

Active-Learning Combined with Topology Optimization for Top-Down Design of Multi-Component Systems

L. Krischer , A. Vazhapilli Sureshbabu and M. Zimmermann

Technical University of Munich, Germany

 lukas.krischer@tum.de

Abstract

In top-down design, optimal component requirements are difficult to derive, as the feasible components that satisfy these requirements are yet to be designed and hence unknown. Meta models that provide feasibility and mass estimates for component performance are used for optimal requirement decomposition in an existing approach. This paper (1) extends its applicability adapting it to varying design domains, and (2) increases its efficiency by active-learning. Applying it to the design of a robot arm produces a result that is 1% heavier than the reference obtained by monolithic optimization.

Keywords: topological optimisation, artificial intelligence (AI), data-driven design, systems engineering (SE)

1. Introduction

Designing systems with many interacting components can be a difficult task due to many disciplines and departments involved throughout the product development process. In classical top-down development, requirements are therefore first formulated on the system level and then passed on to lower levels and finally the respective components (Forsberg and Mooz, 1991). The components are then typically designed by separate engineering groups. This procedure is especially beneficial from a designer's perspective who may work on parts rather than the entire system at once (Eckert and Clarkson, 2005). However, since the detail level information is not a priori known, suboptimal system level decisions may lead to iterations or inferior designs (Zimmermann *et al.*, 2017).

In the context of design optimization, optimization architectures were developed to resemble this distributed development process. Distributed optimization architectures decompose a given system into smaller subproblems allowing for individual design by separate groups. However, most of these optimization architectures are not fully separable and rely on a coordination strategy to maintain consistency between shared quantities of components and the system (Martins and Lambe, 2013). For instance, Albers and Ottnad (2008) carried out a distributed optimization, where a static topology optimization of a humanoid robot arm was combined with a dynamic co-simulation of a multi-body simulation and a control scheme. Beernaert and Etman (2019) presented an automatic way of converting requirements into a distributed optimization problem and solving it for a multidisciplinary two-level power train design. Further, Kim *et al.* (2016) and Wang *et al.* (2019) decomposed structural optimization problems on the system level by decoupling the respective physical components using meta models allowing for independent component development. However, this decomposition is carried out iteratively with a feedback loop that is updating the system level with bottom-up information.

By contrast, complete decoupling, i.e., horizontal decomposition between system and component level and vertical decomposition between different components, makes further coordination between

disciplines or components after the decomposition unnecessary. This kind of decoupling was introduced by Zimmermann and von Hoessle (2013) and was further developed into a design procedure in Zimmermann et al. (2017). However, missing detail level information can lead to suboptimal system level decisions and due to the complete decoupling to an exclusion of optimal designs (Krischer et al., 2020). Krischer and Zimmerman (2021) therefore established a design procedure for complete decoupling that uses meta models containing detail level information about component properties. A regression model estimates the lowest possible mass with respect to component stiffness. The component properties can therefore be assigned in a more mass-optimal way. A classification model allows for estimates of physical feasibility, i.e., the ability to realize a given component performance with a detail level design in a later stage. The developed procedure was referred to as *Informed Decomposition*. However, the process of establishing meta models is computationally expensive. Mechanical similarity analysis allows for higher applicability for same characteristic sizes of the components (Ramu et al., 2013). Nevertheless, sometimes a new training process is inevitable. A possible way to enhance the meta models further is to add input parameters that consider geometrical changes of the components explicitly. In this way, a meta model is valid over a wider range of different geometrical design domains.

In this paper, the existing approach to design multi-component systems by a system optimization and completely decoupled component optimizations is extended to meta models for varying geometrical design domains. The meta models are trained by a 2-phase active-learning strategy, considering not only mechanical stiffness properties, but also different lengths of the components allowing for higher applicability. The procedure is applied to a three-component robot arm, where the components are designed via a 3-dimensional topology optimization. For the components, that all differ in length, only one regression model and one classifier model need to be trained. The paper is organized as follows: In Section 2, the design problem is introduced, Section 3 presents the proposed approach with the developed extension to varying geometrical design domains and the active-learning strategy. In Section 4, results for 3-dimensional topology optimization are presented and compared to a full monolithic optimization. The results are then discussed in Section 5 and the conclusion is presented in Section 6.

2. Design problem

A seven degrees of freedom robot arm is to be designed. In particular, the weight-optimal shape of the components is to be determined. The arm consists of three components: upper arm (1), lower arm (2), and hand (3), with the lengths $l_{(1)} = 100$ mm, $l_{(2)} = 200$ mm and $l_{(3)} = 50$ mm, respectively. The three degrees of freedom shoulder is realized with three subsequent revolute joints 1-3, whereas the elbow 4-5 and the wrist 6-7 possess two revolute joints each for the pitch and roll movement, see Figure 1 (a). The requirement on the system stiffness is: *The robot arm must sustain a vertical load of $F = 50$ N with a maximum tip displacement of $d_c = 0.3$ mm in a straight position.* Under the assumption of rigid joints, the robot arm is clamped on the left side at the last shoulder joint 3, while the payload F is applied on the right end of the robot arm, see Figure 1 (b). Each component is modelled with ABS as linear elastic ($E = 1.91$ GPa, $\nu = 0.36$, $\rho = 1.07$ g/cm³) and has two interfaces. Each interface possesses two degrees of freedom: one translational and one rotational, see Figure 1 (c).

