Cooperative multi-objective topology optimization using clustering and metamodelling

Nivesh Dommaraju, Mariusz Bujny, Stefan Menzel, Markus Olhofer, Fabian Duddeck

2022

Preprint:

This is an accepted article published in IEEE 2022 Congress on Evolutionary Computation. The final authenticated version is available online at:
https://doi.org/[DOI not available]

Cooperative Multi-objective Topology Optimization Using Clustering and Metamodeling

1st Nivesh Dommaraju
Technical University of Munich
Munich, Germany
nivesh.dommaraju@tum.de

2nd Mariusz Bujny
Honda Research Institute Europe
Offenbach, Germany
mariusz.bujny@honda-ri.de

3rd Stefan Menzel
Honda Research Institute Europe
Offenbach, Germany
stefan.menzel@honda-ri.de

4th Markus Olhofer
Honda Research Institute Europe
Offenbach, Germany
markus.olhofer@honda-ri.de

5th Fabian Duddeck
Technical University of Munich
Munich, Germany
duddeck@tum.de

***Abstract*—Topology optimization optimizes material layout in a design space for a given objective, such as crash energy absorption, and a set of boundary conditions. In industrial applications, multi-objective topology optimization requires expensive simulations to evaluate the objectives and generate multiple Pareto-optimal solutions. So, it is more economical to identify preferred regions on the Pareto front and generate only the desired solutions. Clustering methods, a widely used subclass of machine learning methods, provide an unsupervised approach to summarize the dataset, which eases the identification of the preferred set of designs. However, generating solutions similar to the preferred designs based on different metrics is a challenging task. In this paper, we present an interactive method to generate designs similar to a preferred set using one of the state-of-the-art weighted-sum approaches called scaled energy weighting - hybrid cellular automata (SEW-HCA). To avoid unnecessary computations, metamodels are used to predict the desired weight vectors needed by SEW-HCA. We evaluate an application of our method for cooperative topology optimization using a cantilever multi-load-case problem and a crashworthiness optimization problem. Using the proposed method, we could successfully generate designs that are similar to preferred solutions based on geometry and performance. We believe that this is a crucial component that will improve the usefulness of multi-objective topology optimization in real-world applications.**

***Index Terms*—multi-objective optimization, topology optimization, similarity measures, data mining, geometric processing**

I. Introduction

Topology optimization (TO) [1] optimizes the material layout in a design space for a given set of objectives such as structural stiffness or crash energy absorption, given a set of boundary conditions such as loads and supports. In real-world applications, TO uses multiple expensive simulations to iteratively find solutions in a high-dimensional design space. Multi-objective topology optimization (MTO) is even more challenging due to additional, usually conflicting, objectives. Furthermore, methods are needed to analyze the numerous Pareto optima and identify desirable solutions.

Nivesh Dommaraju gratefully acknowledges the financial support from Honda Research Institute Europe (HRI-EU).

Multi-objective evolutionary algorithms (MOEAs) such as NSGA-III are successful in identifying diverse as well as preferred solutions [2]. Since the search space in TO problems is in general high-dimensional, it is difficult or not feasible to use MOEAs. Methods such as the evolutionary level set method [3], [4] alleviate this problem by using feature mapping to reduce the dimensionality of the problem. However, MOEAs still tend to be computationally intensive since they require a high number of fitness calculations with each evaluation requiring one or more expensive simulations.

In contrast, an MTO method using a weighted-sum approach (w-MTO) can handle the high-dimensional decision space by incorporating computationally-efficient TO approaches. For example, weighted-sum methods can be used with two of the start-of-the-art TO algorithms: gradient-based SIMP (solid isotropic material with penalization) [1] and heuristic-based HCA (hybrid cellular automata) [5] methods. In the weighted-sum approach, each objective is assigned a weight and an optimal solution is computed. However, it is difficult to choose the weights such that the resulting structure meets all the design requirements, which are often difficult to formulate mathematically [6]. Furthermore, given the high cost of MTO, it is more efficient to progressively incorporate the user preference to generate only the desired solutions, i.e., using the *interactive* method for multi-objective optimization [7]. So, we investigate if machine learning techniques such as clustering and regression methods can be used to choose the weights and generate desired solutions in collaboration with an engineer.

An unsupervised machine learning method to analyze datasets is clustering which groups designs based on a similarity criterion. Since it is difficult to analyze each optimal solution, designs can be clustered based on performance [8] or geometrical structure [9]. For each cluster of solutions, only the representative design may be analyzed, reducing the effort in the design process considerably [10]. In the literature, preferred regions are defined using reference directions, points, or other supervised methods [2]. In contrast, in this paper, we use cluster analysis to identify a reference set of designs, which is viable even for a large number of objectives. Moreover, to minimize the expensive simulations, metamodels—regressor and classifier models—are used in an evolutionary optimization process to predict the weight-vectors needed by w-MTO to generate the new solutions in the preferred region.

In this paper, we demonstrate the usefulness of our method in design optimization using two test cases: a cantilever multi-load-case and a crashworthiness optimization problem. We use SEW-HCA (scaled energy weighting - hybrid cellular automata) [11] since it is fast and can handle crash-related objectives. However, any other MTO using the weighted-sum approach can be integrated with our proposed approach. For each test problem, we group solutions by similarity in performance or geometry and then generate more solutions in each cluster. We could evaluate our approach successfully by measuring the accuracy of metamodels and the cluster belongingness of the new solutions.

