Proceedings of the ASME 2018 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2018
August 26–29, 2018, Quebec City, Canada

DETC2018-85051 DRAFT

AUTOMATIC POSTPROCESSING OF TOPOLOGY OPTIMIZATION SOLUTIONS BY USING SUPPORT VECTOR MACHINES

Niclas Strömberg
Department of Mechanical Engineering
Örebro University
SE-701 82 Örebro, Sweden
Email: niclas.stromberg@oru.se

ABSTRACT

The postprocessing step from the density result in topology optimization to a parametric CAD model is typically most time consuming and usually involves several hands on maneuvers by an engineer. In this paper we propose an approach in order to automate this step by using soft non-linear support vector machines (SVM). Our idea is to generate the boundaries separating regions of material (elements with densities equal to one) and no material (elements with densities equal zero) obtained from topology optimization automatically by using SVM. The hypersurface of the SVM can then in the long run be explicitly implemented in any CAD software. In this work we generate these hypersurfaces by solving the dual formulation of the SVM with soft penalization and nonlinear kernel functions using quadratic programming or the sequential minimal optimization approach. The proposed SVM-based postprocessing approach is studied on topology optimization results of orthotropic elastic design domains with mortar contact conditions studied most recently in a previous work. The potential energy of several bodies with nonmatching meshes is maximized. In such manner no extra adjoint equation is needed. Intermediate density values are penalized using SIMP or RAMP, and the regularization is obtained by applying sensitivity or density filters following the approaches of Sigmund and Bourdin. The study demonstrates that the SVM-based postprocessing approach automatically generates proper hypersurfaces which can be used efficiently in the CAD modelling.

INTRODUCTION

Today, topology optimization is a standard tool in product development [1]. In particular, the problem of minimization of compliance for a prescribed volume fraction is most established. But still the transfer of the optimal solution of element densities to a parametric CAD geometry is time consuming. In this work, we propose a postprocessing approach of the topology optimization solution by adopting soft non-linear support vector machines. The soft non-linear support vector machine introduced by Cortes and Vapnik [2] defines a paradigm shift in machine learning and the paper has been cited more than 15000 times. By adopting the kernel trick and the soft penalization, we are able to classify non-linear separable data including misclassified data points. In this work, we suggest to use the soft non-linear SVM to classify the optimal element densities into a material and non-material geometry description with smooth separating boundaries which can be used to set up the CAD geometry as shown in Figure 1. In the long run the optimal support vectors could be integrated into the CAD software directly in order to support an automatic postprocessing process of topology optimization solutions. A similar approach for level-set-based topology optimization was presented recently by Chu et al. [3]. Examples of other approaches for interpretation of topology optimization solutions can be found in e.g. [4–9]. For readers not familiar with SVM, an excellent introduction to this machine learning discipline is found in the textbook by Hamel [10]. For readers not familiar with topology optimization we suggest the textbook by Bendsø and Sigmund [11].

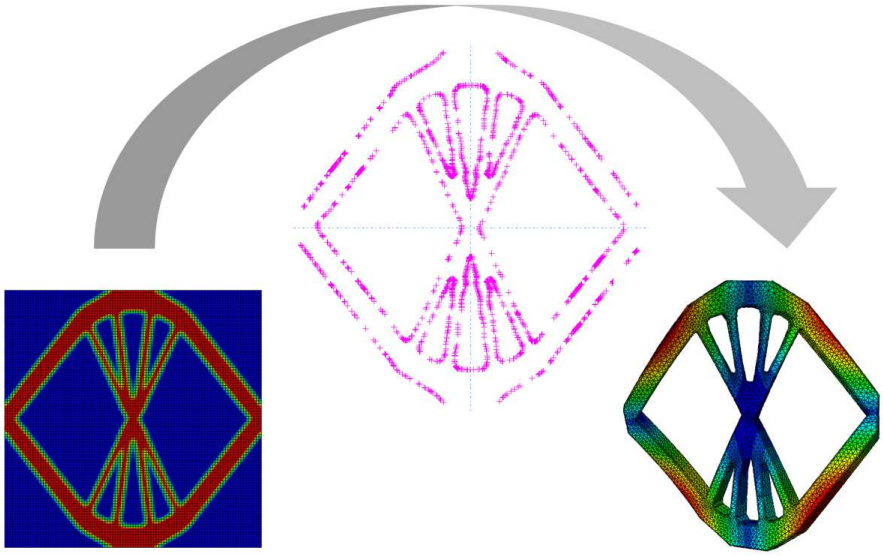

Figure 1. *Automatic postprocessing of topology optimization solution using support vector machines.*

In a standard formulation of topology optimization, such as the SIMP or RAMP model, each element e is equipped with a center point $\mathbf{x}^e$ and a density value ρ_e, where $\rho_e = 0$ (or a small number ε in order to avoid singularities) means no material in the element and $\rho_e = 1$ corresponds to a filled element. Thus, for an optimal solution we can identify one set of points $\mathbf{x}^e$ with density values $\rho_e = 0$ and another set with density values $\rho_e = 1$. By adopting the idea of support vector machines we can classify the data $(\mathbf{x}^e, \rho_e)$ for all elements e of the design domain into these two sets by finding a separating hypersurface that maximizes the distance from this boundary to the closest point of each set, see Figure 2. These points are called the support vectors and will uniquely define the separating boundary between no material and filled regions. By utilizing the kernel trick we can do this for non-linear separable sets and by adding a penalty term to the objective function we also handle misclassified points efficiently. The kernel trick is performed for the dual problem which we then solve by quadratic programming (*quadprog.m* in Matlab) or sequential minimal optimization [12].

The implementation of the proposed SVM-based postprocessing approach is tested on standard compliance problems as well as on optimal solutions obtained for design domains in unilateral contact with non-matching meshes. Most recently this was treated in Strömberg [13] by maximizing the potential energy formulated by using mortar contact conditions for design domains with orthotropic elasticity. It is well-known that the optimal solution strongly depends on the boundary conditions applied on the design domains. In fact, the optimal layout is extra sensitive to unilateral contact conditions. This was demonstrated in Strömberg and Klarbring [14] and Strömberg [15] by performing topology optimization of structures with unilateral contact conditions. In those paper, optimal layouts were obtained for design domains in unilateral contact with matching meshes. Here, we treat design domains with non-matching meshes by adopting the celebrated mortar approach [16, 17]. Furthermore, instead of using the compliance, we choose the potential energy as objective function for the nested problem. In such manner, no extra adjoint adjoint equation is needed in the sensitivity analysis. A similar approach was used for topology optimization of hyperelastic bodies in Klarbring and Strömberg [18], se also the note in [19].

