Exploiting Linear Interpolation of Variational Autoencoders for Satisfying Preferences in Evolutionary Design Optimization

Sneha Saha
Honda Research Institute Europe GmbH
Offenbach, Germany
sneha.saha@honda-ri.de

Leandro L. Minku
School of Computer Science
University of Birmingham
Birmingham, UK
l.l.minku@cs.bham.ac.uk

Xin Yao
School of Computer Science
University of Birmingham, UK
Department of Computer Science and Engineering, SUSTech, China
x.yao@cs.bham.ac.uk

Bernhard Senhoff
Honda Research Institute Europe GmbH
Offenbach, Germany
bernhard.sendhoff@honda-ri.de

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

***Abstract*—In the early design phase of automotive digital development, one of the key challenges for the designer is to consider multiple-criteria like aerodynamics and structural efficiency besides aesthetic aspects for designing a car shape. In our research, we imagine a cooperative design system in the automotive domain which provides guidance to the designer for finding sets of design options or well-performing designs for preferred search areas. In the present paper, we focus on two perspectives for this multi-criteria decision-making problem: First, a scenario without prior information about design preferences, where the designer aims to explore the search space for a diverse set of design alternatives. Second, a scenario where the designer has a prior intuition on preferred solutions of interest. For both scenarios, we assume that historic 3D car shape data exists, which we can utilize to learn a compact low-dimensional design representation based on a variational autoencoder (VAE). In contrast to evolutionary multi-objective optimization approaches where starting populations are randomly initialized, we propose to seed the population more efficiently by exploiting the advantage of linear interpolation in the latent space of the VAE. In our experiments, we demonstrate that the multi-objective optimization converges faster and achieves a diverse set of solutions. For the second scenario, when specifying design preferences by weights, we improve on the weighted-sum method, which simplifies the multi-objective problem and propose a strategy for efficiently adapting the weights towards the preferred design solution.**

***Index Terms*—multi-objective optimization, preference formulation, autoencoders.**

I. Introduction

A current challenge in automotive digital development is the increasing number of multi-disciplinary requirements that have to be considered in the design process. These requirements comprise besides aesthetic design targets also engineering aspects such as aerodynamic and structural efficiency. Even though various tools from computer-aided design (CAD) and engineering (CAE) provide means for modifying shapes, the necessity to simultaneously reach various, potentially conflicting targets leads to considerable complexity in the design process. To support a human designer in handling this complexity, we imagine a cooperative design system (CDS), which can provide guidance for a faster design convergence or suggest potential alternatives to provide inspirations for design exploration.

In the real-world automotive design process, designers have to face multi-criteria decisions (MCDM) at various stages of the design generation. However, providing an efficient optimization method at an earlier design stage would allow higher design innovation potential in MCDM analysis. In this paper, we formulate a 3D multi-criteria design optimization task and consider two perspectives for multi-criteria decision analysis: First, where the designer has no prior information about their design preferences. Second, when the designer has knowledge about the formulation of design preferences.

Often the first perspective is useful in the initial automotive digital development phase, where the designers aim to explore several designs based on multi-criteria for ideation. To support the MCDM process, the problem can be formulated as a multi-objective design optimization to generate a diverse set of design solutions. The multi-objective optimization problem (MOP) formulation to optimize 3D designs requires several simplifications and trade-offs. Further, an efficient representation of the geometry, i.e., capturing essential design features with a possibly low number of parameters, is advantageous to reduce the optimization problem complexity. If prior data, e.g., a large database of design variations, is available, recent

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 766186 (ECOLE).

methods from the field of artificial intelligence (AI) such as deep learning-based (variational) autoencoders (V)AEs [1], [2] allow to design a low-dimensional representation for geometric models. Other than dimensionality reduction of the 3D design representation, autoencoders are capable of abstracting features in an unsupervised fashion, enabling different geometric parameterizations and the generation of novel designs. In the context of 3D design optimization tasks, a typical encoder-decoder structure of the AE can be used for basis transformation, from the design domain to a low-dimensional latent domain, where the operations on the geometry are performed. The most popular methods to solve the MOP are using Multi-objective evolutionary algorithms (MOEAs). These methods start from a set of initial solutions, so-called initial population, and try to improve this solution set iteratively throughout the optimization process. Thus, apart from using the low-dimensional latent domain as a decision variable for our MOP, we propose in this paper an efficient seeding mechanism for generating an initial population for the MOEA to help incorporating the knowledge of the optimization problem in the MOEA to pursue the objectives by exploring the latent space in the AE.

In the second perspective of MCDM analysis, finding the best design solution from an optimization task involves trade-offs between different objectives in MOP. Often experienced designers have prior design preference information that can be incorporated into the optimization algorithm to generate an optimal solution. The weighted-sum method (WSM) is one popular method for MCDM that relies on eliciting designer's preference through weighting the criteria. However, even if with full knowledge of the objectives and satisfactory selection of weights, the final solutions may not necessarily reflect the intended preferences that are supposedly incorporated in the weights [3]. Thus, there is a need for a strategy to adapt the prior weights in a way that the final solution fits the designer's preference criteria.

Our main contributions in this paper are: First, we formulate the 3D multi-objective design optimization problem from two perspectives, where the designer does and does not have prior information about their design preferences. Second, we propose a novel seeding mechanism which is able to initialize the initial population of MOEAs in a more effective manner, utilizing the advantage of latent space interpolations in an AE. We will provide experimental results from our 3D design optimization task formulation and show the effectiveness of our proposed method. Third, we propose a strategy that is able to automatically determine suitable values for the weighted coefficients in WSM, leading to the generation of designer preferred solutions in less function evaluations. Further, we provide a study to investigate how well different types of evolutionary algorithms can solve the problem of preference-based optimization of 3D car designs.

The remainder of the paper is organized as follows: In Section II, we provide a survey of literature related to multi-objective evolutionary algorithms and preference formulation in optimization tasks. Section III discusses the multi-objective problem formulation for modifying 3D shapes, followed by two novel approaches—first a mechanism for generating an initial population of the MOEA and second a strategy to adapt the weights in WSM to match the DM's preference in Section IV. Section V presents the experimental setups. The results of the experiments are discussed in Section VI. Section VII concludes the paper.

II. Prior Art

Industrial design is a complex engineering activity that involves multiple criteria. The design task can often be seen as an optimization problem in which the parameters of the structure describing the best quality design are sought.

