J Optim Theory Appl (2017) 173:628–657
DOI 10.1007/s10957-016-1053-4

Approximate and Widespread Pareto Solutions in the Structure-Control Design of Mechatronic Systems

Miguel G. Villarreal-Cervantes[1]

Received: 30 October 2015 / Accepted: 26 December 2016 / Published online: 12 January 2017

Abstract The structure-control design approach of mechatronic systems requires a different design formulation where the mechanical structure and control system are simultaneously designed. Optimization problems are commonly stated to confront the structure-control design formulation. Nevertheless, these problems are often very complex with a highly nonlinear dependence between the design variables and performance functions. This fact has made the use of evolutionary algorithms, a feasible alternative to solve the highly nonlinear optimization problem; the method to find the best solution is an open issue in the structure-control design approach. Hence, this paper presents a mechanism to exhaustively exploit the solutions in the differential evolution (DE) algorithm in order to find more non-dominated solutions with uniformly distributed Pareto front and better trade-offs in the structure-control design framework. The proposed approach adopts an external population to retain the non-dominated solutions found during the evolutionary process and includes a mechanism to mutate the individuals in their corresponding external population region. As a study case, the structure-control design of a serial-parallel manipulator with its control system is stated as a dynamic optimization problem and is solved by using the proposed approach. A comparative analysis shows that the multi-objective exhaustive exploitation differential evolution obtained a superior performance in the structure-control design framework than a DE algorithm which did not consider the proposal. Hence, the resulting designs provide better trade-offs between the structure-control performance functions.

Communicated by Mauro Pontani.

✉ Miguel G. Villarreal-Cervantes
mvillarrealc@ipn.mx

[1] Postgraduate Department, Instituto Politécnico Nacional, CIDETEC, Mexico City, Mexico

Keywords Structure-control design · Concurrent design · Multi-objective optimization · Exploitation mechanism · Differential evolution

Mathematics Subject Classification 90C29 · 68T20

1 Introduction

Many processes and products in the area of mechanical and electrical engineering are showing an increasing integration of a mechanical system with its embedded control strategy. The integration of such systems significantly improves the performance of the mechatronic system due to the mechanical structure depending on the control system and the control system depending on the mechanical structure. The integration of mechanical structure and control in mechatronic systems has originated a multi-disciplinary framework for their design [1,2]. In this multi-disciplinary design approach, a detailed description of the system behavior is needed, where the mechanical behavior and the control system dynamic performance must be considered. This approach has emerged from space applications [3,4], where the systems are stationary structures. Lately, different complex dynamics are included in the multi-disciplinary design approach. The sequential and simultaneous strategies have been used to face the structure-control design framework [5]. In the sequential strategy [6–8], the mechanical structure is firstly designed, and then, in a second design step the control system is carried out. This can be executed by single pass or iterative processes where it is repetitively continues until the desired performance is fulfilled. It was theoretically and experimentally shown in several works [5,8–11] that the sequential optimization strategies do not guarantee, at least for linear systems, the coupling effects in the combined structure/controller system. For nonlinear systems, this assumption is not theoretically guaranteed. Then, the simultaneous strategies are the best options to merge their corresponding designs. Among the promising simultaneous strategies are those that find the minimum combined performance for the mechatronic system by assuming that the optimum structure occurs when the gain problem is minimized; such approach is called bi-level strategy. Nevertheless, this may not allow the full exploration of the feasible space in the structure-control design framework, so that the best strategy comes from using the all-at-once strategy where all design parameters and performance functions are synergetically combined [5]. Hereinafter, when mentioning simultaneous strategy, it refers especially to the all-at-once strategy.

In simultaneous strategies, multi-objective optimization problems for the optimal balance between the performance of both the basic mechanical structure and the overall control system must be stated. Nevertheless, the mechatronic system design is usually formulated into a scalarization method as in [11–18], where the multi-objective optimization problem is transformed into a single-objective optimization one. The most common scalarization method is the weighted sum approach (WSA) [19], and the main drawback is when a non-convex optimization problem is given (a condition that is not generally satisfied in real problems) such that an evenly distributed set of weights does not necessarily produces an evenly distributed representation of the Pareto-optimal set, i.e., different weights may produce nearly similar objective vectors. Therefore, the obtained solution shows a high sensitivity to the chosen weights, and a

trial-and-error procedure is usually used to find such weights and some non-dominated solutions (Pareto- optimal solutions) could not be found.

In the research work of Portilla, et. al. [16], a pinion-rack continuously variable transmission (CVT) is designed by stating a multi-objective optimization problem. A scalarization method (goal attainment method) and a Pareto-based multi-objective method (modified differential evolution algorithm) were compared. The goal attainment method was very sensitive to the initial point, and few Pareto solutions could be found. Two different weights were chosen with the goal attainment method, and the two Pareto solutions were not distributed according to the weights. On the other hand, with the modified differential evolution algorithm, more Pareto solutions were found such that more diversity in the design of the CVT was obtained. This diversity is useful for the decision making of the designer. In the CVT system, the solutions found by the modified differential evolution algorithm were preferred. As a general point of view, in the scalarization method the synergetic combination of performance functions is not totally fulfilled because of the weight sensitivity.

Multiple criteria problems in simultaneous strategies for the design of mechatronic systems should be preferably formulated as Pareto-based multi-objective optimization problem (PBMOP) [20] where the concept of Pareto optimality is incorporated and the best possible trade-off should be found by evaluating several incommensurable and often conflicting objectives. Optimizing PBMOP does not only mean finding a single optimal solution, but rather a set of alternative solutions. When all objectives are considered, these solutions are optimal in the wider sense that no other solutions in the search space are superior. These solutions are known as Pareto-optimal solutions. Consequently, the final solution may not be unique and the decision-making problem depends on the design specification for the application. PBMOP faces and solves the issues associated to weigh the performance functions in the WSA. Nevertheless, the selection issue related to the optimization technique to be used in the PBMOP is a crucial factor to find a synergetic trade-off in the structure-control design performance for the mechatronic design framework.

Great research efforts were first dedicated to guarantee the optimality conditions in gradient-based optimization techniques [21]. However, the necessity to solve complex engineering problems has promoted the use of evolutionary algorithms (EAs) [22] which inherently explore a set of possible solutions simultaneously. This characteristic enables the search for an entire set of Pareto-optimal solutions in a single run of the algorithm instead of having to carry out a series of separated runs as in the case of gradient-based techniques. In addition, evolutionary algorithms are more susceptible to deal with problem- dependent characteristics (convex, concave or even discontinuous), such as the shape of the Pareto front and the mathematical properties of the search space. In this direction, to the best of the author's knowledge, few works related to the mechatronic design framework have been formulated as PBMOP, e.g., in the design of CVT system [23], the parallel robot [24] and the four bar system [25,26]. Nevertheless, only in Portilla, et. al. [23], modifications in the standard evolutionary algorithm in the mechatronic design framework were considered. In that work, the reconfigurability feature of non-iterative concurrent mechatronic design was enhanced by including into the differential evolution algorithm a mechanism to diversify the solutions. The proposed differential evolution algorithm obtains an extended and well-distributed

Pareto front, where the obtained solutions improve the parametric reconfigurations of the mechanical and control design for two mechatronic systems.

In recent years, the differential evolution algorithm [27] has been widely used to solve complex optimization problems in spite of rising EAs in the literature. This is due to the differential evolution algorithm being a simple and efficient evolutionary algorithm with excellent global search ability and easy implementation [28,29]. These characteristics motivate the research on how to improve the explorative and exploitative search to find better results and hence to avoid stagnation and premature convergence [30,31]. An efficient exploration mechanism in the search space and an effective exploitation mechanism in a region of the search space would be desirable into the optimization algorithm. The exploration can thoroughly search different regions, while the exploitation accelerates the convergence to the optimum solution in the region. Furthermore, the explorative/exploitative mechanism in the differential evolution (DE) algorithm is an open issue and depends on the problem at hand.

Based on the previous literature, the main concern in the structure-control design framework based on simultaneous strategy and stated as PBMOP is to find more and diverse solutions in the Pareto front because the problem is generally larger and difficult to solve due to the strong dependency between the design variables and performance function. However, there is a lack of information about the importance of the optimization technique to fulfill the synergetic trade-off in the structure-control design framework. Therefore, in this paper an exhaustive exploitation mechanism of an external population and the constrained dominance concept is included into the DE algorithm for the purpose of finding a better synergetic integration in the structure-control design framework of mechatronic systems. The superior performance of the proposed multi-objective exhaustive exploitation differential evolution algorithm (MOEEDE) is validated in a particular study case which indicate a strong dependency between the obtained Pareto solutions and how the search in the feasible space is made by the algorithm.

The paper is organized as follows: In the next section, the multi-objective exhaustive exploitation differential evolution algorithm is presented. The design variable vector, performance functions and constraints are stated in Sect. 3 for the study case in the structure-control design framework. The comparative analysis of the results provided by the proposed MOEEDE with a DE version without the exhaustive exploitation mechanism is shown in Sect. 4. Finally, in Sect. 5, the conclusions are given.

2 Exhaustive Exploitation Differential Evolution for Multi-Objective Dynamic Optimization

2.1 Multi-Objective Dynamic Optimization Problem

A multi-objective dynamic optimization problem (MODOP) is defined as finding the n_p-dimensional decision variable vector $\mathbf{p} = [p_1, \ldots, p_n]^T \in \mathbb{R}^{n_p} \subseteq \Omega$ such that optimizes (minimize or maximize) the performance index vector $\bar{\mathbf{J}}(\mathbf{x}(t), \mathbf{p}, t) = [\bar{J}_1(\mathbf{x}(t), \mathbf{p}, t), \ldots, \bar{J}_i(\mathbf{x}(t), \mathbf{p}, t)] \; \forall \, i = 1, \ldots, n_{\bar{J}}$ subject to inequality and equality constraint vector $g_i(\mathbf{x}, \mathbf{p}, t) \leq 0 \; \forall \, i = , 1, \ldots, n_g, h_j(\mathbf{x}, \mathbf{p}, t) = 0 \; \forall \, j = 1, \ldots, n_h$, respectively, and differential algebraic equations $\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}, \mathbf{p}, t)$ that describe

the dynamic behavior of the system with the state vector $\mathbf{x} \in \mathbb{R}^{n_x}$, the input control vector $\mathbf{u} \in \mathbb{R}^{n_u}$ and the initial state vector $\mathbf{x}(0) = \mathbf{x}_{INI} \in \mathbb{R}^{n_x}$. In mathematical notation, the MODOP is formulated as in (1), where $t \in \mathbb{R}$ is the time variable.

$$\underset{\mathbf{p}\in\Omega}{Min}\ \bar{\mathbf{J}}(\mathbf{p}) \tag{1}$$

subject to:

$$\begin{aligned} \dot{\mathbf{x}} = \tfrac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} &= \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{p}, t) \\ \mathbf{g}(\mathbf{x}, \mathbf{p}, t) &< 0, \quad \in \mathbb{R}^{n_g} \\ \mathbf{h}(\mathbf{x}, \mathbf{p}, t) &= 0, \quad \in \mathbb{R}^{n_h} \\ \mathbf{x}(0) &= \mathbf{x}_{INI} \end{aligned}$$

2.2 Multi-Objective Exhaustive Exploitation Differential Evolution Algorithm

The nonlinear dynamic multi-objective optimization problem stated in (1) is solved by using the proposed multi-objective exhaustive exploitation differential evolution algorithm (MOEEDE).

Differential evolution (DE) [27] is a population-based evolutionary algorithm with a simple mutation mechanism and a crossover operator that performs a linear recombination of a number of individuals (normally three) and one parent (which is the subject to be replaced) to create one child. The selection is deterministic between the parent and the child (i.e., the best of them remains in the next population). DE shares similarities with traditional evolutionary algorithms. However, it does not use binary encoding [32] nor a probability density function to self-adapt its parameters as the evolution strategy [33]. Besides, DE has been widely used to solve optimization problems applied to mechanical design [34,35].

DE is generally easy to implement but usually spends more computation time than nonlinear programming techniques [16]. The key parameters are: NP—the population size that is the set of individuals, G—is the generation or iteration in the optimization process, CR—the crossover constant that controls the influence of the parent in the generation of the offspring (higher values mean less influence of the parent), F—the weight applied to the influence of two of the three individuals selected at random in order to generate the offspring (scaling factor).

