


Realistic Constrained Multi-Objective Optimization Benchmark Problems from Design

Cyril Picard  and Jürg Schiffmann

Abstract—Multi-objective optimization is increasingly used in engineering to design new systems and to identify design trade-offs. Yet, design problems often have objective functions and constraints that are expensive and highly non-linear. Combinations of these features lead to poor convergence and diversity loss with common algorithms that have not been specifically designed for constrained optimization. Constrained benchmark problems exist, but they do not necessarily represent the challenges of engineering problems. In this paper, a framework to design electro-mechanical actuators, called MODAct, is presented and 20 constrained multi-objective optimization test problems are derived from the framework with a specific focus on constraints. The full source code is made available to ease its use. The effects of the constraints are analyzed through their impact on the Pareto front as well as on the convergence performance. A constraint landscape analysis approach is followed and extended with three new metrics to characterize the search and objective spaces. The features of MODAct are compared to existing test suites to highlight the differences. In addition, a convergence analysis using NSGA-II, NSGA-III and C-TAEA on MODAct and existing test suites suggests that the design problems are indeed difficult due to the constraints. In particular, the number of simultaneously violated constraints in newly generated solutions seems key in understanding the convergence challenges. Thus, MODAct offers an efficient framework to analyze and handle constraints in future optimization algorithm design.

Index Terms—Multi-objective optimization, constraint handling, evolutionary algorithm, real world problems, test suites.

I. INTRODUCTION

ENGINEERING design problems often imply making a compromise between conflicting objectives (cost over robustness, weight over size) under a set of strict specifications. As such, they can easily be written as constrained multi-objective optimization problems (CMOPs), formulated as follows:

$$\begin{aligned} \min_{\mathbf{x} \in \mathcal{S}} \quad & \mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})]^T \\ \text{subject to} \quad & g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, \dots, p \\ & h_k(\mathbf{x}) = 0, \quad k = 1, 2, \dots, q \end{aligned} \quad (1)$$

where $\mathbf{f} : \mathcal{S} \rightarrow \mathbb{R}^m$ is a vector of m objective functions that map the bounded search space (or design space) \mathcal{S} to the objective space. The search space is defined by lower $\mathbf{x}^{(L)}$ and upper $\mathbf{x}^{(U)}$ bounds such as

$\mathcal{S} = \{\mathbf{x} \in \mathcal{S} \mid x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad i = 1, 2, \dots, n\}$. The specifications are represented by p inequality and q equality constraints. For a solution to be said feasible, it needs to satisfy all $p + q$ constraints. The set of all feasible solutions forms the feasible space $\mathcal{F} \subseteq \mathcal{S}$.

With respect to the definition in (1), engineering problems are not different to commonly used benchmark CMOPs [1], [2]. In practice though, their objective functions and constraints are expensive to calculate, offer no continuity guarantees and their behavior is unknown a priori (simulation-based results). These properties can make engineering problems challenging to solve.

Multi-objective optimization problems can be solved using stochastic meta-heuristics, of which multi-objective evolutionary algorithms (MOEAs) have been widely applied. Yet, most of the available algorithms have been developed primarily for unconstrained optimization problems [3] and are extended by constraint handling strategies to cope with CMOPs [4]–[6].

A. Nature of the issue

Recently reported results show that for some ordinary mechanical design optimisation problems, state of the art constrained optimization algorithms fail to ensure sufficiently reliable optimization outcomes [7], [8], although the same algorithms show good results on benchmark problems. Indeed, the latter have been found to have one or more of the following shortcomings:

- are too simple and can be solved by algorithms disregarding constraints [9];
- offer limited scalability of search space and number of objectives;
- are limited to one or two inequality constraints;
- cannot be adjusted in terms of the complexity of the constraints.

In addition, these benchmark functions have not been extensively characterized. Most often, only information about the number of variables, the number of constraints, the optimal Pareto front and the feasibility ratio (FsR = $\rho = |\mathcal{F}|/|\mathcal{S}|$) is given. While the latter is often thought to be related to the complexity of the problem, recent work shows that this metric is not per se sufficient [9].

B. Goals and objectives

This work presents a new generic framework used to design geared electro-mechanical actuators, called multi-objective design of actuators (MODAct), which is used to derive a realistic

C. Picard and J. Schiffmann are with the Laboratory for Applied Mechanical Design, École polytechnique fédérale de Lausanne, 2000 Neuchâtel, Switzerland e-mail: (see cyril.picard@epfl.ch; jurg.schiffmann@epfl.ch).

Manuscript received February 6, 2020; revised May 13, 2020 and July 14, 2020; accepted August 24, 2020.

multi-objective optimization test suite, focused on constraints. The functions are scalable in terms of decision space, number of objectives and number and type of constraints. In order to better understand how this test suite differs from existing benchmark problems, a constraint landscape analysis approach is applied and extended with three new metrics to give in-depth information on the features of these functions. Two common algorithms – NSGA-II [10] and NSGA-III [11] – are compared to a constrained optimization specific algorithm C-TAEA [12] to define a baseline convergence pattern. The effects of the constraints on convergence are discussed.

C. Scope of the paper

Existing CMOPs are presented in Section II along with constraint landscape analysis tools. Section III introduces the design framework along with 20 new benchmark functions based on a realistic mechanical design problem. Section IV details the extended methods applied to characterize these functions. The parameters for the constraint landscape analysis, as well as the ones used to run the comparative convergence study against other benchmark problems are presented in Section V. The results of both studies are reported in Section VI along with a discussion about the difference between the new design problems and existing benchmark CMOPs.

II. RELATED WORK

A. Constrained multi-objective optimization benchmark problems

Various CMOPs have been proposed over the years. Some of the oldest are the SRN [13], the TNK [14] and the OSY [15] functions. They have been commonly used to benchmark algorithms [10], [16]. These are two objective problems with few decision variables – two for SRN and TNK, and six for OSY – and the same number of inequality constraints. Mostly due to the low dimensionality of their decision space, their complexity is low [17].

Based on this observation, Deb et al. proposed a new scalable framework for CMOPs [17]. Functions derived from this framework have, in theory, an unbounded dimensionality and their complexity can be adjusted by the selection of a helper function. This framework has been used to derive the CTP test suite [17] with 7 functions (CTP1 to CTP7), which are typically limited to two objectives, two decision variables and one or two inequality constraints and remains therefore quite simple. The framework is also the base of the CF test suite [2] created for the IEEE CEC2009 MOEA Competition. Compared to CTP, CF problems have a larger search space ($n = 10$) and three out of the ten functions have three objectives. The results of the competition showed that some of these functions are difficult to solve. The number of constraints is also low (two inequality constraints). More recent variants of this framework have been proposed (e.g. NCTP [18]), but the shortcomings mentioned previously remain.

With the need to solve optimization problems with an increasing number of objectives, many-objective optimizers have been developed along with appropriate test problems. The DTLZ test suite [19], which is scalable also in terms of

objectives, has been extended with constraints to form the C-DTLZ test suite [1]. There are three inequality constraint sets (C1, C2 and C3) that can be combined with the unconstrained DTLZ functions (e.g. C1-DTLZ1 or C1-DTLZ3). C1 and C2 types add one constraint, while the C3 type adds one constraint per objective. While promising, Tanabe and Oyama showed that even algorithms discarding the constraints could solve some of the C-DTLZ problems [9].

In order to represent real-world problems better, Ma and Wang proposed a new framework to build test functions with more inequality constraints and a large infeasible search space [20]. They derived the MW test suite with 14 instances with up to four inequality constraints and an FsR very close to zero. Three instances are scalable in terms of objectives and they cover various front geometries.

The DAS-CMOP and DAS-CMaOP test suites [21] introduce the concept of tunable constraints through a difficulty triplet (η, ζ, γ) , with $\eta, \zeta, \gamma \in [0, 1]$. DAS-CMOP is composed of nine base problems with two or three objectives and 11 or 7 constraints. DAS-CMaOP adds another nine base problems for $m > 3$, extending the WFG framework [22] with $2m + 1$ similarly tunable constraints. 16 given difficulty triplets are suggested, four of which result in equality constraints. In total, this generates 288 test functions, 144 of which are scalable in terms of objectives. This massive test suite offers a great potential that has yet to be evaluated.

Given the strong ties between optimization and engineering applications, real-world like problems have also been proposed as benchmark problems [1], [6], [7], [23], [24]. Among those, the car-side impact [1] and the water problem [6], [23] have more than two objectives. Their decision space is relatively small, but they have a large number of constraints (10 and 7 respectively). It has been shown, however, that most solutions generated during optimization only violate one of the constraints and that unconstrained NSGA-II could also solve them fairly well [9].

Based on these considerations, the following benchmark CMOPs have been selected for this study: the CTP and CF functions to allow for comparison with prior work, the recent MW test suite and a subset of DAS-CMOP with the hard difficulty triplets 9, 10, 11 and 12 and the car-side impact and water problems as existing real-world like problems. They are summarized in Table I.

B. Constraint landscape analysis

The complexity of CMOPs is often discussed using the feasibility ratio, the number of constraints and some descriptive adjectives (e.g. multi-modal, non-linear, rugged, active,...) [9], [16], [25]. The limitations of these means to characterize complexity has already been suggested by various researchers [9], [26]. With C-DTLZ, Jain and Deb [1] proposed a classification scheme for constraints based on the changes they introduce with respect to the unconstrained problem:

- Type-1 constraints introduce an “infeasible barrier” in the objective space, but the Pareto front is not affected.
- Type-2 constraints make a part of the unconstrained Pareto front infeasible.