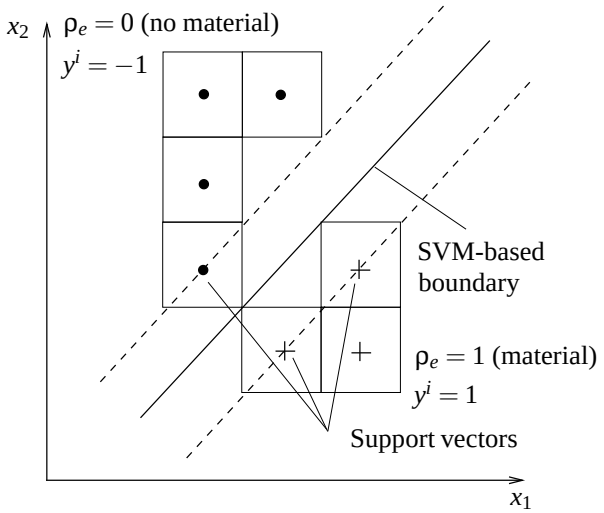

Figure 2. *The basic idea of the proposed SVM-based postprocessing approach.*

The outline of the paper is as follows: in the next section we set up topology optimization for design domains in unilateral contact by maximizing the potential energy, in section 2 we treat non-matching meshes with the mortar approach, in section 3 the

formulation of soft non-linear support vector machines is presented, section 4 presents sequential minimal optimization and, finally, some numerical results are presented together with concluding remarks.

1 TOPOLOGY OPTIMIZATION

Let us consider a system of bodies which are parameterized with the SIMP or RAMP model. The design parameters ρ_e are collected in $\boldsymbol{\rho}$. The stiffness matrix of the system is obtained by the following assembly procedure (SIMP or RAMP):

$$\mathbf{K} = \mathbf{K}(\boldsymbol{\rho}) = \bigcap_{e=1}^{n_{\text{el}}} \rho_e^n \mathbf{k}_e \quad \text{or} \quad \bigcap_{e=1}^{n_{\text{el}}} \frac{\rho_e}{1+n(1-\rho_e)} \mathbf{k}_e, \tag{1}$$

where $\mathbf{k}_e$ is an element stiffness matrix given by orthotropic elasticity (for details see [13]), $n = 3$ or 4, $\bigcap$ represents an assembly operator and n_{el} is the number of elements. The system of bodies is subjected to external forces $\mathbf{F}$ and unilateral contact conditions formulated by the mortar approach presented in the next section.

The state of equilibrium of the system is obtained by minimizing the potential energy subjected to the unilateral constraints between the bodies. The potential energy of the system reads

$$\Pi(\boldsymbol{\rho}, \mathbf{d}) = \frac{1}{2}\mathbf{d}^T \mathbf{K}(\boldsymbol{\rho})\mathbf{d} - \mathbf{F}^T \mathbf{d}, \tag{2}$$

where $\mathbf{d}$ contains nodal displacements. Thus, for a given density distribution $\boldsymbol{\rho} = \hat{\boldsymbol{\rho}}$, the equilibrium state is found by solving

$$\begin{cases} \min\limits_{\mathbf{d}} \Pi(\hat{\boldsymbol{\rho}}, \mathbf{d}) \\ \text{s.t. } \mathbf{C}_S\mathbf{d} + \mathbf{C}_M\mathbf{d} - \mathbf{g} \leq \mathbf{0}, \end{cases} \tag{3}$$

where $\mathbf{C}_S$ and $\mathbf{C}_M$ are defined by the mortar approach in the next section. The corresponding KKT-conditions are given by

$$\mathbf{K}\mathbf{d} + \mathbf{C}_S^T\mathbf{P}_n + \mathbf{C}_M^T\mathbf{P}_n = \mathbf{F} \tag{4}$$

and

$$\mathbf{P}_n \geq 0,\ \mathbf{C}_S\mathbf{d} + \mathbf{C}_M\mathbf{d} - \mathbf{g} \leq \mathbf{0},\ \mathbf{P}_n \circ (\mathbf{C}_S\mathbf{d} + \mathbf{C}_M\mathbf{d} - \mathbf{g}) = \mathbf{0}, \tag{5}$$

where $\mathbf{P}_n$ contains Lagrange multipliers P_n^A, which are interpreted as contact forces, $\mathbf{g}$ is a vector of initial gaps g^A and $\circ$ represents the Hadamard product. By solving these KKT-conditions using the augmented Lagrangian approach and a non-smooth Newton method, we obtain $\mathbf{d} = \mathbf{d}(\boldsymbol{\rho})$ and $\mathbf{P}_n = \mathbf{P}_n(\boldsymbol{\rho})$. An early implementation of the augmented Lagrangian approach using a non-smooth Newton method is found in [20].

For the nested state problem presented above, we maximize the potential energy, i.e.

$$\begin{cases} \max\limits_{\boldsymbol{\rho}} \Pi(\boldsymbol{\rho}, \mathbf{d}(\boldsymbol{\rho})) \\ \text{s.t.} \begin{cases} \sum\limits_{e=1}^{n_{\text{el}}} V_e\rho_e = \hat{V}, \\ \boldsymbol{\varepsilon} \leq \boldsymbol{\rho} \leq \mathbf{1}, \end{cases} \end{cases} \tag{6}$$

where V_e represents the volume of element e for $\rho_e = 1$, $\hat{V}$ is the total amount of material to be distributed, $\boldsymbol{\varepsilon}$=$\{\varepsilon, \ldots, \varepsilon\}^T$ is a vector of small numbers ε and $\mathbf{1}$=$\{1, \ldots, 1\}^T$.

