that the functions are assumed to be periodic

(ii) Parameterization by normal displacement [5, (14)]: Writing $\boldsymbol{n} : \partial\Omega_0 \to \mathbb{R}^d$ for the exterior unit normal vectorfield of Ω_0, we set $X := C^1(\partial\Omega)_0$, and define $\Pi(\xi)$, $\xi \in X$, as the domain with boundary

$$\partial\Pi(\xi) := \{\boldsymbol{x} \in \mathbb{R}^d : \ \boldsymbol{x} = \boldsymbol{x}_0 + \xi(\boldsymbol{x}_0)\boldsymbol{n}(\boldsymbol{x}_0),\, \boldsymbol{x}_0 \in \partial\Omega_0\} \,. \tag{1.4}$$

This is meaningful, $\Pi(\xi) \in \mathcal{A}$, for sufficiently small $\|\xi\|_{C^1(\partial\Omega)}$.

Other options are the description of $\partial\Omega$ by closed (spline/trigonometric) curves or deformations governed by velocity fields [1, Chapter 4].

In any case, parameterization takes us to the more conventional setting of studying the functional

$$\widehat{J} := J \circ \Pi : U \subset X \to \mathbb{R} \,, \tag{1.5}$$

defined on a subset of a normed vector space X and, in light of (1.2), that subset specifically is an ϵ-ball around 0 in X.

1.3. Optimal shape representation. Our main concern is an *efficient* representation of domains in $\mathcal{A}_\epsilon := \{\Omega = \Pi(v) : \ \|v\| < \epsilon\}$ as regards the evaluation of a shape functional J: Given a dimension bound $n \in \mathbb{N}$ we would like to find

$$X_n := \operatorname*{argmin}_{\substack{\dim W_n = n \\ W_n \subset X}} \sup_{\substack{v \in X \\ \|v\| < \epsilon}} \inf_{\substack{w \in W_n \\ \|w\| < \epsilon}} |\widehat{J}(v) - \widehat{J}(w)| \,, \tag{1.6}$$

where "$W_n \subset X$" should be read as "W_n is a subspace of X". In words, among the n-dimensional subspaces of X we seek those that yield the best approximation of J *inside the ϵ-ball of X*.

In order to render the optimization problem (1.6) tractable, we first need to resort to more structure on X.

ASSUMPTION 1.1. *The space $(X, \|\cdot\|)$ is a Hilbert space with inner product $(\cdot,\cdot)_X$.*

Secondly, we relax the optimization problem and seek – existence taken for granted –

$$\boxed{X_n := \operatorname*{argmin}_{\substack{\dim W_n = n \\ W_n \subset X}} \sup_{\substack{v \in X \\ \|v\| < \epsilon}} |\widehat{J}(v) - \widehat{J}(\mathsf{P}_n v)|} \,, \tag{1.7}$$

where $\mathsf{P}_n : X \to W_n$ designates the orthogonal projection onto W_n. It is important that the projection is orthogonal, because this ensures $\mathsf{P}_n v \in \mathcal{A}_\epsilon$.

REMARK 1.2. *Note the importance of the constraint $\|w\| \le \epsilon$ in (1.6). If it was absent, simple scaling arguments would yield one-dimensional spaces X_n in many cases: consider homogeneous functionals $\widehat{J}$.*

REMARK 1.3. *Obviously, the "natural" spaces of C^k-functions for parameterization are not compatible with Assumption 1.1. We have to abandon them for the sake of tractability of the problem.*

REMARK 1.4. *Inevitably, in computations the dimension of X will be finite: $N := \dim X < \infty$. Then solving (1.7) amounts to finding a maximally efficient* reduced model *for the representation of a given shape functional on the set $\mathcal{A}_\epsilon$ of perturbed domains.*

2. Low-Rank Approximation of Quadratic Functionals. Throughout this section we study the case that $\widehat{J} := J \circ \Pi : U \subset X \to \mathbb{R}$ is the restriction of a *quadratic functional* on the Hilbert space X to a subset $U \subset X$. More precisely, we study $\widehat{J}$ of the form

$$\widehat{J}(v) = \tfrac{1}{2}\mathsf{a}(v,v) + \ell(v) + \gamma\,, \quad v \in X\,, \tag{2.1}$$

with a symmetric bilinear form $\mathsf{a} : X \times X \to \mathbb{R}$, a linear form $\ell : X \to \mathbb{R}$, and $\gamma \in \mathbb{R}$.

2.1. Finite-dimensional setting. To elucidate ideas, we consider $X = \mathbb{R}^N$ and its inner product given by a symmetric positive definite (s.p.d.) matrix $\mathbf{M} \in \mathbb{R}^{N,N}$. The quadratic functional has the representation

$$\widehat{J}(\mathbf{x}) = \tfrac{1}{2}\mathbf{x}^\top \mathbf{A}\mathbf{x} + \mathbf{p}^\top \mathbf{x} + \gamma\,, \quad \mathbf{x} \in \mathbb{R}^N\,. \tag{2.2}$$

with a *symmetric* matrix $\mathbf{A} = \mathbf{A}^\top \in \mathbb{R}^{N,N}$, a vector $\mathbf{p} \in \mathbb{R}^N$, and $\gamma \in \mathbb{R}$. In the simplest case $\mathbf{p} = \mathbf{0}$, $\gamma = 0$, that is, for a homogeneous quadratic functional, we can find an explicit solution of the optimization problem (1.7). A key tool is the generalized eigenvalue problem for the matrix pair $(\mathbf{A}, \mathbf{M})$:

$$\mathbf{u} \in \mathbb{R}^N \setminus \{\mathbf{0}\},\ \lambda \in \mathbb{R} : \quad \mathbf{A}\mathbf{u} = \lambda \mathbf{M}\mathbf{u}\,. \tag{2.3}$$

In the sequel we call *generalized eigenvectors* of a matrix pair $(\mathbf{A}, \mathbf{M})$ the components of an $\mathbf{M}$-orthonormal N-tuple $(\mathbf{u}_1, \ldots, \mathbf{u}_N) \in X^N$ that solve (2.3) and whose associated *eigenvalues* λ_j are sorted in descending order (in modulus): $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_N|$.

THEOREM 2.1. *For $X := \mathbb{R}^N$ with norm $\|\mathbf{x}\|^2 := \mathbf{x}^\top \mathbf{M}\mathbf{x}$, $\mathbf{M} \in \mathbb{R}^{N,N}$ s.p.d., and fixed dimension $n < N$, a solution of*

$$X_n := \operatorname*{argmin}_{\substack{\dim W_n = n \\ W_n \subset X}} \sup_{\substack{\mathbf{x} \in X \\ \|\mathbf{x}\| < \epsilon}} |\mathbf{x}^\top \mathbf{A}\mathbf{x} - (\mathsf{P}_n \mathbf{x})^\top \mathbf{A}(\mathsf{P}_n \mathbf{x})|\,, \tag{2.4}$$

P_n the orthogonal projection onto W_n, is the span of n generalized eigenvectors for $(\mathbf{A}, \mathbf{M})$ belonging to n largest (in modulus) generalized eigenvalues.

Proof. By homogeneity, we need consider only the case $\epsilon = 1$ and, after switching to an $\mathbf{M}$-orthonormal basis of $\mathbb{R}^N$ we may even assume $\mathbf{M} = \mathbf{I}_N$ without loss of generality, which means that $\|\cdot\|$ boils down to the Euclidean norm $\|\cdot\|_2$ on $\mathbb{R}^N$.

Recall that for every symmetric matrix $\mathbf{T} \in \mathbb{R}^{N,N}$, $\mathbf{T} = \mathbf{T}^\top$, the diagonalization theorem immediately confirms

$$\sup_{\|\mathbf{x}\|_2 \leq 1} |\mathbf{x}^\top \mathbf{T}\mathbf{x}| = \|\mathbf{T}\|_2 = \sup_{\|\mathbf{x}\|_2 \leq 1} \|\mathbf{T}\mathbf{x}\|_2 \tag{2.5}$$

The singular-value decomposition (SVD) of $\mathbf{A}$ is given by

$$\mathbf{A} = \sum_{k=1}^{N} |\lambda_k| \mathbf{u}_k \mathbf{v}_k^\top\,, \quad \mathbf{v}_k := \operatorname{sgn}(\lambda_k)\mathbf{u}_k\,, \tag{2.6}$$

where $(\mathbf{u}_1, \ldots, \mathbf{u}_N)$ are orthonormal eigenvectors of $\mathbf{A}$ with associated eigenvalues λ_k, $k = 1, \ldots, N$, sorted such that $|\lambda_k| \geq |\lambda_{k+1}|$. The SVD provides the best rank-n approximation of $\mathbf{A}$ [6, Thm. 2.4.8]

$$\operatorname{argmin}\{\|\mathbf{A} - \mathbf{A}_n\|_2 :\ \operatorname{rank}(\mathbf{A}_n) \leq n\} = \sum_{k=1}^{n} |\lambda_k| \mathbf{u}_k \mathbf{v}_k^\top = \sum_{k=1}^{n} \lambda_k \mathbf{u}_k \mathbf{u}_k^\top := \mathbf{A}_n^*\,. \tag{2.7}$$

By virtue of orthonormality of $(\mathbf{u}_1, \ldots, \mathbf{u}_N)$, the orthogonal projection onto $\operatorname{span}\{\mathbf{u}_1, \ldots, \mathbf{u}_n\}$ has the matrix representation

$$\mathsf{P}_n \sim \sum_{k=1}^{n} \mathbf{u}_k \mathbf{u}_k^\top \,, \tag{2.8}$$

which, together with (2.6), implies

$$(\mathsf{P}_n\mathbf{x})^\top \mathbf{A}(\mathsf{P}_n\mathbf{x}) = \sum_{k=1}^{n} \lambda_k (\mathbf{u}_k^\top \mathbf{x})^2 = \mathbf{x}^\top \mathbf{A}_n^* \mathbf{x} \,. \tag{2.9}$$