TABLE I
NUMBER OF OBJECTIVES, SEARCH VARIABLES AND CONSTRAINTS OF THE
SELECTED BENCHMARK CMOPS ALONG WITH THEIR FsR – CALCULATED
AS EXPLAINED IN SECTION IV

	m	n	p	q	FsR
CTP1	2	2	2	0	0.997
CTP2	2	2	1	0	0.990
CTP3	2	2	1	0	0.989
CTP4	2	2	1	0	0.967
CTP5	2	2	1	0	0.989
CTP6	2	2	1	0	0.493
CTP7	2	2	1	0	0.643
CF1	2	10	2	0	0.521
CF2	2	10	2	0	0.994
CF3	2	10	2	0	1.000
CF4	2	10	2	0	0.503
CF5	2	10	2	0	0.512
CF6	2	10	2	0	0.307
CF7	2	10	2	0	0.319
CF8	3	10	2	0	0.004
CF9	3	10	2	0	0.160
CF10	3	10	2	0	0.000
MW1	2	15	1	0	0.000
MW2	2	15	1	0	0.000
MW3	2	15	2	0	0.000
MW4	≥ 3	$12 + m$	1	0	0.000
MW5	2	15	3	0	0.000
MW6	2	15	1	0	0.000
MW7	2	15	2	0	0.000
MW8	≥ 3	$12 + m$	1	0	0.000
MW9	2	15	1	0	0.000
MW10	2	15	3	0	0.000
MW11	2	15	4	0	0.000
MW12	2	15	2	0	0.000
MW13	2	15	2	0	0.002
MW14	≥ 3	$12 + m$	1	0	0.000
DAS-CMOP3_9	2	30	11	0	0.334
DAS-CMOP3_10	2	30	11	0	0.000
DAS-CMOP3_11	2	30	11	0	1.000
DAS-CMOP3_12	2	30	11	0	0.000
DAS-CMOP6_9	2	30	11	0	0.333
DAS-CMOP6_10	2	30	11	0	0.000
DAS-CMOP6_11	2	30	11	0	1.000
DAS-CMOP6_12	2	30	11	0	0.000
DAS-CMOP8_9	3	30	7	0	0.111
DAS-CMOP8_10	3	30	7	0	0.000
DAS-CMOP8_11	3	30	7	0	1.000
DAS-CMOP8_12	3	30	7	0	0.000
Car-side impact	3	7	10	0	0.181
Water	5	3	7	0	0.920

- Type-3 constraints make the full region of the unconstrained Pareto front infeasible and the location of the Pareto front is governed by the constraints.

While, this classification and other similar ones [20], [21] offer insights into the effects of the constraints in the objective space, they do neither quantify these effects nor do they consider the changes due to the constraints in the search space.

In recent work on constrained single-objective continuous and combinatorial optimizations, Malan et al. [26], [27] propose to look at the constraints as defining a “violation landscape” that can be analyzed much in the same fashion as the fitness landscape. This landscape can be described in the search space using the FsR and the ratio of feasible boundary crossing (RFB_×) and in the objective space with the fitness violation correlation (FVC) and the ideal zone (IZ) metrics. The definition of these metrics is given as follows:

- The RFB_× measures the proportion of steps, which imply crossing the feasibility boundary on a progressive random

walk [28] through the search space and quantifies the disjoint nature of the feasible space.

- The FVC is the Spearman’s rank correlation between the fitness and the constraint violation CV (2), measuring the contradiction between the objective and the constraints.
- The IZ quantifies the proportion of points present in the good unconstrained fitness, low violation zone of the fitness-violation plot and represents the likelihood of finding points in that zone.

$$CV(\mathbf{x}) = \sum_{j=1}^p \langle g_j(\mathbf{x}) \rangle + \sum_{k=1}^q |h_k(\mathbf{x})| \quad (2)$$

$$\text{with } \langle \alpha \rangle = \begin{cases} \alpha & \text{if } \alpha > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Malan et al. [26] applied this approach on the constrained single-objective optimization problems of the CEC2010 competition [29] and were able to link the achieved performance of the competing algorithms to the score of these metrics. Thus, they show the potential of this method to better characterize problems and partially address the algorithm selection problem.

While the FsR and the RFB_× can be translated directly to multi-objective optimization (MOO), the FVC and IZ require a single numerical fitness value, which is ambiguous in a multi-objective context. Two alternate metrics adapted to multi-objective problems will be presented in Section IV.

III. ELECTRO-MECHANICAL ACTUATOR DESIGN FRAMEWORK

Electro-mechanical actuators are systems composed of an electric motor and a gear box that are used in numerous applications to rotate other components either in a position-control setup or for motion generation. Their wide application range, the various competing design objectives and the severe constraints make them well adapted to derive realistic benchmark CMOPs.

In this section, the general principles of the parametric design of electro-mechanical actuators are presented along with the derived test suite. While the developed framework (MODAct) resolves accurate domain-specific equations, no domain knowledge is required to use it. The Python source code of MODAct is made available to the community¹ and offers various generic interfaces (Python, C++, MATLAB) to be coupled to any optimizer. The evaluation of one solution (objectives and constraints) takes approximately 20 ms on a regular laptop, making it fast enough for regular benchmarking approaches. Thus, the derived optimization problems could also be of interest to researchers working on parallelization schemes or distributed algorithms.

A. Actuator design overview

In this context, an electro-mechanical actuator is a system composed of a stepper motor, k stages of spur gears (one stage is composed of a pinion and a wheel) and a housing to hold the components, see Fig. 1.

¹<https://github.com/epfl-lamd/modact>

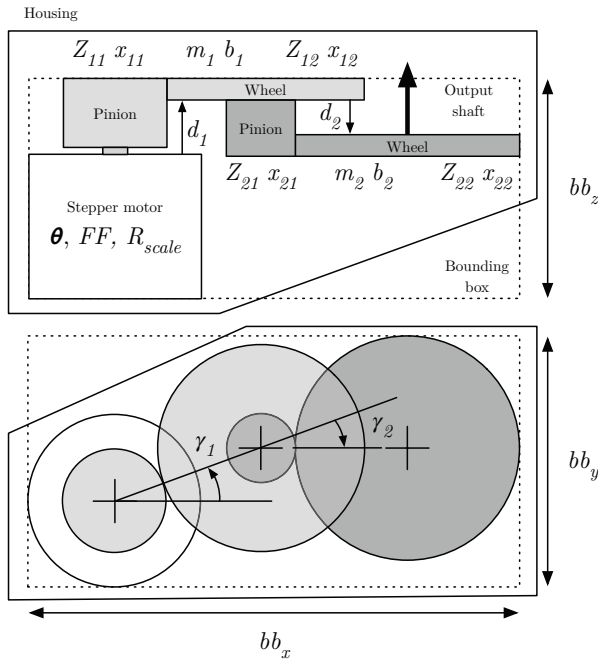


Fig. 1. Schematic representation of a two-stage actuator with its design variables and key dimensions (side and top views)

An actuator is modeled by a chain of $k + 1$ components. Each component is linked to: 1) a physical model that predicts its output speed and torque and component specific constraints, 2) a cost model c_i , and 3) a geometrical model, which is used to create 3D meshes of the components and the assembled system. Objectives and constraints can be formulated on outputs of the physical or cost models or queried on the geometrical models. This modular approach allows to easily extend the framework and build new benchmark functions.

Regarding modeling, the stepper motor's performance are calculated assuming steady-state operation [30, eq. (5.14)] and the parameters defining the motor θ can be chosen among five possible parameter combinations from existing commercial two-phase stepper motors. The stator coil windings can be further adjusted by scaling the fill factor FF or the resistance R_{scale} . All motors are represented by a cylindrical mesh with the diameter and height corresponding to their actual dimensions. Their cost is composed of two parts: a fixed contribution and a variable part adjusted according to the selected windings.

The gears are modeled as standard ISO steel gears, characterized by their number of teeth $Z_{i\{1,2\}}$, profile shift $x_{i\{1,2\}}$, module m_i and thickness b_i . Dimensions, kinematics and mechanical stresses are calculated using the ISO norms relative to spur gears [31]–[33]. A cylindrical mesh is used to represent the gears in 3D. The cost is estimated based on the volume of the gears.

The rotational speed and torque acting on each component are calculated sequentially starting with the motor. The energy flow between components is considered to be perfect (no losses, perfectly rigid connections). For a given input condition (supply voltage U_m , maximum current I_{max} and desired

output rotational speed ω), it is possible to calculate the torque T at the output. A set of supply voltage, current, desired speed and torque forms an operating point. The deviation from the desired torque at the output is called the torque excess.

The 3D position of the gears is specified by two decision variables for each gear stage: a translation d_i along the shaft and an angle γ_i between pinion and wheel (see Fig. 1). The meshes are generated and handled by Trimesh [34]. The combination of all meshes is then used to detect inter-component collisions and to calculate the size of the bounding box of the system. Finally, its convex hull serves as a housing and assuming a fixed wall thickness, the required material cost $c_{housing}$ is added to the total cost.

Using this modeling approach, the design goal consists in finding a set of suitable motors and gears operating at specified operating points, while respecting a set of constraints such as mechanical integrity of the gears, proper spatial configuration or limited space requirements.

The search space corresponds to the combination of the design space of each component (motor, spur gear 1, spur gear 2,...). Both the spur gears and the motors have integer decision variables, but, in order for MODAct problems to be solvable by a larger set of algorithms, these integer variables are combined with related continuous variables into a single number where the integer part codes the integer variable and the fractional part is mapped to the other variable:

- the motor selection variable and the fill factor of the coils FF form the variable mFF ;
- the number of teeth $Z_{i\{1,2\}}$ of a gear and its profile shift $x_{i\{1,2\}}$ form the variable $Zx_{i\{1,2\}}$.

With this transformation, the search space is continuous and has a scalable size of $n = 2 + 6k$.

