2024 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)
July 15-19, 2024. Boston, MA, USA

CAD-based co-optimizations for geometry and motion profile towards energy-optimal point-to-point mechanisms

Brecht Maes[°*‡], Abdelmajid Ben Yahya[°*‡], Nick Van Oosterwyck[*‡], Amélie Chevalier[*], and Stijn Derammelaere[*‡]
[*]Department of Electromechanics, Cosys-Lab, University of Antwerp, Antwerp, Belgium
Email: Brecht.Maes@uantwerpen.be
[‡]AnSyMo/Cosys, Flanders Make, the Strategic Research Center for the Manufacturing Industry, Lommel, Belgium

***Abstract*—Energy consumption is receiving increasing attention due to environmental concerns. There is a high optimization potential concerning the energy consumption of industrial machines. Extensive research has been performed concerning geometry optimizations and motion profile optimizations towards energy-optimal point-to-point mechanisms. A co-optimization approach that integrates geometry and motion profile optimization in one architecture, outperforms methods that optimize motion profiles with suboptimal geometries, or vice versa. However, limited research focuses on co-optimizing geometry and motion profiles in one architecture. Therefore, a simultaneous, sequential, and nested co-optimization architecture is set up and compared. The most optimal motion profile and link lengths are determined for an industrial mechanism, resulting in a root mean square (rms) torque saving of 49.2%. To improve computational speed, a method that uses the derivation of a torque equation from three CAD simulations has been utilized. This method facilitates rapid convergence of the nested co-optimization. As a result, the major difference between the three co-optimization methods lies in the ability to converge rapidly to the minimum. This study demonstrates the nested co-optimization's capacity to identify an enhanced optimum, reducing the computational time by 74.4% compared to the simultaneous co-optimization and by 78.3% compared to three sequential iterations.**

***Index Terms*—Co-optimization; point-to-point; mechanism; energy-optimal; geometry; motion profile; CAD-based**

I. Introduction

The climate crisis has increased the significance of the energy consumption of electric motors [1]. Moreover, energy usage is expected to rise by 50% between 2018 and 2050 [2]. Furthermore, in 2015, electric motors accounted for 53% of the worldwide electricity consumption [3]. These statements emphasize the importance of optimization procedures that decrease the energy demand of electric motor systems.

This paper compares three co-optimization techniques to optimize point-to-point (PTP) mechanisms to reduce energy consumed by electric motors. The co-optimizations aim to combine both geometry and motion profile optimization. Geometry optimization focuses on optimizing the link lengths of the mechanism [4]. Motion profile optimization determines the optimal trajectory of the electric motor to minimize the energy consumption of the point-to-point (PTP) movement [5]. Various architectures for co-optimizations are described in [6]. In this work, three approaches for co-optimization are introduced and applied to an industrial mechanism. The first co-optimization architecture is a simultaneous co-optimization that optimizes the motion profile and geometry by considering all the design parameters simultaneously. The second co-optimization architecture is the sequential approach that firstly optimizes the geometry and, secondly, optimizes the motion profile iteratively. The third co-optimization architecture is a nested co-optimization that optimizes the link lengths in an outer loop, and an inner optimization loop identifies the optimal motion profile for each set of link lengths determined by the outer loop. The optimization objective is the root mean square torque (T_{rms}) of the motor, as [5] shows that T_{rms} is an effective objective to minimize the total energy usage of the system, provided that frictional forces are negligible. The latter is the case in high-dynamical industrial applications, particularly if inertial loads are predominant [7], as in this case.

According to previous research [8]–[10], the simultaneous and nested approach deliver better results than the sequential approach. However, the nested approach is thought to be computationally expensive and it can have potential feasibility issues [11]. Nevertheless, a nested approach can converge faster to an optimal solution than a simultaneous approach, provided that efficient inner loop solving methods are used [12].

In [13], a co-optimization framework based on hierarchical deep reinforcement learning is used to optimize the configuration and motion of a chain-type modular robot. The robot is optimized towards time consumption, task completion rate, and energy consumption to avoid obstacles, find targets, and climb stairs. Another work [14], proposed a bilevel co-design optimization for energy-efficient legged robots. The bilevel optimization framework is similar to a nested co-optimization as it optimizes geometric parameters in an outer loop and motion planning in an inner loop. A similar co-optimization problem is addressed in [15]. In [16], different co-optimization architectures are compared that are suited for the co-design of control systems and physical systems. However, the proposed

[°]These authors contributed equally.

architectures are not proven to be easily applicable and are not validated as they are only conceptual. In [17], co-optimization was used to optimize the structural strength and the controller for the feed drive system of machine tools, improving the overall system dynamic characteristics. A recent study [18], simultaneously optimized the motion and the link shapes to reduce the energy consumption of a six-degrees-of-freedom serial manipulator and a parallel five-bar linkage. A Lie group-based mesh deformation model has been used to model the kinematics and dynamics of the mechanism. However, the model introduces errors in estimating the inertial parameters. In [19], design parameters, including link lengths, multiple actuator placements, and contact forces, are optimized towards minimal torque for multiple robots, including a four-bar mechanism. However, the motion profiles of the actuators are not optimized. Furthermore, in [19] an analytical method, i.e. the implicit function theorem, is used to obtain the objective function and constraints.