A. (Variational) Autoencoders as 3D Shape Representation

Engineering design data are normally in 3D and are of higher complexity. Here, common 3D data representations include voxel, point clouds, or polygon meshes [4]. For our application, we focus on 3D shapes sampled as surface point clouds, as they provide high flexibility and better memory-efficiency, which allows representing finer geometric detail and scale models to large input sizes [5]. Existing popular deep learning models like auto-encoders (AEs), variational auto-encoders (VAEs), and generative adversarial networks (GANs) are capable of an unsupervised learning of 3D shapes and generating low latent representations of 3D data. Popular AE architectures that process point cloud representations are PointNet [6] and PointNet++ [7]. Achlioptas et al. [1] proposed an AE architecture, where the encoder is similar to PointNet and the decoder comprises fully connected layers. However, the generative capabilities of a regular AE in terms of generating novel and realistic 3D shapes are limited [8]. Thus, in addition, Saha et al. [8] utilized a point cloud-based variational autoencoder (PC-VAE) [8] that builds on the point cloud autoencoder originally proposed in [1] and evaluated the model for novel and realistic 3D shape generation for engineering applications. In the context of optimization tasks, previous research [9]–[11] utilized the low-dimensional latent representation of an AE or a VAE for single and multi-objective optimization. Due to better generative abilities of PC-VAE in comparison to AE [1], we intend to use the latent representation of the PC-VAE proposed in [8] as decision variables for our optimization task. This overcomes a major factor contributing to the complexity and difficulty of an optimization problem by reducing the number of decision variables.

B. Evolutionary Algorithms

Evolutionary algorithms facilitate the combination of components in a novel and creative way, making them a good tool for creative design problems, while also having the ability to optimize a parameterized shape for performance criteria. Renner et. al [12] gave an overview of genetic algorithms (GAs) in computer-aided design (CAD) concerning both parametric design and creative design problems. In recent decades, a wide range of evolutionary algorithms (MOEA) for multi-objective optimization has been developed, which are classified into two categories. The first category corresponds to those algorithms that include the use of selection mechanisms based on fitness sharing, but does not include mechanisms for the preservation of good solutions (elitism), e.g., NPGA, NSGA [13], VEGA [14], MOGA [15]. The second category of algorithms are characterized by the use of the elitism strategy, e.g., SPEA [16], PAES [17], SPEA2 [18], NSGA-II [19]. Besides the above mentioned evolutionary algorithms in MOPs, several meta-heuristics have been developed to solve such problems as Ant Colony Optimization (MOACO) [20] and Particle Swarm Optimization (OMOPSO) [21].

C. Seeding the Initial Population of MOEA

One of the main components of most MOEAs is to maintain diversity within a population in order to avoid premature convergence. Promoting diversity is a key feature of an efficient MOEA. A typical MOEA starts from a set of initial solutions and iteratively improves the solutions during the optimization process. Here, MOEA rely on random initialization of the initial populations, but often in large and complex search spaces this random method leads to an initial population which consists of infeasible solutions [22]. Previous research on modifying the initial population of an MOEA [23]–[25] showed that constructing a well-suited initial population can speed up GAs and reduce the convergence time to achieve acceptable results. This method of injecting knowledge about the problem in the initial population of the MOEA is known as *seeding*. Fraser et. al [26] proposed that leveraging target-oriented knowledge about the problem for the initialization of the initial population can considerably improve the algorithm's performance. Opposed to prior research, we aim to incorporate the knowledge of the optimization problem using latent space interpolations of a VAE as advantage to generate a target-oriented initial population for MOEAs to maintain a good diversity of the final non-dominated solution set.

D. Incorporating User Preference in Optimization

The decision-maker (DM) is a person who can express preference information related to conflicting objectives in a MOP. In our optimization task the designer is our DM. There are different methods for DMs to specify preferences [27]: the most common methods include dominance relation, fitness evaluation, termination criteria, constraint limits, etc. Another popular approach is based on adjusting the weights in terms of preferences in a weighted-sum method (WSM). WSM generates a single final solution point by presumably incorporating a single set of weights based on DM preferences. The two most popular approaches are the weighted sum [28] and Tchebycheff approach [29]. However, a weighted sum is capable of the linear approximation of a preference function only when the feasible design space is convex [3]. The weights in WSM need to be set according to the relative magnitude of the objective functions rather than the relative importance of the objectives [30]–[32]. Arbitrary specification of these weights values can lead to an undesired solution. We here propose a strategy to adjust the weights in WSM to obtain the final solution that matches the DM's preference.

In this paper, we formulate a multi-objective problem for modifying 3D shapes using the latent code of a VAE as the decision variable for the optimization task. We propose a method for seeding the initial population of an MOEA and compare our proposed method with MOEAs with random initial populations. Second, we formulate design preferences for a 3D design optimization problem and utilize preference-based multi-objective optimization (weighted-sum or MOEA) approaches for generating solutions based on preference criteria. We propose a method to adapt the DM's formulated prior weights to obtain a final solution that reflects the DM's design preference. In Section III, we provide details on our optimization problem and task formulation before we describe our proposed approach and experimental design setups (Section IV).

III. Problem Formulation

In this section, we first formulate a multi-criteria decision-making (MCDM) problem for modifying 3D shapes. Then, we introduce two scenarios for MCDM analysis.

A. MCDM Problem Formulation

Fig. 1. A digital car represented as 3D polygonal mesh and 3D point cloud.

For designing a car shape, the designer has many objectives in mind, i.e., the designer aims to modify a complete car design based on multiple criteria like aerodynamics efficiency and structural analysis. To solve this multi-criteria problem and also to keep the computational cost low, we formulate a multi-objective design optimization task using two 3D target matching problems for this study. In a target shape matching problem, an initial 3D shape defined by a set of points representing the surface mesh (Fig. 1), has to be altered to match a given 3D target point set. Here, we consider two reference (target) shapes which reflect e.g., an aesthetic design target (S_1) and an aerodynamic target (S_2). Our multi-objective design problem can be formulated as,

$$\min_{x \in D} f_1\left(\vec{x}_j\right), f_2\left(\vec{x}_j\right) \quad j = 1, .., n \tag{1}$$

where f_1, f_2 are the two objective functions. $\vec{x}$ is an n-dimensional latent vector of a VAE used as decision variables for the optimization. The objective functions of the optimization receive an n-dimensional latent vector of each 3D shape from the latent space of a trained VAE. Then the decoder of a VAE converts the latent vector to the 3D domain where the objective function computes the difference between the reference shapes to the current deform shape.