The original DE algorithm [27] can only solve unconstrained optimization problems. Therefore, several modifications of the original DE algorithm are made to improve the Pareto solutions in constrained multi-objective problems. These modifications are mentioned below:

- Constrained dominance is proposed to select between the father and child.
- A mechanism for accepting individuals to external population (external file) is implemented.
- A mechanism to exhaustively exploit the non-dominated solutions in each Pareto front hyper-grid of the external population is included in the mutation process.

In Algorithms 1 and 2, the MOEEDE pseudo-code is presented. Before giving the details of the modifications in the original DE algorithm, some important concepts in multi-objective optimization problems are shown.

Algorithm 1 First part of the MOEEDE algorithm ($G \leq G_{MaxMOEEDE} \times EEF$).

Step 1: $G = 0$
Step 2: Create a random initial base population $\mathbf{x}_G^i \ \forall i = 1, \ldots, BP$.
Step 3: Create an empty external population $\mathbf{z}^k = \varnothing \ \forall k = 1, \ldots, MP$.
Step 4: Create an empty trial population $\mathbf{y}^l = \varnothing \ \forall l = 1, \ldots, BP$.
Step 5: Evaluate $\bar{\mathbf{J}}(\mathbf{x}_G^i, t)$, $\mathbf{g}(\mathbf{x}_G^i, t)$, $\mathbf{h}(\mathbf{x}_G^i, t) \ \forall i = 1, \ldots, BP \wedge \forall t = 0, \ldots, t_f$.
Step 6: **while** $(G \leq G_{MaxMOEEDE} \times EEF)$ **do**
Step 7: **for** $i = 1$ to BP **do**
Step 8: Select randomly $\{r_1 \neq r_2 \neq r_3 \neq i\} \in \mathbf{x}_G^i \ \forall i = 1, \ldots, BP$.
Step 9: j_{rand} =randint(1, D).
Step 10: **for** $j = 1$ to D **do**
Step 11: **if** (rand$_j[0, 1) < CR$ or $j = j_{rand}$) **then**
Step 12: $u_{j,G+1}^i = x_{j,G}^{r_1} + F(x_{j,G}^{r_2} - x_{j,G}^{r_3})$
Step 13: **else**
Step 14: $u_{j,G+1}^i = x_{j,G}^i$
Step 15: **end if**
Step 16: **end for**
Step 17: Evaluate $\bar{\mathbf{J}}(\mathbf{u}_{G+1}^i, t)$, $\mathbf{g}(\mathbf{u}_{G+1}^i, t)$, $\mathbf{h}(\mathbf{u}_{G+1}^i, t) \ \forall t = 0, \ldots, t_f$
Step 18: **if** $(\mathbf{u}_{G+1}^i \ [\preceq] \ \mathbf{x}_G^i)$ **then**
Step 19: $\mathbf{x}_{G+1}^i = \mathbf{u}_{G+1}^i$
Step 20: Store $\mathbf{u}_{G+1}^i$ in $\mathbf{y}$
Step 21: **else**
Step 22: $\mathbf{x}_{G+1}^i = \mathbf{x}_G^i$
Step 23: **end if**
Step 24: **end for**
Step 25: The individual $\mathbf{y}^i$ can be accepted in the external population $\mathbf{z}$ whether the Definition 2.8 is satisfied.
Step 26: $\mathbf{y} = \varnothing$
Step 27: $G = G + 1$
Step 28: **end while**

Definition 2.1 Let $\mathbf{p} \in \Omega$ be all design space solutions, the feasible region is represented as $\hat{\Omega} = \{\mathbf{p} \in \Omega \mid \mathbf{g}(\mathbf{x}, \mathbf{p}, t) < 0, \mathbf{h}(\mathbf{x}, \mathbf{p}, t) = 0\}$

Definition 2.2 A vector $\mathbf{p} \in \mathbb{R}^{n_p}$ dominates a vector $\mathbf{p}' \in \mathbb{R}^{n_p}$ (named as $\mathbf{p} \preceq \mathbf{p}'$) $\iff$ $\mathbf{p}$ is smaller than $\mathbf{p}'$, i. e., $\forall i \in \{1, \ldots, n_{\bar{J}}\}$, $\bar{J}_i(\mathbf{p}) \leq \bar{J}_i(\mathbf{p}') \wedge \exists i \in \{1, \ldots, n_{\bar{J}}\}$: $\bar{J}_i(\mathbf{p}) < \bar{J}_i(\mathbf{p}')$.

Definition 2.3 The continuos time $t \in \mathbb{R}$ is discretized into $n_{\Delta t} = \frac{t_f - t_0}{\Delta t} + 1$ samples, i.e., $t = \{t_0, \ t_1 = \Delta t, \ t_2 = 2\Delta t, \ \ldots, \ t_{n_{\Delta t}} = t_f\} \in \mathbb{R}$. The initial and final time are represented by t_0 and t_f, respectively.

Definition 2.4 A solution $\mathbf{p} \in \hat{\Omega}$ is a Pareto optimum in $\hat{\Omega}$ $\iff$ there is not $\mathbf{p}' \in \hat{\Omega}$ which $\mathbf{p}' \preceq \mathbf{p}$.

Definition 2.5 The set of Pareto optimum $\mathfrak{P}^*$ is defined as:

$$\mathfrak{P}^* := \{\mathbf{p} \in \hat{\Omega} | \nexists \bar{\mathbf{J}}(\mathbf{p}') \preceq \bar{\mathbf{J}}(\mathbf{p})\} \forall \mathbf{p}' \neq \mathbf{p} \in \hat{\Omega}$$

Each solution of the Pareto optimum set is called non-dominated solution.

Algorithm 2 Second part of the MOEEDE algorithm ($G_{MaxMOEEDE} \times EEF < G \leq G_{Max}$).

Step 29: **while** ($G < G_{MaxMOEEDE}$) **do**
Step 30: $\breve{p} = 1$
Step 31: Assign the number of generated offsprings for each occupied grid $n_{iog} = \frac{BP2}{n_{oe}}$.
Step 32: **for** $i = 1$ to $n_{row} \times n_{col}$ **do**
Step 33: **if** ($\mathbf{z} \neq \varnothing$) **then**
Step 34: **for** $k = 1$ to n_{iog} **do**
Step 35: Select randomly $\{r_1 \neq r_2 \neq r_3\} \in \Gamma_i$
Step 36: j_{rand} =randint(1, D)
Step 37: **for** $j = 1$ to D **do**
Step 38: **if** (rand$_j$[0, 1) $< CR$ or $j = j_{rand}$) **then**
Step 39: $u^{\breve{p}}_{j,G+1} = z^{r_1}_j + F(z^{r_2}_j - z^{r_3}_j)$
Step 40: **else**
Step 41: $u^{\breve{p}}_{j,G+1} = z^{r_1}_j$
Step 42: **end if**
Step 43: **end for**
Step 44: Evaluate $\bar{\mathbf{J}}(\mathbf{u}^{\breve{p}}_{G+1}, t)$, $\mathbf{g}(\mathbf{u}^{\breve{p}}_{G+1}, t)$, $\mathbf{h}(\mathbf{u}^{\breve{p}}_{G+1}, t)$ $\forall t = 0, \ldots, t_f$
Step 45: **if** ($\mathbf{u}^{\breve{p}}_{G+1}$ $[\preceq]$ $\mathbf{z}^{r1}$) **then**
Step 46: Store $\mathbf{u}^{\breve{p}}_{G+1}$ in $\mathbf{y}$
Step 47: **end if**
Step 48: $\breve{p} = \breve{p} + 1$
Step 49: **end for**
Step 50: **end if**
Step 51: **end for**
Step 52: Acceptance mechanism to external population
Step 53: $\mathbf{y} = \varnothing$
Step 54: $G = G + 1$
Step 55: **end while**

Definition 2.6 The Pareto front $\mathfrak{PF}$ is defined as:

$$\mathfrak{PF} := \{\bar{\mathbf{J}}(\mathbf{p}) | \mathbf{p} \in \mathfrak{P}^*\}$$

2.3 Selection with Constrained Dominance

The selection between the father (base vector) and the child (mutant vector) in the MOEEDE algorithm is made by using constrained dominance. This is a modification of the definition *domination* between the solution i and j.

Definition 2.7 A solution i dominates with constraint a solution j (named as $\hat{\mathbf{u}}_i [\preceq] \hat{\mathbf{v}}_j$), if any of the following conditions is fulfilled.

1. Solutions i and j are both feasible and solution i dominates solution j.
2. Solution i is feasible and solution j is not feasible.
3. Solutions i and j are both infeasible, but the solution i has a smaller constraint violation number.
4. Solutions i and j are both infeasible with the same number of constraint violations, but solution i has a smaller constraint distance ζ_{g_i}.

It is important to mention that the third point of Definition 2.7 only counts the number of constraints that are violated. But this is not related to the closeness of the violated constraint to the corresponding active region.

The constraint distance ζ_g measures the distance from the violated constraint value to its corresponding active constraint value. The constraint distance of a individual A, named as ζ_{g_A}, is defined as in (2), where $g_{j_{MAX}}$ is the values of the constraints g_j with the maximum values in the design variable vector $\mathbf{p}$.

$$\zeta_{g_A} = \sum_{j=1}^{n_g} \sum_{i=0}^{n_{\Delta t}} \max\left(0, \frac{g_j(\mathbf{x}, \mathbf{p}, t_i)}{g_{j_{MAX}}}\right) \tag{2}$$

Hence, the fourth point of Definition 2.7 includes the constraint distance ζ_g as an additional measure to select the closest individual to the active region. This is more useful in dynamic constraints since if a constraint is violated in only one time interval $[t_{i-1}, t_i]$ $\exists i \in \{1, \ldots, n_{\Delta t}\}$, the third point of Definition 2.7 will only indicate one constraint violation. Meanwhile, the four point will additionally indicate whether the solution is close to the feasible region.

The selection between the father vector and child vector using constrained dominance is shown in rows 18–23 in Algorithm 1, and the selection between the base vector of the mutant vector and the child vector using constrained dominance is shown in rows 45–47 in Algorithm 2.

2.4 Acceptance Mechanism to External Population

An external population or external file is included to store the new non-dominated solutions (new individuals) for each generation. The external population can include a maximum of MP solutions.

In addition, a self-adaptive grid (S-A grid), similar to that adopted in PAES [36], is implemented in the external population. The S-A grid divides the Pareto front of the external population as a multi-dimensional arrays (hyper-grid). For the particular case of two performance functions, which is the study case, two-dimensional matrix is obtained. Then, the grid is set to have $n_{\text{row}} \times n_{\text{col}}$ elements and the length of each element is self-adjusted considering the maximum and minimum limits of the found Pareto front.

Definition 2.8 is considered to accept a new individual in the external population:

Definition 2.8 Let $\mathbf{y}^i \in \hat{\Omega}$, iff $\bar{\mathbf{J}}(\mathbf{y}^i) \preceq \bar{\mathbf{J}}(\mathbf{z}^j)$ $|\exists j = \{1, \ldots, MP\}$, then $\mathbf{y}^i$ is accepted in the external population $\mathbf{z}$ and $\mathbf{z} = \mathbf{z} \setminus \{\mathbf{z}^j\}$.

The new individual is accepted whether it is feasible and dominates at least one individual of the external population and those dominated individuals are removed. On the contrary, the individual is rejected.

Each element of the grid will have at most n_{MaxInd} individuals to not overflow the external file (external population). If more individuals are found, the individuals with less crowding distance grouped in such grid element are eliminated until the maximum limit of individuals per element (n_{MaxInd}) is fulfilled. The crowding distance [37] of the ith individual is the Euclidian distance between the $(i-1)$th individual and the $(i+1)$th individual in the phenotype space.

The pseudo-code of the acceptance mechanism to external population is shown in Algorithm 3.

Algorithm 3 Acceptance mechanism to external population.