Finally, from (2.7) we conclude

$$\sup_{\|\mathbf{x}\|_2 \le 1} |\mathbf{x}^\top \mathbf{A}\mathbf{x} - \mathbf{x}^\top \mathbf{A}_n^* \mathbf{x}| = \min\{ \sup_{\|\mathbf{x}\|_2 \le 1} |\mathbf{x}^\top \mathbf{A}\mathbf{x} - \mathbf{x}^\top \mathbf{A}_n \mathbf{x}| : \ \operatorname{rank}(\mathbf{A}_n) \le n \} \,. \tag{2.10}$$

Since every expression $\mathbf{x} \mapsto (\mathsf{P}\mathbf{x})^\top \mathbf{A}(\mathsf{P}\mathbf{x})$, where P is a projection onto an n-dimensional subspace of $\mathbb{R}^N$, can be written as $\mathbf{x} \mapsto \mathbf{x}^\top \mathbf{A}_n \mathbf{x}^\top$ with a symmetric matrix of rank $\le n$, (2.10) gives the assertion of the theorem. ∎

A closer scrutiny of the proof reveals that the generalized eigenvalues λ_k of $(\mathbf{A}, \mathbf{M})$ allow to predict the model reduction error [6, (2.4.4)]:

$$\min_{\substack{\dim W_n = n \\ W_n \subset X}} \sup_{\substack{\mathbf{x} \in X \\ \|\mathbf{x}\| < \epsilon}} |\mathbf{x}^\top \mathbf{A}\mathbf{x} - (\mathsf{P}_n\mathbf{x})^\top \mathbf{A}(\mathsf{P}_n\mathbf{x})| = \epsilon^2 \lambda_{n+1} \,. \tag{2.11}$$

Another special case is that of a purely linear functional: $\mathbf{A} = \mathbf{O}$, $\mathbf{p} \neq \mathbf{0}$ in (2.2), that is, $\widehat{J}(\mathbf{x}) = \mathbf{p}^\top \mathbf{x}$. The obvious solution in this case is

$$X_n := \operatorname{span}\{\mathbf{q}\} \quad , \quad \mathbf{q} := \mathbf{M}^{-1}\mathbf{p} \in \mathbb{R}^N \,, \tag{2.12}$$

because the $\mathbf{M}$-orthogonal projection P_n onto that X_n is

$$\mathsf{P}_n(\mathbf{x}) = \mathbf{q} \frac{\mathbf{p}^\top \mathbf{x}}{\mathbf{q}^\top \mathbf{p}} \,, \quad \mathbf{x} \in \mathbb{R}^N \,, \tag{2.13}$$

and simple algebra yields

$$\widehat{J}(\mathsf{P}_n\mathbf{x}) = \mathbf{p}^\top \mathbf{x} = \widehat{J}(\mathbf{x}) \quad \forall \mathbf{x} \in \mathbb{R}^N \,. \tag{2.14}$$

Moreover, (2.13) still holds, if only $\mathbf{q} \in X_n$.

Now, let us discuss the generic case. Unfortunately, apparently there is no algorithm to determine solutions for (1.7) for $\widehat{J}$ from (2.2) with $\mathbf{p} \neq \mathbf{0}$ and $\mathbf{A} \neq \mathbf{O}$. Our heuristic appeals to the observation that the leading term is linear

$$\widehat{J}(\mathbf{x}) = \mathbf{p}^\top \mathbf{x} + \gamma + O(\|\mathbf{x}\|^2) \quad \text{for} \quad \mathbf{x} \to \mathbf{0} \,.$$

We conclude that we have to capture the linear term. The considerations of the previous paragraph suggest that we make sure that $\mathbf{q} \in X_n$ a priori. Then, at worst, we are off the optimal choice of the subspace by one dimension. In fact, demanding $\mathbf{q} \in X_n$ already ensures

$$\widehat{J}(\mathbf{x}) - \widehat{J}(\mathsf{P}_n\mathbf{x}) = O(\|\mathbf{x}\|^2) \quad \text{for} \quad \mathbf{x} \to \mathbf{0} \,.$$

On the $\mathbf{M}$-orthogonal complement of $\mathbf{q}$ we face a homogeneous quadratic functional. Thus, we can take the cue from Theorem 2.1 and propose the following choice of X_n in the finite-dimensional setting, for the quadratic functional J from (2.2) with $\mathbf{q} \neq \mathbf{0}$.

$$X_n = \operatorname{span}\{\mathbf{q}, \mathbf{u}_1, \dots, \mathbf{u}_{n-1}\} , \tag{2.15}$$

where $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_N \in \mathbb{R}^N$ are generalized eigenvectors[a] of the matrix pair $(\widetilde{\mathbf{A}}, \mathbf{M})$ and $\widetilde{\mathbf{A}} \in \mathbb{R}^{N,N}$ is the projected matrix

$$\widetilde{\mathbf{A}} := \left(\mathbf{I} - \frac{\mathbf{q}\,\mathbf{p}^\top}{\mathbf{q}^\top \mathbf{p}}\right) \mathbf{A} \left(\mathbf{I} - \frac{\mathbf{q}\,\mathbf{p}^\top}{\mathbf{q}^\top \mathbf{p}}\right) . \tag{2.16}$$

[a] Generalized eigenvectors have been introduced in (2.3).

Recall that the $\mathbf{u}_j$ are non-zero solutions of $\widetilde{\mathbf{A}}\mathbf{u} = \lambda \mathbf{M}\mathbf{u}$, $\lambda \in \mathbb{R}$, *sorted* by descending modulus of the associated generalized eigenvalues $\lambda_1, \lambda_2, \dots, 0$. For the choice (2.15) the eigenvalues provide some information about the model reduction error, because for X_n, $n \geq 2$, according to (2.15) and $\mathsf{P}_n : \mathbb{R}^N \to X_n$ the $\mathbf{M}$-orthogonal projection, we can bound

$$\sup_{\substack{\mathbf{x} \in X,\, \mathbf{x} \perp \mathbf{q} \\ \|\mathbf{x}\| < \epsilon}} \left| \widehat{J}(\mathbf{x}) - \widehat{J}(\mathsf{P}_n \mathbf{x}) \right| = \epsilon^2 |\lambda_n| \quad \forall \epsilon \geq 0 . \tag{2.17}$$

It is crucial to note the $\mathbf{M}$-orthogonality to $\mathbf{q}$ in the supremum; the estimate does not cover the full set of small perturbations.

2.2. Infinite-dimensional (function-space) setting. Having dealt with the case of finite-dimensional spaces X, our insights carry over to $\dim X = \infty$ provided that the quadratic functional $\widehat{J}$ from (2.1) has a "finite-dimensional flavor". This motivates the following assumption.

ASSUMPTION 2.2. *We assume that the bilinear form* $\mathsf{a} : X \times X \to \mathbb{R}$ *is* bounded *and* compact.

Recall, that a bounded bilinear form c on a Hilbert space X is called compact, when the bounded linear operator $\mathsf{T} : X \to X$ defined by $(\mathsf{T}x, y)_X = \mathsf{c}(x, y)$, $x, y \in X$, is compact [10, II.3].

For a compact symmetric bilinear form $\mathsf{c} : X \times X \to \mathbb{R}$ Riesz-Schauder spectral theory for compact operators [10, Theorem VI.3.2] gives us the existence of an *orthonormal basis* $(u_j)_{j \in \mathbb{N}}$ of X and of a sequence of real numbers $\lambda_j \in \mathbb{R}$, with

$$|\lambda_j| \geq |\lambda_{j+1}| \quad \text{and} \quad \mathsf{c}(u_j, v) = \lambda_j \, (u_j, v)_X \quad \forall v \in X . \tag{2.18}$$

This is the infinite-dimensional counterpart of the generalized eigenvalue problem for matrices as introduced in (2.3). We remind that $\lim_{j \to \infty} \lambda_j = 0$.

In order to generalize the construction (2.15), we let $q \in X$ be the Riesz representative of the continuous linear form $\ell : X \to \mathbb{R}$: $(q, x)_X = \ell(x)$ for all $x \in X$. We suppose $\ell \neq 0$, equivalently $q \neq 0$, and introduce the *projected bilinear form*

$$\widetilde{\mathsf{a}}(w, v) := \mathsf{a}\left(w - q\frac{(q, w)_X}{(q, q)_X}, v - q\frac{(q, v)_X}{(q, q)_X}\right) , \quad w, v \in X , \tag{2.19}$$

which is a rank-1 modification of a and, thus, inherits compactness from a. This paves the way for adapting the construction (2.15):

As reduced space we choose

$$X_n := \operatorname{span}\{q, u_1, \dots, u_{n-1}\}\,, \tag{2.20}$$

with an orthonormal basis $(u_j)_{j\in\mathbb{N}}$ of X provided by a sequence of eigenvectors for the generalized eigenvalue problem

$$\lambda_j \in \mathbb{R},\ u_j \in X \setminus \{0\}: \quad \widetilde{\mathsf{a}}(u_j, v) = \lambda_j\,(u_j, v)_X \quad \forall v \in X\,, \quad j \in \mathbb{N}\,, \tag{2.21}$$

sorted such that $|\lambda_j| \geq |\lambda_{j+1}|$.

