Article

Optimization of Variable Stiffness Joint in Robot Manipulator Using a Novel NSWOA-MARCOS Approach

G. Shanmugasundar [1], Vishal Fegade [2], Miroslav Mahdal [3,*] and Kanak Kalita [4,*]

[1] Department of Mechanical Engineering, Sri Sairam Institute of Technology, Chennai 600044, India; shanmugasundar.mech@sairamit.edu.in
[2] Department of Mechanical Engineering, MPSTME, SVKM's Narsee Monjee Institute of Management Studies (NMIMS), Shirpur Campus, Dhule 425405, India; vishal.fegade@nmims.edu
[3] Department of Control Systems and Instrumentation, Faculty of Mechanical Engineering, VSB-Technical University of Ostrava, 17. Listopadu 2172/15, 708 00 Ostrava, Czech Republic
[4] Department of Mechanical Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi 600062, India
* Correspondence: miroslav.mahdal@vsb.cz (M.M.); drkanakkalita@veltech.edu.in (K.K.)

Abstract: Robots and robotic systems have become an inevitable part of modern industrial settings. Robotics systems are being introduced for various household services as well. As the interactions between the workspace of robots and humans increases, there is an increased likelihood of unintended harm being caused by the robots to humans due to collisions or abrupt contact. To mitigate this, active and passive compliant mechanisms must be introduced in these systems. In this study, a design optimization case study is carried out for the optimization of a passive compliance mechanism achieved with variable stiffness joints realized by the use of permanent magnets. Three design parameters of the systems, namely, inner stator width, outer stator width, and magnet height, are considered. The objective is to minimize the weight and maximize the maximum torque. A nature-inspired metaheuristic hybridized with a multi-criteria decision-making method is introduced to achieve this. The Non-dominated Sorting Whale Optimization Algorithm (NSWOA) is used for Pareto optimal front generation and MARCOS (Measurement of Alternatives and Ranking according to COmpromise Solution) is applied to extract the best feasible solution from the Pareto front. We observed 1.8% and 41% improvements as compared to the previous known best design and original design, respectively.

Keywords: optimization; robots; design; modeling; design parameters

check for updates

Citation: Shanmugasundar, G.; Fegade, V.; Mahdal, M.; Kalita, K. Optimization of Variable Stiffness Joint in Robot Manipulator Using a Novel NSWOA-MARCOS Approach. *Processes* **2022**, *10*, 1074. https://doi.org/10.3390/pr10061074

Academic Editor: Raul D.S.G. Campilho

Received: 1 May 2022
Accepted: 26 May 2022
Published: 27 May 2022

1. Introduction

Robotic systems have become an important part of modern industries and are expected to revolutionize the way household chores are done in the coming days. The ability of robotic systems to do repetitive and tedious tasks day in and day out makes them an asset. However, human safety around robotic systems is an important concern. Traditionally, in industrial settings, the robotic system and humans are separated by ensuring appropriate segregation measures in the workspace. However, for service robots involved in household chores, such segregation is difficult to achieve. Robotic systems may unwittingly harm humans around them due to abrupt contact or carelessness of human operators. To mitigate such safety risks, introducing compliance to the joints has been widely theorized and researched. One way to do this is by active compliance, which mimics mechanical compliance by using sensors and actuators [1]. Shetty and Ang [2] used a force feedback mechanism to achieve compliance for a rigid-joint robot. Zinn et al. [3] on the other hand relied on distributed macro–mini actuation. They used two different actuators—a large and a small actuator. High-torque, low-frequency movement was ensured by the large actuator, while the small actuator was responsible for low-torque, high-frequency movements.

Active compliance brings a lot of flexibility to robotic systems. However, many times, due to failure of sensors or low sample frequency, active sensors may become unreliable, and inadvertently the safety of humans may be in question. Passive compliance mechanisms, in general, are more reliable from a safety perspective. Variable stiffness joints are relatively inexpensive and efficient in ensuring no abrupt contact with humans occurs. Upon impact with humans, variable stiffness joints rapidly lower the stiffness of joints to avoid any major damage to the human operator. Tonietti et al. [4] demonstrated a robotic arm comprising linear actuator-aided variable stiffness. Rotary-type permanent magnets were used by Yun et al. [5] to achieve variable stiffness. The variable stiffness joints should have appropriate stiffness during the movement of the manipulator so that the robotic system can carry the payload. According to Yoo et al. [6], in a typical human living environment, variable stiffness joints should be capable of generating more than 10 Nm torque for supporting a 1 kg payload at a 1 m length of robot manipulator. However, using traditional electric motors may make the variable stiffness joints bulky. To mitigate this, Yoo et al. [6] proposed variable stiffness joints made neodymium–iron–boron ring-type permanent magnets. Magnets are sometimes used in robotic systems along with direct current motors [7].