Step 1: Includes the individuals in $\mathbf{y}^i$ that can be accepted in the external population $\mathbf{z}$ based on Definition 2.8.
Step 2: Make the $n_{row} \times n_{col}$ grid to the non-dominated solution vector $\mathbf{z}$ considering the maximum and minimum limits of the found Pareto front.
Step 3: Assign each individual of $\mathbf{z}$ its corresponding grid element.
Step 4: Count the occupied element number n_{oe} in the grid of $\mathbf{z}$ and the individual number at each grid element.
Step 5: **if** individuals of the occupied elements exceed of n_{MaxInd}. **then**
Step 6: Eliminate the excessive individuals in the corresponding element of the grid $\mathbf{z}$ (based on crowding distance in the occupied grid element).
Step 7: **end if**

2.5 Exhaustive Exploitation Mechanism in the Mutation Process

The exhaustive exploitation mechanism consists in carrying out the mutation process in two different modes and depends on the exhaustive exploitation factor value ($EEF \in]0, 1]$). This factor is referred to the porcentage of the maximum generation number G_{Max} where the mutation process is changed.

In the first part of the algorithm, i.e., when $G \leq G_{Max} \times EEF$ (before using the exhaustive exploitation mechanism) shown in Algorithm 1, the mutation process mutates the base population $\mathbf{x}^i_G \; \forall i = 1, \ldots, BP$ to produce a population of BP mutant vectors $\mathbf{v}^i_{G+1} \; \forall \, i \in \{1, \ldots, BP\}$. Therefore, the mutant vector is generated by using three random individuals in $\mathbf{x}_G$ as in the basic differential evolution algorithm [27] and uniform crossover is used. The selection process between the child $\mathbf{u}^i_{G+1}$ and the target vector $\mathbf{x}^i_G$ is based on the constrained dominance, and the non-dominated solutions through generations are stored in $\mathbf{y}$.

The exhaustive exploitation mechanism is activated in the second part of the algorithm (shown in Algorithm 2), i.e., when $G_{Max} \times EEF < G \leq G_{Max}$. The mutation process is generated by using Definition 2.9 where the individuals are mutated with individuals in the same grid element of the external population $\mathbf{z}^k \; \forall \, k = 1, \ldots, MP$ to produce another population of $BP2$ mutant vectors $\mathbf{v}^{\check{p}}_{G+1} \; \forall \, \check{p} \in \{1, \ldots, BP2\}$. The number of generated mutant vector per occupied elements n_{iog} is based on the chosen $BP2$ population and the occupied element number in the grid n_{oe}, which results in $n_{iog} = \frac{BP2}{n_{oe}}$.

Definition 2.9 Let the set $\Gamma_i \subset \mathbf{z}$ the individuals in the ith grid element in the external population $\mathbf{z}$, then the mutant vector is generated as follows:

$$\mathbf{v}^{\check{p}}_{G+1} := \{\mathbf{z}^{r_1} + F(\mathbf{z}^{r_2} - \mathbf{z}^{r_3}) \mid \{r_1 \neq r_2 \neq r_3 \neq \check{p}\} \subseteq \Gamma_i \notin \emptyset\} \tag{3}$$

The random selection of three different individuals ($\mathbf{z}^{r_1}, \mathbf{z}^{r_2}$ and $\mathbf{z}^{r_3} \in \Gamma_i$) in the mutation process exerts a random selection pressure to the external population by regions (grid element) allowing the generation of more and better individuals in each grid element and generation. Two situations can be presented in the selection of the three random individuals in Γ_i for the ith grid element and depends on the number of individuals:

– If $\Gamma_i \in \mathbb{R}^n | 1 \leq n \leq 2$, select one individual from the grid element as the base vector, and the rest by considering the random selection of all individuals in the external population.
– If $\Gamma_i \in \mathbb{R}^n | n > 2$, the random selection considers all individuals in the grid element.

It is important to note that the last consideration shown above, the algorithm exhaustively searches solutions into the occupied grid element. A local selection of individuals into the grid element is carried out. Therefore, the non-dominated solutions in the external population are increased and improve the trade-offs among the objective functions. In addition, with the first consideration, the diversity of the solution is promoted to explore other regions of the search space.

The same uniform crossover, as in the first part of the algorithm, is employed in the second part, and the selection process is made by using constrained dominance between the child $\mathbf{u}_{G+1}^{\breve{p}}$ and the base vector $\mathbf{z}^{r_1}$. The non-dominated solutions are stored in $\mathbf{y}$ and are included in the external population $\mathbf{z}^k \ \forall k = 1, \ldots MP$ following the acceptance mechanism described in Sect. 2.4.

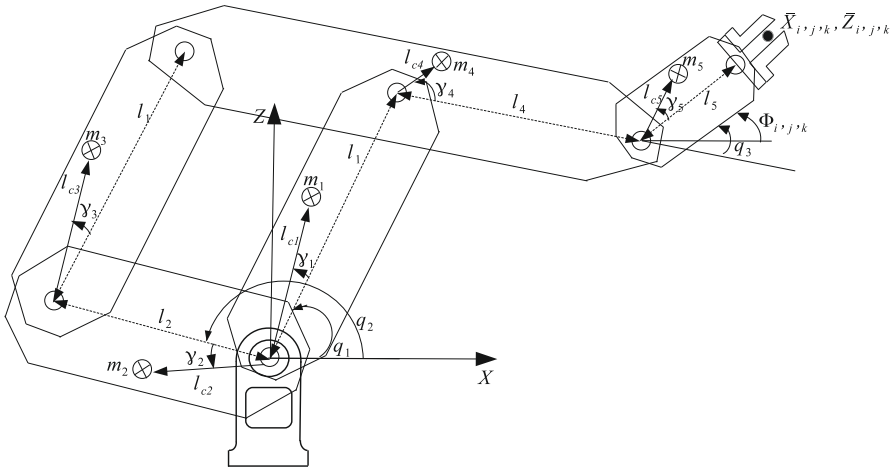
3 Study Case: Structure-Control Design of a 3-d.o.f Manipulator with Parallel Mechanism

As a study case, the design of a three-degree-of-freedom (d.o.f) manipulator with parallel structure and the design of its operational space proportional-integral-derivative (PID) control is formulated as a dynamic optimization problem within the mechatronic design framework. In what follows, the closed-loop robotic system, the performance functions related to the structure and control design as well as the design constraints are detailed.

3.1 Robotic Manipulator and Control System

The three-degree-of-freedom (d.o.f) manipulator with parallel mechanism is shown in Fig. 1, where $\mathbf{q} = [q_1, q_2, q_3]^T$ and $\dot{\mathbf{q}} = [\dot{q}_1, \dot{q}_2, \dot{q}_3]^T$ are the generalized coordinate vector and the generalized velocity vector, respectively; $\tau = [\tau_1, \tau_2, \tau_3]^T$ is the input torque vector; the ith dynamic parameters are the mass m_i, the moment of inertia I_i with respect to the axis y, the length l_{ci} and angle γ_i of the mass center; and the ith kinematic parameters are the link's length l_i. The robotic system includes three degrees of freedom $(d.o.f)$ in the joint space which provide the ability to move the tip of the end effector $(\hat{x}, \hat{z})$ in the plane $X - Z$ with an orientation $\hat{\phi}$ with respect to the axis X of the coordinate system $X - Z$. A particular characteristic of this system is the parallel mechanism. This mechanism provides the ability to achieve higher precision and higher stiffness in the wrist coordinates $(\hat{x}_m, \hat{z}_m)$ of the system such as a parallel robot [38]. The inclusion of the fifth link shows that the robotic system can be considered as a parallel-serial robot.

Considering the formalism of Euler–Lagrange [39], the dynamic model of the robotic system can be derived. Within this formalism, the Lagrangian is formulated as (4), where $K_i(\mathbf{q}, \dot{\mathbf{q}})$ and $P_i(\mathbf{q})$ are the kinetic and potential energy of the ith link.



a)

Fig. 1 Schematic diagram of the three-degree-of-freedom (d.o.f) manipulator with serial-parallel mechanism

$$L(\mathbf{q}, \dot{\mathbf{q}}) = \sum_{i=1}^{5} K_i(\mathbf{q}, \dot{\mathbf{q}}) - \sum_{i=1}^{5} P_i(\mathbf{q}) \tag{4}$$

The kinetic and potential energy of the ith link is computed as in (5)–(6), where $\mathbf{v}_i \in \mathbb{R}^3$ is the linear velocity of the mass center with respect to the inertial frame, ${}^i\mathbf{w}_i \in \mathbb{R}^3$ is the angular velocity of the mass center with respect to the ith frame, ${}^{ci}\mathbf{I}_i \in \mathbb{R}^{3\times 3}$ is the inertia tensor of the ith link with respect to its mass center, h_i is the distance of the mass center with respect to the axis z and g is the gravitational acceleration.

$$\begin{aligned}
K_i &= \frac{1}{2} m_i \mathbf{v}_i^T \mathbf{v}_i + \frac{1}{2} {}^i\mathbf{w}_i^T {}^{ci}\mathbf{I}_i {}^i\mathbf{w}_i \\
K &= \dot{q}_1^2 (m_1 l_{c1}^2 + I_1)/2 + \dot{q}_2^2 (m_2 l_{c2}^2 + I_2)/2 + m_3 (l_2 \dot{q}_2 \cos(q_2) \\
&\quad + l_{c3} \dot{q}_1 \cos(\gamma_3 + q_1))^2/2 + I_5 (\dot{q}_2/2 + \dot{q}_3/2)(\dot{q}_2 + \dot{q}_3) + I_4 \dot{q}_2^2/2 \\
&\quad + m_5 (\dot{q}_2 (l_4 \cos(q_2) + l_{c5} \cos(\gamma_5 + q_2 + q_3)) - l_1 \dot{q}_1 \cos(q_1) \\
&\quad + l_{c5} \dot{q}_3 \cos(\gamma_5 + q_2 + q_3))^2/2 + m_4 (l_1 \dot{q}_1 \sin(q_1) - l_{c4} \dot{q}_2 \sin(\gamma_4 + q_2))^2/2 \\
&\quad + m_3 (l_2 \dot{q}_2 \sin(q_2) + l_{c3} \dot{q}_1 \sin(\gamma_3 + q_1))^2/2 + m_5 (\dot{q}_2 (l_4 \sin(q_2) \\
&\quad + l_{c5} \sin(\gamma_5 + q_2 + q_3)) - l_1 \dot{q}_1 \sin(q_1) + l_{c5} \dot{q}_3 \sin(\gamma_5 + q_2 + q_3))^2/2 \\
&\quad + m_4 (l_1 \dot{q}_1 \cos(q_1) - l_{c4} \dot{q}_2 \cos(\gamma_4 + q_2))^2/2 + I_3 \dot{q}_1^2/2 \qquad (5) \\
P_i &= m_i g h_i \\
P &= m_1 g\, (l_{c1} \sin(q_1 + \gamma_1)) + m_2 g\, (l_{c2} \sin(q_2 + \gamma_2)) + m_3 g\, (l_2 \sin(q_2) \\
&\quad + l_{c3} \sin(q_1 + \gamma_3)) + m_4 g\, (l_1 \sin(q_1) - l_{c4} \sin(q_2 + \gamma_4)) \\
&\quad + m_5 g\, (l_1 \sin(q_1) - l_4 \sin(q_2)) - m_5 g l_{c5} \sin(q_2 + q_3 + \gamma_5) \qquad (6)
\end{aligned}$$

Then, the dynamic model is obtained by satisfying the Euler–Lagrange Eq. (7).

$$\frac{d}{dt}\left[\frac{\partial L}{\partial \dot{\mathbf{q}}_i}\right] - \frac{\partial L}{\partial \mathbf{q}_i} = \tau_i \ \forall\ i = 1, 2, 3 \tag{7}$$