Mathematically the fitness for the two point sets S and S_i is defined by Equation 2. We use the chamfer distance (CD) as an objective function since it is invariant to the permutations of the points in a point cloud which represents the shape [1]. The fitness functions are given as

$$f_i(\vec{x}) = CD(S, S_i) = \sum_{a \in S} \min_{b \in S_i} \|a-b\|_2^2 + \sum_{a \in S_i} \min_{b \in S} \|a-b\|_2^2, \quad \text{where } i = 1, 2 \tag{2}$$

where S_i represents the two reference shapes S_1 and S_2, and S is the shape modified by the optimization algorithm. In a real-world scenario, the objectives are typically in different magnitudes. Thus, there is a need for the normalization of each objective. For normalizing the objective function values, we need to determine the upper bound (Nadir point z_i^N) of the pareto optimal set. The upper bound is obtained by separating i objectives into i sub-problems, and these sub-problems are minimized independently using a single-objective optimization strategy. We normalized each objective function $(\bar{f}_i)$ with the upper bound value of the opposite objective function. The normalized objective functions $(\bar{f}_i)$ are defined as

$$\begin{aligned} z_i^N &= \min\left(f_i\left(\vec{x}_j\right)\right) \\ \bar{f}_i &= \frac{f_i}{z_i^N} \text{ and } 0 \leq \bar{f}_i \leq 1 \end{aligned} \tag{3}$$

This multi-objective optimization problem leads to several

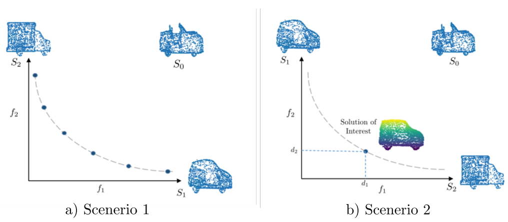

Fig. 2. Two scenarios for multi-criteria decision analysis. S_1, S_2 are two reference shapes for the optimization task and S_0 is the starting shape for the design modification. (a) Scenario 1 for generating a range of diverse solutions. (b) Scenario 2 for generating solution(s) of interest based on DM's design preference.

multi-criteria decision analysis scenarios. In this paper, we describe two scenarios here: First, the DM aims to modify a starting shape S_0 to generate a wide range of design possibilities between two reference shapes S_1 and S_2 (Figure 2a). In the second scenario, we assume that the DM has a preference to generate a solution of interest such that the shape is an intermediate between the two reference shapes S_1 and S_2 (Figure 2b). For example, the DM aims to generate a solution of interest and define the preference using a distance d_1 and d_2 from the reference designs S_1, S_2.

Note that prior to all following experiments, we generated a set of non-dominated solution points as a first approximation of the pareto front. This set is required to evaluate the performance of the different algorithms, e.g., using the IGD measure (Section V.C). For finding this solutions set (best-NDS), we converted the multi-objective optimization problem into a mono-objective function using weighted-sum method and systematically sample weights between $[0, 1]$ to yield best-NDS. Of course this is only possible here to benchmark the algorithms using target shape matching, which has a comparably low function computation time. In practical use cases like car aerodynamic optimization the computational time for a systematic WSM is too high.

B. Generation of Diverse Solutions Using MOEA

The first scenario refers to Figure 2a, where we assume that the DM has no prior preference, and targets to obtain a diverse range of design solutions that approximates the pareto front of the multi-objective optimization well. The objective functions (Eq. 1) are normalized, and the final normalized optimization functions for our MOP are formulated as

$$min \bar{f}_1(\vec{x}), \bar{f}_2(\vec{x}) \tag{4}$$

C. Single Solution Generation Incorporating Preference in MOEA

For the second scenario in Figure 2b, in order to generate a single solution point that reflects preferences, the DM needs to specify comparative preference states for each objective function value. However, since our objective functions are distance functions, we assume that the DM has knowledge about the few best pareto optimal points (best-NDS), and the DM can specify the desired distance values d_1 and d_2 for each objective. Thus, the DM's preference can be defined as a distance ratio (α) function:

$$\begin{gathered} \alpha = \frac{d_1}{d_2} \\ \text{the normalized preference ratio}(\bar{\alpha}) \\ \text{with } \bar{d}_1 = \frac{d_1}{z_1^N} \text{ and } \bar{d}_2 = \frac{d_2}{z_2^N} \\ \bar{\alpha} = \frac{\bar{d}_1}{\bar{d}_2} \end{gathered} \tag{5}$$

The preference relation describes the relative importance over a set of objective functions. We consider three scenarios for 3 different preference relations: 1) when the quality of importance of objective f_1 is higher compared to f_2 ($d_1 > d_2$), so that $\alpha = 2$, 2) when objectives are of equal importance ($d_1 = d_2$), $\alpha = 1$, 3) when the quality of importance of objective f_2 is higher compared to f_1 ($d_1 < d_2$), $\alpha = 0.5$. The normalized preference ratio $(\bar{\alpha})$ is calculated for each of the above three scenarios and depends on the selection of d_1 and d_2.

The DM has to state these additional preferences, and often this additional knowledge of DM preference is integrated as an additional objective within the optimization algorithm. The preference information is incorporated in two ways in our optimization problem:

a) Weighted sum method: The objective functions in MOP (Eq. 4) are summed up with varying weights, and this aggregated objective function is optimized. The weights in WSM need to adjust to obtain the final solution that matches the normalized preference ratio $(\bar{\alpha})$.

b) Preference as a 3rd objective in MOP: In the second approach the preference information ($\bar{\alpha}$) is formulated as an additional objective in MOP.

IV. Proposed Approach

We propose two approaches to address two issues based on our problem formulations: First, in order to improve the convergence of MOEA for our multi-objective design optimization task, we propose a novel mechanism for seeding the initial population of an MOEA. Second, to determine weight values in WSM to match the DM's design preference in our multi-objective design optimization task, we propose a combinative strategy to identify the best weight values.

For both approaches, we trained a PC-VAE [8] on the car class from the ShapeNet data-set [33] (each car design is a point cloud of 2048×3 matrix). The PC-VAE trained with 128 latent dimensions, i.e., each 3D point cloud in converted into a 128-latent vector. This 128-latent vector is used as the decision variable for our multi-objective optimization problem.

A. Proposed Method for Seeding the Initial Population of MOEA

The MOEAs used to solve MOPs are generally initialized with a random initial population. Thus, to improve the diversity within the non-dominated set, we propose an approach to generate a target-oriented initial population for MOEAs in this section. For generating the initial population of an MOEA, the normalized optimization functions in our MOP (Eq. 4) are decomposed into 2 single-objective sub-problems. Each single-objective optimization is performed using the covariance matrix adaptation evolution strategy (CMA-ES) [34]. The CMA-ES method is selected due to its suitability for small populations [34], high convergence ratio, and a low number of hyper-parameters. Next, the final solutions of each single-objective optimization are obtained. The final solutions are in the form of latent vectors, so we interpolate between the final vectors in the latent manifold of the PC-VAE by calculating ratios of the contribution from two final solutions, then enumerate these ratios and construct a vector for each ratio. The 100 latent vectors generated through interpolation (*Lerp-seed*) are used for the initialization of the initial population of an MOEA. The framework of our seeding mechanism is shown in Figure 3.