In several works in the literature, to ensure the proper design of the variable stiffness joints, optimization theory is used. Hyun et al. [8] used a response surface methodology (RSM) to derive empirical relations of torque and weight with respect to various design parameters of permanent magnets like inner stator width, outer stator width, and magnet height. Yoo et al. [6] did a similar analysis and reported a significant improvement over the baseline model. Using a finite element modeling route, Choi and Yoo [9] designed a Halbach magnet array. They used numerical optimization techniques to optimize their design. Very recently, Song et al. [10] discussed a two-step optimization methodology to optimize the maximum speed and impact force reduction capability of variable stiffness robots.

Apart from variable stiffness joints, optimization theory has been applied to other facets of robotic system design as well. Hsiao et al. [11] designed and optimized high-speed robotic arms to achieve weight reduction, moment of inertia reduction, and deformation reduction. They used a finite element analysis to develop the necessary sampling dataset to generate the empirical RSM models. They reported improvements of about 16, 23, and 20% in the weight reduction, moment of inertia reduction, and deformation reduction, respectively. Chau et al. [12] designed and optimize a compliant planar spring using a Kriging model. They deployed a multi-objective genetic algorithm to achieve the optimization task. Zhao et al. [13] used a non-dominated sorting genetic algorithm to design a wall-climbing robot for stability and weight reduction. They were able to reduce the vibration by 6.5% and the weight of the magnetic abortion unit by 9.7%.

From the above literature review, it is clear that recently researchers have started tapping into the potential of optimization methodologies to design and optimize robotic systems. However, the use of nature-inspired optimization techniques is very limited in this field [14,15]. Mostly, nature-inspired optimization techniques are used in path planning applications in robotics [14,15]. In this paper, a very recently developed nature-inspired metaheuristic algorithm called the Non-dominated Sorting Whale Optimization Algorithm (NSWOA) is applied. NSWOA is a previously developed algorithm by Jangir and Jangir [16] in 2017 for continuous optimization problems. However, so far it has not been applied to robot design problems. Since NSWOA generates the optimized solutions in form of Pareto fronts, a multi-criteria decision-making method called MARCOS (Measurement of Alternatives and Ranking according to COmpromise Solution) is applied to extract solutions for predefined application scenarios. The proposed hybrid NSWOA-MARCOS method is then applied to a variable stiffness joint design and optimization problem. The rest of the paper is arranged as follows. The next section details the methodology used. The NSWOA and MARCOS methods are briefly discussed in this section. The description of the case study considered in this paper is discussed in Section 3. Section 4 begins with the understanding of the parametric effect of the design features on the responses. Next, the parametric optimization is carried out by NSWOA-MARCOS. Finally, the knowledge derived based on this study is summarized in the Section 5.

2. Methodology

In this paper, a hybrid NSWOA-MARCOS method for the optimal design of robotic components is proposed. The method comprises two main parts: (1) a nature-inspired metaheuristic algorithm infused with a non-dominated solution sorting strategy to generate a Pareto front and (2) a multi-criteria decision-making method to extract the most viable solutions from the Pareto front depending on the application scenario.

2.1. Whale Optimization Algorithm

The whale optimization algorithm (WOA) was initially propounded by Mirjalili and Lewis [17] in 2016 and is a nature-inspired metaheuristic algorithm that mimics the social behavior of humpback whales.

Humpback whales identify the prey's location and encircle it [18]. Similarly, the whale optimization algorithm assumes that the current best solution is the prey. The position of the other search agents or candidate solutions are updated with reference to the best solution. The following equation represents the mathematical equivalent of prey encirclement by humpback whales

$$\vec{D} = \left| \vec{C}.\vec{X}_{best}(t) - \vec{X}(t) \right| \tag{1}$$

$$\vec{X}(t+1) = \vec{X}_{best}(t) - \vec{A}.\vec{D} \tag{2}$$

where $\vec{X}_{best}$ and $\vec{X}$ are the location vector of the best whale and any other whale, respectively. t indicates the current iteration. At the $(t+1)$ iteration, the location of the whale is given as $\vec{X}(t+1)$. The "." between vectors represent an element-wise product. The "| |" represents the absolute values. The coefficient vectors $\vec{A}$ and $\vec{C}$ can be calculated as

$$\vec{A} = 2\vec{a}.\vec{r} - \vec{a} \tag{3}$$

$$\vec{C} = 2.\vec{r} \tag{4}$$

With the increase in iterations, $\vec{a}$ linearly decreases from 2 to 0. $\vec{r}$ is a random vector having a [0, 1] range. The various locations around the best solution so far are computed by adjusting $\vec{A}$ and $\vec{C}$ vectors.