B. Selected problems

Using the presented framework, a set of 20 problems is built with objectives and constraints directly taken from an industrial application studied by the authors. All problems aim at finding actuators with $k = 3$ stages that have their performance tested at two operating points, Table II. The design problems are constructed around a set of up to five objectives, associated with a capital letter:

- minimize total cost (C)

$$\min \sum_{i=0}^k c_i + c_{housing} \quad (4)$$

- maximize minimum torque excess for each considered operating point j (T)

$$\max \min_{j \in \{1,2\}} \Delta T_j \quad (5)$$

with $\Delta T_j = T_j - T_{desired,j}$

- maximize harmonic mean of the safety factors to bending S_F and to pitting S_H for all gears (S)

$$\max \left(\frac{\sum_{i=1}^k S_{F,i}^{-1} + S_{H,i}^{-1}}{k - 1} \right)^{-1} \quad (6)$$

TABLE II
OPERATING POINT REQUIREMENTS FOR ALL MODACT PROBLEMS

#	Rotational speed	Desired torque	Voltage	Max current
1	1.35 rad/s	0.6 N · m	9 V	2 A
2	0.3 rad/s	1 N · m	12 V	2 A

- maximize electrical to mechanical energy conversion efficiency (E)

$$\max \min_{j \in \{1,2\}} \frac{\omega_j T_j}{U_m I_j} \quad (7)$$

- minimize transmission ratio (I)

$$\min \prod_{i=1}^k \frac{Z_{i,2}}{Z_{i,1}} \quad (8)$$

Based on combinations of these objectives, five problem classes are devised: CS, CT, CTS, CTSE, CTSEI. In practice, cost is essential for such products, while other objectives depend upon the application. For example, the analysis of the trade-off of available torque is important for designing different variants of a series of actuators and the minimization of the transmission ratio is related to the level of emitted noise.

Further, the framework considers a total of 11 constraints which can be added to these problem classes. The implemented constraints are very common for the considered applications and objectives, and are grouped into five general categories:

- gear constraints (sufficient contact ratio, limited sliding speeds, no interference, sufficient mechanical strength in bending and pitting)
- minimum torque excess over all operating points
- upper limits on bounding box dimensions ($bb_y \leq 50$ mm and $bb_z \leq 35$ mm)
- distance from the output shaft to a desired coordinate within 5 mm

The constraints are grouped to form four levels of constraint complexity (1, 2, 3 and 4), which are useful at different stages in the development process of an actuator (early stage: levels 1 and 2) or representing different application specific constraints (levels 3 or 4). Each level can then be combined with any of the problem classes (combination of objectives). This enables the generation of 20 benchmark problems, which are named using the names of the objectives and appending the number of the constraint level. For example, selecting the class CTS with the constraint level 2 forms the problem CTS2. The 20 instances and their constraints are summarized in Table III along with their dimensionality and the search space definition. It is noted that all constraints are independent of the selected objectives, except for the minimum torque excess constraint, for which two different requirements exist.

C. Creating variants of the proposed MODAct instances

Since MODAct is a framework, it can generate many other benchmark problems. An interesting first step would be to change the number of gear stages k . This not only increases the size of the search space, but also affects the constraints,

e.g. it becomes harder to fit all gears within the same bounding box constraint. The specified operating points can be adjusted. In addition, constraints can be changed – e.g. other bounding box geometries, or different output shaft coordinates – and new trade-offs can be explored – e.g. safety factor and efficiency. All these changes require minimal to no domain knowledge and allow to significantly change the benchmark problems to generate easier but also more difficult problems.

IV. MULTI-OBJECTIVE CONSTRAINT LANDSCAPE ANALYSIS

Calculating the objectives and constraints of such an actuator involves several steps. It is therefore not straightforward to predict their mathematical model characteristics. Thus, MODAct problems need to be analyzed with metrics characterizing the effect of the constraints on both the search and objective spaces.

This section presents the metrics for the constraint landscape analysis introduced by [26], as well as additional methods suited for multi-objective problems. The analysis relies on both repeated independent uniform samplings and progressive random walks [28] of the decision space.

Definition 1. Given an independent uniform sampling $\mathcal{U} \subset \mathcal{S}$, the FsR is defined as:

$$\text{FsR} := \frac{|\{\mathbf{u} \in \mathcal{U} \mid \text{feasible}(\mathbf{u})\}|}{|\mathcal{U}|} \quad (9)$$

where $\text{feasible}(\mathbf{x})$ is an indicator function indicating if a solution \mathbf{x} is feasible, i.e. $CV(\mathbf{x}) = 0$.

Definition 2. Given a sequence of s samples \mathcal{W} generated by a random walk of $s - 1$ steps

$\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_s\}$, the RFB_\times is defined as:

$$\text{RFB}_\times := \frac{1}{s-1} \sum_{i=1}^{s-1} \chi(i) \quad (10)$$

$$\chi(i) = \begin{cases} 0 & \text{if } \text{feasible}(\mathbf{w}_i) = \text{feasible}(\mathbf{w}_{i+1}) \\ 1 & \text{otherwise} \end{cases} \quad (11)$$

where the helper function χ indicates when the feasibility boundary is crossed.

The RFB_\times metric should be high for a disjoint feasible space and low for a contiguous feasible space. Yet, the possible values of RFB_\times also depend on the number of feasible points s_f encountered during a walk \mathcal{W} , thus making the comparison between functions difficult. In order to compare, one would need to know, given the ratio of encountered feasible points per walk, how disjoint the space is. In other words, one wants to identify the maximum possible RFB_\times and define this as an upper bound for a given ratio s_f/s . In general, this is equivalent to trying to spread out the feasible or infeasible points (depending on which ones are the minority) in a sequence to maximize the number of transitions. Following this approach, it can be shown that the upper bound of RFB_\times for a given walk is:

$$\text{RFB}_{\times, \max} = \frac{2}{s-1} \cdot \min\{s_f, s - s_f, \frac{s-1}{2}\} \quad (12)$$

TABLE III
SUMMARY OF THE 20 FUNCTIONS OF MODACT INCLUDING DETAILS ABOUT THEIR CONSTRAINTS AND THE SEARCH SPACE

Constraint level	n	m	p	q	Gear constraints (6)	Min. torque excess (1)	3D Collisions (1)	Bounding box (2)	Output shaft (1)	Search space bounds	
										$[mFF, R_{scale}$ $Zx_{11}, Zx_{12}, m_1, b_1, d_1, \gamma_1,$ $Zx_{21}, Zx_{22}, m_2, b_2, d_2, \gamma_2,$ $Zx_{31}, Zx_{32}, m_3, b_3, d_3, \gamma_3]$	
CS1	1	20	2	7	0	Yes	≥ -0.001	No	No	No	$\mathbf{x}^{(L)} = [0, 0.3,$ $9, 30, 0.3, 5, -20, -\pi,$ $9, 30, 0.3, 5, -20, -\pi,$ $9, 30, 0.3, 5, -20, -\pi]$
CS2	2	20	2	8	0	Yes	≥ -0.001	Yes	No	No	
CS3	3	20	2	10	0	Yes	≥ -0.001	Yes	Yes	No	
CS4	4	20	2	9	0	Yes	≥ -0.001	Yes	No	Yes	
CT1, CTS1, CTSE1, CTSE11	1	20	2, 3, 4, 5	7	0	Yes	≥ -0.599	No	No	No	$\mathbf{x}^{(U)} = [5 - 10^{-6}, 2,$ $41 - 10^{-6}, 81 - 10^{-6}, 1, 15, 20, \pi,$ $41 - 10^{-6}, 81 - 10^{-6}, 1, 15, 20, \pi,$ $41 - 10^{-6}, 81 - 10^{-6}, 1, 15, 20, \pi]$
CT2, CTS2, CTSE2, CTSE12	2	20	2, 3, 4, 5	8	0	Yes	≥ -0.599	Yes	No	No	
CT3, CTS3, CTSE3, CTSE13	3	20	2, 3, 4, 5	10	0	Yes	≥ -0.599	Yes	Yes	No	
CT4, CTS4, CTSE4, CTSE14	4	20	2, 3, 4, 5	9	0	Yes	≥ -0.599	Yes	No	Yes	

This upper bound is used to define the normalized ratio of feasible boundary crossing (nRFB_×):

$$\text{nRFB}_{\times} = \begin{cases} 0 & \text{if RFB}_{\times} = 0 \\ \frac{\text{RFB}_{\times}}{\text{RFB}_{\times, \max}} & \text{otherwise} \end{cases} \quad (13)$$

An nRFB_× value of 1 is obtained for walks in the search space that are as disjoint as possible given their s_f/s ratio.

For the analysis of the objective space, two new metrics are introduced: PFd and PFcv. The main idea is to capture the interactions between the image in the objective space of randomly selected points in the search space and the Pareto fronts. Both metrics rely on the previously obtained aggregated set of samples $\mathcal{L} = \mathcal{U}_1 \cup \dots \cup \mathcal{W}_1 \cup \dots$ and the Pareto front \mathcal{PF}^* of the problem.

PFd is constraint independent and represents the average minimal distance from the Pareto front to the cloud of points formed by $f(\mathcal{L})$. It measures the ease of randomly generating points near the Pareto front. In order to make it comparable between functions, the points in the objective space are normalized using the ideal z^* and nadir z^{nadir} vectors of \mathcal{PF}^* , leading to the normalized objective function f^n and the normalized Pareto front \mathcal{PF}_n^* . PFd corresponds to the inverted generational distance (IGD) [35] between \mathcal{PF}_n^* and $f^n(\mathcal{L})$.

Definition 3. Given a set of samples \mathcal{L} and their normalized image $f^n(\mathcal{L})$ and given the normalized Pareto front \mathcal{PF}_n^* , PFd is defined as:

$$\text{PFd} := \frac{1}{|\mathcal{PF}_n^*|} \sum_{z \in \mathcal{PF}_n^*} \min_{l \in f^n(\mathcal{L})} \|z - l\|_2 \quad (14)$$

where $\|\cdot\|_2$ is the Euclidian norm.

PFcv measures the average constraint violation CV value of neighbors of the Pareto front in the sample set. This represents the sensitivity in terms of constraints of solutions on the Pareto front.