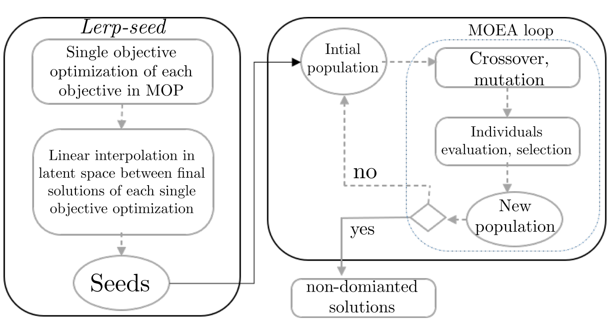

Fig. 3. Proposed seeding mechanism in an MOEA.

B. Proposed Method for Determining the Weights of WSM

For the incorporation of preference information in evolutionary algorithms, we consider one approach as a weighted sum method (WSM). The WSM solves the multi-objective problem by transforming the objective functions of MOP in Equation 4 into a mono-objective one (Eq. 6).

$$F(\vec{x}; w_1) = w_1 \bar{f}_1(\vec{x}) + w_2 \bar{f}_2(\vec{x}) \tag{6}$$

The final solution of the mono-objective optimization depends on the values of the weights w_1 and w_2. The normalized objective functions $\bar{f}_1$ and $\bar{f}_2$ are summed in WSM and the aggregated objective function $F(\vec{x}; w_1)$ in Equation 6 is minimized. w_1 and w_2 are weight factors, and $0 \leq w_{1,2} \leq 1$. It is common practice to choose weights in a way that their sum equals to 1, i.e., $w_2 = 1 - w_1$. Since there is few research on adapting the weights in WSM to match DM's design preference, we propose a combinative method to determine the weights of our WSM approach to generate the final solution that reflects the DM's design preference criteria.

Generating a final solution by selecting a set of weights in WSM may not necessarily reflect the intended preferences. So it is important to determine a strategy to adapt the weights to obtain the preferred solution in the convex pareto front. In this study, we propose a strategy to determine iteratively a global optimum weight w_1 in WSM to generate a final solution of the optimization that is closer to the DM's design preference and utilize a low computational cost. To search for an optimal w_1 within a bound, we propose a two-step method involving a global optimization of the weighted-sum function to minimize a cost function. The global optimizer minimizes the following cost function:

$$C(\vec{x}; w_1) = (\bar{\alpha} - \frac{\bar{f}_1(\vec{x})}{\bar{f}_2(\vec{x})})^2 \tag{7}$$

where $\bar{\alpha}$ is the normalized preference ration (Eq. 5)

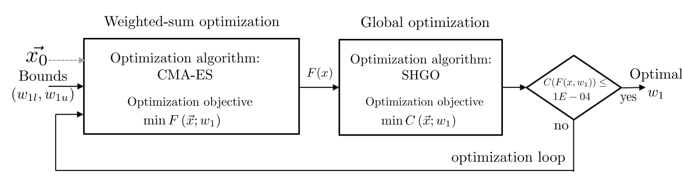

Fig. 4. Optimization strategy to determine weight w_1 in a weighted-sum method to match the DM's design preference.

Our proposed strategy iteratively runs two-steps as shown in Figure 4:

a) Weighted-sum optimization to optimize Eq. 6: In the first step, we consider the CMA-ES strategy for our WSM. CMA-ES often requires providing a feasible solution as a starting point for the initial candidate solution. To select the initial candidate solution in the CMA-ES algorithm, we consider the initial population vectors generated from *Lerp-seed* and calculated their normalized distance ($\hat{d}_1$ and $\hat{d}_2$) from the respective reference shapes (S_1, S_2) of the optimization task. The solution vector in *Lerp-seed* with normalized distances

($\hat{d}_1$ and $\hat{d}_2$) closest to the preferred distance function $\bar{d}_1$ and $\bar{d}_2$ is chosen as the initial solution vector ($\vec{x_0}$) for CMA-ES.

This approach of selecting the initial solution vector ($\vec{x_0}$) is intended to reduce the number of generations required for convergence to the optimum. The WSM using CMA-ES generates a final solution by minimizing Equation 6.

b) Global optimization to determine the weights to be used in Eq. 6: In the second step, we utilize a global optimization algorithm that demonstrates a promising global search capability. There are numerous global optimization techniques, yet only a few of them are derivative-free. The simplicial homology global optimization (SHGO) [35] algorithm is a promising, derivative-free global optimization (GO) algorithm, and it also returns all other local and global minima. A global minimum is a point where the function's value is the minimum of all possible points in the function's domain. The optimizer determines the global minimum of weight w_1 by minimizing the cost function in Equation 7. The final optimal weight w_1 helps the DM to understand the relation between the weights in WSM and the final solution.

V. Experimental Setup

In this section, we present a sequence of experiments that illustrates approaches for generating diverse design solutions and solution(s) of interest using our proposed methods.

A. Model Training of PC-VAE

Data: For all experiments, we used point clouds sampled from the car class of the ShapeNetCore data-set [33]. As input to the models, we sampled 2048 points using random uniform sampling from each shape's surface, resulting in a matrix of 2048×3 with spatial coordinates (x, y, z) for each point. We performed all experiments on a single NVIDIA RTX 2080 Ti GPU.

Training PC-VAE: We trained the PC-VAE [8] on the $80\% - 20\%$ train-test-split, with 128 latent dimensions, using the ADAM optimizer and a learning rate $5e^{-03}$ as proposed in [8]. Thus, the latent code of the PC-VAE used in the optimization experiments as a decision variable is set to a 128-dimensional vector.

B. Problem Instances

To define different problem instances involving considerably different combinations of reference shapes (S_1, S_2), we cluster the latent representations of the shapes in the training set using k-means clustering with Euclidean distance as a metric to evaluate the similarity in the data. The clustering organizes the data according to their characteristics in the latent space of the trained PC-VAE. The reference shapes selection and optimization framework are shown in Figure 5. We select the reference shapes for 3 problem instances (PI) based on 3 scenarios:

- PI 1: Reference shapes from two adjacent clusters
- PI 2: Reference shapes from two different clusters
- PI 3: Reference shapes from same cluster

Once the PC-VAE is trained, the optimizations are performed for each problem instance.