The objective function in (6) can be interpreted by inserting the KKT-conditions from (4) and (5) into $\Pi(\boldsymbol{\rho}, \mathbf{d}(\boldsymbol{\rho}))$. This yields

$$\Pi(\boldsymbol{\rho}, \mathbf{d}(\boldsymbol{\rho})) = -\frac{1}{2}\mathbf{F}^T\mathbf{d} - \frac{1}{2}\mathbf{P}_n^T\mathbf{g}. \tag{7}$$

Thus, maximizing the potential energy is equivalent to minimizing

$$\mathbf{F}^T\mathbf{d} + \mathbf{P}_n^T\mathbf{g}.$$

The first term is the definition of the well-known compliance, the second term implies that P_n^A is minimized for $g^A > 0$ and maximized when $g^A < 0$. Of course, for $\mathbf{g}$=$\mathbf{0}$, the established compliance optimization problem is recovered.

The sensitivity analysis is performed by using the corresponding Lagrangian

$$\mathcal{L}(\boldsymbol{\rho}, \mathbf{d}, \mathbf{P}_n) = \Pi(\boldsymbol{\rho}, \mathbf{d}) + \mathbf{P}_n^T(\mathbf{C}_S\mathbf{d} + \mathbf{C}_M\mathbf{d} - \mathbf{g}). \tag{8}$$

At the state of equilibrium defined by the KKT-conditions in (5) it is clear that the Lagrangian is equivalent to the potential energy, i.e.

$$\mathcal{L} = \mathcal{L}(\boldsymbol{\rho}, \mathbf{d}(\boldsymbol{\rho}), \mathbf{P}_n(\boldsymbol{\rho})) = \Pi(\boldsymbol{\rho}, \mathbf{d}(\boldsymbol{\rho})). \tag{9}$$

This is utilized in following way:

$$\frac{\partial \Pi}{\partial \rho_e} = \frac{\partial \mathcal{L}}{\partial \rho_e} + \left(\frac{\partial \mathcal{L}}{\partial \mathbf{d}}\right)^T \frac{\partial \mathbf{d}}{\partial \rho_e} + \left(\frac{\partial \mathcal{L}}{\partial \mathbf{P}_n}\right)^T \frac{\partial \mathbf{P}_n}{\partial \rho_e}. \tag{10}$$

The first term in (10) equals

$$\frac{\partial \mathcal{L}}{\partial \rho_e} = \frac{1}{2}\mathbf{d}^T \frac{\mathbf{K}}{\partial \rho_e}\mathbf{d}, \tag{11}$$

where

$$\frac{\partial \mathbf{K}}{\partial \rho_e} = n\rho_e^{n-1}\mathbf{k}_e \quad \text{or} \quad \frac{1+n}{(1+n(1-\rho_e))^2}\mathbf{k}_e. \tag{12}$$

The remaining terms are all zeros by the KKT-conditons in (4) and (5). This is verified below.

$$\frac{\partial \mathcal{L}}{\partial \mathbf{d}} = \mathbf{K}\mathbf{d} - \mathbf{F} + \mathbf{C}_S^T\mathbf{P}_n + \mathbf{C}_M^T\mathbf{P}_n = \mathbf{0}, \tag{13a}$$

$$\left(\frac{\partial \mathcal{L}}{\partial \mathbf{P}_n}\right)^T \frac{\partial \mathbf{P}_n}{\partial \rho_e} = (\mathbf{C}_S\mathbf{d} + \mathbf{C}_M\mathbf{d} - \mathbf{g})^T \frac{\partial \mathbf{P}_n}{\partial \rho_e} = 0. \tag{13b}$$

Perhaps, the latter result is not obvious for $\mathbf{P}_n = \mathbf{0}$. However, this is true by taking the derivative of the following formulation of the complementary condition in (5):

$$\frac{\partial}{\partial \rho_e}\left(\mathbf{P}_n^T(\mathbf{C}_S\mathbf{d} + \mathbf{C}_M\mathbf{d} - \mathbf{g}) = 0\right), \tag{14}$$

which yields

$$\left(\frac{\partial \mathbf{P}_n}{\partial \rho_e}\right)^T (\mathbf{C}_S\mathbf{d} + \mathbf{C}_M\mathbf{d} - \mathbf{g}) + \mathbf{P}_n^T(\mathbf{C}_S + \mathbf{C}_M)\frac{\partial \mathbf{d}}{\partial \rho_e} = 0 \tag{15}$$

which in turn proofs (13b) for $\mathbf{P}_n = \mathbf{0}$.

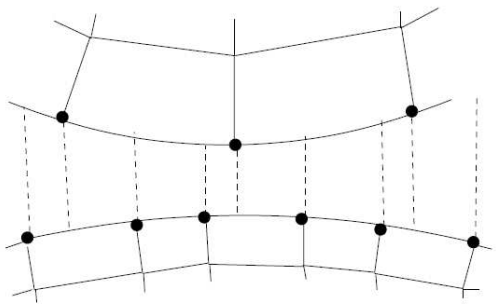

Figure 3. *Non-matching meshes of a contact interface.*

2 THE MORTAR APPROACH

Contact between deformable bodies with non-matching meshes as shown in Figure 3 can efficiently be treated by applying the mortar approach. The mortar approach is briefly presented here in a setting of small displacements. In the case of small displacements, the potential contact zone is identified by two contact surfaces Γ_c^i that are almost coinciding, i.e. $\Gamma_c^1 \approx \Gamma_c^2$. Γ_c^1 belongs to the first body Ω^1 (slave body) and Γ_c^2 is a part of the second one Ω^2 (master body). The virtual power of the total contact pressure $\mathbf{p}$ on this potential contact zone is defined by

$$\mathcal{P}_{\text{int}}^p = \int_{\Gamma_c^1} \mathbf{p}\cdot\mathbf{w}^1\,\mathrm{d}A - \int_{\Gamma_c^2} \mathbf{p}\cdot\mathbf{w}^2\,\mathrm{d}A, \tag{16}$$

where $\mathbf{w}^i$ denotes the virtual velocity field of respectively body. Since, $\Gamma_c = \Gamma_c^1 \approx \Gamma_c^2$, (16) can also be written more compactly as

$$\mathcal{P}_{\text{int}}^p = \int_{\Gamma_c} p_i(\mathbf{w}_i^1 - \mathbf{w}_i^2)\,\mathrm{d}A. \tag{17}$$

By introducing the normal contact pressure p_n and assuming that the tangential forces are zero, we can rewrite (17) to

$$\mathcal{P}_{\text{int}}^p = \int_{\Gamma_c} p_n(\mathbf{w}_i^1 - \mathbf{w}_i^2)n_i\,\mathrm{d}A, \tag{18}$$

where n_i represents the outward unit normal of the slave surface Γ_c^1.