Considering the closed form of motion equations for the three-degree- of-freedom (d.o.f) manipulator with parallel mechanism, the state space $\mathbf{x} = [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9]^T = [\mathbf{q}, \dot{\mathbf{q}}, \int_0^t e_1(t)\mathrm{d}t, \int_0^t e_2(t)\mathrm{d}t, \int_0^t e_3(t)\mathrm{d}t]^T \in \mathbb{R}^9$ and the input vector $\mathbf{u} = [u_1, u_2, u_3]^T = \tau \in \mathbb{R}^3$, the dynamic model of the robotic system can be expressed as in (8), where $\mathbf{M} \in \mathbb{R}^{3\times3}$ is the inertia matrix, $\mathbf{C}[x_3, x_4, x_5]^T \in \mathbb{R}^3$ is the centrifugal and Coriolis force vector, $\mathbf{G} \in \mathbb{R}^3$ is the gravity vector, $M_{ij} \subset M, C_{ij} \subset C$, $G_i \subset G\ \forall\ i, j \in \{1, 2, 3\}$. The ith Cartesian position error and Cartesian velocity error are represented as $e_i = \bar{\xi}_i - \xi_i, \dot{e}_i = \dot{\bar{\xi}}_i - \dot{\xi}_i\ \forall\ i = 1, 2, 3$, respectively, where $\xi = [\xi_1, \xi_2, \xi_3] = [\hat{x}, \hat{z}, \hat{\phi}]$ is the Cartesian coordinate vector of the robotic system's end effector (position and orientation in the Cartesian space) and $\bar{\xi} = [\bar{\xi}_1, \bar{\xi}_2, \bar{\xi}_3] = [\bar{\hat{x}}, \bar{\hat{z}}, \bar{\hat{\phi}}]$ is the desired Cartesian coordinate vector to be followed by the end effector.

$$\begin{aligned}
\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} &= \mathbf{f}\,(\mathbf{x}, \mathbf{u}) \\
\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} &= \begin{bmatrix} \begin{bmatrix} x_4 & x_5 & x_6 \end{bmatrix}^T \\ \mathbf{M}^{-1}\left(\mathbf{u} - \mathbf{C}\begin{bmatrix} x_3 & x_4 & x_5 \end{bmatrix}^T - \mathbf{G}\right) \\ \begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix}^T \end{bmatrix}
\end{aligned} \tag{8}$$

where:

$$\begin{aligned}
M_{11} &= I_1 + I_3 + l_1^2 m_5 + l_{c3}^2 m_3;\ M_{22} = m_3 l_2^2 + m_5 l_4^2 + 2m_5 \cos(q_3 + \gamma_5) l_4 l_{c5} \\
&\quad + m_2 l_{c2}^2 + m_4 l_{c4}^2 + m_5 l_{c5}^2 + I_2 + I_4 + I_5 \\
M_{33} &= m_5 l_{c5}^2 + I_5;\ M_{32} = M_{23} = m_5 l_{c5}^2 + l_4 m_5 \cos(q_3 + \gamma_5) l_{c5} + I_5; \\
M_{31} &= M_{13} = -l_1 l_{c5} m_5 \cos(-q_1 + q_2 + q_3 + \gamma_5) \\
M_{12} &= M_{21} = l_2 l_{c3} m_3 \cos(q_1 - q_2 + \gamma_3) - l_1 l_{c4} m_4 \cos(-q_1 + q_2 + \gamma_4) \\
&\quad - l_1 l_{c5} m_5 \cos(-q_1 + q_2 + q_3 + \gamma_5) \\
&\quad - l_1 l_4 m_5 \cos(q_1 - q_2) \\
C_{12} &= (l_2 l_{c3} m_3 \sin(q_1 - q_2 + \gamma_3) + l_1 l_{c4} m_4 \sin(-q_1 + q_2 + \gamma_4) \\
&\quad + l_1 l_{c5} m_5 \sin(-q_1 + q_2 + q_3 + \gamma_5) \\
&\quad - l_1 l_4 m_5 \sin(q_1 - q_2))\dot{q_2} + l_1 l_{c5} m_5 \sin(-q_1 + q_2 + q_3 + \gamma_5)\dot{q_3} \\
C_{13} &= l_1 l_{c5} m_5 \sin(-q_1 + q_2 + q_3 + \gamma_5)(\dot{q_2} + \dot{q_3}) \\
C_{21} &= l_1 l_4 m_5 \sin(q_1 - q_2)\dot{q_1} - l_1 l_{c5} m_5 \sin(-q_1 + q_2 + q_3 + \gamma_5)\dot{q_1} \\
&\quad - l_2 l_{c3} m_3 \sin(q_1 - q_2 + \gamma_3)\dot{q_1} \\
&\quad - l_1 l_{c4} m_4 \sin(-q_1 + q_2 + \gamma_4)\dot{q_1}
\end{aligned}$$

$$C_{22} = -l_4 l_{c5} m_5 \sin(q_3 + \gamma_5)\dot{q_3};\ C_{23} = -l_4 l_{c5} m_5 \sin(q_3 + \gamma_5)(\dot{q_2} + \dot{q_3});$$
$$C_{31} = -l_1 l_{c5} m_5 \sin(-q_1 + q_2 + q_3 + \gamma_5)\dot{q_1}$$
$$C_{32} = l_4 l_{c5} m_5 \sin(q_3 + \gamma_5)\dot{q_2};\ C_{11} = C_{33} = 0$$
$$G_1 = g(l_1 m_4 \cos(q_1) + l_1 m_5 \cos(q_1) + l_{c1} m_1 \cos(q_1 + \gamma_1) + l_{c3} m_3 \cos(q_1 + \gamma_3));$$
$$G_3 = -g l_{c5} m_5 \cos(q_2 + q_3 + \gamma_5)$$
$$G_2 = g l_{c2} m_2 \cos(q_2 + \gamma_2) - g m_5 (l_4 \cos(q_2) + l_{c5} \cos(q_2 + q_3 + \gamma_5)) - g l_{c4} m_4 \cos(q_2 + \gamma_4) + g l_2 m_3 \cos(q_2)$$

In order to achieve high precision and accuracy control, the control system is designed in the operational space. The proportional-integral-derivative (PID) control in the operational space, shown in (9), is implemented to track a desired trajectory in the Cartesian space. The constants k_{p_i}, k_{i_i}, k_{d_i} are the ith proportional, integral and derivative gains of the PID control.

$$u_i = \breve{\mathbf{J}}^T \left(k_{p_i} e_i + k_{d_i} \dot{e}_i + k_{i_i} \int_0^t e_i(t) \mathrm{d}t \right) \quad \forall\, i = 1, 2, 3 \tag{9}$$

The PID control in the operational space requires adding the last three dynamics in (8). This control system does not require the inverse kinematic of the robotic system but requires the direct kinematic and the Jacobian matrix $\breve{\mathbf{J}} \in \mathbb{R}^{3\times 3}$ to transform the system outputs from the joint to Cartesian space. In addition, the Jacobian matrix is required to transform from the Cartesian force/torque vector to joint torque vector $\mathbf{u}$. The direct kinematic and the elements of the Jacobian matrix are displayed in (10), (11) and (12), respectively.

$$\hat{x} = l_1 \cos(q_1) - l_4 \cos(q_2) - l_5 \cos(q_2 + q_3);$$
$$\hat{y} = l_1 \sin(q_1) - l_4 \sin(q_2) - l_5 \sin(q_2 + q_3);\ \hat{\phi} = q_2 + q_3 - \pi \tag{10}$$
$$\breve{J}_{11} = l_1 \sin(q_1);\ \breve{J}_{12} = l_4 \sin(q_2) + l_5 \sin(q_2 + q_3);$$
$$\breve{J}_{13} = l_5 \sin(q_2 + q_3);\ \breve{J}_{31} = 0;\ \breve{J}_{32} = \breve{J}_{33} = 1 \tag{11}$$
$$\breve{J}_{21} = l_1 \cos(q_1);\ \breve{J}_{22} = -l_4 \cos(q_2) - l_5 \cos(q_2 + q_3);$$
$$\breve{J}_{23} = -l_5 \cos(q_2 + q_3) \tag{12}$$

3.2 Design Parameter Vector

The design parameters of the control system involve the proportional, integral and derivative gains of the PID control in the operational space due to these parameters influencing the behavior of the closed-loop system. The control design parameters are grouped in $\mathbf{p_c} = [k_{p_1},\ k_{i_1},\ k_{d_1},\ k_{p_2},\ k_{i_2},\ k_{d_2},\ k_{p_3},\ k_{i_3},\ k_{d_3}]^T \in \mathbb{R}^9$.

On the other hand, the closed-loop system behavior is also influenced by the mechanical structure of the robotic system. The best selection of dynamic and kinematic parameters of links can improve the closed-loop system performance. As the dynamic and kinematic parameters of links depend on its geometry, a

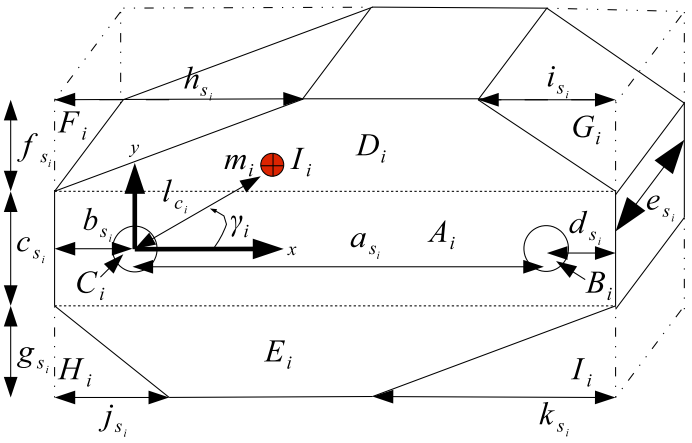


Fig. 2 Schematic representation of the isometric view of the ith link

prism shape with eight sides (see Fig. 2) is assumed. Then, the geometric parameters of links are proposed as the mechanical structure design vector $\mathbf{p_s} = [a_{s_1} \dots a_{s_5}, b_{s_1} \dots b_{s_5}, c_{s_1} \dots c_{s_5}, d_{s_1} \dots d_{s_5}, e_{s_1} \dots e_{s_5}, f_{s_1} \dots f_{s_5}, g_{s_1} \dots g_{s_5}, h_{s_1} \dots h_{s_5}, i_{s_1} \dots i_{s_5}, j_{s_1} \dots j_{s_5}, k_{s_1} \dots k_{s_5}]^T \in \mathbb{R}^{55}$. For more details on how to obtain the dynamic and kinematic parameters of links from its geometric parameters consult [40].

Thus, the structure-control design parameter vector $\mathbf{p}$ is given by (13).

$$\mathbf{p} = [\mathbf{p_s},\ \mathbf{p_c}]^T \in \mathbb{R}^{64} \tag{13}$$

3.3 Structure-Control Design Objectives

One of the main issues related to the design of a robotic system is the precision and accuracy to perform a desired trajectory. Therefore, the integral squared error (ISE) in the Cartesian space ($\xi \in \mathbb{R}^3$) shown in (14) is used as one of the objective functions to be minimized in the MODOP.

$$\bar{J}_1 = \frac{1}{t_f} \sum_{i=1}^{3} \int_0^{t_f} e_i^2(t)\, \mathrm{d}t \tag{14}$$

Another aspect to consider is the energy consumption of the robotic system. The energy consumption can be related to the torque control signal $\mathbf{u} = \tau$ provided by the controller. The torque control signal can be decreased by changing both the control gains and the mechanical structure. Consequently, the performance function given in (15) is used as the second performance function.

$$\bar{J}_2 = \frac{1}{t_f} \int_0^{t_f} \|u(t)\|^2 \mathrm{d}t \tag{15}$$

3.4 Design Constraints

Design constraints involve both static and dynamic constraints. The first is imposed by the desired trajectory to be executed by the robotic system end effector. A hypocycloid trajectory in the $X - Z$ Cartesian space and a cosine trajectory, given in (16)–(18), are selected as the desired because they are difficult tasks (highly nonlinear behavior) and the structure-control design approach can be highlighted.

$$h_1(t) :0.32 + 0.0801\cos(1.2566t) + 0.0178\cos(5.6548t) - \bar{\hat{x}} = 0 \tag{16}$$

$$h_2(t) :0.2 + 0.0801\sin(1.2566t) - 0.0178\sin(5.6548t) - \bar{\hat{z}} = 0 \tag{17}$$

$$h_3(t) :0.4363\cos(1.2566t) - \bar{\hat{\phi}} = 0 \tag{18}$$

The second constraint is the dynamic behavior of the closed-loop system which can be simplified as in (19). The constraint (19) includes, in a single equation, the dynamic model of the three-degree-of-freedom (d.o.f) manipulator (8) with its PID control in the operational space (9). Due to this constraint being provided by a differential equation, an initial condition vector is required. The initial condition is set to $\mathbf{x}(0) = \mathbf{x}_{INI}$, where $\mathbf{x}_{INI} = [1.6, 2.76, 0.82, 0, 0, 0, 0, 0, 0]^T$.