A partial error estimate analogous to (2.17) holds:

$$\sup_{\substack{x\in X,\, x\perp q\\ \|x\|<\epsilon}} \left|\widehat{J}(x) - \widehat{J}(\mathsf{P}_n x)\right| = \epsilon^2\lambda_n \quad \forall \epsilon \geq 0\,. \tag{2.22}$$

2.3. Adaptive algorithm. Shifting focus, we may also tackle the following task of efficient representation of the quadratic functional $\widehat{J}$ from (2.1) with Riesz representative $q \in X$ for the linear functional ℓ:

Given $\epsilon > 0$, $\delta > 0$, seek a subspace $X_\delta \subset X$ of *finite minimal dimension* such that $q \in X_\delta$ and

$$\sup_{\substack{x\in X\\ \|x\|<\epsilon}} \left|\widehat{J}(x) - \widehat{J}(\mathsf{P}_\delta x)\right| \leq \delta\,, \tag{2.23}$$

where $\mathsf{P}_\delta : X \to X_\delta$ is the orthogonal projection onto X_δ.

The heuristic considerations of the previous section, in particular (2.20) and (2.22), offer clear guidance: Choose X_δ as the span of q and all those generalized eigenfunctions u_j of (2.21) for which $\epsilon^2|\lambda_j| > \delta$.

For computing the eigenfunctions belonging to the largest (in modulus) eigenvalues we propose an algorithm based on *subspace power iterations*. Remember that the simple power iteration for a generalized eigenvalue problem (2.21) boils down to the following recursion for $j = 0, 1, 2, \dots$ [6, Sect. 8.2.1]:

$$\widetilde{u} \in X: \quad (\widetilde{u}, v)_X := \widetilde{\mathsf{a}}(u^{(j)}, v) \quad \forall v \in X\,, \quad u^{(j+1)} := \frac{\widetilde{u}}{\|\widetilde{u}\|}\,, \tag{2.24}$$

generating a sequence $u^{(j)}$ starting with some, usually random, initial guess $u^{(0)}$. Barring higher multiplicity of the largest (in modulus) generalized eigenvalue, it converges to an associated eigenvector. The iteration is terminated once the relative or absolute change of the approximate eigenvalue

$$\lambda_{\max}^{(j)} := \frac{\widetilde{\mathsf{a}}(u^{(j)}, u^{(j)})}{\big(u^{(j)}, u^{(j)}\big)_X}\,, \quad j \in \mathbb{N}_0\,, \tag{2.25}$$

drops below a prescribed threshold $\tau_{\rm rel} > 0$:

$$\text{STOP, when} \qquad \frac{|\lambda_{\max}^{(j)} - \lambda_{\max}^{(j-1)}|}{|\lambda_{\max}^{(j)}|} < \tau_{\rm rel} \ . \tag{2.26}$$

If an orthonormal system of eigenvectors belonging to m of the largest eigenvalues is desired, the power iteration can be extended to subspaces [6, Sect. 8.2]. The j-th step of that subspace version starts from $m \in \mathbb{N}$ orthonormal approximate eigenvectors $u_1^{(j)}, \ldots, u_m^{(j)} \in X$ and, as in (2.24), first computes

$$\widetilde{u}_i \in X : \quad (\widetilde{u}_i, v)_X := \widetilde{\mathsf{a}}(u_i^{(j)}, v) \quad \forall v \in X \ , \quad i = 1, \ldots, m \ . \tag{2.27}$$

The simple normalization step in (2.24) is replaced with the solution of a more complicated projected eigenvalue problem the so-called Rayleigh-Ritz step that can be viewed as the Ritz-Galerkin discretization of (2.21) on $\operatorname{span}\{\widetilde{u}_1, \ldots, \widetilde{u}_m\}$. Based on the Galerkin matrices

$$\mathbf{A}_m := [\widetilde{\mathsf{a}}(\widetilde{u}_k, \widetilde{u}_l)]_{k,l=1}^m \in \mathbb{R}^{m,m} \ , \tag{2.28}$$

$$\mathbf{M}_m := [(\widetilde{u}_k, \widetilde{u}_l)_X]_{k,l=1}^m \in \mathbb{R}^{m,m} \ , \tag{2.29}$$

we solve the $m \times m$ generalized matrix eigenvalue problem

$$\mathbf{W} \in \mathbb{R}^{m,m} : \quad \mathbf{A}_m \mathbf{W} = \mathbf{M}_m \mathbf{W} \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_m \end{bmatrix} , \quad \mathbf{W}^\top \mathbf{M}_m \mathbf{W} = \mathbf{I}_m \ , \tag{2.30}$$

where we assume that the real eigenvalues $\lambda_1, \ldots, \lambda_m$ are sorted in descending order: $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_m|$. The entries of the eigenvector matrix $\mathbf{W}$ give the weights of the linear combinations yielding the new approximate eigenvectors

$$u_i^{(j+1)} = \sum_{k=1}^m (\mathbf{W})_{k,i} \, \widetilde{u}_k \ , \quad i = 1, \ldots, m \ , \quad \widetilde{u}_k \text{ from (2.27)}. \tag{2.31}$$

Note that the $\lambda_i \in \mathbb{R}$ as obtained in (2.30) also provide approximations of the eigenvalues, for $m = 1$ we recover (2.24). Therefore they can be used as input for a termination criterion analogous to (2.26). We also point out that it is recommended to choose m larger than the number of eigenvectors/eigenvalues that one is actually interested in.

Algorithm 1: Adaptive resolution of a quadratic functional

```
function adapt(real ϵ, real δ,
               real τ_rel, real τ_abs) {
  Start with orthonormal set {q, u_1, u_2, u_3} ⊂ X.
  L := 0;
  do {
    L := L + 1; // Current dimension of subspace X_δ
    m := L + 2; // Dimension of slightly enlarged subspace
    D := O ∈ R^{m,m}; // Zero matrix
    j := 0;
    set_random := false; // Flag for random reshuffling
```

```
11      do {
12          cont_loop := false;
13          j := j + 1;  // step counter for subspace power iteration
14          Compute ũ_i ∈ X : (ũ_i, v)_X = ã(u_i, w) ∀v ∈ X, i = 1,...,m; // (2.27)
15          Compute A_m := [ã(ũ_k, ũ_ℓ)]_{k,ℓ=1}^m ∈ R^{m,m}; // (2.28)
16          Compute M_m := [(ũ_k, ũ_ℓ)_X]_{k,ℓ=1}^m ∈ R^{m,m};  // (2.29)
17          D_old := D;
18          Rayleigh-Ritz step: solve (2.30), set D := diag(λ_1,...,λ_m);
19          Compute u_i := Σ_{k=1}^m (W)_{k,i} ũ_k ,   i = 1,...,m;  // (2.31)
20          Z := {j ∈ {1,...,m} : |λ_j/λ_1| ≤ 0.01 · δ};
21          if (Z ≠ ∅) {
22              if (!set_random) { // random reshuffling only once
23                  Replace u_j, j ∈ Z, with random vectors
24                  such that {u_1,...,u_m} is M-orthonormal
25                  set_random := true;
26                  cont_loop := true;
27              }
28              else {
29                low_rank := true; // low numerical rank of a
30                L := m − ♯Z;  // detect effective rank
31              }
32          }
33          d_rel := max_{i∈{1,...,L}} |(D)_{i,i} − (D_old)_{i,i}| / |(D)_{i,i}|;  // relative change
34          d_abs := max_{i∈{1,...,L}} |(D)_{i,i} − (D_old)_{i,i}|;  // absolute change
35      }
36      while (cont_loop or (!low_rank and d_abs > τ_abs and
            d_rel > τ_rel));
37      // Enlarge subspace
38      Choose (random) u_{m+1} ∈ X, ‖u_{m+1}‖ = 1, u_{m+1} ⊥ span{u_1, u_2,..., u_m};
39    }
40    while (!low_rank and ε^2 |(D)_{L,L}| > δ);
41    return (u_1,...,u_L);
42  }
```

Algorithm 1 calls for additional explanations. Since $\widetilde{\mathsf{a}}$ need not be definite, some of the Ritz values may be very close to zero. In this case we reset the power iteration by replacing the associated Ritz vectors with random directions, see line 22 of the pseudo-code. We terminate the inner power iteration upon detecting a small relative or absolute change of the Ritz values. We also stop the power iteration, once Ritz values become very small despite a random reshuffling of the subspace. This hints at $\mathbf{A}_m$ having low (numerical) rank, and in this case we should quit the entire iteration and return the Ritz vectors found so far, which will span a subspace orthogonal to the (effective) nullspace of a.

REMARK 2.3. *The symmetric Lanczos process [6, Sect. 10.1-10.3] could be an alternative to subspace power iteration, because, when applied to the generalized eigenvalue problem (2.21), after $> k$ iterations it will have assembled a subspace that is close to the subspace spanned by $u_1, u_2, \ldots, u_k$.*

3. Quadratic Approximation of Shape Functionals. The methods of section 2 are confined to quadratic functional on X of the form (2.1). Yet, most relevant shape functionals $\widehat{J}$ will not be quadratic. Our idea is to apply the techniques of section 2 to a quadratic approximation of the shape functional $\widehat{J}$ from (1.5). This approximation can be computed as third-order "shape Taylor approximation of J at Ω_0" using the tools of *shape calculus*. The validity of this approximation hinges on the assumption of small perturbations as discussed in subsection 1.2.