Nevertheless, the papers mentioned above utilize analytical methods for describing system properties and formulating objective functions, which can be time-consuming, complex, and prone to errors when applied to different mechanisms. A more advantageous approach is to use computer-aided design (CAD) simulations, which are already widely used in the industry to design the machine.

This paper adds the following improvements to the state-of-the-art literature:

- **CAD-based co-optimization**: The literature only uses case-specific and complex analytic formulations of the objective function for co-optimizations [13]–[15], [18], the method in this paper uses CAD and is applicable to all point-to-point mechanisms.
- **Novel nested co-optimization:** A novel nested co-optimization methodology is developed that obtains the objective function more efficiently by extracting system properties from the mechanism with three CAD simulations so a torque equation can be set up in the outer optimization loop. An evident approach would use a CAD simulation in the inner optimization. However, this causes longer solve times.
- **Comparison of co-optimization architectures**: In [18], an energy-optimal co-optimization of geometry and motion profile based on an analytic objective function is presented, however, a comparison of different co-optimization architectures is missing. Previous literature has only compared co-optimization architectures that optimize different parameters than link lengths and motion profiles or different objectives than T_{rms} [8], [17], [19]. This paper compares three co-optimization architectures, simultaneous, sequential, and a novel nested co-optimization aimed at achieving energy-optimal link lengths and motion profiles for an industrial point-to-point mechanism.

This paper is structured as follows. Section II describes the case and explains its important parameters. Subsequently, section III gives information about motion profile optimization and the different scaling factors that are used, followed by a discussion on the determination of the number of design parameters. Section IV discusses how the geometry optimization works. After that, section V provides an extensive explanation of the co-optimization methods. Subsequently, the selection of the optimization algorithm is clarified in section VI. The results of the different optimizations are shown and discussed in section VII.

II. Case study

The co-optimizations are applied to an industrial case study featuring an in-line slider-crank mechanism comprising three interconnected links. As shown in Figure 1, this mechanism includes a green-colored crank l_1, and two pink-colored couplers l_2, which are both parametrized and the end effector indicated in red. The end effector is constrained to a vertical path and paired with a symmetric counterpart – the yellow slider. This yellow end effector moves opposite to the red end effector. The mechanism's motion is driven by an electric motor, positioned at the top of Figure 1 and colored in grey.

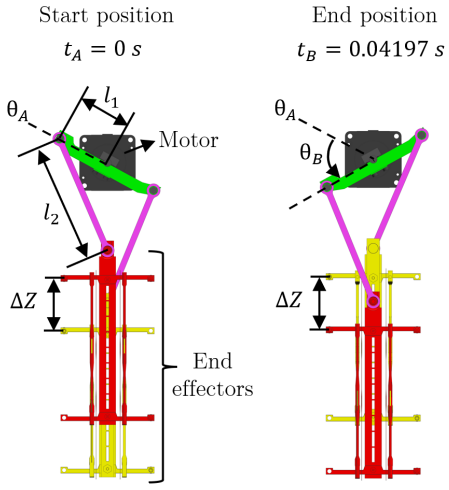

Fig. 1. The motor and the links l_1 and l_2 displace the two end effectors (in red and yellow) to the needed end position. In this case, the absolute motion over ΔZ should be completed in $0.04197s$.

For the application, the distance between the two end effectors ΔZ in start-position and end-position is a design requirement and, consequently, fixed during the optimization (as shown in Figure 1). Consequently, the start angle is denoted as θ_A and the end angle as θ_B. Changes according to the link lengths l_1 and l_2. Another design requirement is the motion starting time t_A and end time t_B, in this case, $t_A = 0\ s$ and $t_B = 0.04197\ s$, which are determined by the manufacturer. In addition to the earth's gravity, the mechanism experiences no external loads.

III. Motion profile optimization

Motion profile optimization optimizes the trajectory of the motor to minimize T_{rms}, as [5] shows that T_{rms} is an effective objective to minimize the total energy usage of the system.

According to [5], the motion profile $\theta(t)$, where $t \in [t_A, t_B]$, is best described by a polynomial with a Chebyshev base of degree n: $\sum_{\ell=0}^{n} c_\ell T_\ell(x)$. To use this representation for the motion profile, a rescaling for the time t from the interval $[t_A, t_B]$ to $[-1, 1]$ of x is needed. To achieve this rescaling, a linear transformation [20] is used:

$$x = \frac{2}{t_B - t_A}\, t - \frac{t_B + t_A}{t_B - t_A} = a\, t - b \tag{1}$$

Furthermore, the position $\theta \in [\theta_A, \theta_B]$ is also rescaled to the interval $\phi \in [-1, 1]$, to obtain bounds on the motion profile coefficients $\mathbf{c} = [c_0, c_1, \ldots, c_n]^T$, for an n-th degree polynomial. The rescaling of the position θ can be achieved through the following transformation [5]:

$$\phi = \frac{2}{\theta_B - \theta_A}\, \theta - \frac{\theta_B + \theta_A}{\theta_B - \theta_A} = e\, \theta - f \tag{2}$$