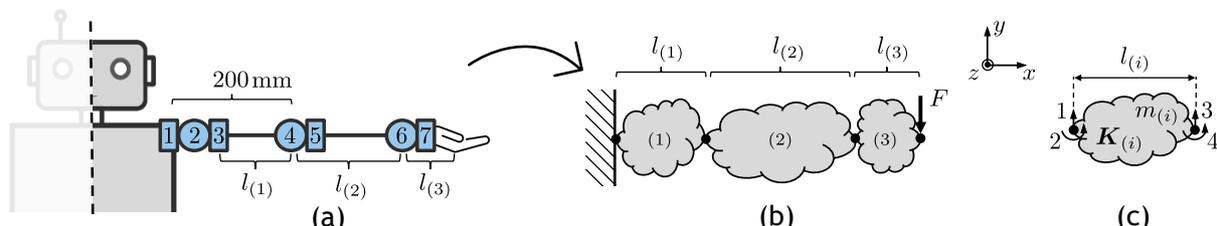


Figure 1. Three-component robot arm with seven degrees of freedom (a), the reference load case for the given requirement (b) and the component description with two interfaces and two degrees of freedom (c)

Figure 2 (a) illustrates the dependencies between all relevant quantities that are needed to solve the given design problem. The design variable vector $x_{(i)}$ includes all design details for component i on the

component-detail level (III). For a given material, the design variable vector $x_{(i)}$ determines the detailed stiffness matrix $K_{d(i)}$ that can be represented by the component stiffness matrix $K_{(i)} \in R^{4 \times 4}$ on the component-performance level (II). Under a given load F , the structure deforms resulting in a system tip displacement d . Similarly, the detailed description $x_{(i)}$ also defines the mass $m_{(i)}$ of each component and consequently the system mass $m = \sum_{i=1}^3 m_{(i)}$. The monolithic optimization problem reads

$$\begin{aligned} \min_{x_{(i)}} \quad & \sum m_{(i)}(x_{(i)}, l_{(i)}) \\ \text{s. t.} \quad & d(x_{(i)}, l_{(i)}) - d_c \leq 0 \\ & x_{lb} \leq x_{(i)} \leq x_{ub} \quad \text{for } i = 1, 2, 3. \end{aligned} \quad (1)$$

The results of this monolithic optimization problem serve us as a comparison for the optimality of the proposed approach in Chapter 3.

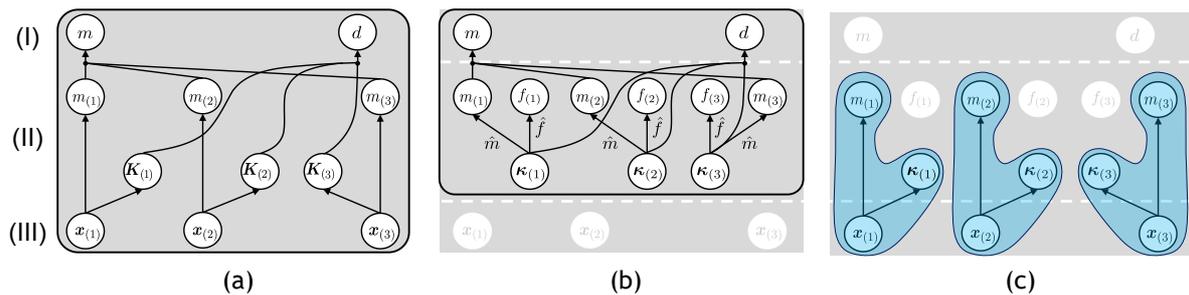


Figure 2. Dependencies between all relevant quantities: on the system level (I), component-performance level (II) and the component-detail level (III) in a monolithic approach (a) and for the proposed approach consisting of a system optimization (b) and component optimizations (c)

3. Informed decomposition

3.1. Systems design

To enable a development process that is separated for three components, the given design problem of Chapter 2 is decomposed. The proposed systems design approach, see Figure 3 (a), consists of a system optimization and respective n completely decoupled component optimizations.

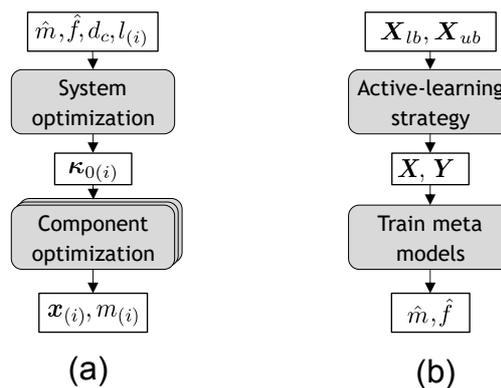


Figure 3. Overview over the proposed approach consisting of systems design (a) and a procedure to establish meta models using an active-learning strategy (b)