We adopt the setting of subsection 1.1. As already announced there, our focus is on domain integral shape functionals $J_f : \mathcal{A} \to \mathbb{R}$ as in (1.1) assuming that the integrand f is piecewise smooth: $f \in C^1_{\mathrm{pw}}(\overline{D})$. We adopt the perturbation of identity approach of shape calculus [2]. Temporarily, we fix a compactly supported perturbation vectorfield $\boldsymbol{\mathcal{V}} \in (W^{1,\infty}_0(D))^d$, and define a one-parameter family of diffeomorphisms.

$$\phi^t : D \mapsto D\ , \quad |t| < \tau\ , \quad \phi^t(\boldsymbol{x}) = \phi(t,\boldsymbol{x}) := \boldsymbol{x} + t\boldsymbol{\mathcal{V}}(\boldsymbol{x})\ , \tag{3.1}$$

where $\tau > 0$ is sufficiently small.

As in subsection 1.2 we choose a reference domain Ω_0 and consider the shape functional along a "curve in shape space":

$$\mathcal{J}_{\boldsymbol{\mathcal{V}}}(t) := J_f(\phi^t(\Omega_0)) = \int_{\phi^t(\Omega_0)} f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} = \int_{\Omega_0} f(\phi(t,\boldsymbol{x}))\,\omega(t,\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}\ , \quad |t| < \tau\ , \tag{3.2}$$

with the (positive) Gram determinant (metric factor)

$$\omega(t,\boldsymbol{x}) := \det \frac{\partial \phi}{\partial \boldsymbol{x}}(t,\boldsymbol{x})\ , \quad \boldsymbol{x} \in \Omega_0\ , \quad |t| < \tau\ . \tag{3.3}$$

Apart from the straightforward derviatives

$$\frac{\partial \phi}{\partial t}(t,\boldsymbol{x}) = \boldsymbol{\mathcal{V}}(\boldsymbol{x})\ , \quad \frac{\partial^2 \phi}{\partial t^2}(t,\boldsymbol{x}) = 0\ , \quad \frac{\partial \phi}{\partial \boldsymbol{x}}(t,\boldsymbol{x}) = \boldsymbol{I} + t\mathsf{D}\boldsymbol{\mathcal{V}}(\boldsymbol{x})\ , \tag{3.4}$$

where $\mathsf{D}\boldsymbol{\mathcal{V}}$ is the Jacobian of the vectorfied $\boldsymbol{\mathcal{V}}$, from [8, Sect. 2.2] we learn

$$\frac{\partial \omega}{\partial t}(0,\boldsymbol{x}) = \operatorname{div} \boldsymbol{\mathcal{V}}(\boldsymbol{x})\ , \tag{3.5}$$

$$\frac{\partial^2 \omega}{\partial t^2}(0,\boldsymbol{x}) = \big(\operatorname{div} \boldsymbol{\mathcal{V}}(\boldsymbol{x})\big)^2 - \operatorname{Tr}\left(\mathsf{D}\boldsymbol{\mathcal{V}}(\boldsymbol{x})^\top \mathsf{D}\boldsymbol{\mathcal{V}}(\boldsymbol{x})\right)\ . \tag{3.6}$$

The first directional shape derivative of J_f in the direction of $\boldsymbol{\mathcal{V}}$ evaluates to

$$\begin{aligned}
\left\langle \frac{dJ_f}{d\Omega}(\Omega_0); \boldsymbol{\mathcal{V}} \right\rangle &:= \frac{d\mathcal{J}_{\boldsymbol{\mathcal{V}}}}{dt}(0) = \int_{\Omega_0} \frac{\partial}{\partial t}\{f(\phi(t,\boldsymbol{x}))\omega(t,\boldsymbol{x})\}\Big|_{t=0}\,\mathrm{d}\boldsymbol{x} \\
&= \int_{\Omega_0} \frac{\partial (f\circ\phi)}{\partial t}(0,\boldsymbol{x})\,\omega(0,\boldsymbol{x}) + f(\phi(0,\boldsymbol{x}))\,\frac{\partial \omega}{\partial t}(0,\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} \\
&= \int_{\Omega_0} \boldsymbol{\nabla} f(\boldsymbol{x})\cdot\boldsymbol{\mathcal{V}}(\boldsymbol{x}) + f(\boldsymbol{x})\,\operatorname{div}\boldsymbol{\mathcal{V}}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} = \int_{\partial\Omega_0} f(\boldsymbol{x})\boldsymbol{\mathcal{V}}(\boldsymbol{x})\cdot\boldsymbol{n}(\boldsymbol{x})\,\mathrm{dS}(\boldsymbol{x})\ .
\end{aligned}$$

The final expression is the so-called Hadamard form of the directional shape gradient, involving only the normal component of $\boldsymbol{\mathcal{V}}$ on $\partial\Omega_0$. Similar though more complicated

manipulations yield the directional shape Hessian [8, Sect. 4.2]:

$$
\begin{aligned}
\left\langle \frac{d^2 J_f}{d\Omega^2}; \boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{V}} \right\rangle &:= \frac{d^2 \mathcal{J}_{\boldsymbol{\mathcal{V}}}}{dt^2}(0) = \frac{d^2}{dt^2} \int_{\Omega_0} f(\phi(t,\boldsymbol{x}))\, \omega(t,\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}\Big|_{t=0} \\
&= \frac{d}{dt} \int_{\Omega_0} \nabla f(\phi(t,\boldsymbol{x})) \cdot \boldsymbol{\mathcal{V}}(\boldsymbol{x})\, \omega(t,\boldsymbol{x}) + f(\phi(t,\boldsymbol{x})) \frac{\partial \omega}{\partial t}(t,\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}\Big|_{t=0} \\
&= \int_{\Omega_0} \Big\{ \left\langle \mathsf{D}^2 f(\boldsymbol{x}); \boldsymbol{\mathcal{V}}(\boldsymbol{x}), \boldsymbol{\mathcal{V}}(\boldsymbol{x}) \right\rangle + 2 \nabla f(\boldsymbol{x}) \cdot \boldsymbol{\mathcal{V}}(\boldsymbol{x})\ \mathrm{div}\, \boldsymbol{\mathcal{V}}(\boldsymbol{x}) + \\
&\qquad f(\boldsymbol{x}) \frac{\partial^2 \omega}{\partial t^2}(0,\boldsymbol{x}) \Big\}\, \mathrm{d}\boldsymbol{x} \\
&= \int_{\Omega_0} \Big\{ \left\langle \mathsf{D}^2 f(\boldsymbol{x}); \boldsymbol{\mathcal{V}}(\boldsymbol{x}), \boldsymbol{\mathcal{V}}(\boldsymbol{x}) \right\rangle + 2 \nabla f(\boldsymbol{x}) \cdot \boldsymbol{\mathcal{V}}(\boldsymbol{x})\ \mathrm{div}\, \boldsymbol{\mathcal{V}}(\boldsymbol{x}) + \\
&\qquad f(\boldsymbol{x}) \left(\mathrm{div}\, \boldsymbol{\mathcal{V}}(\boldsymbol{x})^2 - \mathrm{Tr}\left(\mathsf{D}\boldsymbol{\mathcal{V}}(\boldsymbol{x})^\top \mathsf{D}\boldsymbol{\mathcal{V}}(\boldsymbol{x}) \right) \right) \Big\}\, \mathrm{d}\boldsymbol{x} \\
&= \int_{\Omega_0} \mathrm{div}\left(f(\boldsymbol{x})(\mathrm{div}\, \boldsymbol{\mathcal{V}}(\boldsymbol{x}))\mathbf{I} - \mathsf{D}\boldsymbol{\mathcal{V}}(\boldsymbol{x}) \right) \boldsymbol{\mathcal{V}}(\boldsymbol{x}) + (\nabla f(\boldsymbol{x}) \cdot \boldsymbol{\mathcal{V}}(\boldsymbol{x})) \boldsymbol{\mathcal{V}}(\boldsymbol{x}))\ \mathrm{d}\boldsymbol{x} \\
&= \int_{\partial\Omega_0} \boldsymbol{\mathcal{V}}(\boldsymbol{x}) \cdot \boldsymbol{n}(\boldsymbol{x}) \left(f(\boldsymbol{x})\, \mathrm{div}\, \boldsymbol{\mathcal{V}}(\boldsymbol{x}) + \nabla f(\boldsymbol{x}) \right) - f(\boldsymbol{x}) \boldsymbol{\mathcal{V}}(\boldsymbol{x})^\top \mathsf{D}\boldsymbol{\mathcal{V}}(\boldsymbol{x}) \boldsymbol{\mathcal{V}}(\boldsymbol{x})\, \mathrm{dS}(\boldsymbol{x})\ .
\end{aligned}
$$

Note that for both shape derivatives $\boldsymbol{\mathcal{V}}$ has to be known only in an open neighbourhood of $\partial\Omega_0$.