$$h_4(x, p, t) : \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} - \mathbf{f}(\mathbf{x}, \mathbf{p}) = 0 \tag{19}$$

The third constraint provides the torque limits of the control system $\mathbf{u}$ and these are shown in (20), where $u_{Max} = 10\,\mathrm{Nm}$ represents the torque limit.

$$g_i(\mathbf{x}, \mathbf{p}, t) : |\bar{u}_i(t)| - u_{Max} \leq 0 \; \forall \; i = 1, 2, 3 \tag{20}$$

A bad selection of the design variable vector $\mathbf{p}$ (control gains as well as of geometric parameters) may destabilize the closed-loop system [41]. When this behavior is presented, the robotic system yields undesirable higher-order dynamics which the system cannot control. This effect makes the solver algorithm of the differential equation produce infinity or even an indeterminate number in the objective functions and constraints, such that the solver algorithm can increase its convergence time, or even produce a buffer overflow in the software. In order to face this issue, the constraints given in (21) are proposed, where the maximum value for the state vector is selected as $Tol_{Max} = 1e100$.

$$g_{j+3}(\mathbf{x}, \mathbf{p}, t) : |x_i(t)| - Tol_{Max} < 0, \forall j = 1, 2, \ldots, 9 \tag{21}$$

The last constraints are related to the upper and lower bounds in the design variable vector $\mathbf{p}$. These are stated in (22), where $\mathbf{p}_{Min}$ and $\mathbf{p}_{Max}$ are the minimum and maximum bounds. The minimum and maximum bounds are shown in Table 1.

$$g_{i+12} : \mathbf{p}_{Min} \leq \mathbf{p} \leq \mathbf{p}_{Max}, \forall i = 1, 2, \ldots, 64 \tag{22}$$

Table 1 Design variable vector bound

Geometric parameter	Upper bound (m)	Lower bound (m)
$p_1 - p_5$	0.5	0.35
$p_6 - p_7$	0.3	0.01905
$p_8 - p_{10}$	0.3	0.015
$p_{11} - p_{12}$	0.1	0.0381
$p_{13} - p_{15}$	0.1	0.03175
$p_{16} - p_{19}$	0.3	0.015
p_{20}	0.03	0.015
$p_{21} - p_{25}$	0.03	0.00635
$p_{26} - p_{35}$	0.1	0
p_i	$p_{i-35} + p_{i-30} + p_{i-20}\ \forall i = 36, \ldots, 40$	0
p_i	$p_{i-40} + p_{i-35} + p_{i-25} - p_{i-5}\ \forall i = 41, \ldots, 45$	0
p_i	$p_{i-45} + p_{i-40} + p_{i-30}\ \forall i = 46, \ldots, 50$	0
p_i	$p_{i-50} + p_{i-45} + p_{i-35} - p_{i-5}\ \forall i = 51, \ldots, 55$	0
Control parameter	Upper bound	Lower bound
p_{56}	15000	0
p_{57}	155000	0
p_{58}	50	0
p_{59}	10000	0
p_{60}	55000	0
p_{61}	50	0
p_{62}	1000	0
p_{63}	5000	0
p_{64}	50	0

3.5 Multi-Objective Dynamic Optimization Problem Statement

The synergetic design of both the three-degree-of-freedom manipulator with parallel mechanism and the PID control system in the operational space is established as a multi-objective dynamic optimization problem. Such a problem consists in simultaneously finding the optimum structural geometry of links and the optimum PID control gains [grouped in vector $\mathbf{p} \in \mathbb{R}^{64}$ (13)] such that minimize two performance functions related to the structure design (15) and control (14). This problem is subject to the dynamics of the closed-loop system (19), the maximum torque applied to the motors (20), the desired trajectory to be executed by the end effector of the robotic system (16)–(18), bounds in the structure-control design parameter vector $\mathbf{p}$ (22), as well as, the constraint to avoid undesirable higher-order dynamics in the robotic system (21). The formal statement of the mathematical programming problem for the study case is given in (23).

$$\underset{\mathbf{p}}{Min}\ [\bar{J}_1, \bar{J}_2]^T \tag{23}$$

subject to:

$$\begin{gathered}\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = f(\mathbf{x}, \mathbf{p}, t), \mathbf{x}(0) = \mathbf{x}_{INI} \\ \mathbf{g}(\mathbf{x}, \mathbf{p}, t) < 0, \quad \in \mathbb{R}^{12} \\ \mathbf{h}(\mathbf{x}, \mathbf{p}, t) = 0, \quad \in \mathbb{R}^{3} \\ \mathbf{h}(t) = 0, \quad \in \mathbb{R}^{3} \\ \mathbf{p}_{Min} \leq \mathbf{p} \leq \mathbf{p}_{Max} \in \mathbb{R}^{64}\end{gathered}$$

As the dynamics of the closed-loop system is a nonlinear differential equation, the Euler method is used with an integration step of $\Delta t = 5\,\mathrm{ms}$, final time of $t_f = 10\,\mathrm{s}$ and the initial condition $\mathbf{x}(0) = \mathbf{x}_{INI}$.

4 Results

The experimental results consist in comparing the performance of the proposed multi-objective exhaustive exploitation differential evolution algorithm with the performance of the multi-objective differential evolution without the exhaustive exploitation mechanism (called MODE). The algorithms are used to solve the structure-control design framework presented in Sect. 3. Five experiments are set; in the first four experiments, the exhaustive exploitation factor (EEF) is changed in the MOEEDE to analyze its influence in the generation of non-dominated solutions for the structure-control design already introduced. When the value of EFF approximates to 1, high exploration of the search space and low exploitation of the non-dominated solutions are achieved. Meanwhile, when the value of EFF approximates to 0, high exhaustive exploitation of the non-dominated solutions and low exploration of the search space are carried out. The values of the exhaustive exploitation factor are selected as follows: $EEF = 0.4$ for the Experiment 1, $EEF = 0.54$ for the Experiment 2, $EEF = 0.73$ for the Experiment 3 and $EEF = 0.9$ for the Experiment 4. The fifth experiment consists in using a MODE without the exhaustive exploitation mechanism. The first four experiments are compared with the fifth experiment.

Making a fair comparison, two considerations are taken into account: (i) The scale factor F and crossover probability CR values are obtained in the same way for all experiments. The first parameter is randomly chosen in the intervals $F \in [0.3, 0.9]$, and the second is set to $CR = 0.85$. (ii) The population number and the maximum number of generation for the fifth experiment is chosen such that the objective function evaluation number in the MODE corresponds to the maximum number of objective function evaluation among the first four experiments with MOEEDE. In this case, Experiment 1 with $EEF = 0.4$ evaluates more objective functions (9600 evaluations); then, the same function evaluation is given in Experiment 5. According to the previous discussion, the parameters of the MOEEDE algorithm for the first four experiments are chosen as: maximum number of individuals in the base population $BP = 100$, maximum number of individuals in the second base population $BP2 = 200$, maximum number of individuals in the external population $MP = 1000$, maximum number of individuals in a hyper-grid $n_{MaxInd} = 40$, number of row $n_{\mathrm{row}} = 10$ and column $n_{\mathrm{col}} = 10$ of the self-adaptive grid, $G_{MaxMOEEDE} = 6000$, and the parameters of the MODE are set as: $NP = 100$, $G_{MaxMODE} = 9600$. Furthermore, the stop criterion for

Table 2 Non-dominated solutions per number of iterations and the convergence time of the MOEEDE

Algorithm	Run	Non-dominated solutions per iteration number				D	S	Convergence time (h)
		3000	4000	5000	6000			
Exp. 1 $EEF = 0.40$								
MOEEDE	1	172	498	275	354	0.3952	0.0033	35.65
MOEEDE	2	379	412	491	314	0.3584	0.0063	36.83
MOEEDE	3	306	545	402	470	0.4491	0.0013	36.61
MOEEDE	4	397	466	502	515	0.3819	0.0012	36.00
MOEEDE	5	324	434	468	446	0.4001	0.0030	34.67
Average		316	471	428	420	0.3970	0.0030	36.19
Exp. 2 $EEF = 0.54$								
MOEEDE	1	12	331	462	505	0.3756	0.0010	32.75
MOEEDE	2	16	378	486	531	0.2634	0.0027	31.44
MOEEDE	3	5	305	336	569	0.7407	0.0002	31.17
MOEEDE	4	17	421	330	437	0.4510	0.0007	32.28
MOEEDE	5	14	331	546	535	0.4756	0.0011	32.58
Average		13	353	432	515	0.4612	0.0012	31.78
Exp. 3 $EEF = 0.73$								
MOEEDE	1	15	20	249	423	0.5256	0.0006	26.81
MOEEDE	2	10	13	331	441	0.5525	0.0006	26.58
MOEEDE	3	10	16	340	398	0.5555	0.0007	26.90
MOEEDE	4	13	17	323	342	0.5326	0.0158	26.52
MOEEDE	5	0	9	451	558	0.5698	0.0004	26.99
Average		10	15	339	432	0.5472	0.0036	26.71
Exp. 4 $EEF = 0.90$								
MOEEDE	1	12	20	11	349	0.6902	0.0020	21.84
MOEEDE	2	1	3	8	145	0.7439	0.0020	21.18
MOEEDE	3	18	22	22	382	0.6869	0.0015	21.47
MOEEDE	4	12	25	25	463	0.9240	0.0016	22.31
MOEEDE	5	4	12	17	314	0.6933	0.0026	22.40
Average		9	16	17	331	0.7477	0.0020	21.71

both algorithms is the maximum generation number, $G_{MaxMOEEDE}$ and $G_{MaxMODE}$ for the MOEEDE and MODE, respectively.

Five independent runs are carried out for each experiment using a PC with a 3.5 GHz Intel Core(TM) i7-4770K processor with 16GB of RAM. All runs are programmed in MATLAB. The numerical results are shown in Table 2 for the MOEEDE and in Table 3 for the MODE. The spread of the obtained non-dominated solutions in the objective space is measured by using the spacing metric [42], and the term "S" is used for this purpose. A value of zero for this metric indicates that the non-dominated solutions are equally spaced. The number of non-dominated solutions per single run

Table 3 Non-dominated solutions per number of iterations and the convergence time of the MODE without the exhaustive exploitation mechanism

Algorithm	Run	Non-dominated solutions per iteration number								D	S	Convergence time (h)
		3000	4000	5000	6000	7000	8000	9000	9600			
MODE	1	16	18	15	16	16	17	17	17	1.3097	0.1331	30.22
MODE	2	11	11	19	13	17	19	16	22	1.2980	0.1019	30.30
MODE	3	17	13	15	11	11	14	14	14	1.1980	0.0993	29.31
MODE	4	2	4	7	4	13	22	15	16	0.7258	0.0441	27.87
MODE	5	10	18	20	17	22	21	22	23	1.3009	0.0975	30.44
Average		11	13	15	12	16	19	17	18	1.1665	0.0952	29.65

per some generations (intermediate and final generations) and the convergence time in the last generation are also included in Tables 2 and 3. An additional metric, called distance from the origin (named as "D"), is proposed to show how far the Pareto front is from the minimum possible value of the performance functions, i.e., from the utopia point $[\bar{J}_1, \bar{J}_2] = [0, 0]$. This metric is defined as in (24), where $n_{\mathfrak{P}^*}$ is the number of non-dominated solutions at the end of the search and $d_{\bar{J}_1}$ and $d_{\bar{J}_2}$ are the normalized difference vectors from the fitness of Pareto solutions in $\bar{J}_1 \in \mathfrak{PF}$ and $\bar{J}_2 \in \mathfrak{PF}$ to the origin, respectively. A value of zero for the metric "D" means that the Pareto front is the closest to the origin, i.e., the Pareto front presents the best trade-offs.