In the following, we describe an interactive MTO based on clustering along with an illustration in Section II. An evaluation method to measure the similarity of new solutions to the preferred cluster is presented in Section III. We then present the test problems in Section IV, followed by a conclusion in Section V.

II. Interactive set-based multi-objective optimization (iSMO)

Real-world multi-objective topology optimization (MTO) requires expensive computations. For such problems, researchers recommend the so-called *interactive* approach to generate solutions only in the preferred region [7]. While data analysis methods such as clustering can identify preferred solutions (Section II-A), it is challenging to generate only the desired solutions with a minimal number of expensive computations. In this section, we present an interactive framework using set-based multi-objective optimization (iSMO), which identifies a preferred set of solutions using cluster analysis and generates more such solutions using an evolutionary algorithm (EA) with a metamodel of the MTO.

Given a vector $\mathbf{w}$ of weights for objectives, w-MTO methods such as SEW-HCA can efficiently compute Pareto-optimal solutions. SEW-HCA discretizes the design domain into n elements whose normalized density $\rho \in [0, 1]$ needs to be determined by the the optimization process. Each design $\mathbf{d}_i$ can be then represented by a vector of relative densities of the elements, $[\rho]_{e=1}^{n}$. An *interactive* w-MTO method needs to find the set of weight-vectors $W = \{\mathbf{w}_i\}_{i=1}^{N}$, which results in a diverse set of solutions $\mathbf{D} = \{\mathbf{d}_i(\mathbf{w}_i)\}_{i=1}^{N}$ in the preferred region.

Inspired by MOEAs, we propose to find W using EAs to ensure diversity of solutions while constraining them to the preferred set $\mathbb{S}$. The constraint function evaluates if a given $\mathbf{w}$ results in a solution $\mathbf{d} \in \mathbb{S}$. To avoid expensive TO runs in iSMO, we propose a novel approach using metamodels to predict the objectives (Section II-B) as well as the cluster labels of the TO results (Section II-C) from $\mathbf{w}$. Given the metamodels, an evolutionary optimization method can be inexpensively used to find the desired weight-vectors (Section II-D). We illustrate the complete process using a simple quadratic MOP (multi-objective optimization problem) at the end of this section (Section II-E).

A. Identification of Preferred Regions using Clustering

Several data mining approaches exist to aid the designer in this process of identifying interesting solutions: descriptive statistics, manifold learning techniques, box plots, parallel coordinate plots, and clustering methods can help to analyze the Pareto optima [12]. Clustering methods are particularly useful since they summarize datasets without supervision and can identify a cluster of interesting designs.

In the literature, clustering is used to analyze TO results and identify design groups with similar performance [8] or geometrical structure [9]. By analyzing the cluster properties, one or more clusters can be chosen as the preferred set of designs. This considerably reduces the effort of analyzing the dataset of usually very complex, topologically-optimized designs. For example, we can select a cluster by analyzing a representative solution in it, e.g, the medoid of a cluster [10], [13], which is defined as the design with the least average distance to the other designs within the cluster. A metric, chosen according to the application, measures the distance/dissimilarity between any two designs.

Later in the results (Section IV), we analyze the effect of different clusterings on our method iSMO. For this purpose, we use preferred sets from performance and geometric clustering separately, which are explained below. More sophisticated clusterings of design concepts are possible which consider multiple metrics concurrently [14].

a) Performance clustering: Designs can be clustered based on their objective values. This implicitly considers the Euclidean distance in objective space as the performance metric.

b) Geometrical clustering: 3D object classification, i.e., identifying objects that are geometrically or structurally similar, is a challenging research field [15], [16]. In this paper, we use the method outlined by Dommaraju et al. [13] with the autoencoder (AE) neural-network proposed by Achlioptas et al. [15] to extract *geometric features* ($\mathbf{g}$) that are useful in clustering similar structures. Each TO design is converted to a surface mesh and then points are sampled uniformly on the mesh to obtain a point cloud representation of the design. Given a metric called chamfer distance (CD) [15] which measures the dissimilarity between any two point clouds, AE learns to extract from each design a so-called latent code ($\mathbf{g}$), which can then be used to group similar geometries with ease using a clustering algorithm such as k-means [17].

In the datasets used in Section IV, the resultant TO designs comprise a set of elements whose relative densities range from 0 to 1. So, we use the marching cubes algorithm [18] with a density threshold of 0.1 to convert each TO design to a surface mesh which is then converted to a 2048-dimensional point cloud. An AE model is trained with the initial set of designs to extract a 128-dimensional vector of geometric features (**g**), which is used by k-means to find the geometrical clusters in the design set.

Although we use clustering in this paper, any other method that yields a set of preferred designs can be coupled with our proposed approach to generate more of such designs. Next, we show how metamodels can be used to predict the objectives and cluster labels.