The first phase of WOA is the exploitation phase based on the bubble-net feeding method of whales. Humpback whales make use of a shrinking circle and spiral-shaped path to swim around the prey [18]. Two approaches are considered to model this feeding technique mathematically: shrinking encircling mechanism and spiral position updating [19].

By decreasing the value of $\vec{a}$ in Equation (3), the shrinking encircling mechanism of humpback whales is realized [17].

To update the spiral position [17], the first step is to compute the distance between the whales and prey. The helix-shaped movement of the humpback whale is mathematically represented as

$$\vec{X}(t+1) = \left| \vec{X}_{best}(t) - \vec{X}(t) \right| .e^{bt}.\cos(2\pi l) + \vec{X}_{best}(t) \tag{5}$$

where b is a constant to define the logarithmic spiral's shape, l is a random number within the range of $[-1, 1]$.

At any given instant, the humpback whale may update its position either by the shrinking circle method or by the spiral path method. To model these two methods simultaneously, a 50% probability is assigned to choose between either method, which is mathematically represented as [17].

$$\vec{X}(t+1) = \begin{cases} \vec{X}_{best}(t) - \vec{A}.\vec{D} & \textit{if } p < 0.5 \\ \left|\vec{X}_{best}(t) - \vec{X}(t)\right|.e^{bt}.\cos(2\pi l) + \vec{X}_{best}(t) & \textit{if } p \geq 0.5 \end{cases} \tag{6}$$

where p is a random number within the range of [0, 1], the value of p is responsible for switching between a spiral or circular movement.

For the exploration phase, i.e., to search the prey, the same approach of variation of the $\vec{A}$ vector is used [18]. The humpback whales randomly search for prey. If $\left|\vec{A}\right| \geq 1$, a random whale is chosen. However, if $\left|\vec{A}\right| < 1$, the best solution is selected when changing the location of the search agents. Mathematically this is achieved by [17]

$$\vec{D} = \left|\vec{C}.\vec{X}_{rand} - \vec{X}\right| \tag{7}$$

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{A}.\vec{D} \tag{8}$$

where $\vec{X}_{rand}$ is a random location vector selected from the present population.

2.2. Non-Dominated Sorting Whale Optimization Algorithm

To model the multi-objective version of the WOA an archive is first incorporated into this algorithm. This archive is combined to store the best non-dominated solutions acquired so far. The solution procedure of the multi-objective whale optimization algorithm [20,21] is very analogous to that of the WOA. That is, solutions are established considering the bubble-net feeding method of the humpback whales. A leader selection mechanism is introduced to elect the solutions from the archive [22]. A roulette wheel is also incorporated in the MOWOA to select solutions from the less occupied areas of the archive [23]. The improvement of the distribution of solutions in the archive across all objectives is made using the following probability function [24]

$$P_i = \frac{c}{N_i} \tag{9}$$

where c is a constant and should be greater than 1 and N_i is the number of solutions in the neighborhood of ith solution.

Meanwhile, the archive has a limit to storing non-dominating solutions, and it might become full with the progress of iterations. So, a mechanism is also implemented to eliminate the undesirable solutions from the archive. Undesirable solutions are those solutions that have many adjacent solutions. To discard the undesirable solutions from the archive of NSWOA a probability is employed to give a high value to the undesired solutions. The probability function is the inverse of the previous probability function (Equation (9)), which is used to select the optimal solution from the archive, and it is given as follows [24]

$$P_i' = \frac{N_i}{c} \tag{10}$$

2.3. MARCOS

The relation between any alternative and the ideal and anti-ideal solutions forms the backbone of the MARCOS method. Certain utility functions are determined based on these relationships and the best possible alternative is then determined by ranking all the alternatives.

The Pareto front obtained from the NSWOA serves as the input matrix to the MARCOS method. Any multi-criteria decision-making (MCDM) method has an input matrix or the initial decision matrix of the form $m \times n$. m and n represent the number of alternatives and the criteria. Here, the solutions in the Pareto front are the alternatives, and the two responses, i.e., torque and weight, are the criteria. The initial decision matrix (D) for any MCDM can be written as,

$$D = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \tag{11}$$

here, x_{11}, x_{mn}, etc. are the candidate solutions.