Constraints of zero speed $\dot{\phi}$ and zero acceleration $\ddot{\phi}$ need to be set up for the point-to-point movement in the start and end position:

$$\begin{aligned} \phi(-1) = -1, \quad \dot{\phi}(-1) = 0, \quad \ddot{\phi}(-1) = 0, \\ \phi(1) = 1, \qquad \dot{\phi}(1) = 0, \qquad \ddot{\phi}(1) = 0. \end{aligned} \tag{3}$$

These constraints imply that the lower degree coefficients $[c_0, \ldots, c_5]^T$ can be written as a function of the remaining coefficients $[c_6, \ldots, c_n]^T$. Consequently, the coefficients $\mathbf{c} = [c_6, \ldots, c_n]^T$ are design parameters (DPs).

Citing [5], the rescaling of position θ from $[\theta_A, \theta_B]$ to $[-1, 1]$ enables the determination of bounds for the design parameters $\mathbf{c} = [c_6, \ldots, c_n]^T$:

$$|\mathbf{c}| \leq \frac{4}{\pi} \tag{4}$$

Generally, the motion profile optimization problem can be formulated as follows:

$$\begin{aligned} \underset{\mathbf{c} \in \mathbb{R}^{n-5}}{\text{minimize}} \quad & T_{rms}(\mathbf{c}) \\ \text{subject to} \quad & |\mathbf{c}| \leq \frac{4}{\pi} \end{aligned}$$

A. Motion profile DPs convergence analysis

The savings in T_{rms} are directly influenced by the number of design parameters N_{DP}, utilized in optimizing the motion profile. However, as the number of DPs N_{DP} increases, the solving time t_{sol} for the optimization algorithm also increases. To consider the trade-off involved, a convergence analysis has been conducted to determine the appropriate number of design parameters N_{DP} for the motion profile.

Based on the convergence analysis (Figure 2), four design parameters (DPs) are selected, as a higher N_{DP} does not result in significant savings and would increase the computational cost of the optimization (as shown in Figure 2). Four design parameters result in the following coefficients to be optimized: c_6, c_7, c_8, and c_9, which corresponds to a polynomial degree $n = 9$. The motion profile optimization was executed utilizing a gradient-based optimization algorithm according to [5].

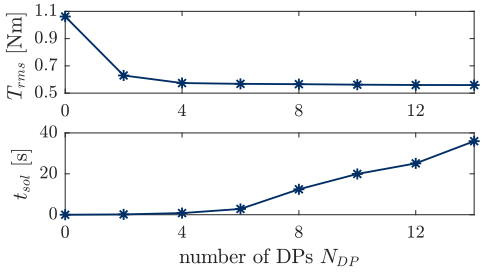

Fig. 2. Four design parameters (DPs) are selected as the T_{rms} has converged around four DPs and an increasing number of DPs N_{DP} results in increasing solve time t_{sol}.

IV. Geometry optimization

Geometry optimization minimizes T_{rms}, as [5] shows that T_{rms} is an effective objective to minimize the total energy usage. The geometry optimization is applicable for a mechanism with m link lengths $\mathbf{r} = [l_1, \ldots, l_m]^T$, which are constrained by specific lower bounds ($l_{j,min}$) and upper bounds ($l_{j,max}$) per link length l_j. The geometry optimization problem can be formulated as follows:

$$\begin{aligned} \underset{\mathbf{r} \in \mathbb{R}^m}{\text{minimize}} \quad & T_{rms}(\mathbf{r}) \\ \text{subject to} \quad & l_{j,min} \leq l_j \leq l_{j,max}, \quad j = 1, \ldots, m \end{aligned}$$

Despite the simple structure of a slider-crank mechanism, an analytical approach to simulate its dynamic behavior has been avoided as it is error-prone and case-specific, which limits its general applicability and usability by machine builders [4]. To calculate the objective T_{rms} based on the link lengths $\mathbf{r}$, CAD simulations are employed that are applicable to any mechanism, given the CAD models provided by machine builders.

As indicated in [4], the initial stage of determining the motor torque T_m for a mechanism with varying geometry parameters involves determining end angle θ_B for the motor. The required displacement of the end effectors allows for the derivation of θ_B through an inverse kinematics simulation (see Figure 3). To obtain θ_B, the mechanism needs to be driven from the end effector with a predefined absolute distance ΔZ. Furthermore, the inverse kinematic simulation acts as a feasibility check, as certain combinations of DPs $\mathbf{r}$, are not able to displace the end effectors to the predefined distance ΔZ. If the link lengths can not provide the mechanism to travel the required distance ΔZ, the design is considered infeasible, and the optimization algorithm will choose new DPs. If the link lengths are feasible, the end angle θ_B is extracted and used in the subsequent dynamic simulation. The dynamic simulation drives the mechanism at the green-colored crank, with a certain motion profile until the end angle θ_B is reached. Subsequently, this simulation calculates the motor torque to return the T_{rms} value as the objective to the optimization algorithm.