$$D = \sum_{i=1}^{n_{\mathfrak{P}^*}} \sqrt{\frac{d_{\bar{J}_1}^2 + d_{\bar{J}_2}^2}{n_{\mathfrak{P}^*}^2}} \tag{24}$$

According to the results for the first four experiments previously presented in Table 2, different findings are observed:

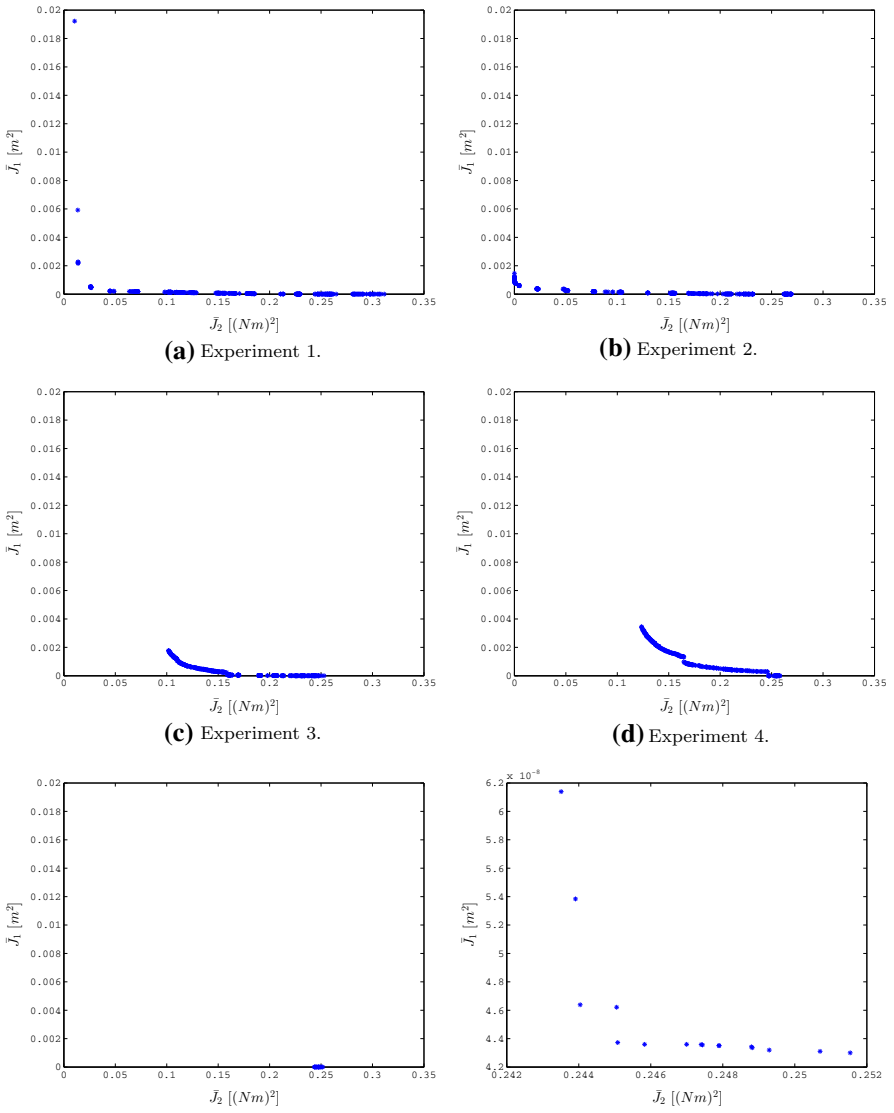
- The average number of non-dominated solutions through the generations (see the column non-dominated solutions per iteration number) increases at least 18 times when the exhaustive exploitation mechanism is activated, i.e., when the generation number is greater than $EEF * G_{MaxMOEEDE}$. Then, the exhaustive exploitation mechanism promotes the generation of more non-dominated solutions.
- The activation moment of the exhaustive exploitation mechanism influences in the search for better trade-offs between the performance functions. While more generations use the exhaustive exploitation mechanism in the MOEEDE, better trade-offs in the Pareto front are obtained. Results from Experiment 1 ($EEF = 0.4$), Experiment 2 ($EEF = 0.54$), Experiment 3 ($EEF = 0.73$) and Experiment 4 ($EEF = 0.9$) include the exhaustive exploitation mechanism at 60, 46, 27 and 10% of their generations, respectively. The average of the distance from the origin to the Pareto front in Experiment 1 ($D = 0.3970$) was the lowest of the four experiments, whereas the highest average distance from the origin to the Pareto front is given in Experiment 4 ($D = 0.7477$). These results are confirmed in

column "D" of Table 2. So, by assigning the exhaustive exploitation factor as $EEF = 0.4$, the average performance of the MOEEDE provides a Pareto front with better trade-offs.
– It is clear that the average convergence time increases as the use of the exhaustive exploitation mechanism increases because it requires more function evaluation. Thus, the average time in Experiment 1 was the highest (36.19 h) of the first four experiments, whereas the corresponding average time in Experiment 4 was the lowest (21.71 h).
– In the experiments, a set of non-dominated solutions with the best average uniformly distributed Pareto front (see column "S" in Table 2) correspond to Experiment 2. However, it is observed that there is not a clear relation between the spacing metric "S" and the exhaustive exploitation factor EEF among the experiments.

Comparing the numerical results between the performance of the MOEEDE (with exhaustive exploitation mechanism; first four experiments) and the performance of the MODE (without the exhaustive exploitation mechanism; fifth experiment) presented in Tables 2 and 3, respectively, the following is found:

– More non-dominated solutions are discovered by using the exhaustive exploitation mechanism. An average of 18 non-dominated solutions are found at the end of the search by using MODE; meanwhile, the MOEEDE presents the highest average of 515 non-dominated solutions in Experiment 2, and the lowest average of 331 non-dominated solutions in Experiment 4. Hence, the MOEEDE increments at least 18 times the number of non-dominated solutions found in MODE. In addition, the average distance from the origin to the Pareto front obtained by the runs of the first four experiments with MOEEDE presents a lower distance than those providing by MODE (see columns "D" in Table 2 and 3). The same happens with the average of the spacing metric (see columns "S" in Tables 2 and 3). This indicates that the exhaustive exploitation mechanism in MOEEDE promotes the generation of more non-dominated solutions with better trade-offs (lower values to the objective functions in a minimization problem) and a spacing between the non-dominated solutions more uniform in the Pareto front than the MODE which does not include this mechanism.
– Comparing the average convergence time of Experiment 1 (with exhaustive exploitation mechanism) with those of Experiment 5 (without exhaustive exploitation mechanism), where the evaluation of the objective function is the same, it is observed that the average convergence time in Experiment 1 (36.19h) by using exhaustive exploitation mechanism increases with respect to the convergence time required without using its mechanism in Experiment 5 (29.65h). Therefore, the convergence time is increased around 18.07% by the use of the exhaustive exploitation mechanism.

In order to show the resulting Pareto front obtained by each experiment, the five fronts given in each run per experiment are filtered into a single front and those are visualized in Fig. 3. The non-dominated solution, the distance from the origin metric and the spacing metric for the filtered results are given in Table 4. Based on the filtered results, some findings are stated:



(a) Experiment 1. **(b)** Experiment 2.

(c) Experiment 3. **(d)** Experiment 4.

(e) Experiment 5. Zoom of Experiment 5.

Fig. 3 Filtered Pareto front for the five experiments

Table 4 Non-dominated solutions in the filtered Pareto fronts per experiment

Experiment	Non-dominated solutions	D	S
1	358	0.4525	0.0027
2	542	0.2996	0.0026
3	434	0.4849	0.0010
4	327	0.5343	0.0016
5	16	0.7258	0.1096

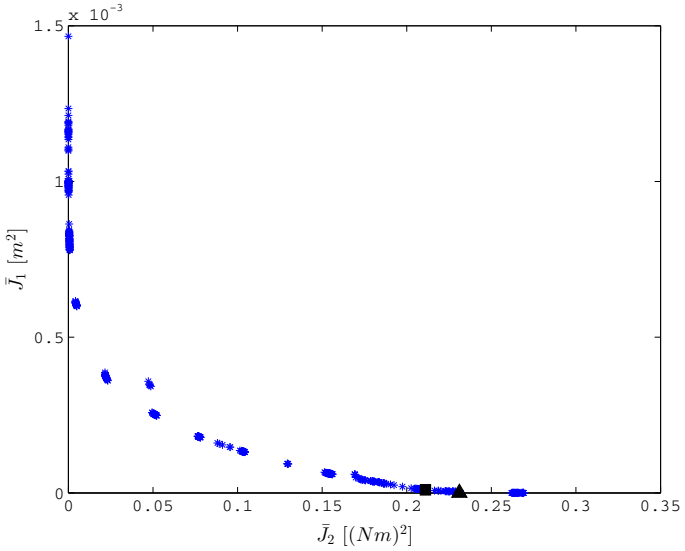


Fig. 4 Selection of two different non-dominated solutions in the filtered Pareto front of Experiment 2. Square marker: $[\bar{J}_1, \bar{J}_2] = [9.9\text{E}{-6}, 0.211]$. Triangle marker: $[\bar{J}_1, \bar{J}_2] = [0.0522\text{E}{-6}, 0.231]$

- Figure 3 confirms the numerical results given by the distance from the origin metric "D" in Table 4, i.e., the best filtered Pareto front is given by Experiment 2 ($D = 0.2996$) and the worst performance in the filtered Pareto front by Experiment 5 ($D = 0.7258$).
- It is observed that the **non**-inclusion of the exhaustive exploitation mechanism in the MODE makes that the Pareto front presents a poor trade-off between the performance functions as is displayed in Experiment 5 of Fig. 3.
- The superior performance of the MOEEDE increases the number of non-dominated solutions, at least 2000%, i.e., from 16 non-dominated solutions in Experiment 5 with MODE to the minimum non-dominated solutions of 327 in Experiment 4 with MOEEDE. In addition, the spacing of the obtained non-dominated solutions in the phenotype space is more uniform with MOEEDE in the first four experiments ($S \in [0.0010, 0.0027]$) than that obtained with MODE in Experiment 5 ($S = 0.11$).
- All obtained non-dominated solutions yielded by MODE at the end of the search are dominated by the non-dominated solutions found in MOEEDE for all experiments.

Previous results show the superior performance of the exhaustive exploitation mechanism in the generation of a better Pareto front which provides the engineer with a broad set of possible design solutions in the structure-control design framework. Then, the decision-making process of the structure-control design solutions (non-dominated solutions) is presented. Two different non-dominated solutions (structure-control design solutions) in the filtered Pareto front of Experiment 2 are taken into account. The filtered Pareto front of Experiment 2 is again presented in Fig. 4, and the selected non-dominated solutions are visualized with a square marker (SM) and a triangle marker (TM). The square marker has the performance $[\bar{J}_1, \bar{J}_2] = [9.9\text{E}{-6}, 0.211]$;