Definition 4. Given a set of samples \mathcal{L} , the Pareto front \mathcal{PF}^* and K the number of neighbors to consider, PFcv is defined as:

$$\text{PFcv} := \frac{1}{CV_{95}|\mathcal{PF}^*|} \sum_{z \in \mathcal{PF}^*} \frac{1}{K} \sum_{x \in \mathcal{B}(z, K)} CV(x) \quad (15)$$

where CV_{95} is the 95th percentile of the CV values found in \mathcal{L} and $\mathcal{B}(z, K)$ is the set of the K closest neighbors of z in \mathcal{L} .

V. METHODS OF THE NUMERICAL INVESTIGATIONS

MODAct problems are compared to five groups of benchmark problems identified from literature in Section II: the CTP, the CF, the MW and some DAS-CMOP functions and two real-world like problems (water and car-side impact). The comparison is done based on the presented constraint landscape analysis approach and on a convergence study. The next sections present the parameters used for the various steps.

A. Constraint landscape analysis

Both uniform samplings and progressive random walks have been performed using the parameters mentioned in [26]: 30 independent uniform sampling \mathcal{U} of $|\mathcal{U}| = 1000n$ points and 30n independent progressive random walks \mathcal{W} of 1000 steps ($|\mathcal{W}| = s = 1001$) each with maximum step size of 1% of the decision space. The reported FsR, RFB_× and nRFB_× scores are obtained by averaging over all independent samplings.

The PFd and PFcv metrics are calculated with the best-known Pareto front obtained from the convergence study when the true Pareto front is unknown. The $K = 20$ closest neighbors are considered for PFcv.

B. Convergence study

In addition to these characteristics, the main interest is to investigate how well these design optimization problems can be solved and understand how the different constraint levels influence convergence. This is achieved by performing a convergence study where the problems are compared among each other and against two kinds of optimizers: the commonly

TABLE IV
PARAMETERS USED TO CONFIGURE EACH RUN OF NSGA-II, NSGA-III
AND C-TAEA

Parameter	NSGA-II/III	C-TAEA
Population size μ	200	210 or 220 ($m = 4$)
Number of function (i.e. solution) evaluations	300 000	
Mutation η_m	20	
Mutation rate	$1/n$	
Crossover η_c	15	
Crossover probability (CXPB)	0.9	

used algorithms NSGA-II [10] and NSGA-III [11] on one side and C-TAEA [12] a recent algorithm developed specifically to tackle CMOPs on the other side. NSGA-II and NSGA-III follow a feasibility first approach and rely in the survival step on the constrained-dominance principle (CDP) introduced in [10]:

Definition 5. Given two solutions \mathbf{x}_1 and \mathbf{x}_2 , \mathbf{x}_1 is said to constrained-dominate \mathbf{x}_2 , if one of the following is true:

- 1) \mathbf{x}_1 is feasible and \mathbf{x}_2 is not;
- 2) \mathbf{x}_1 and \mathbf{x}_2 are infeasible and $CV(\mathbf{x}_1) < CV(\mathbf{x}_2)$;
- 3) \mathbf{x}_1 and \mathbf{x}_2 are feasible and \mathbf{x}_1 (Pareto-)dominates \mathbf{x}_2 .

C-TAEA maintains two separate archives – one for diversity, the other for convergence – and has a special restricted mating approach to balance between the two.

The study is performed using pymoo [36]. NSGA-II is used for all problems with $m = 2$ and NSGA-III for all problems with more objectives. Each optimization is performed 30 times with the parameters specified in Table IV, following common practice for these algorithms. The number of reference directions for NSGA-III is chosen as close to the population size, while following Das and Dennis’s approach [37]. The same approach is used for C-TAEA, except the population size and number of reference directions are set to the same value. An unbounded external archive (UEA) is added to the algorithms, as recommended in [38], to collect all feasible non-dominated solutions along the optimization.

For MODAct problems, the solutions of the archives from all runs are aggregated and sorted to determine the best-known Pareto front. The estimated ideal \mathbf{z}^* and nadir \mathbf{z}^{nad} points are collected to provide per-problem front normalization.

The convergence and diversity are evaluated with the hypervolume indicator [39]. In particular, the exact and fast implementations by the Walking-Fish Group are used [40], [41]. Since the objective functions of the various problems have different scales and since there is a mix of minimization and maximization objectives, the Pareto front and the archives are transformed into minimization only problems and normalized with the corresponding \mathbf{z}^* and \mathbf{z}^{nad} . Only then the hypervolume is calculated using a common reference point $\mathbf{r} = (1.1, \dots, 1.1)^T$. Finally, the comparison is made through the relative hypervolume error ΔHV_n (16) with respect to the best-known Pareto front:

$$\Delta HV_n = \frac{HV(\mathcal{PF}_{m,n}^*, \mathbf{r}) - HV(\mathcal{A}_{m,n}, \mathbf{r})}{HV(\mathcal{PF}_{m,n}^*, \mathbf{r})} \quad (16)$$

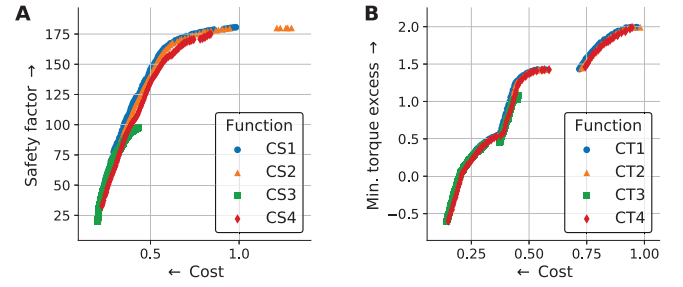


Fig. 2. Best-known Pareto fronts for (A) CS and (B) CT problems

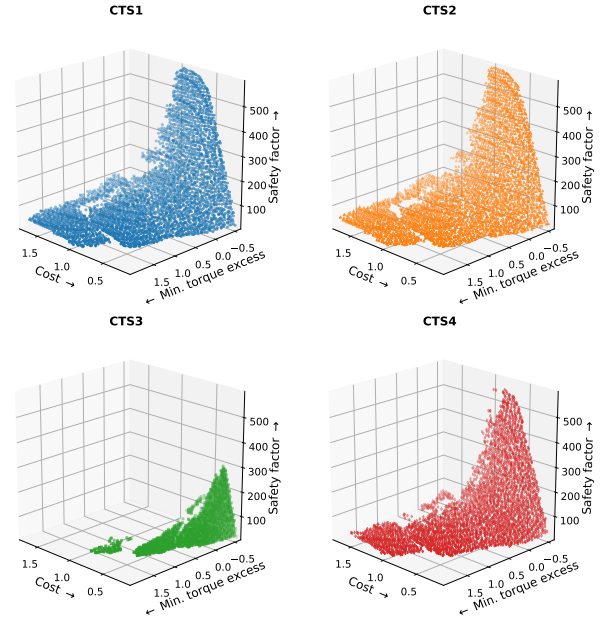


Fig. 3. Best-known Pareto fronts for CTS problems

where \mathcal{A} denotes a given external archive and the subscript m that the problems have been converted to minimization.

For the statistical difference between optimization algorithms, the non-parametric Wilcoxon rank-sum test with the null hypothesis that all algorithms are equal is applied with a confidence interval of 99%.

VI. RESULTS AND DISCUSSION

A. Design trade-offs and constraints

The analysis of MODAct problems starts by looking at some of the best-known Pareto fronts. All best-known Pareto fronts can be downloaded [42]. Fig. 2 shows the best-known Pareto fronts for CS and CT problems, while Fig. 3 represents the CTS problems. The best-known Pareto fronts of CS problems consist of a smooth convex segment, while CT problems have several step-like disconnected segments. CTS problems combine the two features to form a complex surface with concave/convex and disconnected parts.

The discontinuities are indeed expected for these design problems since some variables represent discrete physical choices. In particular, the motor selection is dominating these effects. With different cost and power ranges, the motor

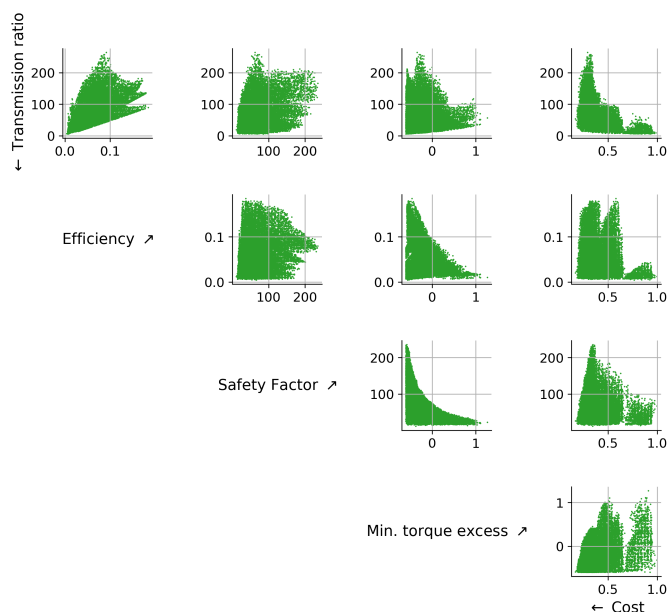


Fig. 4. Best-known Pareto front for CTSEI3, for rendering questions, the numbers of points displayed has been reduced by eliminating points that were too close to each other

designs do not overlap, leading to the visible steps. This feature is common in design problems and should therefore also appear in benchmark problems. For engineers, this is of particular interest since these steps imply important design trade-offs.

Looking more closely at the effects of the constraints, it can be noticed, that constraint levels 2 and 4 slightly shift the found fronts towards higher costs compared to level 1. Constraint level 3 is the most restrictive: the limit on the size of the bounding box of the actuator does not allow large gears or motors, which significantly reduces the available options. While the impact is limited in the objective space, this is not the case in the decision space. An analysis of neighbors in the objective space between CT1 and CT2 confirms important differences, mostly on the number of teeth of the wheels Z_{i2} and the spatial positioning variables d_i and γ_i .