B. Prediction of Objectives Using a Regressor

To offset the high cost of MTO, we build a regressor model ($P_{\mathbf{f}}(\mathbf{w})$) to predict the objective values (**f**) for a given weight vector (**w**) instead of performing the w-MTO with **w** as input. While the regression model needs to be tailored to a given problem [19], the following issues are important while training a metamodel for MTO:

- Dimension of output: If the dimension m of **f** is greater than 1, we need a multi-output regression model.
- Redundant weights: The weights for objectives can be scaled by the same number and the resultant design will not change. So, we normalize the weights such that $\sum_{i=1}^{m} w_i = 1$ and remove a weight to yield the input, $[w_i]_{i=1}^{m-1}$, for training the metamodel.
- Limited data: Since MTO is expensive, it is common to have a small number of designs. So, simpler models might cause less overfitting than the complex models such as neural-network regression models [20].
- Overfitting: A good metamodel should predict correct outputs for weights other than the training weights, i.e., avoid overfitting the data. For our datasets, Gaussian process regressor [21] overfits the least without any hyperparameter tuning unlike random forests, linear regressors, and support vector regressors [20].

C. Prediction of Clusters using a Classifier

Our approach requires a constraint function C to check if a new design **d** obtained using a weight vector **w** belongs to the preferred set $\mathbb{S}$, i.e.,

$$C(\mathbf{w}) = 0 \text{ iff } \mathbf{d}(\mathbf{w}) \in \mathbb{S}. \tag{1}$$

As discussed in Section II-A, we select a preferred cluster whose solutions are representative of the preferred set $\mathbb{S}$. We then evaluate C using a classifier model to predict the cluster label p from **w**:

$$C(\mathbf{w}) = |p(\mathbf{w}) - p_{\text{ref}}|, \tag{2}$$

where p_{ref} is the preferred cluster label and $p(\mathbf{w})$ is the predicted cluster label. In our experiments, we use the k-nearest neighbor classifier with $k = 1$, i.e., the cluster label for a given **w** is the label of the nearest cluster in the training data. For noisy datasets, better classifiers may be needed.

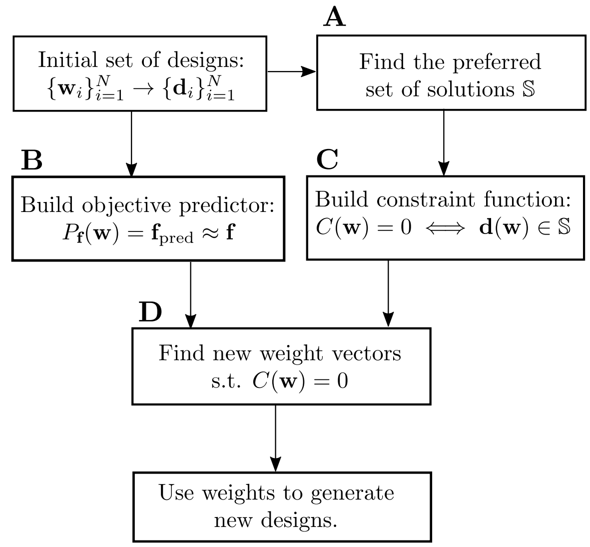


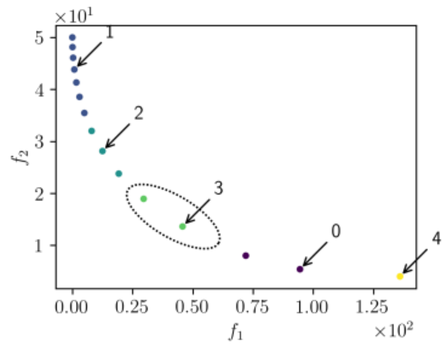
Fig. 1. An iteration of iSMO. Given a preferred cluster (**A**), we use a metamodel to predict objectives from weight vectors (**B**) and a constraint function determines if a weight vector yields a preferred solution (**C**). Using EA, we find the weight vectors that will result in the desired designs (**D**).

D. Identification of Desired Weight-vectors

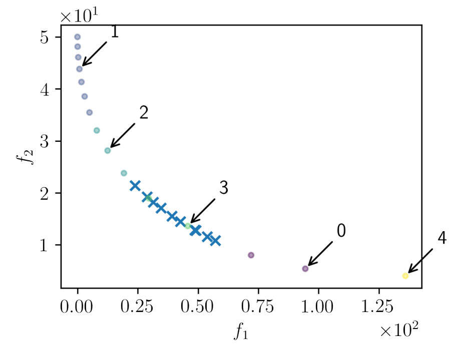
As discussed previously, the $P_{\mathbf{f}}(\mathbf{w})$ can estimate the objectives without requiring expensive simulations and $C(\mathbf{w})$ can be used to restrict solutions to a selected cluster. We now describe our iSMO framework to find the new weights that result in new solutions in the preferred set $\mathbb{S}$.

iSMO follows the general EA framework [2] and uses selection, crossover, and mutation operators to obtain a new generation of solutions that satisfy the cluster constraint C. iSMO uses simulated binary crossover and polynomial mutation [22] but other variations are possible [2]. The input variables for iSMO are **w** and the objectives are predicted using a metamodel $P_{\mathbf{f}}$ of SEW-HCA. Since SEW-HCA is heuristic, the data used to train the $P_{\mathbf{f}}$ may have dominated solutions. Furthermore, since $P_{\mathbf{f}}$ is not perfect, it may generate dominated solutions. If SEW-HCA as well as its metamodel are reasonably accurate, $P_{\mathbf{f}}$ predicts objectives that are still close to the Pareto front. So, we chose not to use the dominance ranking proposed in NSGA-II and only use the crowding distance in the selection operator to ensure diversity of solutions [22]. Otherwise, many useful solutions close to the Pareto front may be ignored, which deteriorates the diversity of new solutions. The overall time and space complexity of iSMO are the same as NSGA-II without the dominance ranking step [22].