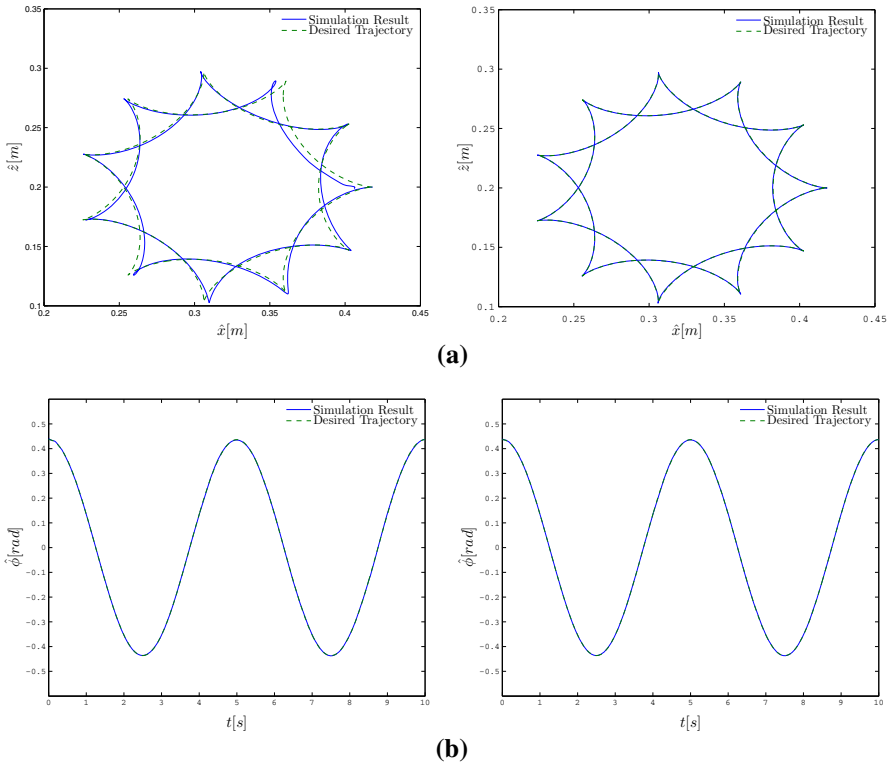


Fig. 5 Behavior in the Cartesian space of the end effector of the parallel robot. *Left column* is with the non-dominated solution of the square marker $[\bar{J}_1, \bar{J}_2] = [9.9\text{E}{-6}, 0.211]$. *Right column* is with the non-dominated solution of the triangle marker $[\bar{J}_1, \bar{J}_2] = [0.0522\text{E}{-6}, 0.231]$. **a** End-effector's linear position and the desired trajectory. **b** End-effector's angular position and the desired trajectory

meanwhile, the triangle marker has the performance $[\bar{J}_1, \bar{J}_2] = [5.22\text{E}{-8}, 0.231]$. The behavior of the parallel robot end effector with the previously chosen structure-control design solutions is displayed in Fig. 5, and the control signals to follow the desired trajectory are shown in Fig. 6. All solutions to the right of the triangle mark in Fig. 4 exhibit a mean error smaller than $0.234\text{E}{-3}$ and $0.564\text{E}{-3}$ for the Cartesian linear and angular position in the trajectory tracking, respectively. Otherwise, the design solution with square mark presents a mean error of $0.560\text{E}{-3}$ and $0.941\text{E}{-3}$. The design solution marked with a triangle fulfills with an acceptable position error in the Cartesian Space such that the trajectory is perfectly traced by the end effector of the parallel robot (see the right column of Fig. 5), but the energy consumption based on the provided torque control is larger than the design solution marked with a square (see the right column of Fig. 6). On the other hand, the design solution marked with a square presents a visible error in the trajectory tracking (see the left column of Fig. 5), in spite of getting close to the chosen non-dominated solutions in the Pareto front. It is clear that the performance functions, i.e., the position error and the energy consumption, of the presented structure-control design are strongly connected, and the

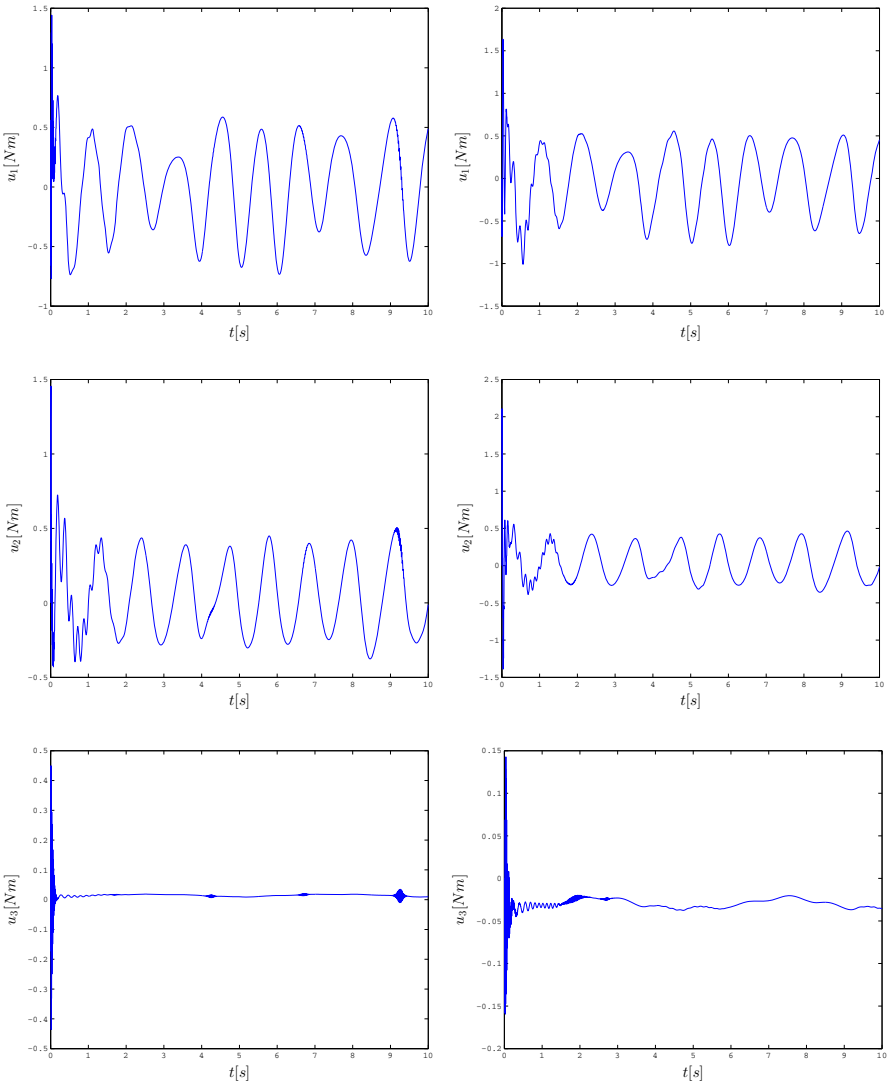


Fig. 6 Control signal behavior of the closed-loop system. *Left column* is with the non-dominated solution of the square marker $[\bar{J}_1, \bar{J}_2] = [9.9\text{E}{-6}, 0.211]$. *Right column* is with the non-dominated solution of the triangle marker $[\bar{J}_1, \bar{J}_2] = [0.0522\text{E}{-6}, 0.231]$

integrated design of the structure and control parameters can improve such trade-offs. Consequently, the decision- making process depends on the robot required precision movement for the specific application.

It is clear that different initial conditions (initial angular configuration of the robot) for the differential equation related to the robotic manipulator dynamic behavior greatly influence in the transient response of the robotic system output error (angular and velocities) to reach the trajectory. However, another question that arises with

Table 5 Performance in both design objectives considering twenty different changes in the initial condition of the robotic system

Design solution	Mean		Median		Std	
	$\bar{J}_1$	$\bar{J}_2$	$\bar{J}_1$	$\bar{J}_2$	$\bar{J}_1$	$\bar{J}_2$
Triangle marker	5.99E−8	0.268	4.59E−8	0.216	2.9E−8	0.109
Square marker	4.58E−6	0.323	3.19E−6	0.209	1.53E−6	0.125

TM: $[\bar{J}_1, \bar{J}_2] = [5.22\text{E}{-8}, 0.231]$ and SM: $[\bar{J}_1, \bar{J}_2] = [9.9\text{E}{-6}, 0.211]$ are the nominal performance (without changes in the initial condition). Std is the standard deviation

different initial conditions is related to the tracking error behavior in the steady state, i.e., once the end effector of the robotic manipulator reaches the trajectory. In order to verify such situation, twenty different changes in the initial conditions are chosen. Those changes provide a proportional increment/decrement from $\pm 10\%$ to $\pm 100\%$ of the initial condition reported in Section 3.4. In Table 5, the statistical analysis of the data given by the Cartesian error $\bar{J}_1$ and the energy consumption $\bar{J}_2$ with those different initial conditions are shown for the two structure-control design solutions marked with a square and a triangle. From the nominal performance (without changes in the initial condition) to the average variation of the performance functions in the steady state, the triangle marker is less affected by initial conditions. In both design solutions, changes in the initial conditions vary the performance functions around 50–60% from their corresponding nominal values. Those variations are represented by the standard deviation in Table 5 and can be considered very small; nevertheless, for applications where the precision is an important issue, robust approach must be included into the structure-control design framework [24] and is beyond the scope of the paper. Then, the initial conditions can vary the nominal performance functions in the steady state response.

The geometric parameters given by the link shapes, and both link dynamic and PID control parameters, corresponding to the previous selected non-dominated solutions, are shown in Fig. 7 and in Table 6, respectively. It is observed that both link dynamic and control parameters simultaneously influence the efficiency of the trajectory tracking and the energy consumption.

From the design engineer's point of view, with the inclusion of the exhaustive exploitation mechanism in the differential evolution algorithm, the decision-making process in the structure-control design framework is more diverse because for a precision of the robot movement (e.g., $\bar{J}_1 \geq 5.22\text{E}{-8}$) there is a set of solutions with different energy consumption. Considering that mechanical parameters (shape of links) are more expensive (due to the manufacture process) than controller implementation, and the robot performance is more influenced by the control parameters, then it is better to have a large set of controller gains than different link shapes. Therefore, the designer can select the design with the minimum required precision which consumes less energy.

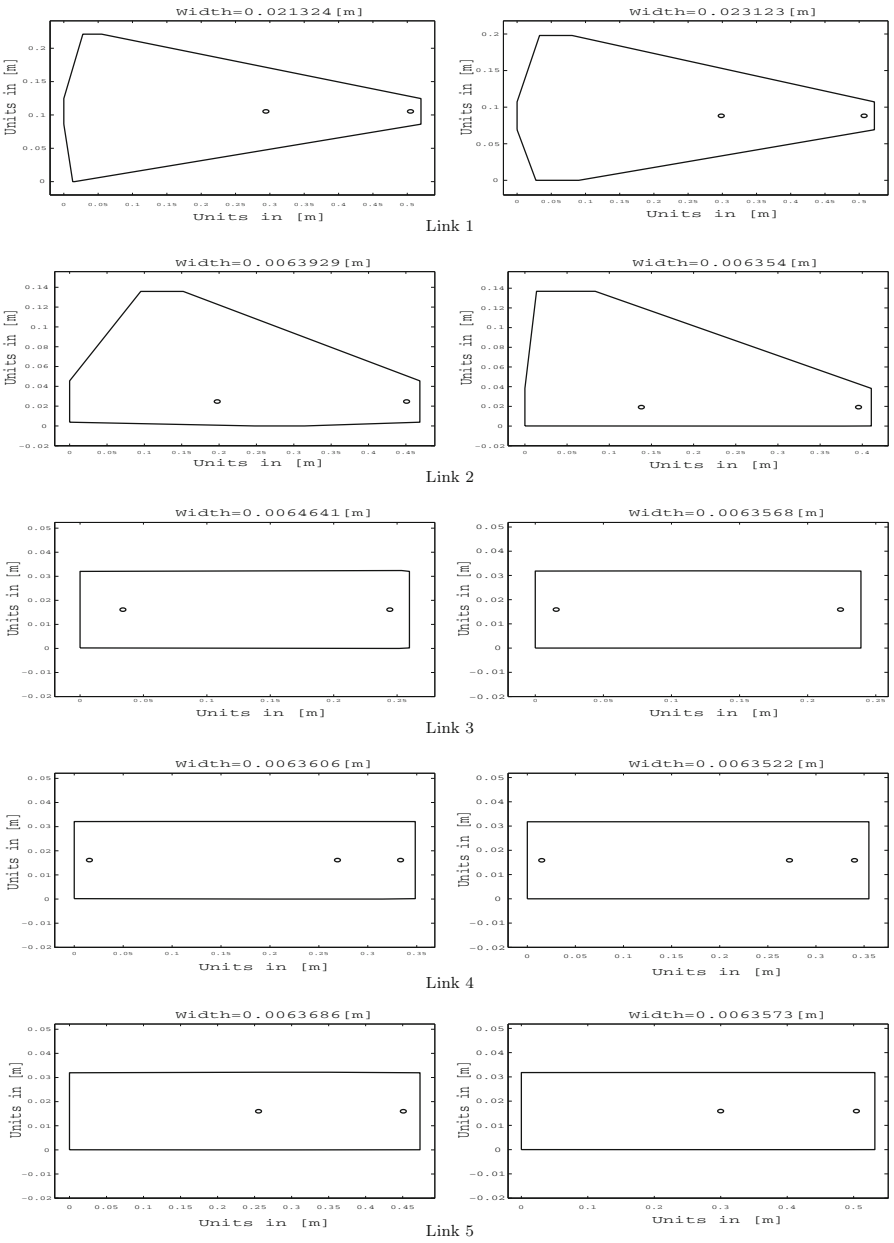


Fig. 7 Shapes of links with the selected non-dominated solutions. *Left column* is with the non-dominated solution of the square marker $[\bar{J}_1, \bar{J}_2] = [9.9\text{E}{-6}, 0.211]$. *Right column* is with the non-dominated solution of the triangle marker $[\bar{J}_1, \bar{J}_2] = [0.0522\text{E}{-6}, 0.231]$