For problems with more objectives, the best-known Pareto front of CTSEI3 is shown as an example in Fig. 4 projected into the various objective planes. It shows that there are indeed up to five competing objectives with different shapes.

B. Convergence analysis

In a second step, the ease of convergence and the repeatability are compared. Fig. 5 represents box plots of the obtained relative hypervolume errors (16) over the 30 optimization runs for each problem by NSGA-II/III and by C-TAEA. Starting with MODAct problems, the results suggest the following:

- 1) The proposed instances pose a wide range of optimization challenges to NSGA-II/III and C-TAEA.
- 2) Within the same classes, constraint levels 1 and 2 are generally equally well solved with rare outliers that may exhibit early convergence to local optima.

- 3) Constraint levels 3 and 4 are increasingly difficult and the various optimization runs achieve very different levels of convergence.
- 4) There is also a clear trend of increasing hypervolume error with the number of objectives, but this is certainly related to the nature of the indicator itself.
- 5) The more restrictive threshold on the minimum torque excess of CS problems seems to negatively impact convergence for all constraint levels. In particular, C-TAEA is barely able to find any solutions within the boundaries of the reference point for CS3 and CS4.

In comparison, the CTP, car-side impact, water, most MW and DAS-CMOP8 problems are effectively solved by the considered algorithms. MW10, MW11 and MW13 are the most difficult of MW and can be compared to MODAct constraint levels 1 and 2. The CF family offers a broader range of challenging problems. The findings that the biobjective problems CF3, CF5 and CF7 are the hardest are consistent with the outcomes of the CEC2009 MOEA Competition [43]. Among the three-objective problems, CF8 and CF10 are the most challenging in particular for NSGA-III. The DAS-CMOP3 problems seem to be challenging for both algorithms. DAS-CMOP6 problems are effectively solved by C-TAEA, while NSGA-II struggles for DAS-CMOP6_11 and DAS-CMOP6_12.

In terms of algorithms, NSGA-II/III is overall significantly better than C-TAEA (better on 30 problems, no difference on 18 and worse on 17), despite using a simple constraint handling strategy. NSGA-II/III is always better for CTP and for the car-side impact problem, although the difference is minor. The results are more balanced for the other benchmark problems. For the MW test suite, C-TAEA performs better on the difficult problems. C-TAEA also performs better on the three-objective CF problems and is on a par for biobjective problems. The opposite is true for DAS-CMOP: NSGA-III is better for three-objective problems, while C-TAEA shows solid performance on biobjective instances. Finally on MODAct problems NSGA-II/III has a clear advantage for biobjective problems and is at least as good as C-TAEA otherwise.

In general, the performance of NSGA-II/III and C-TAEA is insufficient on MODAct instances with constraint levels 3 and 4, although they represent very common and simple mechanical design problems. The observed large variance of the optimization outcomes has important practical consequences. Considering problems CS3 and CS4, more than 75% of the optimization runs of NSGA-II obtain approximate Pareto fronts with a hypervolume of 50% or smaller than the best-known Pareto fronts, while C-TAEA fails to find interesting solutions. As an example, the best run of NSGA-II and of C-TAEA along with two optimization runs are compared to the best-known Pareto front for problem CS4 in Fig. 6 to illustrate the large difference in the proposed solutions with a lack of convergence and diversity. Using partially converged solutions for decision-making can lead to significantly different engineering outcomes.

In order to better understand the optimization process of MODAct problems, the evolution of the relative hypervolume error of the external archives of NSGA-II, C-TAEA

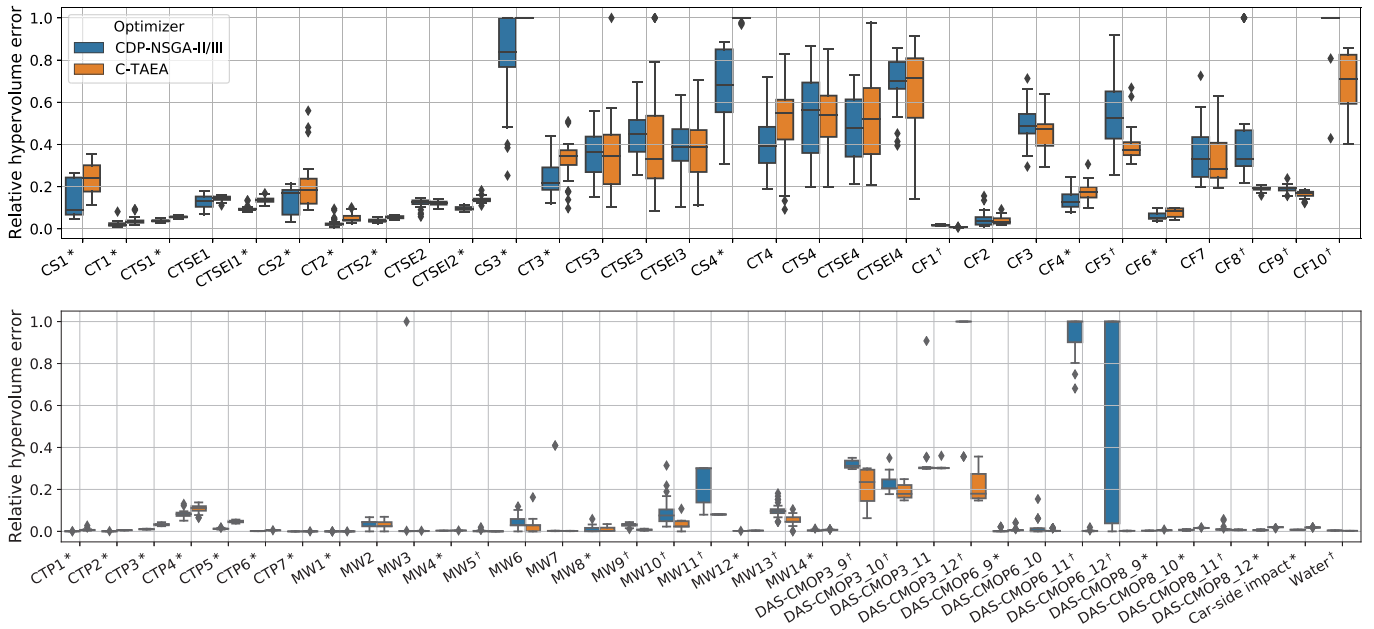


Fig. 5. Box plot of the relative hypervolume errors ΔHV_n of the external archives obtained for each problem by NSGA-II/NSGA-III versus C-TAEA, where the * or † after the function name is used to indicate that C-TAEA is, respectively, significantly worse or better than NSGA-II/NSGA-III

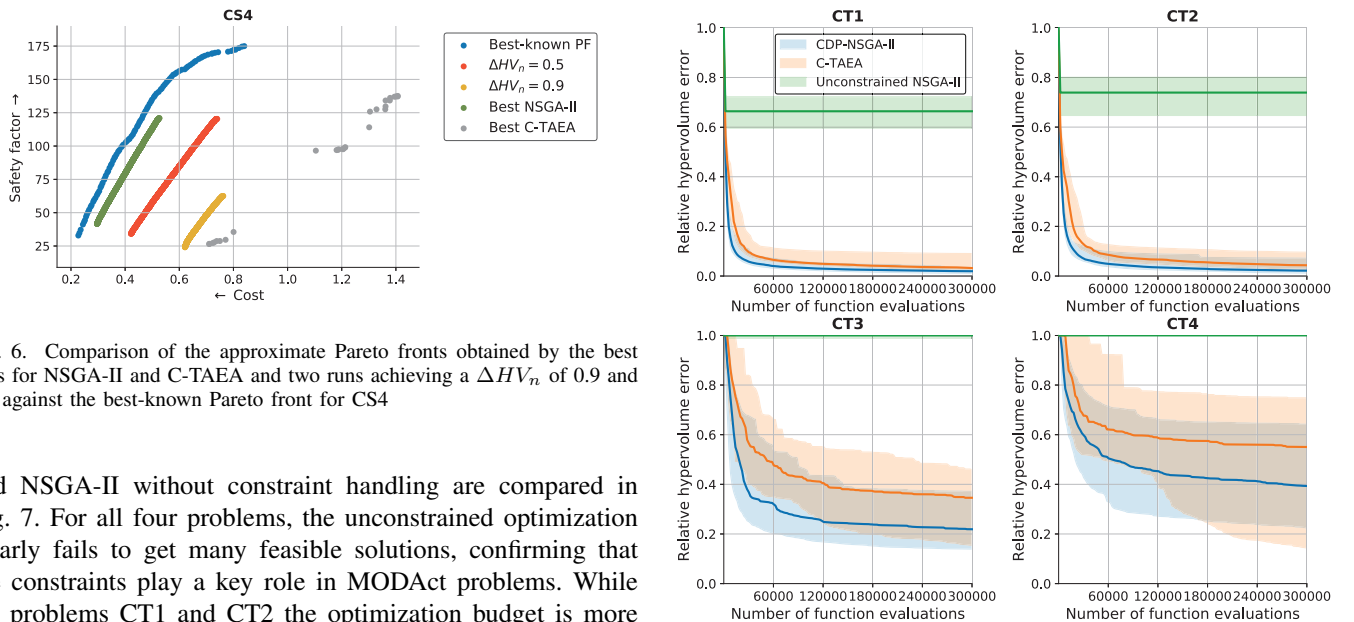


Fig. 6. Comparison of the approximate Pareto fronts obtained by the best runs for NSGA-II and C-TAEA and two runs achieving a ΔHV_n of 0.9 and 0.5 against the best-known Pareto front for CS4

and NSGA-II without constraint handling are compared in Fig. 7. For all four problems, the unconstrained optimization clearly fails to get many feasible solutions, confirming that the constraints play a key role in MODAct problems. While for problems CT1 and CT2 the optimization budget is more than sufficient, better results may be possible for CT3 and CT4. Yet, while the mean relative hypervolume error seems to be decreasing, the spread remains almost constant (NSGA-II) or increasing (C-TAEA). Running the optimization algorithms longer might thus not necessarily address the repeatability issue. It is also interesting to note that while CT3 and CT4 are challenging problems, feasible solutions are rapidly found: on average after 800 evaluated solutions for CT3, and after 2000 for CT4.