The finite element discretization of (18) is done by introducing the following approximations:

$$p_n = \sum_{A=1}^{n} N^A\lambda^A, \tag{19a}$$

$$w_i^1 = \sum_{A=1}^{n} N^A c_i^A, \tag{19b}$$

$$w_i^2 = \sum_{A=1}^{m} M^A c_i^A. \tag{19c}$$

Here, $N^A = N^A(\mathbf{x})$ represents the shape functions on Γ_c^1 which are taken to be the corresponding trace functions of the global shape functions on Ω^1. The total number of shape functions N^A on Γ_c^1 is n. In a similar way, $M^A = M^A(\mathbf{x})$ represents the shape functions on Γ_c^2, which are taken to be m in number. By inserting (19) into (18), one gets

$$\mathcal{P}_{\text{int}}^p = \sum_{A=1}^{n}\sum_{B=1}^{n}\int_{\Gamma_c} N^A N^B n_i\,\mathrm{d}A\lambda^A c_i^B - \sum_{A=1}^{n}\sum_{B=1}^{m}\int_{\Gamma_c} N^A M^B n_i\,\mathrm{d}A\lambda^A c_i^B \tag{20}$$

or written as

$$\mathcal{P}_{\text{int}}^p = \sum_{A=1}^{n}\sum_{B=1}^{n} C_{Si}^{AB}\lambda^A c_i^B + \sum_{A=1}^{n}\sum_{B=1}^{m} C_{Mi}^{AB}\lambda^A c_i^B, \tag{21}$$

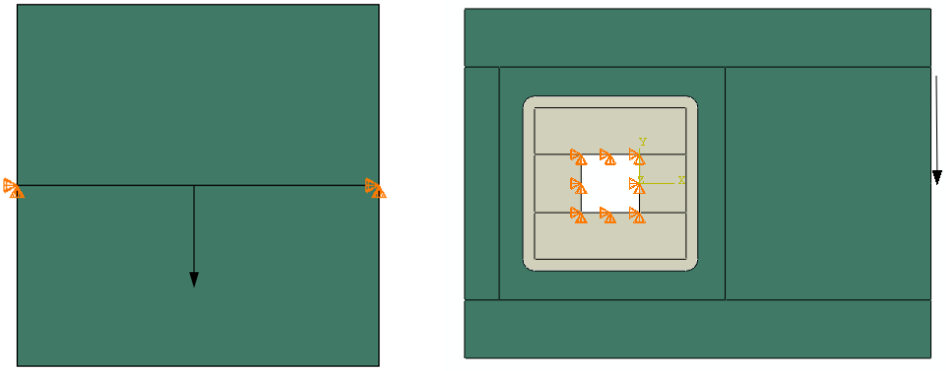

Figure 4. *Design domains with boundary conditions.*

where

$$C_{Si}^{AB} = \int_{\Gamma_c} N^A N^B n_i \, \mathrm{d}A, \tag{22a}$$

$$C_{Mi}^{AB} = -\int_{\Gamma_c} N^A M^B n_i \, \mathrm{d}A. \tag{22b}$$

The latter integral is known as the mortar integral. It is tricky to solve this integral because it cannot in general be dived into sub-domains defined by the finite elements depending on the non-matching meshes. One way of fixing this problem is to us a quadrature rule with many integration points such as the Lobatto rule with 10 points presented in Table 1. This is utilized in the present paper.

Table 1. *Lobatto rule with n_{int}=10 integration points.*

n_{int}	ξ_i	W_i
10	± 0.1652789577	0.3275397612
	± 0.4779249498	0.2920426836
	± 0.7387738651	0.2248894320
	± 0.9195339082	0.1333059908
	± 1	0.0222222222

3 SUPPORT VECTOR MACHINE

In this section, we present the dual formulation of the soft non-linear support vector machine. First we introduce the original linear SVM, which actually was suggested already in the 60s by Vapnik, then we apply the kernel trick and, finally, regularize the problem.

Let us consider N sampling points $\mathbf{x}^i$, which take values $y^i = 1$ or $y^i = -1$. In this work, we let the sampling points be the center points of the elements $\mathbf{x}^i = \mathbf{x}^e$, $y^i = 1$ corresponds to $\rho_e = 1$ and we set $y^i = -1$ for $\rho_e = 0$. Furthermore, we assume that it exists a hyper-plane

$$\mathbf{w} \cdot \mathbf{x} + b = 0, \tag{23}$$

which separate these sampling points into two subsets; one that only takes values $y^i = 1$ (material) and the other one with values $y^i = -1$ (no material). This is shown in Figure 2, where the basic idea of the proposed SVM-based postprocessing approach is illustrated. We also assume that the following constraints are satisfied:

$$y^i(\mathbf{w} \cdot \mathbf{x}^i + b) \geq 1, \qquad i = 1, \ldots, N. \tag{24}$$

The shortest distances to $\mathbf{x}^i$ from a hyper-plane defined in (23) is given by

$$\mathbf{x}^i = \mathbf{x} + \gamma^i \frac{\mathbf{w}}{\|\mathbf{w}\|}. \tag{25}$$

(25) inserted in (24) yields

$$y^i(\mathbf{w} \cdot \mathbf{x} + b + \gamma^i \|\mathbf{w}\|) \geq 1. \tag{26}$$

By utilizing (23), one obtains

$$\gamma^i \geq 1/\|\mathbf{w}\| \qquad \text{for } y^i = 1, \tag{27a}$$

$$\gamma^i \leq -1/\|\mathbf{w}\| \qquad \text{for } y^i = -1. \tag{27b}$$

Thus, the lower bound on the shortest distance $|\gamma^i|$ is maximized by minimizing $\|\mathbf{w}\|$. This is the key idea of the original linear support vector machine formulation, which reads

$$\begin{cases} \min\limits_{(\mathbf{w},b)} \dfrac{1}{2}\|\mathbf{w}\|^2 \\ \text{s.t. } 1 - y^i(\mathbf{w} \cdot \mathbf{x}^i + b) \leq 0, \qquad i = 1, \ldots N. \end{cases} \tag{28}$$

Obviously, the closest sampling points to the optimal hyper-plane

$$\mathbf{w}^* \cdot \mathbf{x} + b^* = 0 \tag{29}$$

are obtained when

$$y^i(\mathbf{w}^* \cdot \mathbf{x}^i + b^*) = 1. \tag{30}$$

Sampling points satisfying (30) are called support vectors, see Figure 2. It is also obvious that in the region between the optimal hyper-plane defined by (29) and the support vectors is empty of sampling points. Thus, the support vector machine formulation in (28) finds a hyper-plane that maximizes the size of this region and in our SVM-based postprocessing approach finds a hyper-plane in the transition between material and non material. Thus, the proposed SVM-based postprocessing approach finds an hyper-surface that is positioned at the center of the blurry region caused by the density filtering.