The system optimization decomposes the given design problem in a mass-optimal way utilizing meta models for estimates on mass m and feasibility f , while satisfying the given system stiffness requirement on d , see Figure 2 (b). In contrast to [Krischer and Zimmermann \(2021\)](#), κ consists of three diagonal entries instead of the eigenvalues of the component stiffness matrix $K \in R^{4 \times 4}$

$$\kappa = [k_{11}, k_{22}, k_{44}]. \quad (2)$$

It can be shown that for a given design domain and two interfaces with two degrees of freedom, this is a sufficient representation of \mathbf{K} . k_{11} is hereby the stiffness with respect to translational deformations of the first interface, whereas k_{22} and k_{44} represent the stiffnesses with respect to rotational deformations of both interfaces, as shown in Figure 1 (c). The system optimization problem reads

$$\begin{aligned} \min_{\boldsymbol{\kappa}_{(1)}, \boldsymbol{\kappa}_{(2)}, \boldsymbol{\kappa}_{(3)}} \quad & \sum \hat{m}(\boldsymbol{\kappa}_{(i)}, l_{(i)}) \\ \text{s. t. :} \quad & d(\boldsymbol{\kappa}_{(i)}, l_{(i)}) - d_c \leq 0 \\ & -\hat{f}(\boldsymbol{\kappa}_{(i)}, l_{(i)}) \leq 0 \\ & \boldsymbol{\kappa}_{lb} \leq \boldsymbol{\kappa}_{(i)} \leq \boldsymbol{\kappa}_{ub} \quad \text{for } i = 1, 2, 3. \end{aligned} \quad (3)$$

The system optimization does not solve the high dimensional system equations of the detailed stiffness matrices $\mathbf{K}_{d(i)}$ related to $\mathbf{x}_{(i)}$, but only computes the displacement $d(\boldsymbol{\kappa}_{(i)}, l_{(i)})$ for the low dimensional component stiffness matrices $\mathbf{K}_{(i)} \in \mathbb{R}^{4 \times 4}$ with the design variables $\boldsymbol{\kappa}_{(i)}$, making the system optimization less expensive. For a given length $l_{(i)}$, the functions \hat{m} and \hat{f} map the stiffness entries on mass and feasibility estimates for all three components. A particle swarm optimization is utilized to solve the given system optimization problem (Eberhart and Kennedy, 1995). The solutions $\boldsymbol{\kappa}_{(i)}$ of Equation 3 are used as reference stiffnesses $\boldsymbol{\kappa}_{0(i)}$ of the subsequent $n=3$ completely decoupled component optimizations.

The component optimization between level (II) and (III), as shown in Figure 2 (c), determines the optimal design variable vector $\mathbf{x}_{(i)}$ for each component with minimum mass $m_{(i)}$ for a given $\boldsymbol{\kappa}_{0(i)}$. The associated optimization problem reads

$$\begin{aligned} \min_{\mathbf{x}_{(i)}} \quad & m_{(i)}(\mathbf{x}_{(i)}, l_{(i)}) \\ \text{s. t. :} \quad & |\boldsymbol{\kappa}_{(i)}(\mathbf{x}_{(i)}, l_{(i)}) - \boldsymbol{\kappa}_{0(i)}| \leq 0 \\ & \mathbf{x}_{lb} \leq \mathbf{x}_{(i)} \leq \mathbf{x}_{ub} \quad \text{for } i = 1, 2, 3 \end{aligned} \quad (4)$$

where \mathbf{x}_{lb} and \mathbf{x}_{ub} are the lower and upper bounds on the detailed design variables $\mathbf{x}_{(i)}$, $\boldsymbol{\kappa}_{(i)}$ is the representation of the candidate component stiffness matrix $\mathbf{K}_{(i)}$ associated with $\mathbf{x}_{(i)}$ for a given length $l_{(i)}$. The stiffness constraint $|\boldsymbol{\kappa}_{(i)}(\mathbf{x}_{(i)}, l_{(i)}) - \boldsymbol{\kappa}_{0(i)}|$ is formulated for numerical processing as

$$-\epsilon \leq \left[\begin{array}{c} \kappa_{j,0} - \kappa_{j,0} \\ \kappa_{j,0} \end{array} \right] \leq \epsilon \quad \text{for } j = 1, 2, 3. \quad (5)$$

In the following, the design domain for each component is restricted to a height of $h = 100$ mm and a width of $w = 25$ mm, the connection to the interface is modelled as rigid. The modelling process consists of three steps (Figure 4):

- Discretization of the design domain with 3-dimensional brick elements \mathbf{K}_e ,
- Static condensation of the design domain with respect to the left and right side of the domain,
- Kinematic condensation with respect to the four interface degrees of freedom.

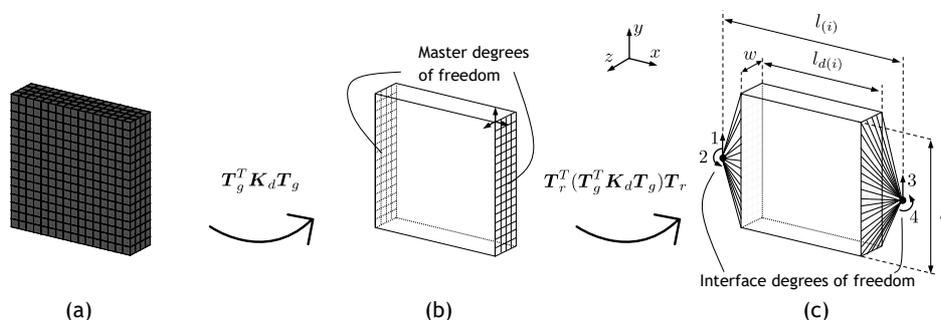


Figure 4. Modelling process of the component stiffness matrix consisting of discretization (a), static condensation (b) and kinematic condensation (c)