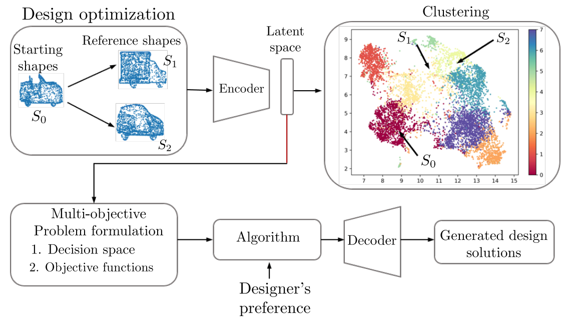

Fig. 5. Optimization workflow with the procedure to select different problem instances.

C. Experimental Setup for Diversity Solution Generation Experiments

This experimental setup refers to Scenario 1 (Figure 2a) of the multi-criteria decision analysis to generate a range of design solutions. In this experiment, we considered three multi-objective optimization tasks based on 3 problem instances. The MOEAs chosen for the optimization tasks comprises one that uses elitism strategy (NSGA-II) and another that follows a meta-heuristic approach (OMOPSO), both MOEAs are initialized with random seeds. The third chosen MOEA is *Lerp-seed*-OMOPSO, where we applied our proposed approach of seeding for an improved initialization of the initial population of OMOPSO, since OMOPSO outperforms NSGA-II in initial experiments (not shown).

a) Parameterization of MOEAs: We have chosen a set of parameter settings to guarantee a fair comparison among the algorithms. The GA (NSGA-II) uses an internal population size equal to 100, and OMOPSO has been configured with 100 particles. To assess the search capabilities of the algorithms, we perform 10 independent runs of each experiment for 100 generations, and we compare different metrics of performance.

TABLE I
Parameterization of MOEA algorithms.

Parameterization used in NSGA-II	
Population size	100 individuals
Selection of parents	binary tournament
Parameterization used in OMOPSO	
Swarn size	100 particles
Mutation	uniform+non-uniform
Leader size	100

b) Seeding of MOEA: To generate the seeds for our MOEA with our proposed approach (Figure 3), we converted the objectives of our MOP in Equation 4 into 2 single objective sub-problems. Next, each single objective sub-problem is optimized using CMA-ES. The approach implemented follows the proposal of Hansen et al. [34], considering a (3,10) CMA-ES strategy with an initial step size of 0.01 and 100 generations. The objective functions $\bar{f}_1$ and $\bar{f}_2$ in Equation

4 are minimized separately, where $\vec{x}$ is the 128-dimensional latent vector of the trained PC-VAE. The final solution of each single objective optimization is 128-dimensional latent vector that represents a 3D car design. Next, we performed a 100 step linear interpolation in the latent space of the PC-VAE between the two final solution vectors. All these interpolated solution vectors (*Lerp-seed*) are added as an individual to the initial population of an MOEA. We analyzed the effect of our seeding strategy (*Lerp-seed*-OMOPSO) with OMOPSO and compare our results with no seed MOEAs using the performance metrics discussed below.

c) Metric of performance: For comparing the behavior of the MOEAs, we considered the following four metrics:

Hyper-volume indicator (HV) [16]: measures the volume enclosed by a solution set and a specified reference point and can provide a quality of convergence and diversity. A high HV value is preferable, reflecting the set having good comprehensive quality. Since we normalize the objective functions, the reference point is chosen as $(1, 1)$.

Inverted generational distance (IGD) [36]: indicates how far are the non-dominated solutions produced by the algorithm from the reference points in the true pareto front (best-NDS) of the problem. A smaller IGD indicates that all the elements generated are in the true pareto front.

Spacing (SP): Spacing as suggested in [37] measures the distance variance of neighboring vectors in the pareto front. Lower SP shows that all the non-dominated solutions found are equidistantly spaced.

Number of function evaluation (NFEs): We considered total number of simulation (experimental) runs as a performance indicator. The total number of function evaluations performed by the algorithm equals the product of the population size (or the number of particles) and the number of generations.

D. Experimental Setup for Solutions of Interest Experiments

This experimental setup refers to Scenario 2 (Figure 2b) of the multi-criteria decision analysis for generating solution(s) of interest based on DM's design preference. For each problem instance, we consider 3 preference scenarios ($\alpha = 2, 1$ and 0.5) (Section III-C) and the DM is asked to choose (d_1, d_2) for all the preference scenarios. Next, we normalize the preference ratio ($\bar{\alpha}$) using Equation 5. This normalized preference ratio (α) is considered as additional objective or constraint in the optimization approaches. Two different optimization approaches (WSM or 3-objective MOEA) are used to solve our MOP (Eq. 4) and generate a solution that satisfies the DM's preference criteria in Equation 5.

a) Parameterization of WSM: In the first WSM approach, the composite objective function in Equation 6 is minimized for different values of w_1 until the final solution satisfies the preference criteria in Equation 5. We utilized our proposed method (Section IV-B) for determining the weights (w_1, w_2) to generate a final solution to match the DM's design preference. In each problem instance, for each preference scenario ($\alpha = 2, 1$ and 0.5), we provided an initial bound $[0, 1]$ at the start of each optimization process. The global optimizer attempts to find the global minimum (Fig. 4) and the process continues until the cost function (Eq. 7) converges. The global minimum is considered as the final weight (w_1) suggested by our proposed approach, such that the final solution generated with this optimal weights (w_1, w_2) in WSM is closer or matches the DM's design preference.

b) Parameterization of 3-objective MOEA: In the second approach of preference incorporation in MOEA as 3rd objective function (3-objective MOEA), we minimize the following objective functions

$$\min \bar{f}_1(\vec{x}), \bar{f}_2(\vec{x}), \bar{\alpha} - \frac{\bar{f}_1(\vec{x})}{\bar{f}_2(\vec{x})} \tag{8}$$

The normalized preference ratio $\bar{\alpha}$ is the third objective of the multi-objective optimization task. The MOEA chosen for this optimization task is OMOPSO. Further, to compare the solution generated by these two approaches, we compared a range of performance metrics.

c) Metric of comparison: For our comparative study, we evaluate three unary metrics:

Number of function evaluations (NFEs): The total number of function evaluations for our combination method depends on the number of calculations of objective functions. In WSM, we propose to search weights w_i using a global optimizer. Each iteration of the optimization loop needs 500 function evaluations for performing the CMA-ES optimization. Further, modifying the initial solution of CMA-ES ($\vec{x_0}$) needs 2000 NFEs. Thus, the total NFEs for our proposed approach of determining weights w_i for WSM is $(2000 + n \times 500)$ NFEs. The total number of function evaluations exhausted by the MOEA algorithm (OMOPSO) equals the product of the number of particles (100) and the number of generations(100).