The Karush-Kuhn-Tucker conditions of the support vector machine in (28) are given by

$$\mathbf{0} = \mathbf{w} - \sum_{i=1}^{N} \lambda_i y^i \mathbf{x}^i, \tag{31a}$$

$$0 = \sum_{i=1}^{N} \lambda_i y^i, \tag{31b}$$

$$\lambda_i \geq 0, \tag{31c}$$

$$1 - y^i(\mathbf{w} \cdot \mathbf{x}^i + b) \leq 0, \tag{31d}$$

$$\lambda_i \left(1 - y^i(\mathbf{w} \cdot \mathbf{x}^i + b)\right) = 0. \tag{31e}$$

The corresponding Lagrangian function is

$$\mathcal{L} = \mathcal{L}(\boldsymbol{\lambda}, \mathbf{w}, b) = \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{N} \lambda_i \left(1 - y^i(\mathbf{w} \cdot \mathbf{x}^i + b)\right). \tag{32}$$

Furthermore, the dual formulation of the support vector machine in (28) reads

$$\max_{\boldsymbol{\lambda} \geq \mathbf{0}} \min_{(\mathbf{w},b)} \mathcal{L}(\boldsymbol{\lambda}, \mathbf{w}, b). \tag{33}$$

By inserting (31a) in (32), one obtains

$$\mathcal{L}(\boldsymbol{\lambda}, \mathbf{w}(\lambda), b) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \lambda_i \lambda_j y^i y^j \mathbf{x}^i \cdot \mathbf{x}^j + \sum_{i=1}^{N} \lambda_i - b \sum_{i=1}^{N} \lambda_i y^i. \tag{34}$$

In addition, the latter part is zero by (31b). In conclusion, the dual support vector machine formulation is given by

$$\begin{cases} \min\limits_{\boldsymbol{\lambda}} \dfrac{1}{2}\displaystyle\sum_{i=1}^{N}\sum_{j=1}^{N} \lambda_i \lambda_j y^i y^j \mathbf{x}^i \cdot \mathbf{x}^j - \sum_{i=1}^{N} \lambda_i \\ \text{s.t.} \begin{cases} \displaystyle\sum_{i=1}^{N} \lambda_i y^i = 0, \\ \lambda_i \geq 0, \qquad i = 1, \ldots, N. \end{cases} \end{cases} \tag{35}$$

From the optimal solution $\boldsymbol{\lambda}^*$ of the dual support vector machine in (35), we obtain the corresponding support vector machine solution from the Karush-Kuhn-Tucker conditions in (31) as

$$\mathbf{w}^* = \sum_{i=1}^{N} \lambda_i^* y^i \mathbf{x}^i \tag{36}$$

and

$$b = 1/y^i - \mathbf{w}^* \cdot \mathbf{x}^i \tag{37}$$

for any $\lambda_i^* > 0$. Notice, by using (36), the optimal hyper-plane in (29) can be written as

$$\sum_{i=1}^{N} \lambda_i^* y^i \mathbf{x}^i \cdot \mathbf{x} + b^* = 0. \tag{38}$$

Notice also that you only need to do the summation over support vector indices, because otherwise λ_i^* equals zero by the KKT-conditions.

For non-separable sets of sampling points $\mathbf{x}^i$, the support vector machine approach presented above will of course not work. However, one might transform the sample set to a new space where it become separable, let say by $\boldsymbol{\xi} = \boldsymbol{\xi}(\mathbf{x})$. In this new space, the only difference in the derivations of the dual support vector machine in (35) and (38) is the appearance of a new scalar product $< \boldsymbol{\xi}^i, \boldsymbol{\xi}^j >$ instead of $\mathbf{x}^i \cdot \mathbf{x}^j$. Thus, we do not have to know the explicit expression of the transformation $\boldsymbol{\xi} = \boldsymbol{\xi}(\mathbf{x})$, but only the expression of the scalar product of the new space. The explicit expression of this scalar product is known to be the kernel function, i.e.

$$k(\mathbf{x}^i, \mathbf{x}^j) = < \boldsymbol{\xi}^i(\mathbf{x}^i), \boldsymbol{\xi}^j(\mathbf{x}^j) > . \tag{39}$$

Consequently, by using an appropriate kernel function in (35) instead of $\mathbf{x}^i \cdot \mathbf{x}^j$, e.g. the Gaussian kernel

$$k = k(\mathbf{x}, \mathbf{z}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{z}\|^2}{2\sigma^2}\right), \tag{40}$$

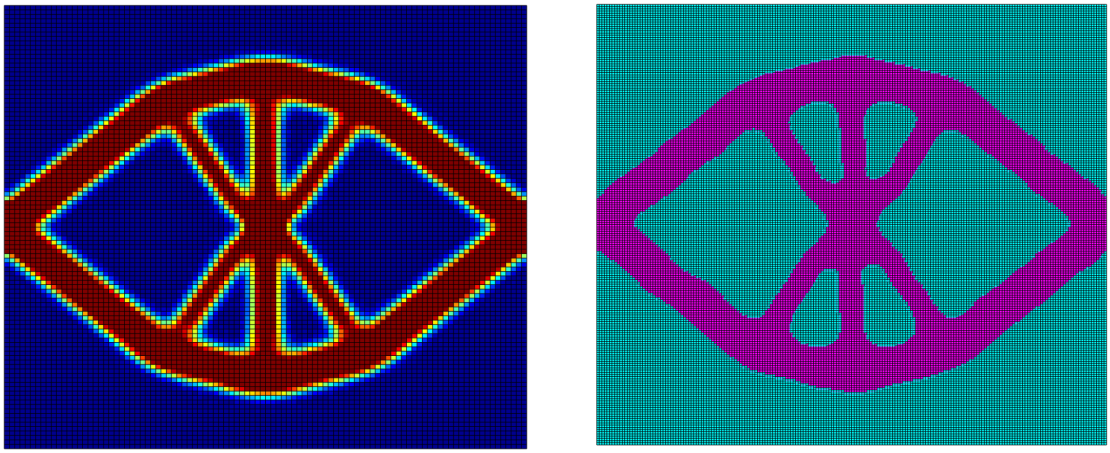

Figure 5. *Element densities and the corresponding soft non-linear support vector machine for Michell's problem.*

the sample set can be separated by

$$\sum_{i=1}^{N} \lambda_i^* y^i k(\mathbf{x}^i, \mathbf{x}) + b^* = 0. \tag{41}$$