C. Link between convergence and constraints

Finally, the objective is to use the results from the constraint landscape analysis to identify the underlying characteristics

Fig. 7. Evolution of the median relative hypervolume error including 5th and 95th percentiles for CT problems comparing: NSGA-II with the constrained-dominance strategy (CDP-NSGA-II), C-TAEA and NSGA-II discarding constraints (Unconstrained NSGA-II)

that affect convergence most and how the newly introduced problems differ from the existing benchmarks functions.

The obtained metrics for all problems are summarized in Table V. To begin, the advantages of nRFB_× are evaluated. CF1, CTP6 and CTP7 have the three highest RFB_× scores. They also have an FsR close to 0.5. For problems with a high or a low FsR, the definition of RFB_× necessarily decreases its possible values, thus masking the level of discontinuity

TABLE V
CALCULATED METRICS RESULTING FROM THE CONSTRAINT LANDSCAPE ANALYSIS

Function	FsR	nRFB _×	nRFB _×	PfD	PfCv
CS1	0.0022	0.0016	0.1155	0.0015	0.4158
CT1	0.0342	0.0278	0.3081	0.0071	1.7115
CTS1	0.0343	0.0281	0.3095	0.0788	0.8317
CTSE1	0.0342	0.0283	0.3050	0.1235	0.7521
CTSE11	0.0347	0.0287	0.3075	0.1369	1.2519
CS2	0.0010	0.0007	0.0593	0.0021	0.4703
CT2	0.0160	0.0156	0.3097	0.0068	1.6274
CTS2	0.0158	0.0158	0.2947	0.0796	0.7688
CTSE2	0.0160	0.0161	0.3075	0.1242	0.7226
CTSE12	0.0162	0.0161	0.3110	0.1411	1.1255
CS3	0.0000	0.0000	0.0000	0.0012	0.6763
CT3	0.0000	0.0001	0.0126	0.0029	2.2518
CTS3	0.0000	0.0001	0.0112	0.0551	0.9973
CTSE3	0.0000	0.0001	0.0178	0.0643	1.5531
CTSE13	0.0000	0.0001	0.0259	0.0336	8.4287
CS4	0.0000	0.0000	0.0000	0.0013	0.7802
CT4	0.0000	0.0000	0.0000	0.0039	0.7694
CTS4	0.0000	0.0000	0.0000	0.0483	0.7807
CTSE4	0.0000	0.0000	0.0000	0.0569	0.9021
CTSE14	0.0000	0.0000	0.0000	0.0277	1.6211
CF1	0.5199	0.1539	0.2007	0.0301	0.7801
CF2	0.9942	0.0033	0.1682	0.1114	0.0116
CF3	1.0000	0.0000	0.0000	1.9014	0.0000
CF4	0.5001	0.0154	0.0169	0.3835	0.1432
CF5	0.5068	0.0080	0.0090	2.8356	0.0574
CF6	0.3056	0.0089	0.0222	0.4670	0.0306
CF7	0.3151	0.0169	0.0325	2.5682	0.0896
CF8	0.0042	0.0012	0.0427	0.7449	0.1984
CF9	0.1597	0.0340	0.1259	0.3445	0.2047
CF10	0.0001	0.0001	0.0298	2.8237	0.1391
CTP1	0.9972	0.0048	0.7769	0.0429	0.0001
CTP2	0.9899	0.0100	0.4902	0.0505	0.0771
CTP3	0.9892	0.0100	0.4659	0.0481	0.0653
CTP4	0.9656	0.0287	0.4402	0.0363	0.5128
CTP5	0.9893	0.0099	0.4695	0.0509	0.0117
CTP6	0.4888	0.4108	0.4208	0.0206	0.0133
CTP7	0.6458	0.4171	0.5856	0.0220	0.0222
DAS-CMOP3_9	0.3337	0.0999	0.1496	0.7515	0.4267
DAS-CMOP3_10	0.0000	0.0024	0.0795	0.0241	0.0002
DAS-CMOP3_11	0.9996	0.0137	0.1214	0.5173	0.1614
DAS-CMOP3_12	0.0000	0.0001	0.0177	0.0138	0.0065
DAS-CMOP6_9	0.3330	0.1000	0.1501	23.7690	0.3444
DAS-CMOP6_10	0.0000	0.0000	0.0000	17.5318	0.1394
DAS-CMOP6_11	1.0000	0.0000	0.0000	14.7336	0.0000
DAS-CMOP6_12	0.0000	0.0000	0.0000	22.9593	0.1433
DAS-CMOP8_9	0.1114	0.0614	0.2835	22.0217	0.6104
DAS-CMOP8_10	0.0000	0.0000	0.0000	21.0716	0.1383
DAS-CMOP8_11	1.0000	0.0000	0.0000	22.1774	0.0000
DAS-CMOP8_12	0.0000	0.0000	0.0000	21.3521	0.1358
MW1	0.0000	0.0000	0.0000	8.9253	0.5888
MW2	0.0000	0.0000	0.0000	0.6848	0.0549
MW3	0.0000	0.0000	0.0000	0.5878	0.0704
MW4	0.0000	0.0000	0.0000	4.3189	0.6268
MW5	0.0000	0.0000	0.0000	7.8080	0.3784
MW6	0.0000	0.0000	0.0000	0.6881	0.0079
MW7	0.0000	0.0000	0.0000	0.4555	0.0178
MW8	0.0000	0.0000	0.0000	0.8002	0.0080
MW9	0.0000	0.0000	0.0000	4.2789	0.0246
MW10	0.0000	0.0000	0.0000	0.7595	0.0003
MW11	0.0000	0.0000	0.0017	0.1222	0.0000
MW12	0.0000	0.0000	0.0000	4.4956	0.3126
MW13	0.0018	0.0029	0.1765	0.0587	0.0000
MW14	0.0003	0.0002	0.0391	0.1815	0.0218
Car-side impact	0.1827	0.0048	0.0284	0.1054	0.0281
Water	0.9198	0.0058	0.0612	0.0505	0.0012

of small infeasible or feasible spaces. In such cases, the nRFB_× metric acts as an amplifier of the scores through the normalization. The nRFB_× values indicate that all CTP problems have a relatively disjoint search space, which is known a priori from their definition [17]. MODAct problems have nRFB_× values ranging between the CTP and the CF problems, suggesting a rather disjoint search space. The same applies to DAS-CMOP problems with difficulty triplet 9, DAS-CMOP3_11 and MW13. nRFB_× values can also remain low such as for the car-side impact and water problems, suggesting a contiguous feasible space. It is noted that some limitations remain, in particular for highly feasible or infeasible search spaces, where boundaries are hard to find. This is the case for the rest of DAS-CMOP and MW.

In order to get a better overview of the differences between functions and function groups, each problem is shown in the

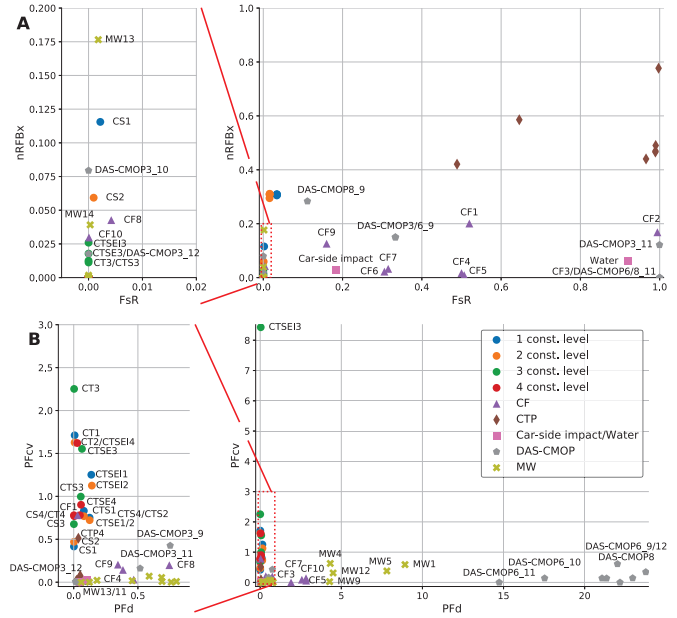


Fig. 8. Map of the problems based on their (A) FsR-nRFB_× and (B) PfD-PfCv scores grouped by constraint levels or test suites

FsR-nRFB_× plane, Fig. 8(A), and in the PfD-PfCv plane, Fig. 8(B). The FsR-nRFB_× plane confirms the initial analysis of Table V. In addition, the following points are highlighted:

- 1) MODAct, MW, DAS-CMOP with triplets 10 and 12, CF8 and CF10 are all located near FsR = 0.
- 2) The feasibility ratio of MODAct problems is decreasing with increasing constraint level.
- 3) The CTP family has a unique signature with high FsR and high nRFB_×.

Yet, there is no clear relationship between these metrics and convergence. In particular, the MW test suite is a good example that very low FsR is not directly related to convergence challenges.

Analyzing Fig. 8(B), the newly introduced metrics exhibit the following problem clusters:

- 1) MODAct problems in the low PfD, high PfCv zone;
- 2) three groups with high PfD: (i) DAS-CMOP6 and DAS-CMOP8, (ii) MW1, MW4, MW5, MW9 and MW12 and (iii) CF3, CF5, CF7 and CF10;
- 3) a large group with low PfD and low PfCv, including most CTP, the car-side impact and water problems, MW11, MW13 and DAS-CMOP3_12.