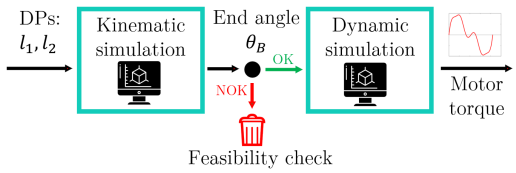

Fig. 3. The workflow to obtain the motor torque for certain design parameters (DPs): l_1 and l_2.

V. Co-optimizations

In order to increase energy saving, the geometry and motion profile optimization are combined in a co-optimization architecture. Three different approaches are proposed and compared, a simultaneous co-optimization, a sequential co-optimization, and a nested co-optimization. The co-optimizations minimize T_{rms} by optimizing link lengths $\mathbf{r}$, which are constrained by lower bound $l_{j,min}$ and upper bound $l_{j,max}$ and by optimizing the motion profile coefficients $\mathbf{c}$, of which the constraints are determined in section III. The co-optimization problem is formulated as follows:

$$\begin{aligned} \underset{\mathbf{r}\in\mathbb{R}^m,\, \mathbf{c}\in\mathbb{R}^{n-5}}{\text{minimize}} \quad & T_{rms}(\mathbf{r},\mathbf{c}) \\ \text{subject to} \quad & l_{j,min} \le l_j \le l_{j,max}, \quad j = 1,\dots,m \\ & |\mathbf{c}| \le \frac{4}{\pi} \end{aligned}$$

To solve an optimization problem, a communication needs to be set up between an optimization algorithm and the objective function. Journaling plays a vital role in establishing the communication of optimizations that utilize CAD simulations. The journal is a script that can send parameters, e.g. link lengths and motion profile coefficients to the CAD software and it enables the extraction of results from the CAD software.

The first step of all three optimization methods is a kinematic simulation that determines the end position θ_B and checks the feasibility of the design, as explained in section IV. The following steps are dependent on the type of optimization method and are therefore discussed in the following subsections.

A. Simultaneous co-optimization

The simultaneous co-optimization optimizes the motion profile and geometry by simultaneously considering all the design parameters $\mathbf{s} = [\mathbf{r}; \mathbf{c}]$ (as shown in Figure 4). This type of architecture is believed to have a lower chance of getting stuck in local minima than sequential optimization (described in subsection V-B) [8] .

The motion profile is sent as coefficients $[c_0,\dots,c_n]^T$ to the CAD software together with the link lengths $[l_1,\dots,l_m]^T$. To drive the mechanism at the motor, a built-in polynomial motion function with a Chebyshev base is used in the CAD software where a time variable t is used in the interval $[t_A, t_B]$, which delivers a motion profile $\theta(t)$, where $\theta \in [\theta_A, \theta_B]$. The coefficients that are sent to the CAD software are determined for a motion profile $\phi(x)$, where $\phi \in [-1,1]$ and $x \in [-1,1]$. Consequently, the time variable t in the CAD motion function must be rescaled by rewriting the scaling factors in Equation (1), and the angles need to be rescaled by rewriting the scaling factors in Equation (2). The driver can now simulate the mechanism with the chosen link lengths and correct motion profile $\mathbf{s}$ in CAD software and calculate the motor torque T_m.

The optimization algorithm cannot find an optimal solution as the difference in the order of magnitude between the DPs of the geometry $\mathbf{r}$ and the DPs of the motion profile $\mathbf{c}$ is too big. Therefore the geometry DPs: l_1 and l_2, expressed in $[mm]$ for the CAD simulation, are rescaled for the optimization algorithm with a factor $SF_g = 10^{-6}$ to the rescaled link lengths l_1, r and l_2, r (see Table I).

B. Sequential co-optimization

Another common method is sequential co-optimization, this method first optimizes the geometry, and subsequently, the optimized link lengths are used as input parameters for motion profile optimization. This order is preferred as the geometry influences the behaviour of the system more than the motion profile [17]. After an optimal motion profile is found, another iteration can start with the optimized profile (as shown in Figure 4), until the T_{rms} is converged. The optimization of the motion profile involves extracting the system properties to formulate a torque equation (Equation (5)), which accelerates the solve time t_{sol}. This equation will be discussed in detail in subsection V-C *Nested co-optimization*.

C. Nested co-optimization

A novel nested co-optimization architecture is proposed to ensure fast convergence to an optimal solution. A nested architecture optimizes the link lengths in an outer loop, and an inner optimization loop identifies the optimal motion profile for each set of link lengths determined by the outer loop (as shown in Figure 4). Separating the motion profile optimization from the geometry for a co-optimization makes it possible to obtain the objective function in several ways for the motion profile optimization. It would be obvious to directly use a CAD simulation to calculate T_{rms} as the objective function, such as in simultaneous co-optimization. However, this would lead to long solve times t_{sol} when the CAD simulations are executed inside the inner optimization loop of the motion profile. To reduce the solve time t_{sol}, a novel nested co-optimization is developed that obtains the objective function more efficiently by extracting system properties load torque T_l and inertia J with three CAD simulations [21] so a torque equation [22] can be set up in the outer optimization loop. This is the torque equation [22]:

$$T_m(t) = T_l(\theta) + J(\theta)\ddot{\theta} + \frac{1}{2}\frac{\mathrm{d}J(\theta)}{\mathrm{d}\theta}(\dot{\theta})^2 + T_f(\dot{\theta}) \qquad (5)$$