For MARCOS, an extended decision matrix (X) of the following form is used,

$$X = \begin{bmatrix} x_{aa1} & x_{aa2} & \cdots & x_{aan} \\ x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \ddots & x_{mn} \\ x_{ai1} & x_{ai2} & \cdots & x_{ain} \end{bmatrix} \tag{12}$$

here, x_{aa1}, x_{aan}, etc. are the anti-ideal (AAI) solutions and x_{ai1}, x_{ain}, etc. are the ideal (AI) solutions.

The anti-ideal (AAI) solutions and the ideal (AI) solutions in the extended decision matrix are determined using the following relations,

$$\begin{aligned} AAI &= \min_i x_{ij} \text{if} j \in B \text{and} \max_i x_{ij} \text{if} j \in C \\ AI &= \min_i x_{ij} \text{if} j \in B \text{and} \max_i x_{ij} \text{if} j \in C \end{aligned} \tag{13}$$

here, B and C are the benefit and the cost criteria respectively. In terms of this paper, the torque is the benefit criteria and the weight is the cost criteria.

The extended decision matrix (X) is then normalized using the following relations to form the normalized decision matrix (N).

$$n_{ij} = \frac{x_{ai}}{x_{ij}} \text{ if} j \in C n_{ij} = \frac{x_{ij}}{x_{ai}} \text{ if} j \in B \tag{14}$$

The extended decision matrix (X) is of the following form,

$$N = \begin{bmatrix} n_{aa1} & n_{aa2} & \cdots & n_{aan} \\ n_{11} & n_{12} & \cdots & n_{1n} \\ n_{21} & n_{22} & \cdots & n_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ n_{m1} & n_{m2} & \ddots & n_{mn} \\ n_{ai1} & n_{ai2} & \cdots & n_{ain} \end{bmatrix} \tag{15}$$

The weighted decision matrix (V) is then determined by multiplying the weight vector with the normalized decision matrix (N). The weight vector is expressed as,

$$W = \begin{bmatrix} w_1 & w_2 & \cdots & w_n \end{bmatrix} \tag{16}$$

here, w_1, w_n, etc. are the weights assigned to each of the criteria. Additionally, the sum of the weights must be equal to unity, i.e.,

$$w_1 + w_2 + \cdots + w_n = 1 \tag{17}$$

The weighted decision matrix (V) can be written as,

$$V = \begin{bmatrix} v_{aa1} & v_{aa2} & \cdots & v_{aan} \\ v_{11} & v_{12} & \cdots & v_{1n} \\ v_{21} & v_{22} & \cdots & v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{m1} & v_{m2} & \ddots & v_{mn} \\ v_{ai1} & v_{ai2} & \cdots & v_{ain} \end{bmatrix} \tag{18}$$

Next, the utility degree of the alternatives (K_i) is determined using the following two equations,

$$K_i^- = \frac{S_i}{S_{aai}} \tag{19}$$

$$K_i^+ = \frac{S_i}{S_{ai}} \tag{20}$$

here, S_i is the sum of the elements of the weighted decision matrix (V). S_i is determined using the following equation,

$$S_i = \sum_{j=1}^{n} v_{ij} \tag{21}$$

The utility function of the alternatives $f(K_i)$ is then computed,

$$f(K_i) = \frac{K_i^+ + K_i^-}{1 + \frac{1-f\left(K_i^+\right)}{f\left(K_i^+\right)} + \frac{1-f\left(K_i^-\right)}{f\left(K_i^-\right)}} \tag{22}$$

here, $f\left(K_i^-\right)$ and $f\left(K_i^+\right)$ are the utility functions in relation to anti-ideal (AAI) solutions and the ideal (AI) solutions.

$$f\left(K_i^-\right) = \frac{K_i^+}{K_i^+ + K_i^-} \tag{23}$$

$$f\left(K_i^+\right) = \frac{K_i^-}{K_i^+ + K_i^-} \tag{24}$$

Finally, the alternatives are ranked from rank 1 to n in descending order of $f(K_i)$ values.