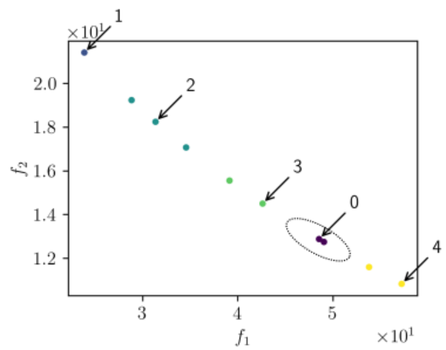
Figure 1 outlines the complete method. Since the metamodels are used to predict the objectives as well as the cluster labels, no additional TO runs with simulations are required to find the desired weight-vectors, which are then used in the final step to generate actual designs. In the next section, we illustrate our method using a simple quadratic MOP.



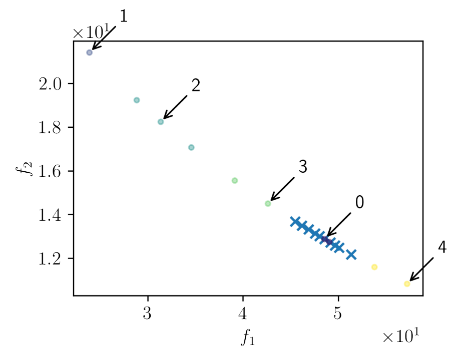
(a) Initial Pareto front with clusters based on objective values. Each solution is colored as per its cluster and a preferred cluster is chosen (dotted ellipse).



(b) New solutions (cross-marked) are generated in the preferred cluster.



(c) Partial Pareto front with only the new solutions from subfigure b. The solutions are clustered again and a new preferred cluster is chosen (dotted ellipse).



(d) New solutions (cross-marked) are again created in the preferred cluster from subfigure c.

Fig. 2. Illustration of iSMO. The axes labels f_1, f_2 represent the two objectives of BNH problem. In each row, Pareto optima are clustered and new solutions are generated in a selected cluster using iSMO.

E. Illustration of iSMO

To illustrate our method, we consider the BNH [23] function with two decision variables, two objectives $[f_1, f_2]$ and no constraints. To mimic a w-MTO, we use the weighted-sum method to aggregate the objectives of BNH.

For a set of weights $\mathbf{w} = [w_1, w_2]$, the resultant objective $f = \mathbf{w} \cdot \mathbf{f} = w_1 f_1 + w_2 f_2$ is minimized using the differential evolution algorithm [24], where $w_1 \geq 0$ and $w_2 = 1 - w_1 \geq 0$. To generate an initial set of solutions, $\mathbf{W} = \{\mathbf{w}_i\}_{i=1}^N$ are uniformly sampled using the Das-Dennis approach [25] to obtain the corresponding set of objectives $\mathbf{F} = \{\mathbf{f}_i\}_{i=1}^N$ (Figure 2a). Since the objectives are of different scales, uniform sampling of weight vectors gives a slightly non-uniform Pareto front. A Gaussian regressor model for the MOO is built using w_1 as input. Using iSMO, new solutions are progressively found in the preferred clusters. Figure 2 shows two iterations of (1) choosing a preferred cluster, and (2) generating a dense set of solutions for each of the clusters.

III. Evaluation method

The new solutions generated by iSMO should belong to the preferred set of designs $\mathbb{S}$ compared to other designs in the initial Pareto-optimal solutions $\mathbb{I}$. The constraint function ensures that the new solutions belong to the preferred cluster. But this can be verified objectively using the silhouette score [26] to measure the cluster belongingness.

Silhouette sample score s_i measures the belongingness of i-th solution to a given set $\mathbb{S}$ relative to other solutions and is defined as follows:

$$s_i = \frac{b_i - a_i}{\max(b_i, a_i)}, \tag{3}$$

where a_i is the closest distance to other solutions in $\mathbb{S}$, b_i is the closest distance to the other solutions in $\mathbb{I}$ excluding $\mathbb{S}$, i.e., $\mathbb{I} - \mathbb{S}$. The sample score ranges from $[-1, 1]$. In this calculation, the metric used for clustering should be used to measure the distances (Section II-A). The average silhouette score for a set of new solutions indicates the quality of belongingness to the preferred set $\mathbb{S}$. While a high score ($s \approx 1$) is preferred, the silhouette score is expected to be low for contiguous data, where the clusters are not well separated. In particular, the score is low for samples on the boundary of clusters.

IV. Topology optimization using iSMO

In this section, we provide results using exemplary MTO problems. For each example, we generate an initial set of solutions which are then clustered based on geometry and performance separately. We then use our proposed approach, iSMO, to find new solutions in each of the clusters. Although our method is supposed to generate solutions for a single chosen cluster, we show here solutions in all the clusters to evaluate iSMO with different preferred sets.