The frictional forces $T_f(\dot{\theta})$ are negligible in high-dynamical industrial applications, especially if the inertial loads are predominant [7]. The system properties T_l and J remain constant across all motion profiles, allowing the extraction process to

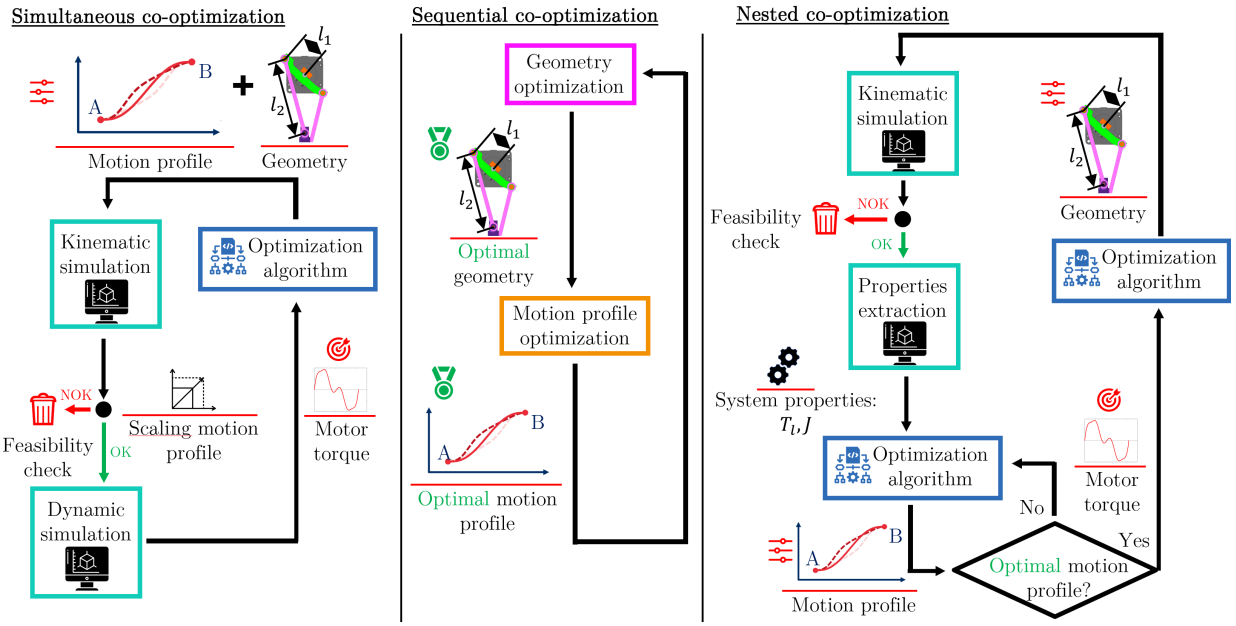

Fig. 4. Workflow of the simultaneous and nested co-optimization architecture for one sample and the sequential optimization for one iteration.

be positioned outside the motion profile optimization loop. A detailed explanation of the system properties extraction can be found in [21]. After obtaining the torque equation, it takes less than one second to determine the optimal motion profile [5]. Once the optimal motion profile is found, the associated torque is sent back to the outer loop optimization algorithm that will choose new link lengths until the geometry converges to a minimum. Due to the ability to obtain an optimal motion profile rapidly, it becomes viable to optimize more DPs for the motion profile, which further reduces T_{rms}, with a slight increase in solve time t_{sol} (as shown in Figure 2).

VI. Optimization Algorithm Settings

All three optimization methods use the same optimization algorithm settings from the MATLAB Optimization Toolbox, which are shown in Table I. For the optimization algorithm, a gradient-based method, namely SQP, has been selected as it is computationally cheap, even if a substantial amount of design parameters are chosen [23]. It is also one of the most effective methods for nonlinearly constrained optimizations and can be utilized for both small and large problems [24]. Furthermore, SQP has shown that it is an excellent optimization algorithm for both geometry optimizations [4] and motion profile optimizations [5].

VII. Results

The three different co-optimizations have been tested on the in-line slider-crank mechanism starting from two different initial designs to evaluate their performance. In addition, the geometry and the motion profile optimization are executed separately as a performance reference for the co-optimizations. The optimizations were performed using MATLAB and Siemens NX, with licensed access through an ethernet connection on an

TABLE I
The settings concerning the optimization algorithm used for all the optimizations.

Setting	Value
Optimization algorithm	SQP
Finite difference type	Forward
Finite difference step size ϵ	10^{-7}
Lower bound $[l_1, r;\ l_2, r]$	$[26 \cdot 10^{-6};\ 40 \cdot 10^{-6}]$
Upper bound $[l_1, r;\ l_2, r]$	$[100 \cdot 10^{-6};\ 350 \cdot 10^{-6}]$
Lower bound $[c_6;\ c_7;\ c_8;\ c_9]$	$[-4/\pi;\ -4/\pi;\ -4/\pi;\ -4/\pi]$
Upper bound $[c_6;\ c_7;\ c_8;\ c_9]$	$[4/\pi;\ 4/\pi;\ 4/\pi;\ 4/\pi]$
Step tolerance	10^{-14}
Function tolerance	10^{-6}
Optimality tolerance	10^{-14}
Constraint tolerance	10^{-6}

i9-7960X 16C CPU @ 2.80GHz. All the optimizations were performed with the same settings for the CAD simulations, the optimization algorithm, and the scaling factor SF_g.