Further, for the problem class CS, there is a relation between increasing PfCv values and decreasing convergence as shown in Fig. 5. Other MODAct problems do not exhibit the same trend across constraint levels. The difficult CF functions (CF3, CF5, CF7 and CF10) have a high PfD. The opposite is true for the most difficult DAS-CMOP and MW problems. Hence, these metrics seem to capture different optimization challenges, but certainly not all of them.

In order to better understand the qualitative difference between PfD and PfCv scores and convergence, four particular functions – CT3 (low PfD, high PfCv), CF7 (high PfD, low

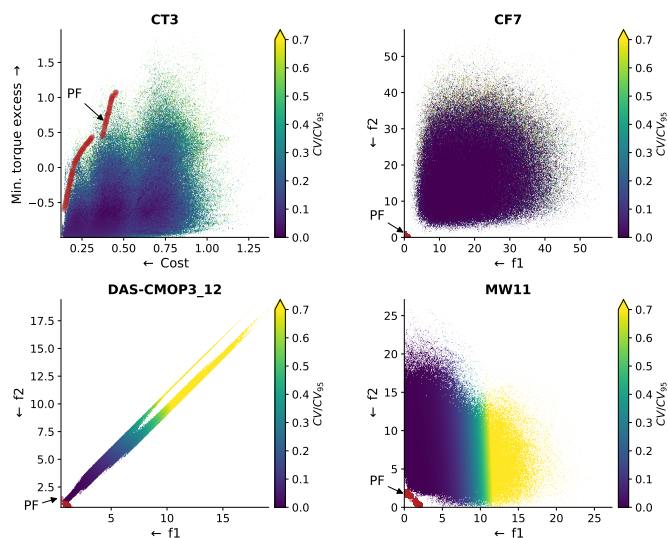


Fig. 9. Color-maps of the normalized constraint violation values as a function of the objective values based on the random samples in \mathcal{L} for CT3, CF7, DAS-CMOP3_12 and MW11 with their respective (best-known or true) Pareto front in red dots. For the generation of the color-map, the constraint violation values of points very close to each others in the objective space are aggregated using the lowest value

PFcv), DAS-CMOP3_12 (low PFd, low PFcv) and MW11 (low PFd, low PFcv) – are further analyzed. Fig 9 shows color-maps of the minimum constraint violation values in the objective space based on the samples from \mathcal{L} along with the respective best-known (CT3) or true Pareto front. It becomes apparent that PFd measures the ease of randomly sampling points in the vicinity of the Pareto front and PFcv the constraint violation of the closest known points. These particular problems represent four different situations:

- 1) While it is possible to find samples near the best-known Pareto front for CT3, it is harder to find feasible solutions.
- 2) For CF7, the sampling near the Pareto front is challenging, but it seems easier to find feasible solutions scattered all over the objective space, in particular in the direction of the Pareto front.
- 3) Random samples of DAS-CMOP3_12 are located in a narrow band reaching to the Pareto front and some constraints clearly guide the search process towards the optimum. Finding points outside this band is a clear challenge for diversity.
- 4) More uniformly distributed feasible samples near the Pareto front can be found for MW11 and here as well, some constraints clearly guide in the right direction.

The optimization challenges of the four situations are different and might thus require a different set of tools to be addressed efficiently. CF7 might, for example, be better solved by specific search operators regardless of constraints, while a specific constraint handling strategy or diversity generating operators would be needed for the others. And indeed, C-TAEA was better at solving DAS-CMOP3_12 and MW11.

Finally, an additional specificity of MODAct problems is their high number of constraints. Fig. 10(A) represents the relative share of solutions from the random samples \mathcal{L} grouped

by number of simultaneously violated constraints for all the investigated problems with more than two constraints. The results suggest that MODAct problems are heavily constrained and have at least 50% of the search space where at least two constraints are simultaneously violated, or even at least three for the more challenging problems. For these levels, up to five constraints can be violated simultaneously in a non negligible part of the search space.

With that respect, the car-side impact problem is comparable to problems with constraint level 1. The search space of the water problem, despite its six constraints, is mostly unconstrained and when constrained, there is only one violated constraint. DAS-CMOP problems have 11 or 7 constraints most of which, however, are not simultaneously violated. The exceptions are DAS-CMOP problems with triplet 12, which have up to three simultaneous violations. MW problems have less constraints, but nonetheless, MW10 and MW11 show two violated constraints throughout the samples.

This analysis is based on the whole search space and is therefore not necessarily representative of what an optimizer would encounter during its search. Fig. 10(B) shows the average share of simultaneously violated constraints encountered during the optimizations performed by NSGA-II/III. Following the results reported by Tanabe and Oyama [9], only a small fraction of the generated solutions for the car-side impact and water problems – about 5.4% and 7.0% respectively – yield one or more simultaneously violated constraints. While the number is similar to what is shown in Fig. 10(A) for the water problem (8.1%), the difference is significant for the car-side impact problem (80%) and suggests that in practice, not all constraints are as critical as expected.

This reduction is also observed to a more limited extent for the other problems. Yet, three and more simultaneous constraint violation have only been encountered on MODAct problems with constraint levels 3 and 4 and their share still represents 5-10%. This is a key and unique feature algorithms have to be able to deal with.

VII. CONCLUSION

In this paper, the design of electro-mechanical actuators has been used to derive realistic benchmark problems for constrained multi-objective optimization. The proposed new test suite MODAct is composed of 20 problems built around four different constraint levels and up to five objectives. With discontinuities in the objective space and a focus on constraints, these problems are representative of many common mechanical design applications. In addition, a constraint landscape analysis method applicable to CMOPs has been presented. It relies on existing metrics (FsR and RFB $_{\times}$) as well as three new ones (nRFB $_{\times}$, PFd and PFcv).

MODAct has been studied through this constraint landscape analysis approach and by running a convergence study with NSGA-II, NSGA-III and C-TAEA. The same methods were applied to existing benchmark problems from literature: CTP, CF, car-side impact, water, MW and selected DAS-CMOP. The results have been compared to highlight the differences, leading to the following observations:

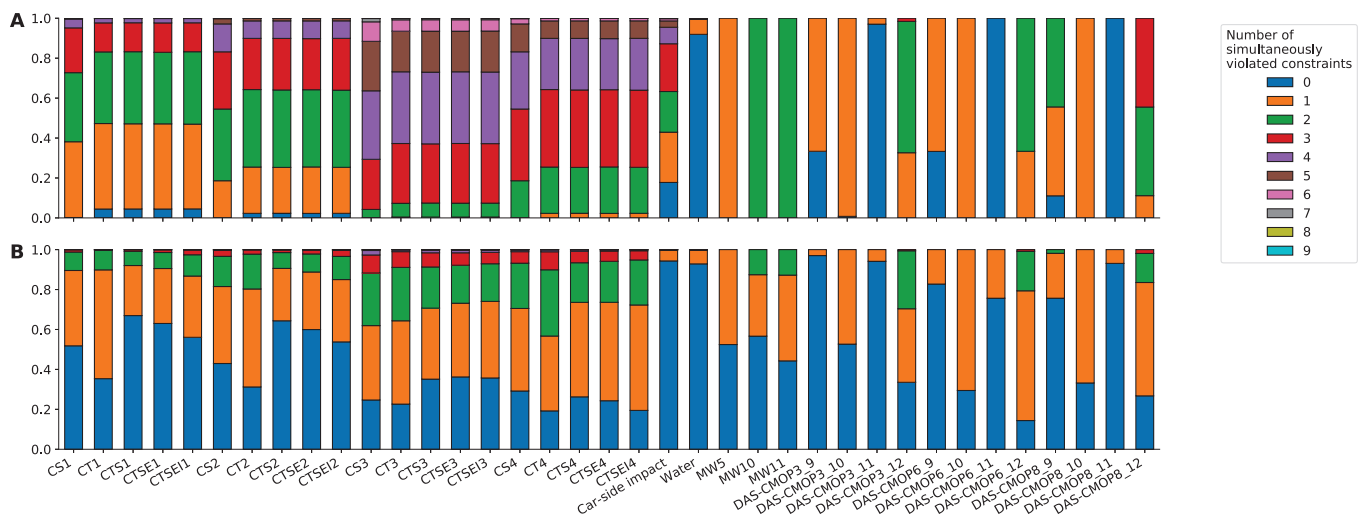


Fig. 10. (A) Relative share of samples from \mathcal{L} and (B) average share of evaluated solutions by NSGA-II or NSGA-III by number of simultaneously violated constraints for all considered problems with more than two constraints

- For MODAct, the convergence quality of NSGA-II, NSGA-III and C-TAEA decreases as tighter constraints are added, up to the point where finding a front close to the optimum is rather unlikely. As such, MODAct covers a wide range of complexity levels.
- While C-TAEA demonstrates better performance than NSGA-II or NSGA-III on many existing benchmark problems, this is not the case for MODAct.
- MODAct problems are characterized by a very small FsR. However, this analysis suggests that a low FsR alone does not necessarily imply difficult problems for optimization algorithms.
- Certain combinations of PFd and PFcv highlight potentially different optimization challenges: finding feasible solutions, promoting diversity and moving towards the Pareto front. Yet, they might not all be related to the constraints.

Finally, the number of simultaneously violated constraints in solutions generated along the various optimizations further emphasized the importance of the constraints and their combinations in MODAct problems, since a large majority of the found solutions are violating several constraints.

The fact that established and modern optimization algorithms struggle with constraints clearly is an important motivation for further research on constrained multi-objective optimizers. With its tunable level of complexity (objectives and constraints), the MODAct test suite is well designed to promote advanced constraint handling strategies and the development of the required tools to further push multi-objective optimization in engineering problems.

Depending on their priorities, researchers can focus on a single class and multiple constraint levels or explore multiple objectives with fewer constraint levels. The framework also offers a pathway to explore the impact of a larger search space or more challenging problems through new constraint combinations as algorithms evolve.