3. Problem Description

The design and optimization of robotic components is an essential task due to the growing demand for robotic systems in modern world applications. However, given the intricate nature of associated systems, this task is a challenging one. In this work, a nature-inspired evolutionary algorithm, namely, the whale optimization algorithm, is employed for the design optimization of a variable stiffness joint in a robot manipulator. The case study considered here is adapted from Yoo et al. [6]. Figure 1a shows the typical location of variable stiffness joints in the robot manipulator arm and Figure 1b shows the architecture of the three-ring permanent magnet-type variable stiffness joint design considered by Yoo et al. [6]. The variable stiffness joint in a robot manipulator is aimed to be used in a robot arm that can support a 1 kg payload. The objective of the optimization task is to minimize the overall weight of the variable stiffness joint. However, it is assumed that the rotational stiffness of the variable stiffness joint must be at least 10 Nm. In single-objective optimization, this can be achieved by modeling the torque requirement as a constraint to the problem and formulating the problem as a constrained optimization problem. However, in this paper, the optimization problem is formulated as a Pareto optimization problem where maximizing rotational stiffness is the second objective. Further, in the archiving step of the non-dominated solutions, along with Equation (10), an additional constraint of rejecting solutions of rotational stiffness values < 10 Nm is applied. This ensures the entry of only those solutions to the non-dominated solution archive that met the rotational stiffness constraint criteria.

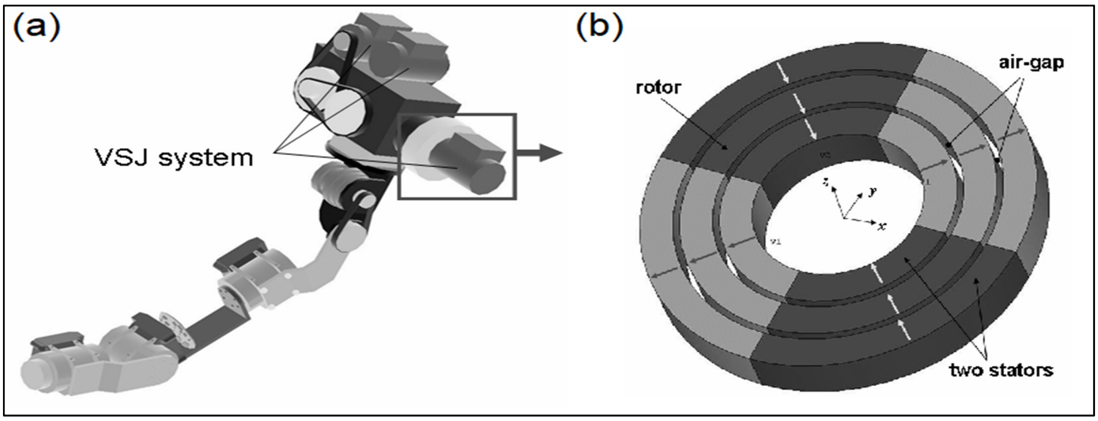

Figure 1. (**a**) Typical locations of variable stiffness joints in the robot manipulator arm. (**b**) Three-ring permanent magnet-type variable stiffness joint design considered by Yoo et al. [6], copyright 2010, Springer Nature.

Further, the variable stiffness joint system is assumed to be composed of the stator and the rotor connected with the joint. The inner stator width (W_i), outer stator width (W_o), and magnet height (H) are considered as the design parameters. For the optimization phase, the following constraints are applied to the design variables

$$6 \text{ mm} \leq W_i \leq 10 \text{ mm} 5 \text{ mm} \leq H \leq 15 \text{ mm} 6 \text{ mm} \leq W_o \leq 10 \text{ mm} \tag{25}$$

Based on a set of experiments conducted as per central composite design, Yoo et al. [6] proposed the following second-order empirical relationships for the stated design variables and maximum torque and weight. It should be noted that W_i, W_o, and H are in mm, whereas maximum torque and weight are in Nm and kg, respectively. However, Equations (26) and (27) are provided in coded form. Thus, the range of all the variables in Equations (26) and (27) is ±1.

$$\begin{aligned} T = 8.8949 + 0.9555W_i + 0.8100W_o + 5.7616H - 0.0703{W_i}^2 - 0.1246{W_o}^2 \\ +0.2499H^2 + 0.0292W_i.W_o + 0.6722W_i.H + 0.6028W_o.H \end{aligned} \tag{26}$$

$$\begin{aligned} W = 0.4954 + 0.0483W_i + 0.0520W_o + 0.2483H - 0.0009{W_i}^2 - 0.0009{W_o}^2 \\ +0.0019W_i.W_o + 0.0242W_i.H + 0.0260W_o.H \end{aligned} \tag{27}$$

The empirical relationships in Equations (26) and (27) are used as the objective functions for the NSWOA algorithm. Yoo et al. [6] have reported excellent estimation power of the equations with R^2 of 0.97 and 1 for Equations (26) and (27), respectively. R^2, i.e., the coefficient of determination, indicates the amount of variance in the data captured by the models.