Even if we perform a suitable kernel trick, we might have some misclassified points such that (35) does not converge to a solution. This can be treated by applying a regularization of (35). The established soft regularization of (28) is

$$\begin{cases} \min\limits_{(\mathbf{w},b,\mathbf{v})} \dfrac{1}{2}\|\mathbf{w}\|^2 + C\sum\limits_{i=1}^{N} v_i \\ \text{s.t.} \begin{cases} 1 - v_i - y^i(\mathbf{w}\cdot\mathbf{x}^i + b) \le 0, & i = 1,\dots N, \\ v_i \ge 0, & i = 1,\dots N. \end{cases} \end{cases} \tag{42}$$

The Karush-Kuhn-Tucker conditions in (31) then modify by adding the following conditions:

$$C - \lambda_i \ge 0, \tag{43a}$$
$$v_i \ge 0, \tag{43b}$$
$$v_i(C - \lambda_i) = 0. \tag{43c}$$

The corresponding Lagrangian becomes

$$\mathcal{L} = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \lambda_i\lambda_j y^i y^j \mathbf{x}^i \cdot \mathbf{x}^j + \sum_{i=1}^{N}\lambda_i - \sum_{i=1}^{N}\lambda_i v_i + C\sum_{i=1}^{N} v_i, \tag{44}$$

where the two latter terms cancel out due to (43c). Thus, the only difference of the dual support vector machine in (35) for this regularization is the appearance of an upper bound on λ_i, i.e.

$$\begin{cases} \min\limits_{\boldsymbol{\lambda}} \dfrac{1}{2}\sum\limits_{i=1}^{N}\sum\limits_{j=1}^{N} \lambda_i\lambda_j y^i y^j k(\mathbf{x}^i, \mathbf{x}^j) - \sum\limits_{i=1}^{N}\lambda_i \\ \text{s.t.} \begin{cases} \sum\limits_{i=1}^{N}\lambda_i y^i = 0, \\ 0 \le \lambda_i \le C, \qquad i = 1,\dots,N. \end{cases} \end{cases} \tag{45}$$

Finally, $0 < \lambda_i < C$ must be satisfied in order for (37) to be valid. Here, we have also introduced the kernel $k(\mathbf{x},\mathbf{y})$ in the objective function. The soft non-linear SVM in (45) is solved using *quadprog.m* in Matlab or the sequential minimal optimization approach presented in the next section..

4 Sequential Minimal Optimization

A most simple approach, called the coordinate descent method, for solving the unconstrained problem

$$\min_{\mathbf{x}} f(\mathbf{x}) \tag{46}$$

is to minimize $f = f(\mathbf{x})$ with respect to only one component x_i while keeping the remaining components of $\mathbf{x}$ constant. After convergence f is minimized with respect to a new component keeping the other coordinates at constant values. This procedure is then continued in sequence until convergence is obtained. If we also include a linear constant as

$$\alpha_0 + \alpha_1 x_1 + \ldots + \alpha_N x_N = 0, \tag{47}$$

then we can extend the coordinate descent method by minimizing f with respect to two components, say x_1 and x_2, while keeping $x_3,\dots,x_N$ constant. By eliminating x_1 from (47) and inserting $x_2 = x_2(x_1)$ into (46) we recover the original coordinate descent

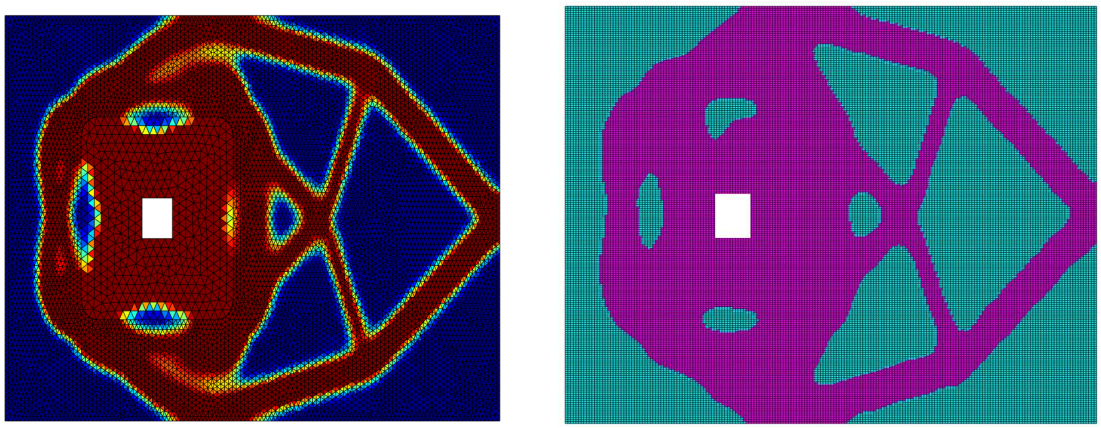

Figure 6. *Element densities and the corresponding soft non-linear support vector machine for the contact problem with non-matching meshes.*

approach. This is the main idea of sequential minimal optimization for solving the dual soft non-linear support vector machine in (45).