Two starting designs, *base design* and *seed design*, have been used for the optimizations. *Base design* refers to the original design from the manufacturer, in which the crank moves $171°$ due to the small crank length of $27\ mm$. This *base design* is located close to a feasibility border as the crank length l_1 cannot be reduced much further while still being able to displace the end effector to the predefined end distance ΔZ. The pink-colored coupler l_2 has a length equal to $239\ mm$. In contrast, *seed design* is located further from such an infeasible design as it has a longer crank l_1, which is equal to $28.5\,mm$. *Seed design* also has a shorter pink-colored coupler of $102\ mm$. For both starting designs, a 5-th degree polynomial motion profile has been selected as a reference for the geometry optimization. Strikingly, starting from *base design* did not result in a fully optimal solution for the nested

co-optimization. Starting from *base design* obtained a T_{rms} saving of $46.2\,\%$ while the optimal solution achieves a T_{rms} saving of $49.2\,\%$.

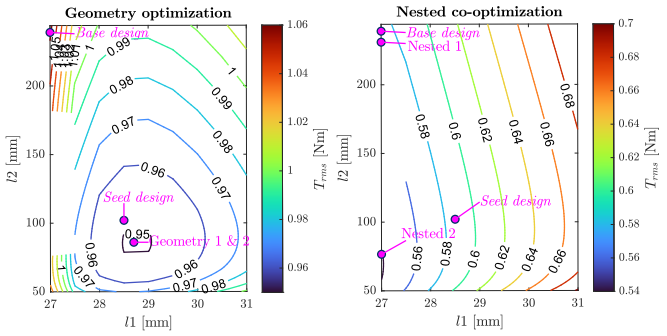

Fig. 5. The two plots are contour plots of the objective function of the geometry optimization (left) and the nested co-optimization (right). A starting design close to the feasibility border can be suboptimal for the nested co-optimization.

The inability of the nested co-optimization to optimize the design can be explained by examining the plots in Figure 5. Figure 5 shows the objective function of the geometry optimization and nested co-optimization displayed as contour plots. Noticeably, the shape of these objective functions is massively affected by different motion profiles, as seen in Figure 5. The usage of gradient-based optimization algorithms implies that the algorithm searches for the minimum by searching in the direction of the steepest descent of the objective function. This implies that if the nested co-optimization is initiated from *base design*, the search direction leads toward infeasible designs characterized by a small l_1. Due to this search direction, the nested co-optimization is unable to find the minimum, which is located at a smaller l_2. In contrast, the geometry optimization is able to find the minimum as the search direction is away from the feasibility border and towards the minimum indicated as 'Geometry 1 & 2'. As indicated, *base design* is not a good starting point, therefore the co-optimizations are performed starting from *seed design*, the results are shown in Table II. In Figure 5, it is evident that the solution for the nested approach starting from *seed design*, denoted as 'Nested 2', is located at the minimum. This is in contrast to the solution resulting from initiating the nested approach from *base design*, indicated by 'Nested 1'.

Table II contains the optimized design parameters indicated in bold and blue and their respective T_{rms} and T_{rms} saving compared to *base design*. The time to obtain a solution is denoted as t_{sol} and the number of required samples is denoted as N_s. In order to find the minimum, the sequential co-optimization needed three iterations (3 it.). The three co-optimization architectures are roughly able to find the optimal design (shown in Figure 6) with a $T_{rms} = 0.539\,Nm$. The co-optimizations' solution is significantly better than a separate geometry or motion profile optimization (see Table II). This emphasizes the importance of co-optimizing geometry and motion profile. The optimal design, along with its associated motor torque and motion profile, is presented in Figure 6, showing both its start and end positions.

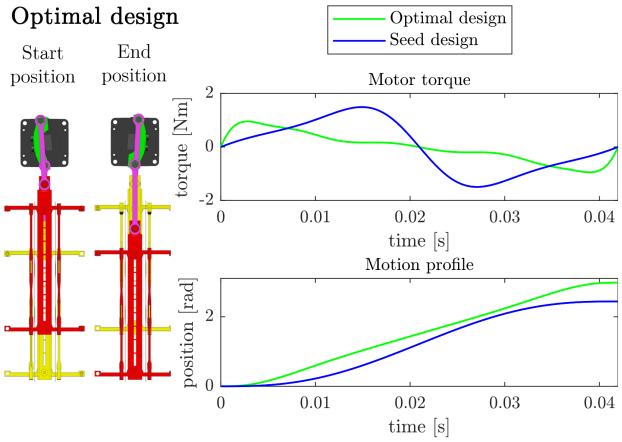

Fig. 6. The optimal design achieved by the nested co-optimization in both its start and end positions, along with the corresponding motor torque and motion profile, is compared to that of *seed design*.