We focus on the special case of deformation of Ω_0 by normal displacement of the boundary according to (1.4). This can be modeled by a perturbation vector field

$$
\boldsymbol{\mathcal{V}}(\boldsymbol{x}) = \xi(\mathsf{N}\boldsymbol{x}) \boldsymbol{n}(\mathsf{N}\boldsymbol{x})\ , \quad \mathrm{dist}(\boldsymbol{x}, \partial\Omega_0) < \delta\ , \tag{3.7}
$$

for $\delta > 0$ sufficiently small, $\xi \in C^1(\partial\Omega_0)$, and N the projection onto $\partial\Omega_0$ in normal direction. For this special choice we find for $\boldsymbol{x} \in \partial\Omega_0$

$$
\mathsf{D}\boldsymbol{\mathcal{V}}(\boldsymbol{x}) \boldsymbol{\mathcal{V}}(\boldsymbol{x}) = 0 \quad \text{and} \quad f(\boldsymbol{x})\, \mathrm{div}\, \boldsymbol{\mathcal{V}}(\boldsymbol{x}) + \nabla f(\boldsymbol{x}) \cdot \boldsymbol{\mathcal{V}}(\boldsymbol{x}) = \nabla f(\boldsymbol{x}) \cdot \boldsymbol{n}(\boldsymbol{x}) + \kappa(\boldsymbol{x}) f(\boldsymbol{x})\ ,
$$

with $\kappa(\boldsymbol{x})$ the additive curvature of $\partial\Omega_0$ in $\boldsymbol{x}$. The C^1-smoothness of $\partial\Omega_0$ is essential in all these manipulations. The final formulas for the directional shape derivatives, expressed in terms of the size ξ of the normal deformation, are

$$
\left\langle \frac{dJ_f}{d\Omega}(\Omega_0); \xi \right\rangle = \int_{\partial\Omega_0} \xi(\boldsymbol{x})\, f(\boldsymbol{x})\, \mathrm{dS}(\boldsymbol{x})\ , \tag{3.8}
$$

$$
\left\langle \frac{d^2 J_f}{d^\Omega}; \xi, \xi \right\rangle = \int_{\partial\Omega_0} \xi^2(\boldsymbol{x}) \left(\nabla f(\boldsymbol{x}) \cdot \boldsymbol{n}(\boldsymbol{x}) + \kappa(\boldsymbol{x}) f(\boldsymbol{x}) \right)\, \mathrm{dS}(\boldsymbol{x})\ . \tag{3.9}
$$

They enter the third-order Taylor approximation

$$
\begin{aligned}
J(\Pi(\xi)) \approx J(\Omega_0) &+ \int_{\partial\Omega_0} \xi(\boldsymbol{x})\, f(\boldsymbol{x})\, \mathrm{dS}(\boldsymbol{x}) + \\
&\frac{1}{2} \int_{\partial\Omega_0} \xi^2(\boldsymbol{x}) \left(\nabla f(\boldsymbol{x}) \cdot \boldsymbol{n}(\boldsymbol{x}) + \kappa(\boldsymbol{x}) f(\boldsymbol{x}) \right)\, \mathrm{dS}(\boldsymbol{x})\ .
\end{aligned} \tag{3.10}
$$

This suggests that we replace $\widehat{J} := J \circ \Pi$ with the following quadratic functional in ξ:

$$\widetilde{J}(\xi) := \frac{1}{2} \underbrace{\int_{\partial\Omega_0} \xi^2(\boldsymbol{x}) \left(\nabla f(\boldsymbol{x}) \cdot \boldsymbol{n}(\boldsymbol{x}) + \kappa(\boldsymbol{x}) f(\boldsymbol{x})\right) \mathrm{dS}(\boldsymbol{x})}_{=:\mathsf{a}(\xi,\xi)} + \underbrace{\int_{\partial\Omega_0} \xi(\boldsymbol{x})\, f(\boldsymbol{x})\, \mathrm{dS}(\boldsymbol{x})}_{=:\ell(\xi)} + \underbrace{J(\Omega_0)}_{=:\gamma} , \tag{3.11}$$

which we related to the formula (2.1). For smooth f and $\partial\Omega_0$ we expect the asymptotic accuracy

$$|\widehat{J}(\xi) - \widetilde{J}(\xi)| = O(\|\xi\|_\infty^3) . \tag{3.12}$$

It remains to establish a function space framework compliant with subsection 2.2. Both a and ℓ from (3.11) are bounded on $L^2(\partial\Omega)$. Owing to the Rellich embedding theorem [7, Theorem 3.27], choosing X as the Sobolev space $H^1(\partial\Omega_0)$ [7, Pp. 96], $X := H^1(\partial\Omega_0)$, makes Assumption 2.2 hold. The generalized eigenvalue problem (2.21) is exlicitly given by: seek $\mu \in H^1(\partial\Omega_0) \setminus \{0\}$, $\lambda \in \mathbb{R}$ such that

$$\begin{aligned} &\int_{\partial\Omega_0} \mu(\boldsymbol{x})\nu(\boldsymbol{x}) \left(\nabla f(\boldsymbol{x}) \cdot \boldsymbol{n}(\boldsymbol{x}) + \kappa(\boldsymbol{x}) f(\boldsymbol{x})\right) \mathrm{dS}(\boldsymbol{x}) \\ &\qquad = \lambda\, (\mu, \nu)_{H^1(\partial\Omega_0)} := \lambda \int_{\partial\Omega_0} \nabla_\Gamma \mu(\boldsymbol{x}) \cdot \nabla_\Gamma \nu(\boldsymbol{x}) + \nu(\boldsymbol{x})\nu(\boldsymbol{x})\, \mathrm{dS}(\boldsymbol{x}) \end{aligned} \tag{3.13}$$

for all $\nu \in H^1(\partial\Omega_0)$, where ∇_Γ is the tangential gradient.

4. Numerical Experiments. Lacking a complete rigorous theory we demonstrate the performance of the approximations and algorithms in a few two-dimensional ($d = 2$) test cases. As model reference domain Ω_0 we use a "kite-shaped" domain displayed in Figure 1, whose boundary can be parameterized as follows

$$\boldsymbol{\gamma}(\theta) := \begin{bmatrix} 3.5\cos(\theta) + 1.625\cos(2\theta) \\ 3.5\sin(\theta) \end{bmatrix} , \quad \theta \in [0, 2\pi[.$$

4.1. Discretization. We equip the parameter interval $[0, 2\pi]$ with a mesh $\mathcal{M}$ consisting of $N \in \mathbb{N}$ equi-sized cells and write $\mathcal{S}^0_{1,\mathrm{per}}(\mathcal{M}) \subset H^1(0, 2\pi)$ for the classical finite-element space of $\mathcal{M}$-piecewise linear continuous 2π-periodic functions $[0, 2\pi] \to \mathbb{R}$: $\dim \mathcal{S}^0_{1,\mathrm{per}}(\mathcal{M}) = N$. We define N-dimensional subspaces of $H^1(\partial\Omega_0)$ by pullback under the parameterization $\boldsymbol{\gamma}$

$$X_N := \{\nu : \partial\Omega_0 \to \mathbb{R} : \ \boldsymbol{\gamma}^*\nu \in \mathcal{S}^0_{1,\mathrm{per}}(\mathcal{M})\} . \tag{4.1}$$

As basis $\{b_N^1, \ldots, b_N^N\}$ for X_N we use the locally supported functions whose pullbacks $\boldsymbol{\gamma}^* b_N^j$ are the usual nodal basis functions ("tent functions") of $\mathcal{S}^0_{1,\mathrm{per}}(\mathcal{M})$. Thus we can introduce the Galerkin matrices and vectors

$$\mathbf{A}_N := \left[\mathsf{a}(b_N^i, b_N^j)\right]_{i,j=1}^N \in \mathbb{R}^{N,N} , \tag{4.2}$$

$$\mathbf{M}_N := \left[\left(b_N^i, b_N^j\right)_{H^1(\partial\Omega_0)}\right]_{i,j=1}^N \in \mathbb{R}^{N,N} , \tag{4.3}$$

$$\mathbf{q}_N := \left[\ell(b_N^i)\right]_{i=1}^N \in \mathbb{R}^N , \tag{4.4}$$

for a and ℓ as introduced in (3.11). All occurring integrals were evaluated by means of "overkill" numerical quadrature using the Gauss Legendre quadrature rule of order 16. After these steps we have arrived at the setting of subsection 2.1; the notations should make clear the relationship. In particular, the discrete version of the generalized eigenvalue problem (3.13), the counterpart of (2.3), becomes

$$\mathbf{u} \in \mathbb{R}^N \setminus \{\mathbf{0}\}, \lambda \in \mathbb{R}: \quad \mathbf{A}_N \mathbf{u} = \lambda \mathbf{M}_N \mathbf{u} \,. \tag{4.5}$$

As functions f in the definition of the domain functionals J_f we selected divergences of known vectorfields: $f(\boldsymbol{x}) = \operatorname{div} \mathbf{F}(\boldsymbol{x})$, for some $\mathbf{F} : \mathbb{R}^2 \to \mathbb{R}^2$. Thus integration can be reduced to the boundary

$$J_f(\Omega) = \int_\Omega f(\boldsymbol{x}) \,\mathrm{d}\boldsymbol{x} = \int_{\partial\Omega} \mathbf{F}(\boldsymbol{x}) \cdot \boldsymbol{n} \,\mathrm{dS}(\boldsymbol{x}) \,,$$

and that latter integral can be computed approximately by pullback to the parameter domain $[0, 2\pi[$ and "overkill" numerical quadrature.

Concretely, our numerical tests rely on the following four integrands ($\boldsymbol{x} = \binom{x_1}{x_2}$), rendered in Figure 1:

1. $f_1(\boldsymbol{x}) := 3 \left\|\boldsymbol{x}\right\|^2$, a rotationally symmetric function, globally supported,
2. $f_2(\boldsymbol{x}) := 3x_1^2 + 2x_2$, a globally supported anisotropic function,
3. $f_3(\boldsymbol{x}) := -2(x_1+x_2-5)e^{-((x_1-5)^2+x_2^2)}$, a smooth, effectively locally supported function
4. $f_4(\boldsymbol{x}) := \begin{cases} 1+\cos(\pi r) - r\frac{\pi}{2}\sin(\pi r) & \text{, if } r \le 1 \,, \\ 0 & \text{elsewhere} \end{cases}$, $r := \left((x_1-5)^2 + x_2^2\right)^{\frac{1}{2}}$, non-smooth and locally supported.