A. Test Case 1: Cantilever Beam with Two Static Loads

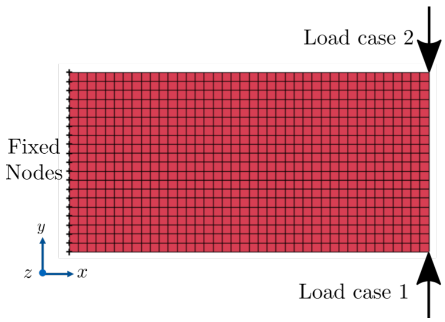


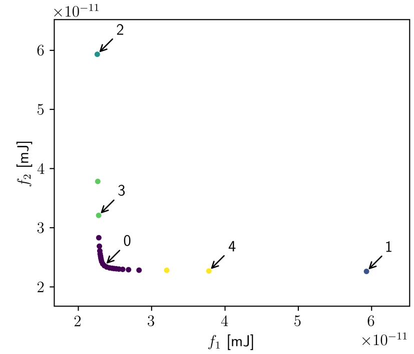
Fig. 3. A cantilever beam is optimized to support two static loads acting independently.

A cantilever beam is optimized to support two static loads, each with a magnitude of 0.2 N (Figure 3). For each static load $i \in \{1, 2\}$, we minimize the structural compliance f_i—the inverse of stiffness—which is measured by the total internal strain energy stored. Lower stored energy indicates lower compliance of the structure for the applied load. Each static load case is simulated separately using the implicit solver of the commercial software LS-DYNA.

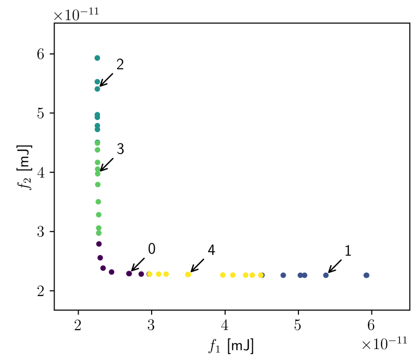
The design space of dimensions, 400 mm $\times$ 200 mm $\times$ 10 mm, is discretized into $40 \times 20 \times 1 = 800$ elements. Bilinear elasto-plastic model (MAT_024 in LS-DYNA) of an aluminium material is used with the following properties: maximum mass density $\rho_{\max} = 2.7 \times 10^3$ kg/m^3, Young's modulus $E = 70$ GPa, Poisson's ratio $\nu = 0.33$, yield strength $\sigma_Y = 117$ MPa, and hardening modulus $E_{\tan} = 49$ GPa. For a given **w**, we use an SEW-HCA run with a maximum of 25 iterations, an allowed volume fraction of 0.4, a move limit of 0.1, and a penalization factor of 3.

A Gaussian process regressor was trained using initial solutions to predict the objectives from the weight-vector. The k-fold cross-validation method [20] with $k = 5$ yields a score $R^2 = 0.77 \pm 0.08$. Since the size of data is small, the scores can be quite low depending on the test split. However, test accuracy measured on the new solutions generated by iSMO is high: $R^2 = 0.97 \pm 0.02$. A k-nearest neighbor classifier with $k = 1$ is built to define the constraint function, i.e., a new solution is labeled according to the cluster label of its closest solution in the initial dataset.

a) Performance clusters: Using k-means method, solutions are grouped based on performance by using objective values as the features for clustering (Figure 4a). Using iSMO, we generated new solutions in each of the performance clusters. Figure 4b shows that the new solutions belong to their selected clusters. The average silhouette score of the new solutions is 0.61, which is reasonable since the clusters are contiguous.



(a) Initial Pareto front



(b) Pareto front containing exclusively the new solutions in each cluster of (a).

Fig. 4. Using iSMO, new solutions are generated for each performance cluster in test case 1 (Figure 3).

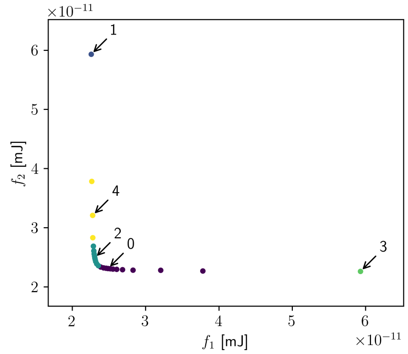
b) Geometric clusters: The Pareto-optimal designs are clustered based on the geometric features (dim=128) obtained using an autoencoder (Section II-A). Using iSMO, we generated new solutions in each of the clusters (Figure 5b) with an average silhouette score of 0.53. Figure 6 compares the structure of new solutions in the geometric clusters with the initial solutions.

Comparing the new solutions to the corresponding clusters, one can see that the new designs are reasonably spread even for clusters with 1 or 2 initial designs. Furthermore, the solutions spread up to the cluster boundaries.

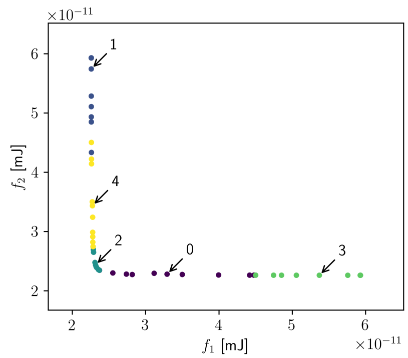
B. Test Case 2: Simply Supported Beam with a Crash and a Static load

A simply supported beam is optimized using MTO with two objectives: minimizing compliance f_1 for a static load and maximizing energy absorbed f_2 for a crash load (Figure 7). The static and crash load cases are simulated using the implicit and explicit solvers in LS-DYNA respectively. The beam is fixed at two locations at the bottom, as shown in Figure 7. The crash load is applied using a rigid hollow cylinder with a prescribed displacement of 100 mm in 0.1 s. The static load of 10^4 N is used.