The design domain for each component of length $l_{d(i)}$ is discretized with $n_e = 1024$ 3-dimensional brick elements \mathbf{K}_e , see Figure 4 (a). For the design of the components, a 3-dimensional topology optimization based on the SIMP method is utilized (Bendsøe and Sigmund, 2004). The design variable vector for the component optimization of Equation 4 is therefore defined as the element densities

$$\mathbf{x} = [\rho_e], \quad (6)$$

with a penalty factor of $p = 3$ and a sensitivity filter radius of $r = \sqrt{3}$. To ensure a planar deformation, planar-symmetry with respect to the x-y and x-z-plane is enforced. The detailed stiffness matrix reads

$$\mathbf{K}_d = \sum_{e=1}^{n_e} \rho_e^p \mathbf{K}_e. \quad (7)$$

In order to determine the elastic deformation behaviour with respect to the interface degrees of freedom, first a static condensation (Guyan, 1965) is carried out to the master degrees of freedom for the left and right side of the design domain, Figure 4 (b). The remaining degrees of freedom of the design domain are then connected rigidly to the interface degrees of freedom using a kinematic condensation with a multi-point constraint, e.g. Liu and Quek (2013), see Figure 4 (c). The final component stiffness matrix is

$$\mathbf{K}_{rg} = \mathbf{T}_r^T (\mathbf{T}_g^T \mathbf{K}_d \mathbf{T}_g) \mathbf{T}_r, \quad (8)$$

with the transformation matrices \mathbf{T}_g for the static and \mathbf{T}_r for the kinematic condensation. The optimization problem is solved utilizing the Method of Moving Asymptotes by Svanberg (1987).

The derivatives with respect to the entries $\boldsymbol{\kappa} = [k_{11}, k_{22}, k_{44}]$ of the component stiffness matrix $\mathbf{K}_{rg} \in \mathbb{R}^{4 \times 4}$ are

$$\frac{\partial \mathbf{K}_{rg}}{\partial \rho_e} = \mathbf{T}_r^T \left(\mathbf{T}_g^T \frac{\partial \mathbf{K}_d}{\partial \rho_e} \mathbf{T}_g \right) \mathbf{T}_r, \quad \frac{\partial \mathbf{K}_d}{\partial \rho_e} = p \rho_e^{p-1} \mathbf{K}_e \quad (9)$$

and the mass gradients can be computed as

$$\frac{\partial m}{\partial \rho_e} = \rho L_e H_e W_e, \quad (10)$$

with ρ as the density of ABS and L_e, H_e , and W_e as the element dimensions. Since the problem was completely decoupled by the system optimization, the system stiffness measured by the total system displacement d is assumed to satisfy the requirement $d \leq d_c$, if the component optimizations are all feasible.

3.2. Establishing meta models using an active-learning strategy

When the feasibility estimator \hat{f} and the mass estimator \hat{m} are not available, meta models can be established following the procedure depicted in Figure 3 (b). First, the input sample data needs to be created by sampling the input space $[k_{11} \times k_{22} \times k_{44} \times l] \in \mathbb{R}^4$

$$\mathbf{X}_A = [k_{11}, k_{22}, k_{44}, l]_A, \quad (11)$$

within the bounds $[\mathbf{X}_{lb}, \mathbf{X}_{ub}]$. The sample output vector contains information about feasibility f_A and mass m_A

$$\mathbf{Y}_A = [f, m]_A. \quad (12)$$

The sample data $[\mathbf{X}, \mathbf{Y}] \in \mathbb{R}^{N \times 6}$, can be used to train the feasibility estimator \hat{f} and the mass estimator \hat{m} for the system optimization problem of Equation 3. However, most combinations of stiffness entries $\boldsymbol{\kappa} = [k_{11}, k_{22}, k_{44}]_A$ for a given length l_A cannot be realized physically, i.e., are infeasible. According to Krischer and Zimmermann (2021), the following requirements must be satisfied to have a feasible input sample \mathbf{X}_A :

1. \mathbf{K} must be symmetric,
2. rigid body deformations should result in zero forces,
3. \mathbf{K} must be positive semi-definite,
4. a detailed geometry \mathbf{x} must exist with an associated stiffness $\boldsymbol{\kappa}$ at length l .

For each \mathbf{X}_A a detail level design \mathbf{x}_A is computed using the component optimization problem of Equation 4. The requirements are satisfied if there exists a corresponding component-detail design \mathbf{x}_A , i.e., if the optimization algorithm converges. The feasibility flag is then set to $f_A = 1$ and added to the input sample together with the corresponding mass m_A , otherwise the sample point is infeasible $f_A = -1$ and no mass is assigned, $m_A = []$. Due to the imposed requirements on the stiffness matrix, any data set is assumed to have much more infeasible than feasible data points. This is not only disadvantageous for the training process of the classifier, but also makes the process less efficient, because infeasible data points can only be used for the feasibility estimator, but not for the mass estimator. Since each point is evaluated carrying out an optimization involving the high dimensional component-detail level, the sampling is computationally expensive.

For this reason, [Krischer and Zimmerman \(2021\)](#) developed a special parametrization of stiffness matrices using eigenvalues and eigenvectors to reduce the amount of infeasible data. However, a generalization to arbitrary mechanical components in higher dimensions can be difficult. Another way of dealing with imbalanced training data are active-learning sampling strategies. Active-learning refers here to algorithms that automatically choose data points from which they learn, hence feasible regions can be sampled efficiently ([Kremer et al., 2014](#); [Ertekin et al., 2007](#)). One specific class of active-learning strategies is called uncertainty sampling, where the sample points are selected in regions where the classifier is most uncertain about, i.e., closest to the classification boundary. Support Vector Machines (SVM) are especially suited for this kind of sampling due to the underlying mathematical training concept of minimizing the distances to the separating classification hyperplane ([Ertekin et al., 2007](#); [Kremer et al., 2014](#); [Corinna Cortes and Vladimir Vapnik, 1995](#)). Based on uncertainty sampling a two-phase active-learning strategy is proposed consisting of (i) classification sampling and (ii) regression sampling, see Figure 5 (a).