Preference ratio: measures the ratio of the distance in $\bar{f}_1$ to $\bar{f}_2$ of the final generated solution.

Quality: of the final generated solutions is measured by Euclidean distance between the generated solution and preferred solution $(\bar{d_1}, \bar{d_2})$. The lower the Euclidean distance, the closer the final generated solution to the DM's preferred solution.

VI. Results

In this section, the results obtained by the experimental evaluations are discussed in detail.

A. Multi-Objective Optimization for Generating Diverse Design Proposals

The results of the comparative study of the experiments in Section V-C related to Scenario 1 (Figure 2a) are described here. We firstly evaluated the different MOEAs (NSGA-II, OMOPSO) with random seeds, and assessed whether our proposed seeding strategy *Lerp-seed* in OMOPSO can outperform the existing MOEAs with random seeds. We first explain the results of multi-objective optimization with reference shapes from problem instance 1 (Section V-B).

TABLE II

RSULTS OBTAINED FROM OPTIMIZATION OF PROBLEM INSTANCE 1 FOR NSGA-II, OMOPSO AND *Lerp-seed*-OMOPSO.

Methods	**NFEs**	**HV** (mean)	**SP** (mean)	**IGD** (mean)
NSGA-II	10000	0.473	**0.186**	0.060
OMOPSO	10000	0.484	0.229	0.059
Lerp-seed OMOPSO	10000	**0.490**	0.211	**0.055**

Table II presents the mean values of HV, IGD, and SP of the optimization task using NSGA-II, OMOPSO and *Lerp-seed*-OMOPSO. *Lerp-seed*-OMOPSO approach achieved the best results with respect to mean HV and mean IGD values for our design optimization task. We tested the HV difference using Mann-Whitney test in order to prove the difference between *Lerp-seed*-OMOPSO and other MOEAs (NSGA-II, OMOPSO) algorithms are statistically significant. The statistical analysis reveals significant statistical differences between *Lerp-seed*-OMOPSO and other MOEAs ($p < 0.05{*}{*}{*}$). Thus, an MOEA with high HV and low IGD values provides better distribution of the generated non-dominated solutions. The NSGA-II returns a higher number of non-dominated solutions and has lower spacing (SP) between the solutions. However, the OMOPSO (with or without seeds) provides a better spread of solutions than NSGA-II. Besides the quantitative comparison, the non-dominated solutions plot obtained for one run of problem instance 1 (Fig. 6), shows that the *Lerp-seed*-OMOPSO has a better spread of solutions than NSGA-II and OMOPSO with random seeds.

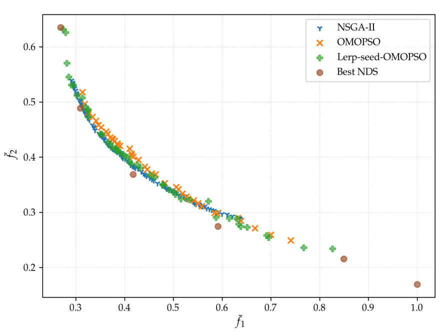

Fig. 6. Pareto fronts obtained by the NSGA-II, OMOPSO and *Lerp-seed*-OMOPSO for problem instance 1 (shown for 1 run).

In addition, we performed multi-objective optimizations with reference shapes from problem instance 2 and 3 and took a close look at how the mean values of HV and IGD change with respect to NFEs as the search evolves. Figure 7 and 8 show the mean HV and mean IGD of the non-dominated solution set for 10000 NFEs over 10 runs for 3 optimization tasks. Our proposed method (*Lerp-seed OMOPSO*) requires initial 2000 function evaluations to run the two single objective CMA-ES, so we start the mean HV and mean IGD plot of our proposed approach after 2000 NFEs to make fair comparisons.

In all the experiments, the OMOPSO using *Lerp-seed* for initial population construction outperformed a random-based approach on average. This leads to the conclusion that spending some additional computational effort in constructing the initial population of an MOEA helps to improve the convergence in the optimization. However, the exploitation of this low-dimensional latent domain knowledge comes with an additional cost of training the PC-VAE with 3D shapes.

Further, we observe that for problem instance 3, the optimization algorithms provide a high scale of HV value and a low scale of IGD value. While for problem instance 2, the algorithms provide lower scale of mean HV and IGD values. This is because reference shapes in problem instance 3 are from similar clusters (less Euclidean distance between the reference shapes in the latent subspace of the trained PC-VAE), so the optimization algorithms generated a better set of diverse non-dominated solutions compared to problem instance 2, where the reference shapes are from two different clusters (high Euclidean distance between the reference shapes in the latent subspace of the trained PC-VAE). However, for all the problem instances, the optimization algorithms can achieve feasible car shapes. Thus, we can conclude that different MOEAs can solve our problem of multi-objective optimization for generating diverse 3D car designs.

B. Generating Solutions Based on DM Preference

The results of the comparative study of the experiments in Section V-D related to Scenario 2 (Figure 2b) are described here. For each problem instance (Section V-B), we tested three different preference scenarios ($\alpha = 2, 1$ and 0.5), using WSM and 3-objective MOEA.

Table III presents the number of function evaluations (NFEs), preference ratio, and quality of the final solutions generated by WSM and 3-objective MOEA. The optimization using the 3-objective MOEA generates more than one final solution, so we chose one/two final solutions nearer to the normalized preference ratio ($\bar{\alpha}$). Also, for 3-objective MOEA, the NFEs are constant for each optimization task.

For all problem instances and normalized preference ratios in table III, both of the optimization methods converge and generated feasible car shapes. However, the WSM method with a global optimizer generates a shape that is closer (lower quality metrics in table III) to the DM-defined preference and also with much fewer NFEs. The WSM generates different weights w_1 depending on each problem instance and normalized preference ratio ($\bar{\alpha}$).

Thus, we conclude that our proposed strategy (Fig. 4) adapted the initial bound $[0, 1]$ of weight (w_1) to an optimal weight value (Optimal w_1), and the final solution of WSM with this optimal w_1 is the closest to the DM's preference criteria. Thus, WSM provides better results for linear approximation of preference criteria.

VII. CONCLUSIONS AND OUTLOOK

In the present paper, we discussed a real-world inspired 3D multi-objective optimization problem related to two scenarios for coping with designer preferences.