The design space of dimensions, 600 mm$\times 50$ mm$\times 50$ mm, is discretized into $120 \times 10 \times 10 = 12000$ elements. The material properties are as in the test case 1 (Section IV-A).



(a) Initial Pareto front



(b) Pareto front containing exclusively the new solutions in each cluster of (a).

Fig. 5. Using iSMO, new solutions are generated for each geometric cluster in test case 1 (Figure 3).

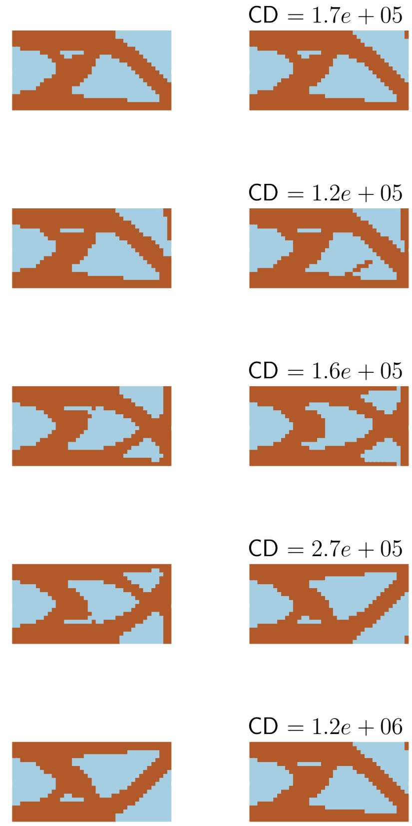


Fig. 6. A comparison of initial and new solutions in geometric clusters. Each design comprises of elements which are classified into high density elements (brown) if $\rho \geq 0.1\ \rho_{\max}$ or low density elements (light-blue), otherwise. In each row, we show the representative design in a cluster i of initial Pareto front (Figure 5a), followed by the most dissimilar new solution for cluster i (Figure 5b). CD between the designs is shown above the new design.

For a given $\mathbf{w}$, we use SEW-HCA with a maximum of 25 iterations, an allowed volume fraction of 0.4, a move limit of 0.05, and a penalization factor of 3.

A Gaussian process regressor was trained using initial solutions. The k-fold cross validation method with $k = 5$ yields a score $R^2 = 0.86 \pm 0.04$. Test accuracy measured on the new solutions is also high: $R^2 = 0.84 \pm 0.05$. A k-nearest neighbor classifier with $k = 1$ is once again built to define the constraint function.

a) Performance clusters: Using the k-means method, initial solutions are clustered based on performance by using objective values as the features (Figure 8a). Due to the heuristic nature of SEW-HCA, some of the solutions are not Pareto-optimal but are near the Pareto front. Using iSMO, we generated new solutions in each of the clusters. Figure 8b shows the new solutions which belong to their selected clusters in the objective space. The average silhouette score of the new solutions is 0.69.

b) Geometric clusters: The initial solutions are clustered based on the geometric features (dim=128) obtained using an autoencoder (Figure 9a). Using iSMO, we generated new solutions (Figure 9b) in each of geometric clusters. Figure 9b shows that the new solutions seem to belong to their selected clusters as seen in the objective space, which is expected for this dataset where similar geometries tend to have similar performance. The average silhouette score of the new solutions is 0.61, which is measured using CD as the metric. Figure 10 compares the structure of initial and final solutions.

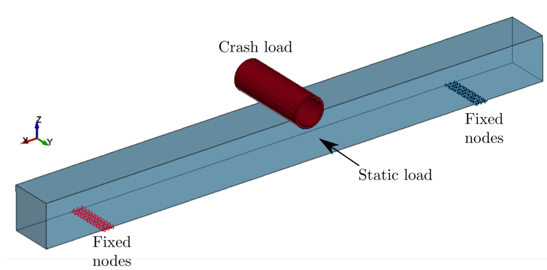
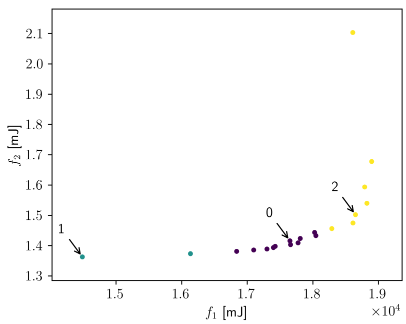


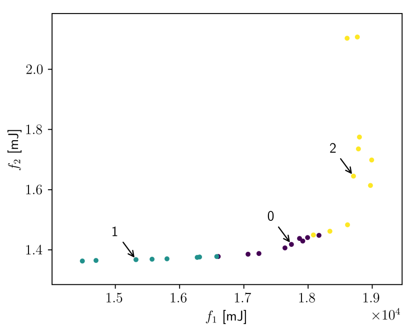
Fig. 7. A simply supported beam is optimized for a crash load from a cylinder (top) and a static load (right face).