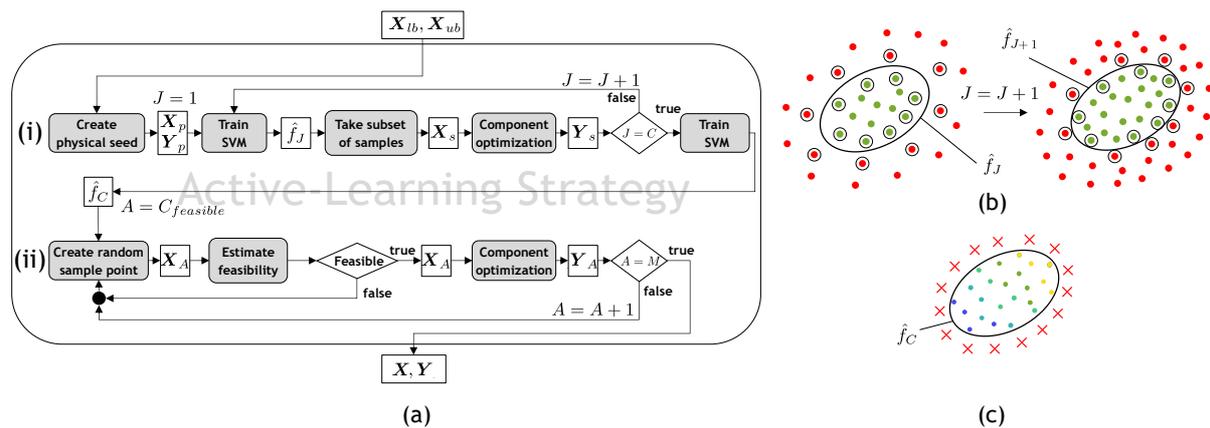


Figure 5. Active-learning strategy: process (a), sample designs for training the feasibility estimator (b), and sample designs for training the mass estimator (c). Green and red dots in (b) show feasible and infeasible sample points, respectively, circles in (b) indicate proximity to the feasibility boundary, and red crosses in (c) mark infeasible designs that are ignored

The classification sampling (i) approximates the hyperplane between feasible and infeasible designs by selecting sample points according to temporarily trained feasibility estimators \hat{f}_J . In each iteration, \hat{f}_J is trained on an increasing set of training data to improve the selection quality of the sample points $\mathbf{X} \in \mathbb{R}^{N_c \times 4}$ gradually. For the first iteration $J = 1$, a physical seed \mathbf{X}_p is created, by randomly sampling the detail level design variables \mathbf{x}_p and the length l_p . Thus, all sample points of the first iteration exist and are therefore feasible, $\mathbf{Y}_p = \mathbf{1}, \forall \mathbf{X}_p$. Since only samples of one class (feasible designs) are created, a one-class SVM is trained to select the next input samples. For the consecutive iterations a binary SVM is trained (feasible/infeasible points). As illustrated in Figure 5 (b), for each iteration J , a random set of samples $\mathbf{X}_t \in \mathbb{R}^{N_t \times 4}$, is created and evaluated by the respective estimator \hat{f}_J (green and red dots). Afterwards, a subset of sample points $\mathbf{X}_s \in \mathbb{R}^{N_s \times 4}$, is taken (black circled dots) that have the closest distance to the estimated hyperplane, with $N_t \gg N_s$. For each iteration, the

sample size N_t of the temporary samples \mathbf{X}_t is increased while the sample size N_s of the subset \mathbf{X}_s is kept constant. Since the ratio N_t/N_s increases for each iteration, the selected subset of sample points \mathbf{X}_s of iteration $J + 1$ are likely to be closer to the trained hyperplane than the sample points of iteration J . Each selected sample point \mathbf{X}_s is then evaluated using the component optimization of Equation 4. After a prescribed number of iterations C , a last classifier \hat{f}_C is trained on the whole sample data and phase (i) of the proposed active-learning strategy is completed.

Next, the regression sampling (ii) is initiated, where N_M mass samples are selected inside the feasible region of classifier \hat{f}_C , as shown in Figure 5 (c). Note, that all feasible points of the previous phase (i) can be already added to the set of sample points. Each remaining sample point \mathbf{X}_A is created randomly within the bounds of the input space and is evaluated by the classifier \hat{f}_C . If the sample point is classified as infeasible, the design is neglected, and a new point is created and evaluated. If it is feasible, the expensive component optimization is carried out to compute the respective mass and recheck the feasibility. If the predefined number of sample point N_M is achieved, phase (ii) is completed, and the sample data can be utilized for the training of the final meta models \hat{m} and \hat{f} with respect to the whole data set $[\mathbf{X}, \mathbf{Y}]$, as illustrated in Figure 3 (b).