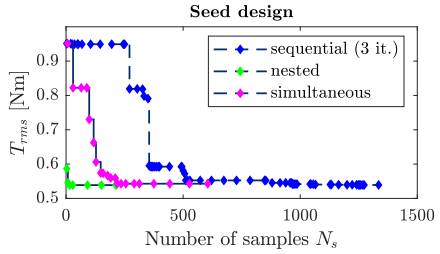

Fig. 7. The number of samples are plotted with their corresponding T_{rms} values, starting from *seed design*. Nested co-optimization is the fastest.

The major difference between the three co-optimizations lies in their ability to converge, as shown in Table II. The nested co-optimization reduced the computational time t_{sol} with 74.4% compared to the simultaneous co-optimization, and with 78.3% compared to the sequential co-optimization, due to rapidly determining the optimal motion profile per geometry by using the torque equation (5). The sequential approach also uses the torque equation but the geometry is optimized with a suboptimal motion profile, and vice versa, leading to a longer solve time t_{sol}. For the simultaneous co-optimization, the torque equation is not a viable option, instead, it uses CAD simulations as the objective function, which increases the computational cost. Consequently, the same trend is seen for the number of samples that is required (see Table II and Figure 7). The nested co-optimization required 213 samples, while the simultaneous and sequential approach required respectively, 605 and 1335 samples.

VIII. Conclusion

This study proposes a comparison of energy-optimal co-optimization architectures to obtain optimal link lengths and motion profile for point-to-point mechanisms. The proposed method is a nested co-optimization that determines the most

TABLE II
RESULTS OF THE DIFFERENT OPTIMIZATIONS STARTING FROM *seed design*.

Optimization method	$\mathbf{l_1}$ **[mm]**	$\mathbf{l_2}$ **[mm]**	$\mathbf{c_6}$ **[1]**	$\mathbf{c_7}$ **[1]**	$\mathbf{c_8}$ **[1]**	$\mathbf{c_9}$ **[1]**	$\mathbf{T_{rms}}$ **[Nm]**	$\mathbf{T_{rms}}$ **saving [%]**	$\mathbf{t_{sol}}$ **[h]**	$\mathbf{N_s}$ **[1]**
Base design	27.0	239	0	0	0	0	1.06	-	-	-
Geometry	**28.7**	**86.0**	0	0	0	0	0.949	10.6	3.57	215
Motion profile	28.5	102	$\mathbf{-2.18 \cdot 10^{-6}}$	$\mathbf{3.53 \cdot 10^{-4}}$	$\mathbf{3.52 \cdot 10^{-6}}$	$\mathbf{3.34 \cdot 10^{-3}}$	0.587	38.3	$\mathbf{11 \cdot 10^{-5}}$	544
Simultaneous	**27.0**	**94**	$\mathbf{-1.11 \cdot 10^{-3}}$	$\mathbf{1.32 \cdot 10^{-4}}$	$\mathbf{2.50 \cdot 10^{-4}}$	$\mathbf{3.94 \cdot 10^{-3}}$	0.543	48.9	6.24	605
Nested	**27.0**	**76.6**	$\mathbf{5.69 \cdot 10^{-4}}$	$\mathbf{1.03 \cdot 10^{-5}}$	$\mathbf{-2.60 \cdot 10^{-4}}$	$\mathbf{3.96 \cdot 10^{-3}}$	0.539	49.2	1.60	213
Sequential (3 it.)	**27.0**	**81.7**	$\mathbf{4.08 \cdot 10^{-4}}$	$\mathbf{5.48 \cdot 10^{-5}}$	$\mathbf{-1.87 \cdot 10^{-4}}$	$\mathbf{3.94 \cdot 10^{-3}}$	0.539	49.2	7.39	1335

All values indicated in bold and blue are design parameters for the relevant optimization: e.g. **x.xx**.

optimal motion profile for every geometry, utilizing a torque equation obtained from three CAD simulations. This co-optimization method is compared to a conventional sequential iteration method and a simultaneous co-optimization. A convergence analysis is performed on the polynomial degree of the motion profile to ensure the co-optimizations are efficient and accurate. Using CAD simulations enables the co-optimization methods to be implemented on all mono-actuated mechanisms. Five different optimizations were performed on two different starting designs to thoroughly evaluate the performance of the co-optimization methods. A starting design close to the feasibility border causes problems for the nested co-optimizations to find the optimum. Consequently, a starting design further from the feasibility border enables the nested co-optimization to find the optimum. The nested co-optimization and the three sequential iterations resulted in a T_{rms} saving of 49.2% compared to the *base design*. The simultaneous co-optimization nearly reached a similar saving of 48.9%. The major advantage of the proposed nested co-optimization is the ability to converge rapidly, it delivers a time saving of 74.4% compared to the simultaneous co-optimization and a time saving of 78.3% compared to three sequential iterations. The major time savings indicate the significance of the proposed co-optimization architecture.