Discussion

Since the two test cases discussed in this section consider only two objectives, our proposed method can be qualitatively analyzed in addition to evaluating quantitatively using silhou-

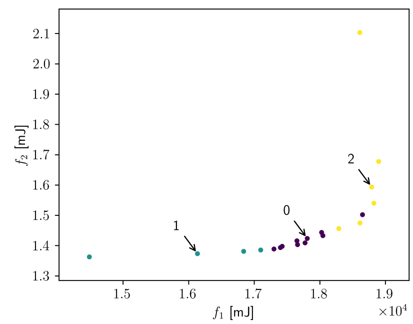


(a) Initial set of solutions colored according to the performance cluster.

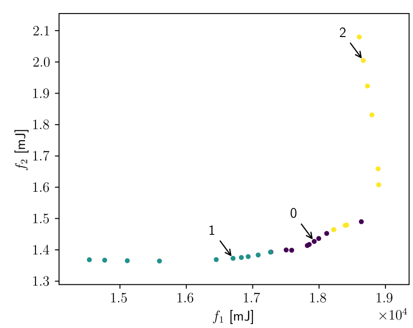


(b) Pareto front containing exclusively the new solutions in each cluster of (a).

Fig. 8. Using iSMO, new solutions are generated for each performance cluster in test case 2 (Figure 7).



(a) Initial set of solutions colored according to the geometric cluster.



(b) Pareto front containing exclusively the new solutions in each cluster of (a).

Fig. 9. Using iSMO, new solutions are generated for each geometric cluster in test case 2 (Figure 7).

ette scores. Since the Pareto front is two-dimensional, it is easy to see that the new solutions belong to their preferred performance clusters (Figures 4 and 8). Given that the clusters are not separated well, the average silhouette scores, ranging from 0.5 to 0.6, indicate that the new solutions are within their target clusters. The same argument is true for geometric clusters as well since the solutions along the Pareto front gradually change in geometry for these datasets. The new solutions should be closer to the preferred geometric cluster as seen in the objective space as well, which is indeed the case as seen in Figures 5 and 9.

V. Conclusion

In this paper, we addressed some of the challenges faced in structural design optimization. We recommended the use of a weighted-sum approach for multi-objective topology optimization (w-MTO) as in SEW-HCA (scaled energy weighting - hybrid cellular automata) since they can efficiently find the solutions in the high-dimensional decision space. However, such methods need to be rerun with different weight-vectors (**w**) to yield multiple Pareto optima and each run needs expensive simulations. So, we proposed an efficient approach called iSMO (interactive set-based multi-objective optimization) that iteratively generates the desired solutions.

In iSMO, first, we generate an initial set of solutions using an appropriate weight sampling method for a given w-MTO. Then, we analyze the solutions using clustering based on a metric to identify a preferred set of solutions (Section II-A). Using the initial solutions and clustering data, iSMO predicts the weights required to generate more solutions similar to a preferred cluster (Sections II-B, II-C, II-D). Since iSMO uses metamodels, it does not require expensive w-MTO runs. Finally, we evaluate our method by measuring the cluster belongingness of the new solutions using silhouette scores (Section III). Although we use iSMO as an interactive method for an MTO based on the weighted-sum approach, it can be adapted to MTOs or other expensive MOPs that use other scalarization functions or reference directions.

In our experiments with two test problems (Section IV), SEW-HCA generated reasonably diverse initial solutions from uniformly sampled weights. The initial solutions are clustered based on performance and geometry separately. For each cluster, iSMO could successfully generate new, diverse solutions. The cluster belongingness of the new solutions, measured using the silhouette score, is reasonably high, given that the input clusters are close to each other. While the new solutions seem to be well-spread within their cluster, we would like to evaluate this more quantitatively in the future. In this paper, we used two-objective MTO problems since they are more intuitive to understand. We believe that the underlying methods of iSMO are general enough to handle many-objective TO problems. It would be interesting to use iSMO with other TO algorithms and with large-scale industrial problems, e.g., for generating specific geometrical concepts in open datasets such

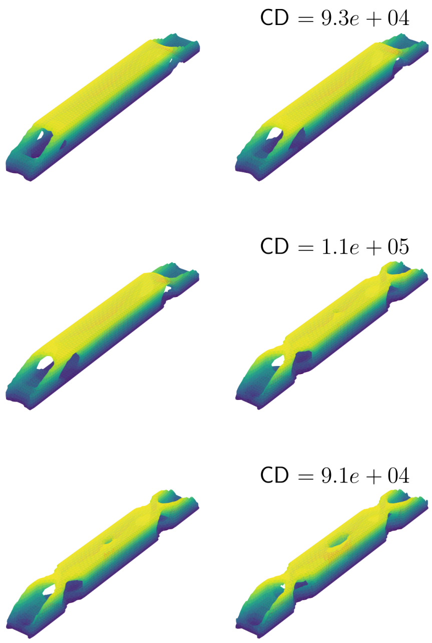


Fig. 10. A comparison of initial and new solutions in geometric clusters. Each design is represented by the surface mesh enclosing only the high density elements $\rho \geq 0.1\ \rho_{\max}$. In each row, we show the representative design in a cluster i of initial solutions (Figure 9a), followed by the most dissimilar new solution for cluster i (Figure 9b). CD between the designs is shown above the new design.

as the CarHoods10k dataset [27].

Multi-objective optimization in the context of topology optimization is a challenging field of research. New methods are needed to effectively generate new designs while minimizing the expensive simulations. In this paper, we leveraged machine learning methods to support the optimization process. At the core, we generated new solutions similar to a set of initial solutions. We believe this approach of progressively recommending new solutions to the designers will enhance the use of topology optimization in the industry and product design process.