4. Results

The proposed Informed Decomposition was implemented in MATLAB© and was applied to the design problem of Chapter 2. The results are then compared to the monolithic optimization of Equation 1. First, new meta models were established using the active-learning strategy of Chapter 0. The sample data of the classification sampling can be seen in Figure 6 (i). For $C = 10$ iterations, $N_C = 4800$ samples points were created. The final classification data set consists of 3091 feasible and 1709 infeasible data points resulting in even more feasible than infeasible data points, while still being sufficiently well-balanced. During the classification sampling, points were mostly selected at the vicinity of the classification boundary, hence many infeasible regions remained unsampled. For the regression sampling, see Figure 6 (ii), $N = 5000$ predicted mass samples were utilized, whereas $N_M = 4967$ mass samples were feasible after evaluating them. Having a closer look at the feasible region, one can observe that the mass increases for higher stiffness values approaching the boundary to the infeasible region. Additionally, in accordance with the expected deformation behaviour of the components, the stiffness characteristics for constant component height and width decreases for greater length l and increases for shorter l . Finally, fewer sample data is available in high stiffness regions, e.g. $[k_{11}, k_{22}]$, indicating for a thinner feasible design space, that is difficult to sample for the developed active-learning strategy.

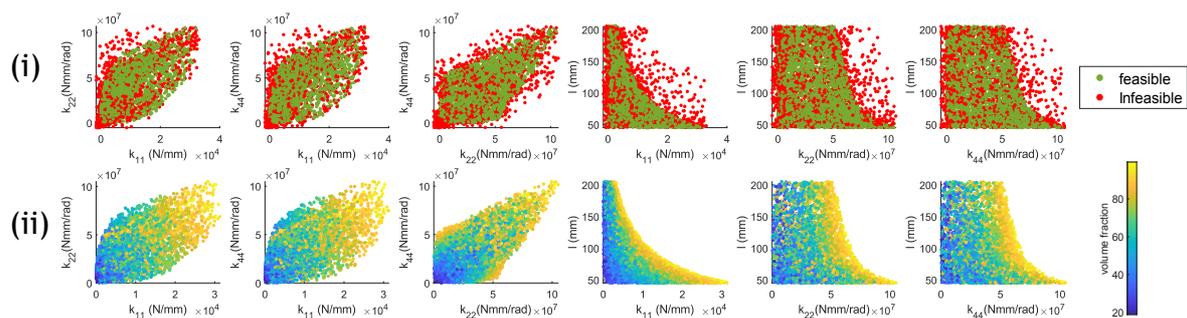


Figure 6. Sampling results of the developed active-learning strategy for the classification sampling (i) and the regression sampling (ii)

Next, the meta models were trained. For the feasibility estimator, the data is split into 80% training and 20% test samples. The hyperparameters of the SVM are determined using a Bayesian optimization with a five-fold cross-validation on the training data. The resulting classifier has a false positive rate of $FP = 1.7\%$, a true positive rate of $TP = 95\%$ and an accuracy of $acc = 96\%$. The mass estimator is realized with a Feedforward Artificial Neural Network (ANN). The sample data is split into 80% training, 10% validation and 10% test samples. The ANN possesses $n_{hl} = 4$ hidden layers and

$nn = 13$ neurons with a R^2 -value of $R^2 = 0.996$ and a mean squared error of $mse = 1.35$ with respect to the volume fraction of the initial design domain.

Having established the meta models, Informed Decomposition can be utilized to design the robot arm. First, the system optimization of Equation 3 was solved using the trained meta models. The results of the system optimization were compared to the ones of the monolithic optimization of Equation 1. The respective first three columns of Table 1 show the component performances κ , represented by the stiffness matrix entries, for all components for both approaches. The results for component (1) and (2) of the system optimization and the monolithic optimization only slightly deviate. The results of component (3) show higher deviations in the last rotational stiffness entry k_{44} .

Table 1. Optimized component performances of the robot arm for the Informed Decomposition and the benchmark monolithic optimization and the respective component and system masses

Com P.	Informed Decomposition				Monolithic Optimization			
	k_{11} (N/mm)	k_{22} (Nmm/rad)	k_{44} (Nmm/rad)	m (kg)	k_{11} (N/mm)	k_{22} (Nmm/rad)	k_{44} (Nmm/rad)	m (kg)
(1)	$1.94 \cdot 10^3$	$4.14 \cdot 10^7$	$3.43 \cdot 10^7$	0.157	$1.83 \cdot 10^3$	$4.27 \cdot 10^7$	$3.43 \cdot 10^7$	0.164
(2)	646	$2.33 \cdot 10^7$	$7.32 \cdot 10^6$	0.249	552	$2.22 \cdot 10^7$	$7.30 \cdot 10^6$	0.242
(3)	$3.64 \cdot 10^3$	$9.39 \cdot 10^6$	$2.18 \cdot 10^6$	0.037	$3.38 \cdot 10^3$	$8.80 \cdot 10^6$	$4.10 \cdot 10^5$	0.035
	Σ				Σ			
	0.443				0.441			

The component performances κ are then subsequently used as reference values for the respective component optimizations of Equation 4. The resulting topologies are shown in Figure 7, whereas the masses are contained in the last column of Table 1. The robot arm designed by the Informed Decomposition shows only small deviations with respect to the topologies of component (1) and (2), whereas especially component (2) shows more intermediate element densities ρ_e for the proposed approach. In accordance with the deviating stiffness entry k_{44} of component (3), also the topology deviates showing an additional horizontal reinforcement for the right interface strengthening the rotational stiffness. However, since the third component possesses the smallest design domain, the deviation affects the overall weight only slightly. In conclusion, the total mass of the robot arm for the proposed Informed Decomposition is $m = 0.443$ kg which is a deviation of 0.5% with respect to the benchmark monolithic optimization, while both approaches satisfy the given system requirement $d \leq d_c$.