Now, let us treat the SVM in (45) by only consider two components of $\boldsymbol{\lambda}$ as variables, let say λ_1 and λ_2, and the remaining components as constants, i.e. $\lambda_3,\ldots,\lambda_N$. Then, we have instead the following problem:

$$\begin{cases} \min\limits_{(\lambda_1,\lambda_2)} \mathcal{L}(\lambda_1,\lambda_2) \\ \text{s.t.} \begin{cases} \lambda_1 y^1 + \lambda_2 y^2 = -\sum\limits_{i=3}^{N} \lambda_i y^i, \\ 0 \le \lambda_1 \le C, 0 \le \lambda_2 \le C, \end{cases} \end{cases} \tag{48}$$

where

$$\begin{aligned} \mathcal{L}(\lambda_1,\lambda_2) = \frac{1}{2}k_{11}\lambda_1^2 + \frac{1}{2}k_{22}\lambda_2^2 + y^1y^2k_{12}\lambda_1\lambda_2 + \ldots \\ \lambda_1 y^1 \sum_{i=3}^{N} \lambda_i y^i k_{1i} + \lambda_2 y^2 \sum_{i=3}^{N} \lambda_i y^i k_{2i} - \lambda_1 - \lambda_2 + D, \end{aligned} \tag{49}$$

$k_{ij} = k(\mathbf{x}^i,\mathbf{x}^j)$ and D is a constant. Notice that

$$w_i = \sum_{j=3}^{N} \lambda_j y^j k_{ij} = f(\mathbf{x}_i) - b - \lambda_1 y^1 k_{1i} - \lambda_2 y^2 k_{2i} \tag{50}$$

for $i = 1,2$, where

$$f(\mathbf{x}) = \sum_{i=1}^{N} \lambda_i y^i k(\mathbf{x}^i,\mathbf{x}) + b. \tag{51}$$

We will use w_i in the derivation presented below.

The idea of sequential minimal optimization is now to solve (48) for pairwise components of $\boldsymbol{\lambda}$ and to do that in sequence for different pairs of $\boldsymbol{\lambda}$ until the global KKT-conditions for (45) are satisfied, which read

$$\lambda_i \ge 0, \tag{52a}$$
$$1 - y^i f(\mathbf{x}^i) - v_i \le 0, \tag{52b}$$
$$\lambda_i \left(1 - y^i f(\mathbf{x}^i) - v_i\right) = 0, \tag{52c}$$
$$v_i \ge 0, \tag{52d}$$
$$\lambda_i - C \le 0, \tag{52e}$$
$$v_i(\lambda_i - C) = 0. \tag{52f}$$

Thus, if $0 < \lambda_i < C$, then

$$y^i f(\mathbf{x}^i) = 1. \tag{53}$$

Furthermore,

$$y^i f(\mathbf{x}^i) \text{ is } \begin{cases} \ge 1 \text{ if } \lambda^i = 0, \\ \le 1 \text{ if } \lambda^i = C. \end{cases} \tag{54}$$

The coordinate descent is now adopted by defining $\lambda_1 = \lambda_1(\lambda_2)$ from the constraint

$$\lambda_1 + y^1y^2\lambda_2 = \gamma, \tag{55}$$

where

$$\gamma = -y^1 \sum_{i=3}^{N} \lambda_i y^i = \lambda_1^{\text{old}} + y^1y^2\lambda_2^{\text{old}} \tag{56}$$

and λ_i^{old} is representing starting values of λ_i $(i = 1,2)$. Notice that $(y^i)^2 = 1$ is utilized in (55). From (55), we have

$$\lambda_1 = \lambda_1(\lambda_2) = \gamma - y^1y^2\lambda_2. \tag{57}$$

By using this in (49), we obtain $\mathcal{L} = \mathcal{L}(\lambda_2) = \mathcal{L}(\lambda_1(\lambda_2), \lambda_2)$. Thus, the problem in (48) becomes

$$\begin{cases} \min\limits_{\lambda_2} \mathcal{L}(\lambda_2) \\ \text{s.t. } \lambda^l \leq \lambda_2 \leq \lambda^u, \end{cases} \tag{58}$$

where λ^l and λ^u represent the lower and upper limits on λ_2, respectively. These limits are derived in detail below.

The necessary optimality condition for (58) reads

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}\lambda_2} = \frac{\partial \mathcal{L}}{\partial \lambda_1}\frac{\partial \lambda_1}{\partial \lambda_2} + \frac{\partial \mathcal{L}}{\partial \lambda_2}, \tag{59}$$

where

$$\frac{\partial \mathcal{L}}{\partial \lambda_1} = \lambda_1 k_{11} + \lambda_2 y^1 y^2 k_{12} + y^1 w_1 - 1, \tag{60a}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_2} = \lambda_2 k_{22} + \lambda_1 y^1 y^2 k_{12} + y^2 w_2 - 1, \tag{60b}$$

$$\frac{\partial \lambda_1}{\partial \lambda_2} = -y^1 y^2. \tag{60c}$$

Inserting (60) into (59) yields

$$\lambda_2 = \frac{\gamma y^1 y^2 (k_{11} - k_{12}) + y^2 (w_1 - w_2) - y^1 y^2 + 1}{k_{11} + k_{22} - 2k_{12}}. \tag{61}$$

The box constraint $0 \leq \lambda_i \leq C$ must also be satisfied. The constraint on λ_1 implies

$$0 \leq \gamma - y^1 y^2 \lambda_2 \leq C. \tag{62}$$

Thus, if $y^1 y^2 = 1$, then

$$\gamma - C \leq \lambda_2 \leq \gamma, \tag{63}$$

but if $y^1 y^2 = -1$, then

$$-\gamma \leq \lambda_2 \leq C - \gamma. \tag{64}$$

In addition,

$$0 \leq \lambda_2 \leq C \tag{65}$$

must of course also be satisfied.

5 NUMERICAL EXAMPLES

The SVM-based postprocessing approach is implemented in our in-house toolbox TopoBox[1] for topology optimization. This first implementation is done for 2D-problems and so far it seems to work very well. This is demonstrated here by studying the problems for the design domains presented in Figure 4. The left picture in this figure shows the design domain and the boundary conditions for one of Michell's classical benchmarks. The center nodes of the left and right side of the design domain are fixed and a vertical force is applied at the center of the design domain.

A typical solution of element densities for Michell's problem is presented in the left plot of Figure 5, clearly showing the transition between densities of zeros and ones depending on the filter radius. This is sometimes removed by applying a heaviside filter. This is not done in this work. The idea of the SVM-based postprocessing approach is instead to find a boundary in this region automatically. This is obtained by training the soft non-linear SVM for the solution of densities according to the approach outlined in the previous sections and the result is plotted to the right in Figure 5. The resemblance of the densities and the SVM is very clear. The number of elements is 10000 and the number of support vectors is 233.