REFERENCES

[1] I. (2022), "World energy outlook 2022," *IEA*, 2022. [Online]. Available: https://www.iea.org/reports/world-energy-outlook-2022

[2] U. D. of Energy, "International energy outlook 2019 with projections to 2050," p. 85, 2019.

[3] A. De Almeida, J. Fong, C. U. Brunner, R. Werle, and M. Van Werkhoven, "New technology trends and policy needs in energy efficient motor systems - a major opportunity for energy and carbon savings," *Renewable and Sustainable Energy Reviews*, vol. 115, no. 109384, 2019.

[4] A. Ben Yahya, N. Van Oosterwyck, J. Herregodts, S. Herregodts, S. Houwen, B. Vanwalleghem, and S. Derammelaere, "An industrial applicable approach towards design optimization of a mechanism: a coronaventilator case study," *arXiv preprint arXiv:2207.14537*, 2022.

[5] N. V. Oosterwyck, F. Vanbecelaere, F. Knaepkens, M. Monte, K. Stockman, A. Cuyt, and S. Derammelaere, "Energy optimal point-to-point motion profile optimization," *Mechanics Based Design of Structures and Machines*, vol. 0, no. 0, pp. 1–18, 2022. [Online]. Available: https://doi.org/10.1080/15397734.2022.2106241

[6] J. R. Martins and A. B. Lambe, "Multidisciplinary design optimization: a survey of architectures," *AIAA journal*, vol. 51, no. 9, pp. 2049–2075, 2013.

[7] J. Park, "Motion profile planning of repetitive point-to-point control for maximum energy conversion efficiency under acceleration conditions," *Mechatronics*, vol. 6, no. 6, pp. 649–663, 1996.

[8] Z. Affi, E. Badreddine, and L. Romdhane, "Advanced mechatronic design using a multi-objective genetic algorithm optimization of a motor-driven four-bar system," *Mechatronics*, vol. 17, no. 9, pp. 489–500, 2007.

[9] J. T. Allison, T. Guo, and Z. Han, "Co-design of an active suspension using simultaneous dynamic optimization," *Journal of Mechanical Design*, vol. 136, no. 8, p. 081003, 2014.

[10] A. P. Deshmukh and J. T. Allison, "Multidisciplinary dynamic optimization of horizontal axis wind turbine design," *Structural and Multidisciplinary Optimization*, vol. 53, pp. 15–27, 2016.

[11] J. T. Allison and S. Nazari, "Combined plant and controller design using decomposition-based design optimization and the minimum principle," in *International design engineering technical conferences and computers and information in engineering conference*, vol. 44090, 2010, pp. 765–774.

[12] D. R. Herber and J. T. Allison, "Nested and simultaneous solution strategies for general combined plant and control design problems," *Journal of Mechanical Design*, vol. 141, no. 1, p. 011402, 2019.

[13] J. Sun, M. Yao, X. Xiao, Z. Xie, and B. Zheng, "Co-optimization of morphology and behavior of modular robots via hierarchical deep reinforcement learning," in *Robotics: Science and Systems (RSS)*, vol. 2023, 2023.

[14] T. Dinev, C. Mastalli, V. Ivan, S. Tonneau, and S. Vijayakumar, "A versatile co-design approach for dynamic legged robots," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 10 343–10 349.

[15] A. Spielberg, B. Araki, C. Sung, R. Tedrake, and D. Rus, "Functional co-optimization of articulated robots," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 5035–5042.

[16] J. T. Allison and D. R. Herber, "Special section on multidisciplinary design optimization: multidisciplinary design optimization of dynamic engineering systems," *AIAA journal*, vol. 52, no. 4, pp. 691–710, 2014.

[17] C.-Y. Chen and C.-C. Cheng, "Integrated design for a mechatronic feed drive system of machine tools," in *Proceedings, 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics.* IEEE, 2005, Conference Proceedings, pp. 588–593.

[18] J. Kwon, S. Kim, and F. C. Park, "Physically consistent lie group mesh models for robot design and motion co-optimization," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9501–9508, 2022.

[19] S. Ha, S. Coros, A. Alspach, J. Kim, and K. Yamane, "Computational co-optimization of design parameters and motion trajectories for robotic systems," *The International Journal of Robotics Research*, vol. 37, no. 13-14, pp. 1521–1536, 2018.

[20] W. J. Thompson, "Chebyshev polynomials: After the spelling the rest is easy," *Computers in physics*, vol. 8, no. 2, pp. 161–165, 1994.

[21] N. Van Oosterwyck, F. Vanbecelaere, M. Haemers, D. Ceulemans, K. Stockman, and S. Derammelaere, "CAD enabled trajectory optimization and accurate motion control for repetitive tasks," in *2019 IEEE 15th international conference on control and automation (ICCA)*. IEEE, 2019, pp. 387–392.

[22] H. Dresig and F. Holzweißig, *Dynamics of machinery: theory and applications*. Springer Science & Business Media, 2010.

[23] P. T. Boggs and J. W. Tolle, "Sequential quadratic programming for large-scale nonlinear optimization," *Journal of computational and applied mathematics*, vol. 124, no. 1-2, pp. 123–137, 2000.

[24] J. N. S. J. Wright, "Numerical optimization," 2006.