References

[1] M. P. Bendsøe and O. Sigmund, *Topology optimization*. Springer Berlin Heidelberg, 2004.

[2] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 32–49, 2011.

[3] M. Bujny, N. Aulig, M. Olhofer, and F. Duddeck, "Identification of optimal topologies for crashworthiness with the evolutionary level set method," *International Journal of Crashworthiness*, vol. 23, no. 4, pp. 395–416, 2017.

[4] M. Bujny, M. Olhofer, N. Aulig, and F. Duddeck, "Topology Optimization of 3D-printed joints under crash loads using Evolutionary Algorithms," *Structural and Multidisciplinary Optimization*, vol. 64, no. 6, pp. 4181–4206, 2021.

[5] A. Tovar, N. M. Patel, G. L. Niebur, M. Sen, and J. E. Renaud, "Topology optimization using a hybrid cellular automation method with local control rules," *Journal of Mechanical Design, Transactions of the ASME*, vol. 128, no. 6, pp. 1205–1216, 2006.

[6] M. S. Yousaf, M. Bujny, N. Zurbrugg, D. Detwiler, and F. Duddeck, "Similarity control in topology optimization under static and crash loading scenarios," *Engineering Optimization*, vol. 53, no. 9, pp. 1523–1538, 2021.

[7] K. Miettinen, J. Hakanen, and D. Podkopaev, "Interactive nonlinear multiobjective optimization methods," in *Multiple Criteria Decision Analysis*. Springer New York, 2016, pp. 927–976.

[8] Y. Sato, K. Izui, T. Yamada, and S. Nishiwaki, "Data mining based on clustering and association rule analysis for knowledge discovery in multiobjective topology optimization," *Expert Systems with Applications*, vol. 119, pp. 247–261, 2019.

[9] N. Dommaraju, M. Bujny, S. Menzel, M. Olhofer, and F. Duddeck, "Simultaneous Exploration of Geometric Features and Performance in Design Optimization," in *16th International LS-DYNA Conference*, 2020, p. 12.

[10] A. Hagg, A. Asteroth, and T. Bäck, "Prototype Discovery Using Quality-Diversity," in *Parallel Problem Solving from Nature – PPSN XV*. Springer International Publishing, 2018, pp. 500–511.

[11] N. Aulig, E. Nutwell, S. Menzel, and D. Detwiler, "Preference-based topology optimization for vehicle concept design with concurrent static and crash load cases," *Structural and Multidisciplinary Optimization*, vol. 57, no. 1, pp. 251–266, 2018.

[12] S. Bandaru, A. H. C. Ng, and K. Deb, "Data mining methods for knowledge discovery in multi-objective optimization: Part A - Survey," *Expert Systems with Applications*, vol. 70, pp. 139–159, 2017.

[13] N. Dommaraju, M. Bujny, S. Menzel, M. Olhofer, and F. Duddeck, "Identifying Topological Prototypes using Deep Point Cloud Autoencoder Networks," in *2019 International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2019, pp. 761–768.

[14] F. Lanfermann, S. Schmitt, and S. Menzel, "An Effective Measure to Identify Meaningful Concepts in Engineering Design optimization," in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2020.

[15] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning Representations and Generative Models for 3D Point Clouds," in *Proceedings of the 35th International Conference on Machine Learning*, vol. 80. PMLR, 2018, pp. 40–49.

[16] Z.-X. Yang, L. Tang, K. Zhang, and P. K. Wong, "Multi-view CNN feature aggregation with ELM auto-encoder for 3d shape recognition," *Cognitive Computation*, vol. 10, no. 6, pp. 908–921, 2018.

[17] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*, vol. 07-09-Janu, 2007, pp. 1027–1035.

[18] T. Lewiner, H. Lopes, A. W. Vieira, and G. Tavares, "Efficient Implementation of Marching Cubes' Cases with Topological Guarantees," *Journal of Graphics Tools*, vol. 8, no. 2, pp. 1–15, 2003.

[19] L. Fahrmeir, T. Kneib, S. Lang, and B. Marx, *Regression*. Springer Berlin Heidelberg, 2013.

[20] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, 2019.

[21] C. E. Rasmussen and H. Nickisch, "Gaussian processes for machine learning (GPML) toolbox," *The Journal of Machine Learning Research*, vol. 11, pp. 3011–3015, 2010.

[22] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[23] T. B. To and B. Korn, "MOBES: A Multiobjective Evolution Strategy for Constrained Optimization Problems," in *The Third International Conference on GeneticAlgorithms (Mendel 97)*, vol. 25, 1997, p. 27.

[24] K. V. Price, "Differential evolution," in *Handbook of Optimization*. Springer Berlin Heidelberg, 2013, pp. 187–214.

[25] I. Das and J. E. Dennis, "Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems," *SIAM Journal on Optimization*, vol. 8, no. 3, pp. 631–657, 1998.

[26] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.

[27] P. Wollstadt, M. Bujny, S. Ramnath, J. J. Shah, D. Detwiler, and S. Menzel, "CarHoods10k: An industry-grade data set for representation learning and design optimization in engineering applications," *IEEE Transactions on Evolutionary Computation*, pp. 1–1, 2022.