Table 6 Dynamic parameter and PID control gains of the two selected non-dominated solutions

Link i	m_i (m)	l_{ci} (m)	γ_i (rad)	I_i (kg m^2)
Non-dominated solution of TM				
1	4.3842	0.0826	3.0655	0.0868
2	0.9204	0.0421	0.6387	0.0112
3	0.3225	0.1079	0.0001	0.0027
4	0.9623	0.0123	0.0001	0.0779
5	0.4141	0.0077	3.1406	0.0086
Controller u_i	k_{p_i}	k_{i_i}	k_{d_i}	
Non-dominated solution of TM				
1	12589.11	54987.95	165.46	
2	14996.48	1002.54	108.39	
3	649.91	125.73	1.41	
Link i	m_i (m)	l_{ci} (m)	γ_i (rad)	I_i (kg m^2)
Non-dominated solution of SM				
1	4.2309	0.0835	3.0894	0.0854
2	1.0101	0.0304	0.7951	0.0141
3	0.3376	0.1049	0.0001	0.0030
4	0.9598	0.0112	−0.0009	0.0751
5	0.3849	0.0040	0.0088	0.0062
Controller u_i	k_{p_i}	k_{i_i}	k_{d_i}	
Non-dominated solution of SM				
1	250.60	41.33	248.00	
2	4866.94	2558.66	50.68	
3	363.02	394.08	0.13	

5 Conclusions

In this paper, an exhaustive exploitation mechanism is proposed in the differential evolution algorithm to promote both diverse non-dominated solutions and a uniformly distributed Pareto front in the structure-control design framework.

The EEM considers the thorough search in regions (hyper-grid) of the Pareto front, where the mutation process requires individuals in a region such that the exploitation of such individuals improves the Pareto front trade-offs.

The EEM in the optimization process of the structure-control design framework can benefit the synergetic trade-offs of its design, and in addition, good approximation to the Pareto front (sub-optimal) with extended and well-distributed Pareto solutions can be obtained.

The comparative analysis of the MOEEDE with a DE algorithm that does not include such mechanism (MODE) shows the superior performance of the proposal

which results in finding better structure-control design trade-offs. From a computational point of view, the convergence time of the MOEEDE is larger than the MODE. Nevertheless, from a design engineering point of view, the found non-dominated solutions are considerably improved, and more sets of possibilities can be used by the designer. Hence, the decision- making process will depend on the required performance for a specific application.

Then, the structure-control design approach not only requires using the simultaneous optimization strategy, but also an analysis of how more, and better trade-offs in the design solutions are obtained by the optimization technique. Due to the high trade-offs in the structure-control design problem, the analysis and improvements of the optimization technique is an important task to find more and diverse solutions with better coupling effects in the combined structure/controller system.

Future research involves the inclusion of mechatronic system initial condition uncertainties in the structure-control design framework based on simultaneous strategies.

Acknowledgements The author acknowledges support from the COFAA of the Instituto Politécnico Nacional and from SEP-CONACyT, via the project numbers 20160826 and 182298, respectively. Furthermore, I would like to express my appreciation for the help provided in the grammatical expressions by English professor Lawrence Whitehill.

References

1. Youcef-Toumi, K.: Modeling, design, and control integration: a necessary step in mechatronics. IEEE/ASME Trans. Mecatronics **1**(1), 29–38 (1996)
2. Isermann, R.: Modeling and design methodology for mechatronic systems. IEEE/ASME Trans. Mechatron. **1**(1), 16–28 (1996)
3. Bodden, D.S., Junkins, J.L.: Eigenvalue optimization algorithms for structure/controller design iterations. J. Guidance Control Dyn. **8**(6), 697–706 (1985)
4. Ling, J., Kabamba, P., Taylor, J.: Multicriterion structure/control design for optimal maneuverability and fault tolerance of flexible spacecraft. J. Optim. Theory Appl. **82**(2), 219–251 (1994)
5. Reyer, J.A., Papalambros, P.Y.: Combined optimal design and control with application to an electric DC motor. J. Mech. Des. **124**(2), 183–191 (2002)
6. Pil, A., Asada, H.: Integrated structure/control design of mechatronics systems using a recursive experimental optimization method. IEEE/ASME Trans. Mechatron. **1**(1), 191–203 (1996)
7. Zhang, W.J., Li, Q., Guo, L.S.: Integrated design of mechanical structure and control algorithm for a programmable four-bar linkage. IEEE/ASME Trans. Mechatron. **4**(4), 354–362 (1999)
8. Li, Q., Zhang, W.J., Chen, L.: Design for control—a concurrent engineering approach for mechatronics systems design. IEEE/ASME Trans. Mechatron. **6**(2), 161–169 (2001)
9. Fathy, H.K., Reyer, J.A., Papalambros, P.Y., Ulsov, A.G.: On the coupling between the plant and controller optimization problems. In: American Control Conference, 2001. Proceedings of the 2001, vol. 3, pp. 1864–1869 (2001)
10. Yan, H.S., Yan, G.J.: Integrated control and mechanism design for the variable input-speed servo four-bar linkages. Mechatronics **19**(2), 274–285 (2009)
11. Villarreal-Cervantes, M., Cruz-Villar, C., Alvarez-Gallegos, J.: Synergetic structure-control design via a hybrid gradient-evolutionary algorithm. Optim. Eng. **16**(3), 511–539 (2015)
12. Yang, Y.P., Chen, Y.A.: Multiobjective optimization of hard disk suspension assemblies: part ii—integrated structure and control design. Comput. Struct. **59**(4), 771–782 (1996)
13. Li, Q., Wu, F.X.: Control performance improvement of a parallel robot via the design for control approach. Mechatronics **14**(8), 947–964 (2004)
14. Ouyang, P.R., Li, Q., Zhang, W.J.: Integrated design of robotic mechanisms for force balancing and trajectory tracking. Mechatronics **13**(8–9), 887–905 (2003)

15. Fu, K., Mills, J.K.: A convex approach solving simultaneous mechanical structure and control system design problems with multiple closed-loop performance specifications. J Dyn. Syst. Meas. Control **127**(1), 57–68 (2005)
16. Portilla-Flores, E.A., Mezura, E., Alvarez-Gallegos, J., Coello-Coello, C.A., Cruz-Villar, C.A.: Integration of structure and control using an evolutionary approach: an application to the optimal concurrent design of a CVT. Int. J. Numer. Methods Eng. **71**(8), 883–890 (2007)
17. Cruz-Villar, C.A., Alvarez-Gallegos, J., Villarreal-Cervantes, M.G.: Concurrent redesign of an underactuated robot manipulator. Mechatronics **19**, 178–183 (2009)
18. Alyaqout, S., Papalambros, P., Ulsoy, A.: Combined robust design and robust control of an electric DC motor. IEEE/ASME Trans. Mechatron. **16**(3), 574–582 (2011)
19. Osyczka, A.: Multicriterion Optimization in Engineering with Fortran Programmes. Halsted Press, Ultimo (1984)
20. Coello, C.A.C., Lamont, G.B.: Applications of Multi-Objective Evolutionary Algorithms. World Scientific Publishing Co, Singapore (2004)
21. Bazaraa, M.S., Sherali, H.D., Shetty, C.M.: Nonlinear Programming: Theory and Algorithms. Wiley, New York (2006)
22. Conway, B.: A survey of methods available for the numerical optimization of continuous dynamic systems. J. Optim. Theory Appl. **152**(2), 271–306 (2012)
23. Portilla-Flores, E.A., Mezura-Montes, E., Alvarez-Gallegos, J., Coello-Coello, C.A., Cruz-Villar, C.A., Villarreal-Cervantes, M.G.: Parametric reconfiguration improvement in non-iterative concurrent mechatronic design using an evolutionary-based approach. Eng. Appl. Artif. Intell. **24**(5), 757–771 (2011)
24. Villarreal-Cervantes, M.G., Cruz-Villar, C.A., Alvarez-Gallegos, J., Portilla-Flores, E.A.: Robust structure-control design approach for mechatronic systems. IEEE/ASME Trans. Mechatron. **18**(5), 1592–1601 (2013)
25. Affi, Z., EL-Kribi, B., Romdhane, L.: Advanced mechatronic design using a multi-objective genetic algorithm optimization of a motor-driven four-bar system. Mechatronics **17**(9), 489–500 (2007)
26. EL-Kribi, B., Houidi, A., Affi, Z., Romdhane, L.: Application of multi-objective genetic algorithms to the mechatronic design of a four bar system with continuous and discrete variables. Mech. Mach. Theory **61**(1), 68–83 (2013)
27. Price, K.V., Storn, R.M., Lampinen, J.A.: Differential Evolution: A Practical Approach to Global Optimization. Springer, Berlin (2005)
28. Mezura-Montes, E., Reyes-Sierra, M., Coello, C.: Multi-objective optimization using differential evolution: a survey of the state-of-the-art. In: Chakraborty, U. (ed.) Advances in Differential Evolution, Studies in Computational Intelligence, vol. 143, pp. 173–196. Springer, Berlin (2008)
29. Neri, F., Tirronen, V.: Recent advances in differential evolution: a survey and experimental analysis. Artif. Intell. Rev. **33**(1–2), 61–106 (2010)
30. Weber, M., Neri, F., Tirronen, V.: Distributed differential evolution with explorative-exploitative population families. Genet. Program. Evolvable Mach. **10**(4), 343–371 (2009)
31. Epitropakis, M., Tasoulis, D., Pavlidis, N., Plagianakos, V., Vrahatis, M.: Enhancing differential evolution utilizing proximity-based mutation operators. IEEE Trans. Evol. Comput. **15**(1), 99–119 (2011)
32. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Boston (1989)
33. Schwefel, H.P.: Evolution and Optimization Seeking. Wiley, New York (1995)
34. Deb, K., Kain, S.: Multi-speed gearbox design using multi-objective evolutionary algorithms. J. Mech. Des. **125**(3), 609–619 (2003)
35. Shiakolas, P., Koladiya, D., Kebrle, J.: On the optimum synthesis of six bar linkages using differential evolution and the geometric centroid of precision positions technique. Mech. Mach. Theory **40**(3), 319–335 (2005)
36. Knowles, J.D., Corne, D.W.: The pareto archived evolution strategy: a new baseline algorithmfor pareto multiobjective optimisation. In: Proceedings of the 1999 Congress on Evolutionary Computation, vol. 1, pp. 98–105 (1999)
37. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm NSGA-II. IEEE Transactions on Evolutionary Computation **6**(2), 182–197 (2002)
38. Lung-Wen, T.: Robot Analysis: The Mechanics of Serial and Parallel Manipulators. Wiley, New York (1999)
39. Spong, M.W., Vidyasagar, M.: Robot Dynamics and Control. Wiley, New York (2004)

40. Villarreal-Cervantes, M.G., Cruz-Villar, C.A., Alvarez-Gallegos, J., Portilla-Flores, E.A.: Differential evolution techniques for the structure-control design of a five-bar parallel robot. Eng. Optim. **42**(6), 535–565 (2010)
41. Villarreal-Cervantes, M.G., Alvarez-Gallegos, J.: Off-line PID control tuning for a planar parallel robot using de variants. Expert Syst. Appl. **64**, 444–454 (2016)
42. Van Veldhuizen, D., Lamont, G.: On measuring multiobjective evolutionary algorithm performance. Congr. Evolut. Comput. **1**, 204–211 (2000)

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH ("Springer Nature").

Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users ("Users"), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use ("Terms"). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.

These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.

We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.

While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.

These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.

To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.

Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.

If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com