4.2. Error of quadratic approximation. In this section we investigate the quadratic approximation errors

$$E_Q(\epsilon) := \sup_{\|\xi\|_{H^1(\partial\Omega_0)} \le \epsilon} |\widehat{J}(\xi) - \widehat{J}(\xi)| \,,$$

with $\widehat{J}$ defined in (1.5) and $\widetilde{J}$ in (3.11). We approximate the supremum by randomly sampling from $\{\xi : \|\xi\|_{H^1(\partial\Omega_0)} \le \epsilon\}$. We construct samples by (rescaled) (i) random combinations of generalized eigenfunctions (solutions of (2.3)) and (ii) random combinations of "tent functions". The combination weights are drawn from uniform distributions on $[-1, 1]$. The estimated errors E_Q are reported in Figure 2 for $f_1, \ldots, f_4$ and mesh resolutions $N = 100, 200, 400$.

For smooth integrands, that is, for f_1, f_2, and f_3, we observe the expected algebraic convergence $E_Q(\epsilon) = O(\epsilon^3)$ as $\epsilon \to 0$. The function f_4 is just continuous and piecewise smooth and this obviously reduces the convergence of the quadratic approximation: asymptotically we observe $E_Q(\epsilon) = O(\epsilon)$.

4.3. Model reduction error. In this section we examine the low rank approximation error and the full model reduction error (which includes both quadratic approximation and low rank approximation)

$$E(\epsilon) := \sup_{\|\xi\|_{H^1(\partial\Omega_0)} \le \epsilon} |\widehat{J}(\xi) - \widetilde{J}(\mathsf{P}_n \xi)| \,,$$

with P_n as defined in section 2 based on (2.15) and (2.16).

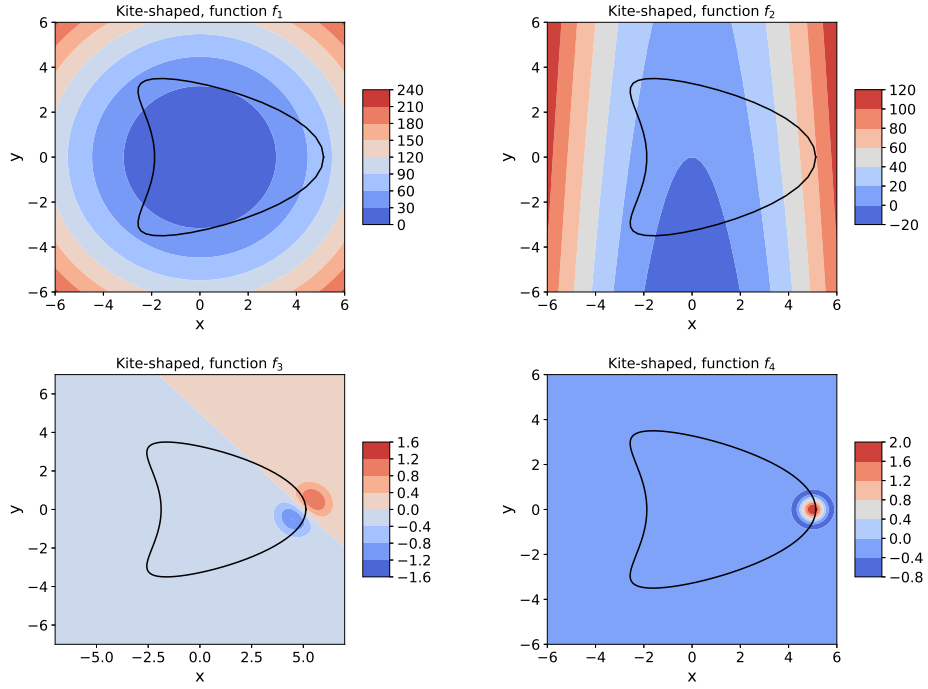


Fig. 1: Contour plots of the sample functions $f_1, \ldots, f_4$. The black line shows the boundary of the "kite-shaped" domain Ω_0.

We compute the errors for perturbations of a fixed size $\epsilon = 0.1$ by means of the same random sampling approach as introduced in subsection 4.2. Concerning low-rank approximation, we monitor the error for reduced models of dimension $n \in \{1, \ldots, 20\}$. The experiment is performed for different mesh resolutions, for $N = 100, 200, 400$ panels. In each case we also report the moduli of the generalized eigenvalues for $\widetilde{\mathbf{A}}$ and $\mathbf{M}$, see (2.16), because their decay determines the behavior of the low-rank approximation error, *cf.* (2.17).

For $\epsilon = 0.1$ we observe a quadratic approximation error $\approx 10^{-3}$, which is a lower bound for the total error. For integrands f_1 and f_2 the bilinear form a boils down to a weighted $L^2(\partial\Omega)$-type pairing. Thus, by Weyl's law we expect a decay of the generalized eigenvalues like $\lambda_\ell \sim \ell^{-2}$, which we see in Figure 3 and Figure 4. This behavior carries over to the model reduction errors. The localized integrands f_3 and f_4 lead to many practically vanishing generalized eigenvalues, because the supports of many basis functions cover only regions where the integrand is very small or even zero. Thus, the low-rank approximation error virtually disappears for moderate values of n and the full error reflects the quadratic approximation error. However, the large eigenvalues, of which there will be more and more on fine meshes, still reveal a decay according to Weyl's law.

In Figure 7–10 we visualize the generalized eigenfunctions for the largest (in modulus) generalized eigenvalues for $N = 200$ panels. These functions form a basis of the reduced parameter space X_n. For integrands f_3 and f_4 these functions are well localized in the vicinity of the (practical) support of the integrands.

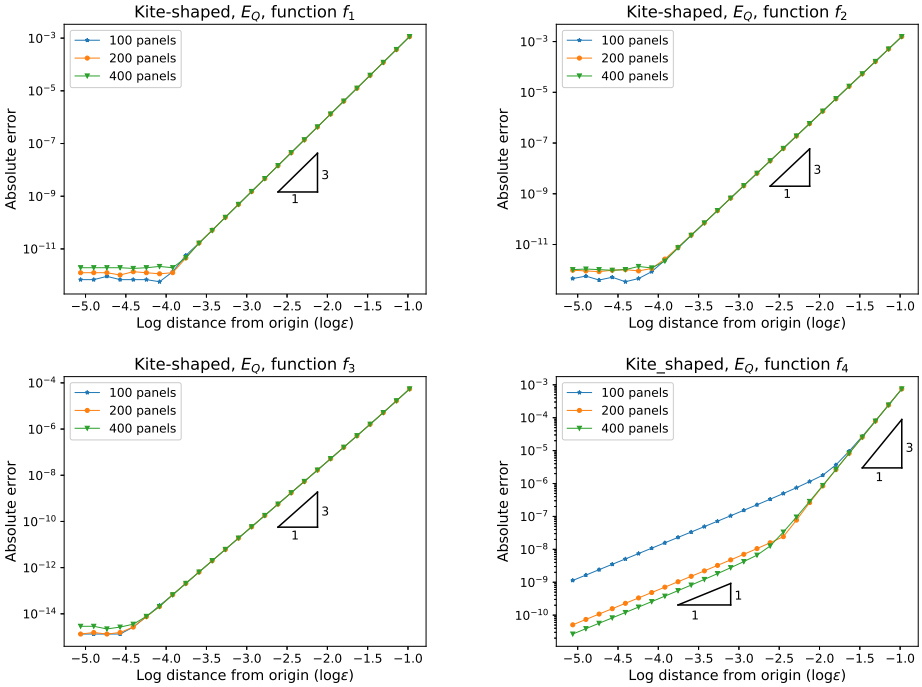
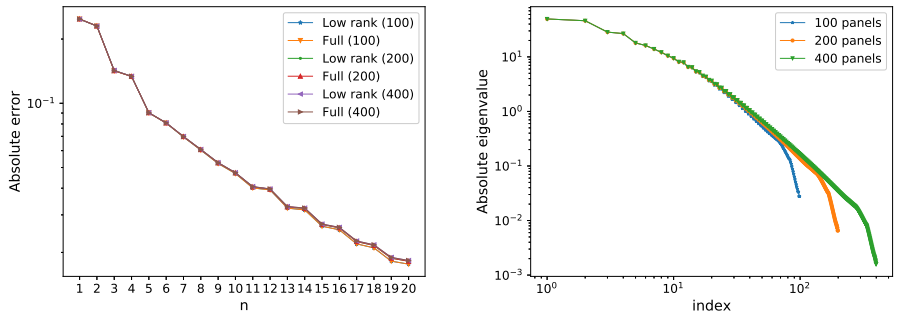


Fig. 2: Estimated errors $E_Q = E_{Q(\epsilon)}$ for different functions and mesh resolutions