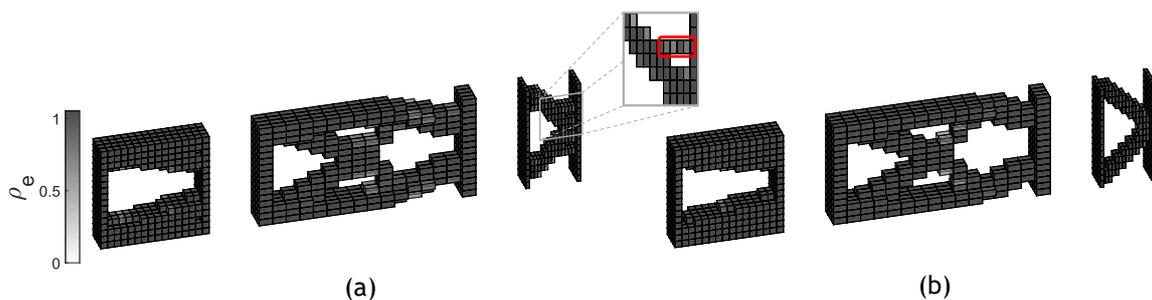


Figure 7. Final topologies for the Informed Decomposition (a) and the benchmark monolithic optimization (b)

5. Discussion

The proposed approach was capable of completely decoupling a robot arm with varying component lengths enabling an independent and separate design process. Despite the complete decoupling, which supports the classical product development process, the system mass m only slightly differs from the benchmark design with a deviation below 1%. This was achieved by meta models that provide information of the components to the system level before the components are actually designed.

The generation of the sample data for the meta models, i.e., mass and feasibility estimator, is the critical step within the proposed approach. Each sample point requires a computationally expensive component optimization leading to high sampling cost. The extension to varying geometrical design domains is therefore an important step for a higher applicability of each model. Eventually, the idea is to set up an off-line data base containing multiple meta models that can be used for arbitrary mechanical design problems making the need for a new training process very unlikely. So far, only the length can be changed while height and width remain constant. This limits the current approach since the design domain cannot be altered arbitrarily.

The implemented active-learning strategy helps to enable an efficient sampling procedure while also providing a sufficiently well-balanced data set. However, the initial physical seed influences the course of the sampling strategy, and in general it is not guaranteed that the whole design space is in fact captured during the sampling procedure. For instance, as pointed out in Chapter 4, high stiffness regions seem to be not fully captured in the current procedure.

Finally, topology optimization results often need a post-processing step to transfer the computed discretized, but continuous results (Figure 7) into a discrete physical product (Figure 8). Small changes in the obtained solution, however, might cause an invalid component behaviour violating consequently the system requirement. Hence, e.g., a consecutive shape optimization on the parameterized model could be carried out to ensure the validity of the approach.

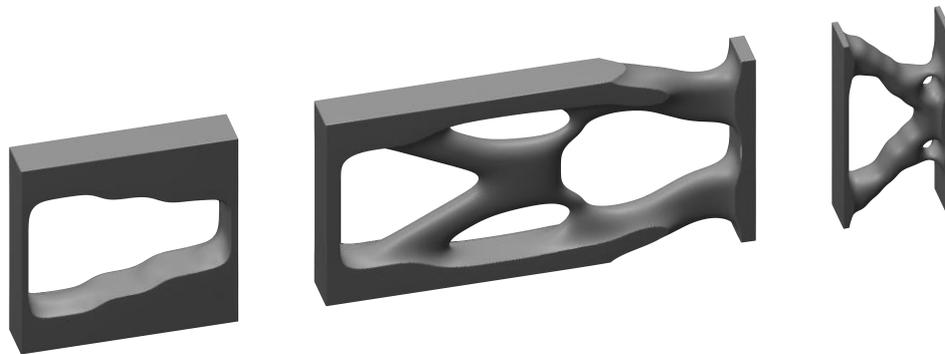


Figure 8. Topology optimized and post-processed three-component robot arm

6. Conclusion

In top-down design of multi-component systems, missing detail information often leads to suboptimal system level decisions and hence inferior designs. Meta models can be used to provide this information to the system level. In this paper, the top-down approach introduced in [Krischer and Zimmermann \(2021\)](#), consisting of an informed system optimization using meta models and completely decoupled component optimizations, was extended to meta models that are also valid over a range of varying component lengths. Hence, mechanical design problems can be solved without the need of carrying out the training process of the meta models every single time. If a new model is needed, an active-learning strategy, consisting of a classification and regression sampling phase, was developed. This new two-phase sampling strategy ensures an efficient sampling process together with a sufficiently well-balanced training data.

The extended approach was then applied to a robot arm with three components of different length and compared to a monolithic system optimization, where the entire mechanical system is designed at once. The parts are designed utilizing a 3-dimensional topology optimization scheme. For all three components, only one mass and one feasibility estimator were trained. It was shown that the proposed approach can assign completely decoupled component requirements in a feasible and mass-optimal manner, hence enabling an efficient and separated design process. The resulting robot arm's weight only differs below 1% from the results of the monolithic optimization while satisfying the system requirement. In the future, the approach will be enhanced to problems with more degrees of freedom per interface.