Future work should evaluate more existing and new op-

timization algorithms and constraint handling strategies to identify features that help efficiently solve MODAct and other problems. In addition, additional constraint analysis tools are needed to continuously improve the understanding of new and published constrained multi-objective problems. This valuable information should then inspire future algorithm development.

ACKNOWLEDGMENT

The authors would like to thank Dr. E. de Lange from Johnson Electric International AG for his support and for providing the considered application.

This work is sponsored by Innosuisse (the Swiss Innovation Agency).

REFERENCES

- [1] H. Jain and K. Deb, "An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 602–622, Aug. 2014.
- [2] Q. Zhang, A. Zhou, S. Zhao, P. N. Suganthan, W. Liu, and S. Tiwari, "Multiobjective optimization test instances for the CEC 2009 special session and competition," *Univ. of Essex, Colchester, UK and Nanyang Technol. Univ., Singapore*, no. Rep. CES-487, 2008.
- [3] E. Mezura-Montes and C. A. Coello Coello, "Constraint-handling in nature-inspired numerical optimization: Past, present and future," *Swarm Evol. Comput.*, vol. 1, no. 4, pp. 173–194, Dec. 2011.
- [4] H. Fukumoto and A. Oyama, "A Generic Framework for Incorporating Constraint Handling Techniques into Multi-Objective Evolutionary Algorithms," in *Appl. of Evol. Comput.* Springer, Cham, Apr. 2018, pp. 634–649.
- [5] C. A. Coello Coello, "Use of a self-adaptive penalty approach for engineering optimization problems," *Comput. in Industry*, vol. 41, no. 2, pp. 113–127, Mar. 2000.
- [6] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Jul. 2001.
- [7] C. Picard and J. Schiffmann, "Impacts of Constraints and Constraint Handling Strategies for Multi-objective Mechanical Design Problems," in *Proc. Genetic and Evol. Comput. Conf. (GECCO)*. New York, NY, USA: ACM, 2018, pp. 1341–1347.
- [8] H. Fukumoto and A. Oyama, "Benchmarking Multiobjective Evolutionary Algorithms and Constraint Handling Techniques on a Real-world Car Structure Design Optimization Benchmark Problem," in *Proc. Genetic and Evol. Comput. Conf. (GECCO) Companion*. New York, NY, USA: ACM, 2018, pp. 177–178.

- [9] R. Tanabe and A. Oyama, "A note on constrained multi-objective optimization benchmark problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2017, pp. 1127–1134.
- [10] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
- [11] K. Deb and H. Jain, "An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2014.
- [12] K. Li, R. Chen, G. Fu, and X. Yao, "Two-Archive Evolutionary Algorithm for Constrained Multiobjective Optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 2, pp. 303–315, Apr. 2019.
- [13] N. Srinivas and K. Deb, "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms," *Evol. Comput.*, vol. 2, no. 3, pp. 221–248, Sep. 1994.
- [14] M. Tanaka, H. Watanabe, Y. Furukawa, and T. Tanino, "GA-based decision support system for multicriteria optimization," in *Proc. IEEE Int. Conf. Syst. Man Cybern.*, vol. 2, Oct. 1995, pp. 1556–1561 vol.2.
- [15] A. Osyczka and S. Kundu, "A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm," *Structural Optim.*, vol. 10, no. 2, pp. 94–99, Oct. 1995.
- [16] Y. G. Woldeesenbet, G. G. Yen, and B. G. Tessema, "Constraint Handling in Multiobjective Evolutionary Optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 3, pp. 514–525, Jun. 2009.
- [17] K. Deb, A. Pratap, and T. Meyarivan, "Constrained Test Problems for Multi-objective Evolutionary Optimization," in *Proc. 1st Int. Conf. Evol. Multi Criterion Optim. (EMO)*, 2001, pp. 284–298.
- [18] J. Li, Y. Wang, S. Yang, and Z. Cai, "A comparative study of constraint-handling techniques in evolutionary constrained multiobjective optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 4175–4182.
- [19] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable Test Problems for Evolutionary Multiobjective Optimization," in *Proc. Evol. Multiobjective Optim.* Springer, 2005, pp. 105–145.
- [20] Z. Ma and Y. Wang, "Evolutionary Constrained Multiobjective Optimization: Test Suite Construction and Performance Comparisons," *IEEE Trans. Evol. Comput.*, vol. 23, no. 6, pp. 972–986, Dec. 2019.
- [21] Z. Fan, W. Li, X. Cai, H. Li, C. Wei, Q. Zhang, K. Deb, and E. Goodman, "Difficulty Adjustable and Scalable Constrained Multiobjective Test Problem Toolkit," *Evol. Comput.*, pp. 1–40, May 2019.
- [22] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 477–506, Oct. 2006.
- [23] T. Ray, K. Tai, and K. C. Seow, "Multiobjective Design Optimization by an Evolutionary Algorithm," *Eng. Optim.*, vol. 33, no. 4, pp. 399–424, Apr. 2001.
- [24] K. Deb and S. Jain, "Multi-Speed Gearbox Design Using Multi-Objective Evolutionary Algorithms," *J. Mech. Design*, vol. 125, no. 3, p. 609, 2003.
- [25] Z.-Z. Liu and Y. Wang, "Handling Constrained Multiobjective Optimization Problems With Constraints in Both the Decision and Objective Spaces," *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 870–884, Oct. 2019.
- [26] K. M. Malan, J. F. Oberholzer, and A. P. Engelbrecht, "Characterising constrained continuous optimisation problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*. Sendai, Japan: IEEE, May 2015, pp. 1351–1358.
- [27] K. M. Malan and I. Moser, "Constraint Handling Guided by Landscape Analysis in Combinatorial and Continuous Search Spaces," *Evol. Comput.*, vol. 27, no. 2, pp. 267–289, Mar. 2018.
- [28] K. M. Malan and A. P. Engelbrecht, "A progressive random walk algorithm for sampling continuous fitness landscapes," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2014, pp. 2507–2514.
- [29] R. Mallipeddi and P. N. Suganthan, "Problem Definitions and Evaluation Criteria for the CEC 2010 Competition on Constrained Real-Parameter Optimization," *Nanyang Technol. Univ., Singapore*, 2010.
- [30] P. Acarnley, *Stepping Motors: A Guide to Theory and Practice*. IET, Jan. 2002.
- [31] International Organization for Standardization, "Vocabulary of gear terms – Part I: Definitions related to geometry," ISO, Geneva, Switzerland, ISO 1122-1:1998, 1998.
- [32] —, "Calculation of load capacity of spur and helical gears," ISO, Geneva, Switzerland, ISO 6336:2006, 2006.
- [33] —, "Gears – Cylindrical involute gears and gear pairs – Concepts and geometry," ISO, Geneva, Switzerland, ISO 21771:2007, 2007.
- [34] Dawson-Haggerty et al., "trimesh," Dec. 2019. [Online]. Available: <https://trimesh.org/>
- [35] C. A. C. Coello and N. C. Cortés, "Solving Multiobjective Optimization Problems Using an Artificial Immune System," *Genetic Program. Evolvable Mach.*, vol. 6, no. 2, pp. 163–190, Jun. 2005.
- [36] J. Blank and K. Deb, "Pymoo: Multi-Objective Optimization in Python," *IEEE Access*, vol. 8, pp. 89 497–89 509, 2020.
- [37] I. Das and J. Dennis, "Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems," *SIAM J. Optim.*, vol. 8, no. 3, pp. 631–657, Aug. 1998.
- [38] D. Brockhoff, T.-D. Tran, and N. Hansen, "Benchmarking Numerical Multiobjective Optimizers Revisited," in *Proc. Annu. Conf. on Genetic and Evol. Comput. (GECCO)*, ser. GECCO '15. New York, NY, USA: ACM, 2015, pp. 639–646.
- [39] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999.
- [40] W. Cox and L. While, "Improving and Extending the HV4D Algorithm for Calculating Hypervolume Exactly," in *Proc. Advances in Artificial Intell.* Springer, 2016, pp. 243–254.
- [41] L. While, L. Bradstreet, and L. Barone, "A Fast Way of Calculating Exact Hypervolumes," *IEEE Trans. Evol. Comput.*, vol. 16, no. 1, pp. 86–95, Feb. 2012.
- [42] C. Picard and J. Schiffmann. (2020, May) Multi-objective design of actuators: Pareto fronts. Zenodo. (Version 1.1.0). [Online]. Available: <https://doi.org/10.5281/zenodo.3824302>
- [43] Q. Zhang and P. N. Suganthan, "Final report on CEC '09 MOEA competition," The School of CS and EE, University of Essex, UK, Tech. Rep., 2009.

Cyril Picard received the B.Sc. and M.Sc. degrees in mechanical engineering from the École polytechnique fédérale de Lausanne (EPFL), Lausanne, Switzerland in 2013 and 2015, respectively.

He is currently pursuing the PhD degree at the Laboratory for Applied Mechanical Design at EPFL, Neuchâtel, Switzerland, under the supervision of Prof. Schiffmann. His current research interests include automated and integrated design methodology, and algorithms and constrained evolutionary optimization.

Jürg Schiffmann received his Ing. degree in mechanical engineering from the École polytechnique fédérale de Lausanne (EPFL), Lausanne, Switzerland in 1999. He received his PhD degree in mechanical engineering from EPFL in 2008.

In 1999, he co-founded a start-up company dedicated to the design of gas-lubricated bearings. In 2005 he joined Fischer Engineering Solutions, Herzogenbuchsee, Switzerland, where he led the development of small scale, high-speed turbomachinery. In 2008, he then joined the Gas Turbine Lab at the Massachusetts Institute of Technology (MIT), Boston, MA, USA as a postdoctoral associate where he worked on foil bearings. In 2013, he was nominated assistant professor at EPFL, Neuchâtel, Switzerland.

Dr. Schiffmann has received several awards from the American Society of Mechanical Engineers (ASME).