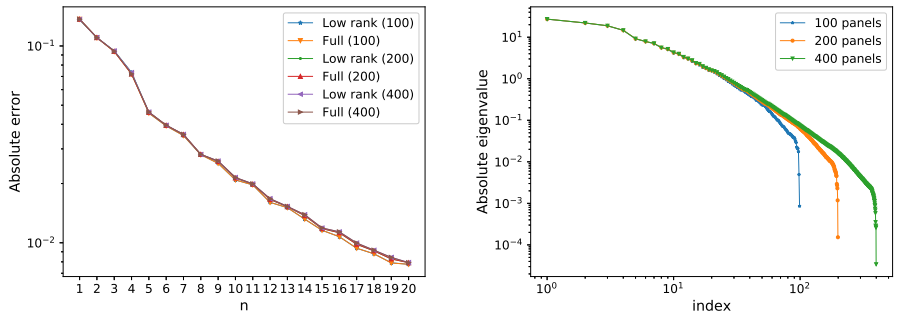
(a) Errors: full and low-rank

(b) Generalized eigenvalues

Fig. 3: Model reduction for kite-shaped domain, J_f with integrand f_1

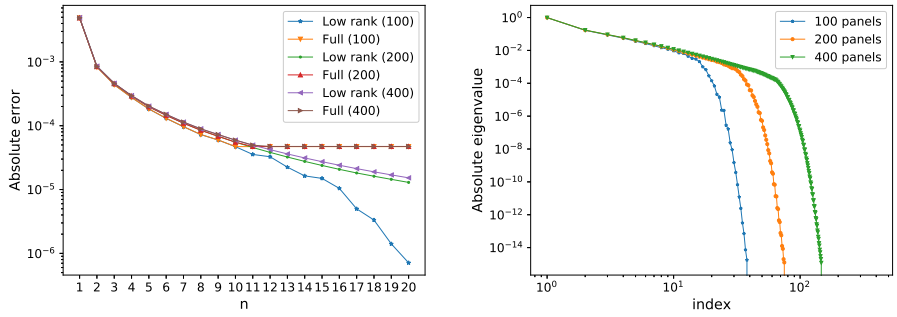
4.4. Performance of the adaptive algorithm. We retain the setting of the previous sections: For the kite-shaped domain and the shape functionals J_f, $f = f_1, \ldots, f_4$, we report the performance of the adaptive Algorithm 1 introduced in subsection 2.3. For all the results that are presented in this section, ϵ is fixed at 0.1. Throughout, the control parameters for the termination of the inner iteration are fixed at $\tau_{\text{rel}} = 0.01$ and $\tau_{\text{abs}} = 0.0001$.

In Figure 11, we report all the eigenvalues obtained from Algorithm 1 in the first



(a) Errors: full and low-rank (b) Generalized eigenvalues

Fig. 4: Model reduction for kite-shaped domain, J_f with integrand f_2



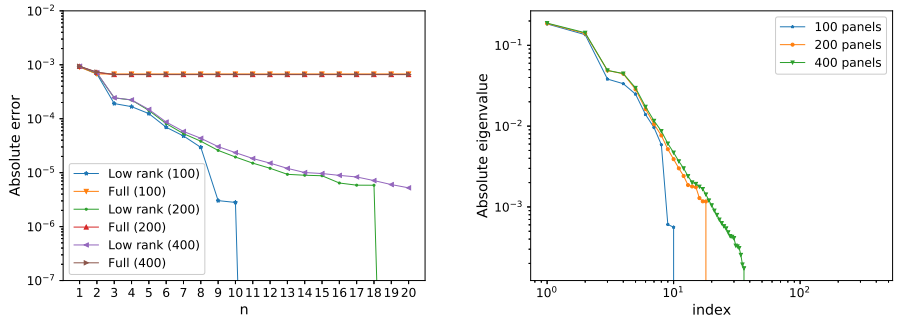
(a) Errors: full and low-rank (b) Generalized eigenvalues

Fig. 5: Model reduction for kite-shaped domain, J_f with integrand f_3

30 steps. We use a mesh with 400 panels and $\delta = 10^{-8}$. For each step of the outer iteration the moduli of the obtained Ritz values are plotted vertically. The modulus $|\lambda_L|$ of the smallest Ritz value of interest is marked in red. Grey lines indicate the moduli of the exact generalized eigenvalues. Obviously, they are well approximated by the Ritz values.

Next, we study the *total* number of inner iterations and the number of outer iterations in the adaptive algorithm. The number of outer iterations is one less than the dimension of obtained space in case of normal termination and two less in case of low rank termination. Computations are done at 3 tolerance levels: $\delta = 10^{-3}, 10^{-4}$ and 10^{-5}.

The tables send the expected message that tightening the tolerance δ entails computing more eigenfunctions, in particular for integrands f_1 and f_2, where the generalized eigenvalues decay according to Weyl's law. Setting $\delta = 10^{-5}$ for f_1/f_2 essentially enforces full approximation, whereas for the localized integrands f_3/f_4 we still gain from low-rank approximation. The rather large numbers of inner iterations suggest that there is probably much room for relaxing or improving the termination



(a) Errors: full and low-rank

(b) Generalized eigenvalues

Fig. 6: Model reduction for kite-shaped domain, J_f with integrand f_4

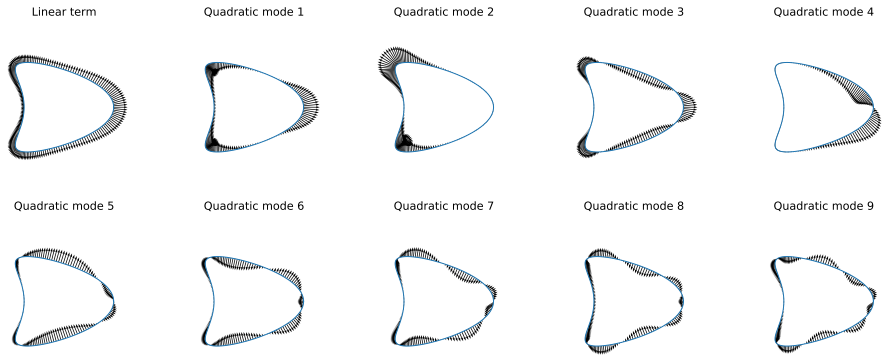


Fig. 7: Integrand f_1, kite-shaped Ω_0: generalized eigenfunctions

criteria used in Algorithm 1.