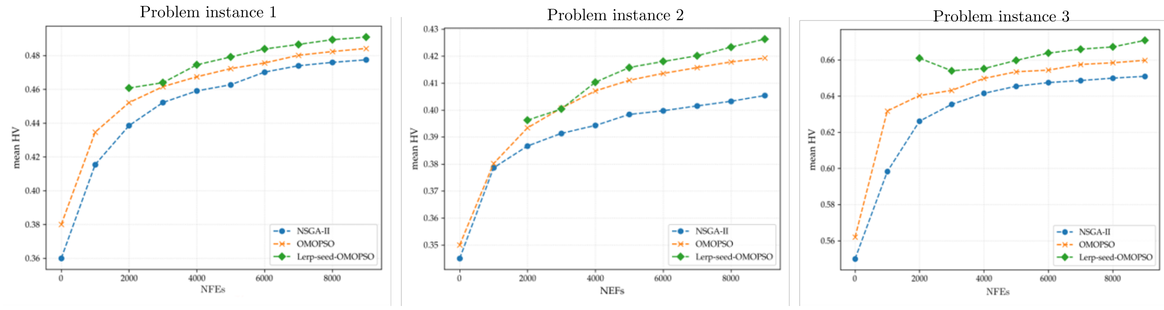

Fig. 7. The change of mean HV (10 runs) with respect to the number of function evaluations (NFEs) on 3 problem instances.

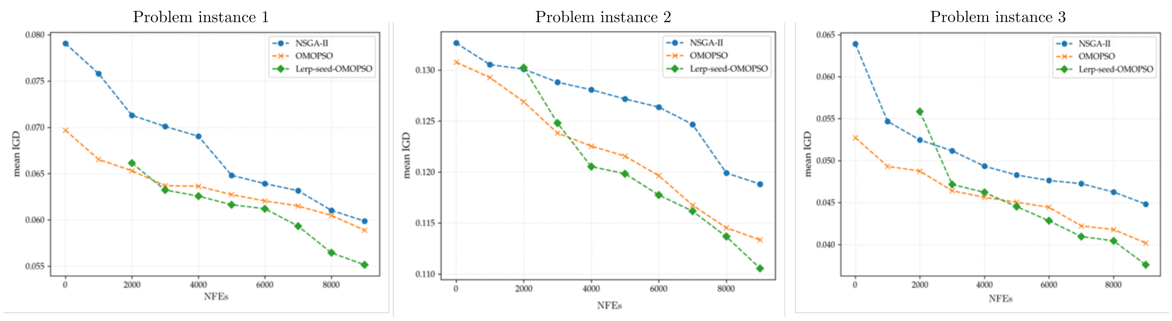

Fig. 8. The change of mean IGD (10 runs) with respect to the number of function evaluations (NFEs) on 3 problem instances.

TABLE III

METRIC COMPARISONS (NFES, PREFERENCE RATIO, AND QUALITY) FOR GENERATING SOLUTION (S) OF INTEREST USING WSM WITH THE GLOBAL OPTIMIZER AND 3-OBJECTIVE MOEA. EACH OF THE PROBLEM INSTANCES (PI) IS EVALUATED FOR EACH OF THE PREFERENCE SCENARIOS. FOR WSM, WE REPORTED AS NFES (n), WHERE n IS THE NUMBER OF ITERATIONS THROUGH THE OPTIMIZATION LOOP.

Methods	**Metrics**	**Preference ratio (α = 2)**			**Preference ratio (α = 1)**			**Preference ratio (α = 0.5)**		
		PI 1 $\bar{\alpha}$ =7.94	PI 2 $\bar{\alpha}$ = 3.62	PI 3 $\bar{\alpha}$=8.66	PI 1 $\bar{\alpha}$=1.6	PI 2 $\bar{\alpha}$=0.91	PI 3 $\bar{\alpha}$=1.06	PI 1 $\bar{\alpha}$=0.56	PI 2 $\bar{\alpha}$=0.22	PI 3 $\bar{\alpha}$=0.085
Search w_1 with global optimizer in WSM	NFEs (n)	**4000 (4)**	**6000 (8)**	**4000 (4)**	**5000 (6)**	**3500 (3)**	**4000 (4)**	**5500 (7)**	**5000 (6)**	**5000 (6)**
	Normalized preference ratio ($\bar{\alpha}$)	7.96	3.48	7.72	1.10	0.92	1.09	0.48	0.20	0.095
	Quality	**0.019**	**0.085**	**0.089**	**0.022**	**0.003**	**0.036**	**0.045**	**0.054**	**0.050**
3-objective MOEA	NFEs	10000	10000	10000	10000	10000	10000	10000	10000	10000
	Normalized preference ratio ($\bar{\alpha}$)	6.30	3.92, 3.35	5.30	1.10, 1.03	1.06	1.04, 0.95	0.56, 0.64	0.24	0.64
	Quality	0.114	0.194, 0.139	0.10	0.121, 0.120	0.054	0.24, 0.19	0.28, 0.27	0.16	0.40

In the context of multi-objective design optimization, to improve the performance of the multi-objective algorithms, we propose a seeding strategy using the latent space knowledge of a PC-VAE. However, it should be noted that our proposed method requires an existing data set for training the PC-VAE, which is not always available prior to the optimization. Additionally, defining the data-set is already a challenging task, since it should contain the complete set of geometric features that the optimization search should be able to reach during the optimization. Nevertheless, we demonstrated that if prior data is available then spending computational budget on training a PC-VAE provides an additional advantage in improving the diversity of the generated design solutions and lowers the convergence cost. Currently, we utilized NSGA-II and particle swarm optimization algorithms in our framework but plan to extend our studies for further state-of-the-art methods, like SPEA2 and MOEA/D.

Further, for the second multi-criteria decision-making scenario, we concluded that the weighted-sum method (WSM) is easy to use. However, it provides only a linear approximation of the preference criteria. We quantitatively showed that our proposed strategy to determine the optimal weights in WSM performs better than other optimization algorithms, while being faster to generate the final design with less number of function evaluations. We demonstrated in an experimental setup that the weights in WSM with our proposed strategy generated a feasible final design solution closer to the designer's preference criteria.

Finally, this is the first study to explore the feasibility of latent representations of a point cloud-based variational autoencoder (PC-VAE) as a geometric representation for multi-objective optimization problems of 3D designs. The latent representation of the PC-VAE trained on ShapeNet data-set showed an advantage for performing a multi-objective design optimization by generating feasible 3D car designs with less computational budget in both multi-criteria decision analysis scenarios. These generated designs may be given back as suggestions to the designer in a cooperative design system. Thus, spending additional costs for training a PC-VAE is advantageous for performing a multi-objective design optimization task.

REFERENCES

[1] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3d point clouds," *35th International Conference on Machine Learning, ICML*, vol. 1, pp. 67–85, 2018.

[2] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *2nd International Conference on Learning Representations, ICLR*, 2014, pp. 1–14.