References

- Albers, A. and Otnad, J. (2008), “System based topology optimization as development tools for lightweight components in humanoid robots”, in *Humanoids 2008: 2008 8th IEEE-RAS International Conference on Humanoid Robots*; Daejeon, South Korea, 1 - 3 December 2008, 12/1/2008 - 12/3/2008, Daejeon, IEEE, Piscataway, NJ, pp. 674–680
- Beernaert, T.F. and Etman, L.F.P. (2019), “Multi-level Decomposed Systems Design: Converting a Requirement Specification into an Optimization Problem”, *Proceedings of the Design Society: International Conference on Engineering Design*, Vol. 1 No. 1, pp. 3691–3700. <https://doi.org/10.1017/dsi.2019.376>
- Bendsøe, M.P. and Sigmund, O. (2004), *Topology Optimization: Theory, Methods, and Applications*, Second Edition, Corrected Printing, Springer, Berlin, Heidelberg. <https://doi.org/10.1007/978-3-662-05086-6>
- Corinna Cortes and Vladimir Vapnik (1995), “Support-vector networks”, *Machine Learning*, Vol. 20 No. 3, pp. 273–297. <https://doi.org/10.1007/BF00994018>
- Eberhart, R. and Kennedy, J. (1995), “A new optimizer using particle swarm theory”, in *Proceedings of the Sixth International Symposium on Micro Machine and Human Science: Nagoya Municipal Industrial Research Institute*, October 4 - 6, 1995, 4-6 Oct. 1995, Nagoya, Japan, IEEE Service Center, Piscataway, NJ, pp. 39–43
- Eckert, C. and Clarkson, J. (2005), “The reality of design”, in Clarkson, J. and Eckert, C. (Eds.), *Design process improvement: A review of current practice*, SpringerLink Bücher, Springer London, London, pp. 1–29. https://doi.org/10.1007/978-1-84628-061-0_1
- Ertekin, S., Huang, J., Bottou, L. and Giles, L. (2007), “Learning on the border”, in Silva, M.J., Falcão, A.O., Laender, A.A.F., Baeza-Yates, R., McGuinness, D.L., Olstad, B. and Olsen, Ø.H. (Eds.), *Proceedings of the 2007 ACM Conference on Information and Knowledge Management*, 11/6/2007 - 11/10/2007, Lisbon, Portugal, ACM, New York, NY, p. 127
- Forsberg, K. and Mooz, H. (1991), “The Relationship of System Engineering to the Project Cycle”, *INCOSE International Symposium*, Vol. 1 No. 1, pp. 57–65. <https://doi.org/10.1002/j.2334-5837.1991.tb01484.x>
- Guyan, R.J. (1965), “Reduction of stiffness and mass matrices”, *AIAA Journal*, Vol. 3 No. 2, p. 380. <https://doi.org/10.2514/3.2874>
- Kim, B.J., Yun, D.K., Lee, S.H. and Jang, G.-W. (2016), “Topology optimization of industrial robots for system-level stiffness maximization by using part-level metamodelling”, *Structural and Multidisciplinary Optimization*, Vol. 54 No. 4, pp. 1061–1071. <https://doi.org/10.1007/s00158-016-1446-x>
- Kremer, J., Steenstrup Pedersen, K. and Igel, C. (2014), “Active learning with support vector machines”, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Vol. 4 No. 4, pp. 313–326. <https://doi.org/10.1002/widm.1132>
- Krischer, L., Sureshbabu, A.V. and Zimmermann, M. (2020), “Modular Topology Optimization of a Humanoid Arm”, 2020 3rd International Conference on Control and Robots (ICCR). <https://doi.org/10.1109/ICCR51572.2020.9344316>
- Krischer, L. and Zimmermann, M. (2021), “Decomposition and optimization of linear structures using meta models”, *Structural and Multidisciplinary Optimization*. <https://doi.org/10.1007/s00158-021-02993-1>
- Liu, G.-R. and Quek, S.S. (2013), *The finite element method: A practical course*, Second, Butterworth-Heinemann, Oxford, UK
- Martins, J.R.R.A. and Lambe, A.B. (2013), “Multidisciplinary Design Optimization. A Survey of Architectures”, *AIAA Journal*, Vol. 51 No. 9, pp. 2049–2075. <https://doi.org/10.2514/1.J051895>
- Ramu, M., Prabhu Raja, V. and Thyla, P.R. (2013), “Establishment of structural similitude for elastic models and validation of scaling laws”, *KSCE Journal of Civil Engineering*, Vol. 17 No. 1, pp. 139–144. <https://doi.org/10.1007/s12205-013-1216-x>
- Svanberg, K. (1987), “The method of moving asymptotes—a new method for structural optimization”, *International Journal for Numerical Methods in Engineering*, Vol. 24 No. 2, pp. 359–373. <https://doi.org/10.1002/nme.1620240207>
- Wang, X., Zhang, D., Zhao, C., Zhang, P., Zhang, Y. and Cai, Y. (2019), “Optimal design of lightweight serial robots by integrating topology optimization and parametric system optimization”, *Mechanism and Machine Theory*, Vol. 132, pp. 48–65. <https://doi.org/10.1016/j.mechmachtheory.2018.10.015>
- Zimmermann, M. and Hoessle, J.E. von (2013), “Computing solution spaces for robust design”, *International Journal for Numerical Methods in Engineering*, Vol. 94 No. 3, pp. 290–307. <https://doi.org/10.1002/nme.4450>
- Zimmermann, M., Königs, S., Niemeyer, C., Fender, J., Zeherbauer, C., Vitale, R. and Wahle, M. (2017), “On the design of large systems subject to uncertainty”, *Journal of Engineering Design*, Vol. 28 No. 4, pp. 233–254. <https://doi.org/10.1080/09544828.2017.1303664>