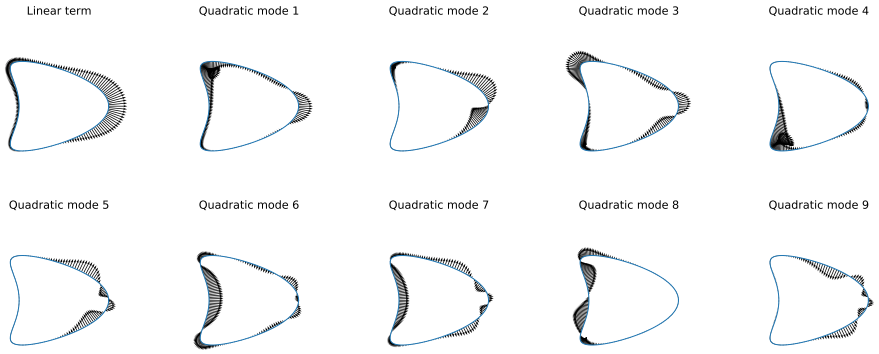


Fig. 8: Integrand f_2, kite-shaped Ω_0: generalized eigenfunctions

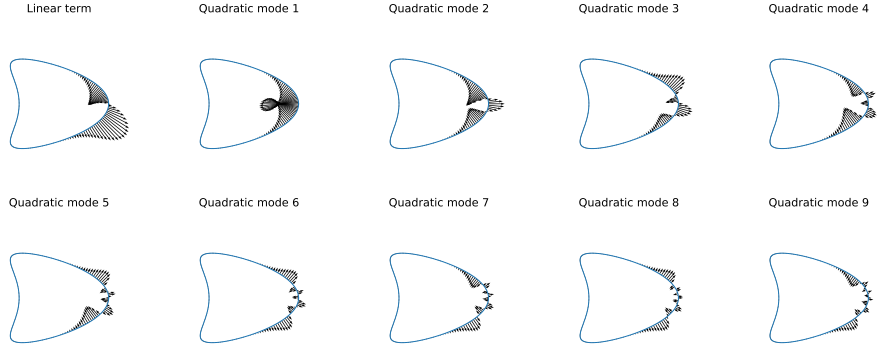


Fig. 9: Integrand f_3, kite-shaped Ω_0: generalized eigenfunctions

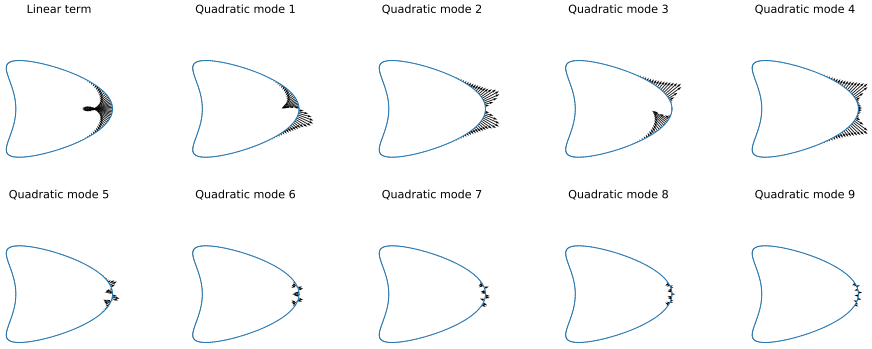


Fig. 10: Integrand f_4, kite-shaped Ω_0: generalized eigenfunctions

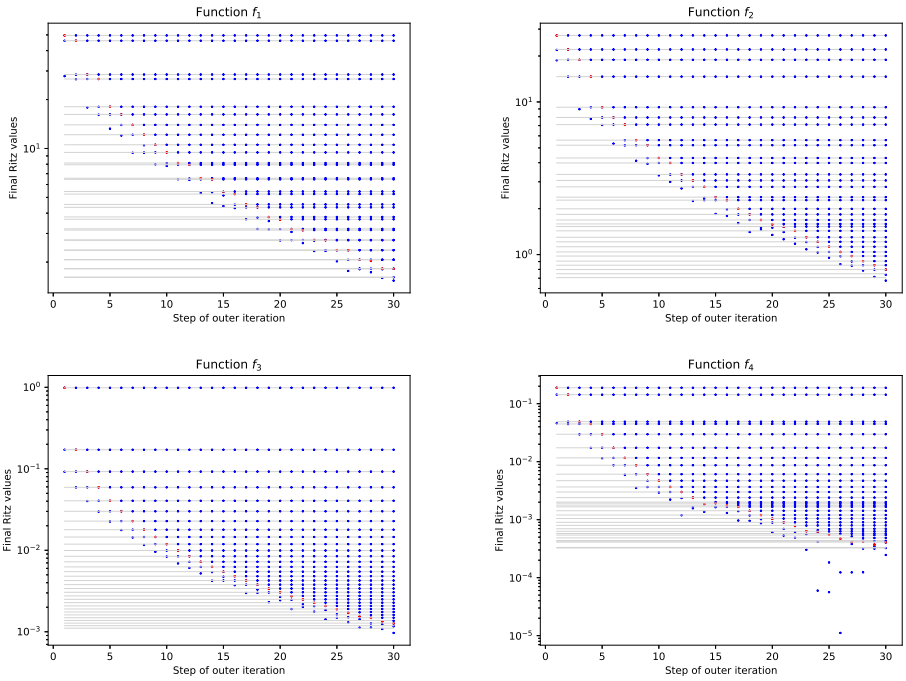


Fig. 11: Generalized eigenvalues as computed during a run of the adaptive algorithm from subsection 2.3, $\epsilon = 0.1$, $N = 400$, $\delta = 10^{-8}$, $\tau_{\text{rel}} = 10^{-2}$, $\tau_{\text{abs}} = 10^{-4}$

	f_1	f_2	f_3	f_4
100	767	864	15	22
200	1072	945	20	18
400	1108	1121	23	24

(a) Total no. of inner iterations

	f_1	f_2	f_3	f_4
100	86	68	3	3
200	115	82	3	3
400	125	87	3	3

(b) No. of outer iterations

Table 1: Total no. of inner and outer iterations for kite domain ($\delta = 10^{-3}$)

	f_1	f_2	f_3	f_4
100	827	1251	80	40
200	1682	2304	88	61
400	2756	3130	91	62

(a) Total no. of inner iterations

	f_1	f_2	f_3	f_4
100	98	98	10	7
200	190	174	11	8
400	317	234	11	8

(b) No. of outer iterations

Table 2: Total no. of inner and outer iterations for kite domain ($\delta = 10^{-4}$)

	f_1	f_2	f_3	f_4
100	827	1251	108	46
200	1712	2573	174	103
400	3195	4385	204	120

(a) Total no. of inner iterations

	f_1	f_2	f_3	f_4
100	98	98	17	9
200	198	198	28	17
400	398	389	34	21

(b) No. of outer iterations

Table 3: Total no. of inner and outer iterations for kite domain ($\delta = 10^{-5}$)

5. Conclusions. We have explored a combined quadratic approximation and adaptive low-rank approximation strategy for model model reduction in shape representation with respect to a given shape functional. We focused on a particular model problem for which we observed a rather disappointing efficacy of low-rank approximation that we could attribute it to a merely algebraic decay of singular values of the shape Hessian in a chosen Sobolev space framework.

The question is whether relevant shape functionals share this awkward trait of our model problem. Results by Eppler and Harbrecht [3, 4, 5] indicate that shape Hessians for so-called Bernoulli problems and shape optimization problems based on tracking-type functionals are continuous and sometimes even coercive in Sobolev spaces $H^s(\Gamma)$ for suitable $0 < s \leq 1$. Consequently a slow algebraic decay of singular values can again be expected when using some Sobolev space $H^m(\Gamma)$ as function space framework.

Couldn't we expect an exponential decay of the eigenvalues in an H^m-setting for *far-field-type* shape functionals? Uforntunately not: use the line segment $\Gamma_0 = [0,1] \times \{0\} \subset \mathbb{R}^2$ as the reference shape, and let Γ stand for another open C^1 Jordan curve with endpoints $\binom{0}{0}$ and $\binom{1}{0}$. The reader may viewed it as the variable part of the boundary of some domain Ω. Let us consider the shape functional

$$J(\Gamma) := \int_\Gamma G(\boldsymbol{x}, \boldsymbol{y}) \, \mathrm{d}S(\boldsymbol{y}) \,, \quad \boldsymbol{x} \notin \Gamma_0 \text{ fixed} \,, \quad G(\boldsymbol{x}, \boldsymbol{y}) = \exp(- \|\boldsymbol{x} - \boldsymbol{y}\|^2) \,.$$

We employ a normal displacement parameterization of the shape space of "perturbed line segments"

$$\Pi(\xi) := \left\{ \begin{bmatrix} t \\ \xi(t) \end{bmatrix}, \, 0 \leq t \leq 1 \right\} \,, \quad \xi \in X := C_0^1([0,1]) \,.$$

This satisfies $\Pi(0) = \Gamma_0$ and yields the mapped functional ($\dot{\xi} = \frac{d\xi}{dt}$)

$$\widehat{J}(\xi) := J(\Pi(\xi)) = \int_0^1 G\left(\boldsymbol{x}, \left[\begin{smallmatrix} t \\ \xi(t) \end{smallmatrix}\right]\right) \sqrt{1 + |\dot{\xi}(t)|^2} \, \mathrm{d}t \,, \quad \xi \in X \,.$$

We can plug in the Taylor expansions

$$G\left(\boldsymbol{x}, \left[\begin{smallmatrix} t \\ \xi(t) \end{smallmatrix}\right]\right) = G\left(\boldsymbol{x}, \left[\begin{smallmatrix} t \\ 0 \end{smallmatrix}\right]\right) + \frac{\partial G}{\partial y_2}\left(\boldsymbol{x}, \left[\begin{smallmatrix} t \\ 0 \end{smallmatrix}\right]\right) \xi(t) + \frac{1}{2} \frac{\partial^2 G}{\partial y_2^2}\left(\boldsymbol{x}, \left[\begin{smallmatrix} t \\ 0 \end{smallmatrix}\right]\right) \xi(t)^2 + O(\|\xi\|_X^3) \,,$$

$$\sqrt{1 + |\dot{\xi}(t)|^2} = 1 + \tfrac{1}{2}\dot{\xi}(t)^2 + O(\|\xi\|_X^4) \,,$$

for $\xi \to 0$. The terms that are quadratic in ξ give us the second derivative

$$\left\langle \frac{d^2 \widehat{J}}{d\xi^2}(0); \xi, \eta \right\rangle = \int_0^1 \tfrac{1}{2} G\left(\boldsymbol{x}, \left[\begin{smallmatrix} t \\ 0 \end{smallmatrix}\right]\right) \dot{\xi}(t)\dot{\eta}(t) + \tfrac{1}{2} \frac{\partial^2 G}{\partial y_2^2}\left(\boldsymbol{x}, \left[\begin{smallmatrix} t \\ 0 \end{smallmatrix}\right]\right) \xi(t)\eta(t) \, \mathrm{d}t \,.$$

Up to a compact perturbation the induced bilinear form is equivalent to the inner product of $H^1(]0,1[)$, and the generalized eigenvalues with respect to an H^m-inner product, $m \geq 1$, will decay only algebraically.

Acknowledgment. The first and the second author would like to thank the Isaac Newton Institute (INI), Cambridge, UK, for inviting them in the framework of the 2019 programme on *Geometry, compatibility and structure preservation in computational differential equations*. Seminal discussions for this work took place at INI.

[1] M. C. Delfour and J.-P. Zolésio, *Shapes and geometries*, vol. 22 of Advances in Design and Control, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second ed., 2011. Metrics, analysis, differential calculus, and optimization.
[2] K. Eppler, *Fréchet-differentiability and sufficient optimality conditions for shape functionals*, in Optimal control of partial differential equations (Chemnitz, 1998), vol. 133 of Internat. Ser. Numer. Math., Birkhäuser, Basel, 1999, pp. 133–143.
[3] K. Eppler and H. Harbrecht, *Coupling of FEM and BEM in shape optimization*, Numer. Math., 104 (2006), pp. 47–68.
[4] K. Eppler and H. Harbrecht, *Tracking Neumann data for stationary free boundary problems*, SIAM J. Control Optim., 48 (2009/10), pp. 2901–2916, https://doi.org/10.1137/080733760, https://doi.org/10.1137/080733760.
[5] K. Eppler and H. Harbrecht, *Shape optimization for free boundary problems—analysis and numerics*, in Constrained optimization and optimal control for partial differential equations, vol. 160 of Internat. Ser. Numer. Math., Birkhäuser/Springer Basel AG, Basel, 2012, pp. 277–288, https://doi.org/10.1007/978-3-0348-0133-1_15, http://dx.doi.org/10.1007/978-3-0348-0133-1_15.
[6] G. H. Golub and C. F. Van Loan, *Matrix computations*, Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, Baltimore, MD, fourth ed., 2013.
[7] W. McLean, *Strongly Elliptic Systems and Boundary Integral Equations*, Cambridge University Press, Cambridge, UK, 2000.
[8] A. Schiela and J. Ortiz, *Second order directional shape derivatives of integrals on submanifolds*, Preprint urn:nbn:de:bvb:703-epub-3251-1, EPub Bayreuth, 2017.
[9] J. Sokolowski and J.-P. Zolesio, *Introduction to shape optimization*, vol. 16 of Springer Series in Computational Mathematics, Springer, Berlin, 1992.
[10] D. Werner, *Funktionalanalysis*, Springer, Berlin, 1995.

Appendix A. Code. The C++ codes with which all numerical experiments reported in this article have been conducted can be accessed through the Git repository https://gitlab.ethz.ch/ppanchal/adaptiveshapeapprox.git. The accompanying `README.md` file explains installation and use of these codes.