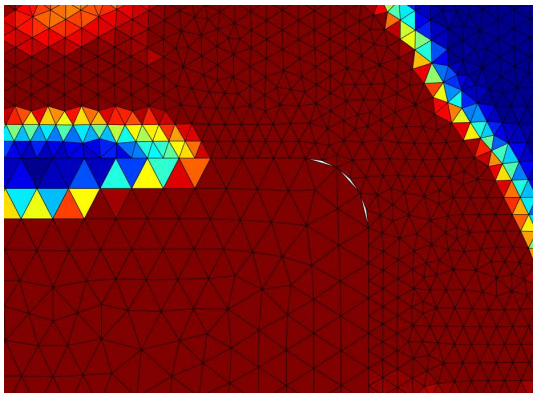

Figure 7. *The upper right corner of the contact region is zoomed in showing the non-matching meshes clearly, which are treated by using the mortar approach.*

The right picture of Figure 4 shows two design domains in unilateral contact with non-matching meshes which we treat with the mortar approach. In Figure 7, the upper right corner of the contact region is zoomed in, clearly showing the non-matching meshes used in the problem. The smaller domain of the two is fixed at the center and the other domain is subjected to a vertical force at the center node of the right side. This force is treated by applying two load cases: one with the force point upwards and

[1] `www.fema.se`

the other one with the load pointing downwards. The solution of densities is plotted to the left in Figure 6 and the corresponding SVM is shown to the right in the same figure. The resemblance of the SVM and the solution of densities is also clear for this example. The number of elements is 18328 and the number of support vectors is now 543.

CONCLUDING REMARKS

An approach for postprocessing topology optimization solutions automatically by using soft non-linear support vector machines is proposed and implemented. The implementation is done in our in-house toolbox TopoBox (`www.fema.se`) for two-dimensional design domains. Preliminary results are most promising showing that the SVM represents the transition boundary between densities with zeros and ones most accurately. One should remark that no additional filtering has been utilized. In a near future the implementation will be done for three-dimensional design domains. Our believe is that this also will produce promising results and that the next step then would be to integrate this SVM-based postprocessing approach with a CAD software. The CAD geometry is then simply defined by the support vectors. This is a topic for future work.

Acknowledgements
I am grateful for the financial support from Vinnova (www.vinnova.se) in the project Digi3D (www.digi3d.org).

REFERENCES

[1] J.-H. Zhu, W.-H. Zhang & L. Xia, Topology Optimization in Aircraft and Aerospace Structures Design, *Archives of Computational Methods in Engineering*, **23**, 595–622, 2016.

[2] C. Cortes & V. Vapnik, Support-Vector Networks, *Machine Learning*, **20**, 273–297, 1995.

[3] S. Chu, M. Xiao, L. Gao, L. Gui & H. Li, An Effective Structural Boundary Processing Method Based on Support Vector Machine for Discrete Topology Optimization, in the proceedings of *the 2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design*, 4–6 May, Nanchang, China, 2016.

[4] C Lin & L. Chao, Automated Image Intepretation for Integrated Topology and Shape Optimization, *Structural and Multidisciplinary Optimization*, **20**, 125–137, 2000.

[5] A. Koguchi & N. Kikuchi, A Surface Reconstruction Algorithm for Topology Optimization, *Engineering with Computers*, **22**, 1-10, 2006.

[6] A.R. Yildiz, N. Öztürk, N. Kaya & F. Öztürk, Integrated Optimal Topology Design and Shape Optimization using Neural Networks, *Structural and Multidisciplinary Optimization*, **25**, 251-260, 2003.

[7] Poh-Soong Tang & Kuang-Hua Chang, Integration of Topology and Shape Optimization for Design of Structural Components, *Structural and Multidisciplinary Optimization*, **22**, 65-82, 2001.

[8] Gang-Won Jang, Kyung Joo Kim & Yoon Young Kim, Integrated Topology and Shape Optimization Software for Compliant MEMS Mechanism Design, *Advances in Engineering Software*, **39**,1-14, 2008.

[9] Yeh-Liang Hsu, Ming-Sho Hs & Chuan-Tang Chen, Interpreting Results from Topology Optimization using Density Contours, *Computers & Structures*, **79**, 1049–1058, 2001.

[10] L.H. Hamel, *Knowledge Discovery with Support Vector Machines*, Wiley-Blackwell, Hoboken, New Jersey, 2009.

[11] M.P. Bendsøe. & O. Sigmund, *Topology Optimization: Theory, Methods, and Applications*, Berlin: Springer, 2003.

[12] J. Platt, Fast training of SVMs using Sequential Minimal Optimization. *Advances in Kernel Methods Support Vector Machine, Hoboken*, MIT Press, Cambridge, 185–208, 1999.

[13] N. Strömberg, Topology Optimization of Orthotropic Elastic Design Domains with Mortar Contact Conditions, in *Advances in Structural and Multidisciplinary Optimization*, 1427–1438, 2018.

[14] N. Strömberg and A. Klarbring, Topology Optimization of Structures in Unilateral Contact, *Structural and Multidisciplinary Optimization*, **41**, 57–64, 2010.

[15] N. Strömberg, Topology Optimization of Structures with Manufacturing and Unilateral Contact Constraints by Minimizing an Adjustable Compliance-Volume Product, *Structural and Multidisciplinary Optimization*, **42**, 341–350, 2010.

[16] A. Popp, M. Gitterle, M.W. Gee and W.A. Wall, A Dual Mortar Approach for 3D Finite Deformation Contact with Consistent Linearization, *International Journal for Numerical Methods in Engineering*, **83**, 1428–1465, 2010.

[17] C. Wilking and M. Bischoff, Alternative Integration Algorithms for Three-Dimensional Mortar Contact, *Computational Mechanics*, **59**, 203–218, 2017.

[18] A. Klarbring and N. Strömberg, Topology Optimization of Hyperelastic Bodies including Non-Zero Prescribed Displacements, *Structural and Multidisciplinary Optimization*, **47**, 37–48, 2013.

[19] A. Klarbring and N. Strömberg, A Note on the Min-Max Formulation of Stiffness Optimization including Non-Zero Prescribed Displacements, *Structural and Multidisciplinary Optimization*, **45**, 147–149, 2012.

[20] N. Strömberg, An Augmented Lagrangian Method for Fretting Problems, *European Journal of Mechanics, A/Solids*, **16**, 573–593, 1997.