[3] R. T. Marler and J. S. Arora, "The weighted sum method for multi-objective optimization : new insights," *Structural and Multidisciplinary Optimization*, vol. 41, no. 6, pp. 853–862, 2010.

[4] T. Friedrich, N. Aulig, and S. Menzel, "On the Potential and Challenges of Neural Style Transfer for Three-Dimensional Shape Data," in *EngOpt 2018 Proceedings of the 6th International Conference on Engineering Optimization*. Springer International Publishing, 2019, pp. 581–592.

[5] T. Rios, P. Wollstadt, B. V. Stein, T. Back, Z. Xu, B. Sendhoff, and S. Menzel, "Scalability of Learning Tasks on 3D CAE Models Using Point Cloud Autoencoders," in *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2019, pp. 1367–1374.

[6] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2017, pp. 652–660.

[7] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Advances in Neural Information Processing Systems*, 2017, pp. 5099–5108.

[8] S. Saha, L. L. Minku, X. Yao, B. Sendhoff, and S. Menzel, "Exploiting Linear Interpolation of Variational Autoencoders for Satisfying Preferences in Evolutionary Design Optimization," in *EEE Congress on Evolutionary Computation (CEC)*.

[9] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik, "Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules," *ACS Central Science*, vol. 4, no. 2, pp. 268–276, 2018.

[10] R. R. Griffiths and J. M. Hernández-Lobato, "Constrained Bayesian optimization for automatic chemical design using variational autoencoders," *Chemical Science*, vol. 11, no. 2, pp. 577–586, 2020.

[11] J. D. Cunningham, D. Shu, T. W. Simpson, and C. S. Tucker, "A sparsity preserving genetic algorithm for extracting diverse functional 3D designs from deep generative neural networks," *Design Science*, vol. 6, pp. 1–33, 2020.

[12] G. Renner and A. Ekárt, "Genetic algorithms in computer aided design," *CAD Computer Aided Design*, vol. 35, no. 8 SPEC., pp. 709–726, 2003.

[13] N. Srinivas and K. Deb, "Muiltiobjective Optimization Using Nondominated Sorting in Genetic Algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221–248, 1994.

[14] J. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," *The 1st international Conference on Genetic Algorithms*, 1985.

[15] C. M. Fonseca and P. J. Fleming, "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization," *The Fifth International Conference in Genetic Algorithms*, vol. 93, pp. 416–425, 1993.

[16] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

[17] J. Knowles and D. Corne, "The Pareto archived evolution strategy: A new baseline algorithm for Pareto multiobjective optimisation," in *Proceedings of the 1999 Congress on Evolutionary Computation, CEC*, 1999, pp. 98–105.

[18] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm," *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, p. 103, 2001.

[19] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II," in *International conference on parallel problem solving from nature*, 2000, pp. 849–858.

[20] C. García-Martínez, O. Cordón, and F. Herrera, "An empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria TSP," in *International Workshop on Ant Colony Optimization and Swarm Intelligence*, 2004, pp. 61–72.

[21] J. J. Durillo, J. García-Nieto, A. J. Nebro, C. A. Coello Coello, F. Luna, and E. Alba, "Multi-objective particle swarm optimizers: An experimental comparison," in *International conference on evolutionary multi-criterion optimization.*, 2010, pp. 495–509.

[22] C. Haubelt, J. Gamenik, and J. Teich, "Initial population construction for convergence improvement of MOEAs," in *International Conference on Evolutionary Multi-Criterion Optimization*, 2005, pp. 191–205.

[23] S. Poles, Y. Fu, and E. Rigoni, "The effect of initial population sampling on the convergence of multi-objective genetic algorithms," *In Multiobjective programming and goal programming*, pp. 123–133, 2009.

[24] E. Khaji and A. S. Mohammadi, "A Heuristic Method to Generate Better Initial Population for Evolutionary Methods," pp. 1–8, 2014. [Online]. Available: http://arxiv.org/abs/1406.4518

[25] T. Chen, M. Li, and X. Yao, "On the effects of seeding strategies: A case for search-based multi-objective service composition," in *GECCO 2018 - Proceedings of the 2018 Genetic and Evolutionary Computation Conference*, 2018, pp. 1419–1426.

[26] G. Fraser and A. Arcuri, "The seed is strong: Seeding strategies in search-based software testing," in *Proceedings - IEEE 5th International Conference on Software Testing, Verification and Validation, ICST*, 2012, pp. 121–130.

[27] L. Li, I. Yevseyeva, V. Basto-Fernandes, H. Trautmann, N. Jing, and M. Emmerich, "An Ontology of Preference-Based Multiobjective Metaheuristics," no. March, 2016. [Online]. Available: http://arxiv.org/abs/1609.08082

[28] R. Wang, Z. Zhou, H. Ishibuchi, T. Liao, and T. Zhang, "Localized Weighted Sum Method for Many-Objective Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 1, pp. 3–18, 2016.

[29] S. Mardle and K. M. Miettinen, "Nonlinear Multiobjective Optimization," *The Journal of the Operational Research Society*, vol. 51, no. 2, p. 246, 2000.

[30] S. Phelps and M. Köksalan, "An Interactive Evolutionary Metaheuristic for Multiobjective Combinatorial Optimization," *Management Science*, vol. 49, no. 12, pp. 1726–1738, 2003.

[31] M. Koksalan and I. Karahan, "An interactive territory defining evolutionary algorithm: ITDEA," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 5, pp. 702–722, 2010.

[32] R. Wang, R. C. Purshouse, and P. J. Fleming, "Preference-inspired co-evolutionary algorithms using weight vectors," *European Journal of Operational Research*, vol. 243, no. 2, pp. 423–441, 2015.

[33] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An Information-Rich 3D Model Repository," Stanford University - Princeton University - Toyota Technological Institute at Chicago, Tech. Rep., 2015. [Online]. Available: http://arxiv.org/abs/1512.03012

[34] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies." *Evolutionary computation*, vol. 9, no. 2, pp. 159–195, 2001.

[35] S. C. Endres, C. Sandrock, and W. W. Focke, "A simplicial homology algorithm for Lipschitz optimisation," *Journal of Global Optimization*, vol. 72, no. 2, pp. 181–217, 2018.

[36] D. A. Van Veldhuizen and G. B. Lamont, "On measuring multiobjective evolutionary algorithm performance," *Proceedings of the 2000 Congress on Evolutionary Computation, CEC 2000*, vol. 1, pp. 204–211, 2000.

[37] Jason R. Schott, "Fault tolerant design using single and multicriteria genetic algorithm optimization," Ph.D. dissertation, Massachusetts Institute of Technology, 1995.