4. Results and Discussion

4.1. Parametric Study

Before the optimization of the variable stiffness joint can be carried out, it is important to look at the relationships of the design parameters with the design responses. For this, a five-level full factorial design consisting of 125 sampling points is constructed. The corresponding values of maximum torque and weight are calculated using Equations (26) and (27). Figure 2 shows the mean of the maximum torque for level-wise aggregation of the design parameters. It is seen that on an average the maximum torque increases with an increase in inner stator width, outer stator width, and magnet height. However, the order of increment is more in the case of magnet height as opposed to the other two parameters. A similar trend for level-wise averaged weight is seen in Figure 3. The level-wise average weight of the variable stiffness joint increases with an increase in inner stator width, outer stator width, and magnet height. This is because an increase in the width in most cases indicates more material used and thus the weight increases. Interestingly, from Figures 2 and 3, it is clear that both the objectives, i.e., maximum torque and weight, are conflicting in nature. It means torque improvement, in general, is not possible without increasing the weight of the system. Thus, this kind of problem is apt to be solved by Pareto optimization, which generates a set of non-dominated solutions that are compromise solutions on the two objectives.

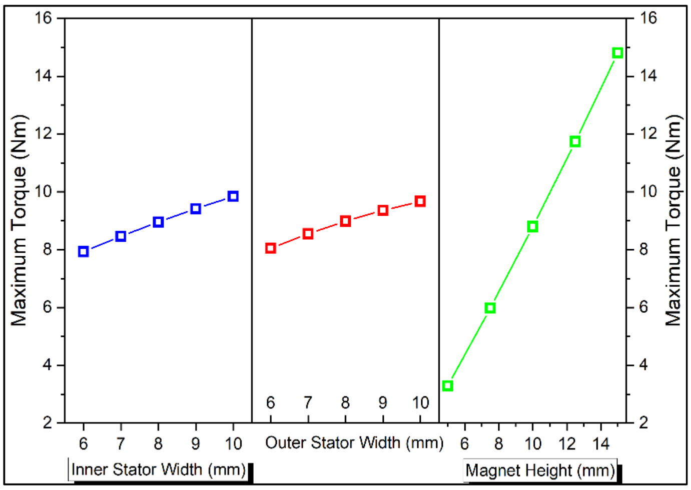

Figure 2. Level-wise average of maximum torque for each design parameter.

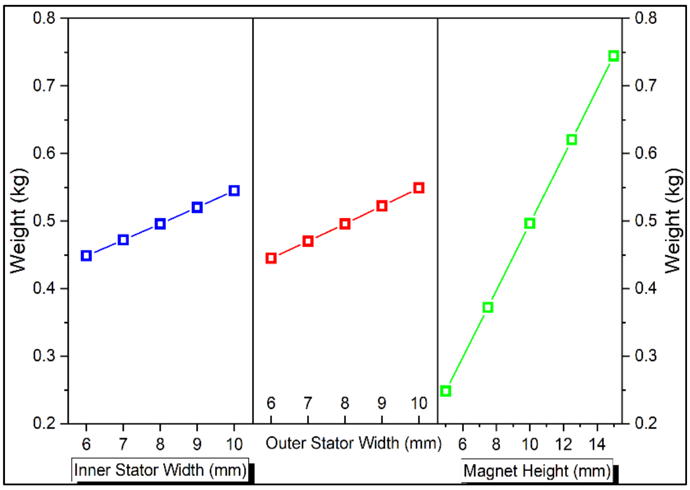

Figure 3. Level-wise average of weight for each design parameter.

To further see the effects of the process parameters on the maximum torque and weight, a parallel plot is shown in Figure 4. Parallel plots are an excellent visual tool to study the features of many individual observations over the range of the problem. To keep the parallel plot manageable, only five levels for each of the three design variables are considered, i.e., 125 combinations of the three design variables are studied. The corresponding values of maximum torque and weight are computed using (25) and (26). Figure 4 shows the complex interaction of the design variables with the response features. It is evident that the high torques are accompanied by higher weights. Moreover, the combined effect of the parameters is seen to be complex and thus, the optimal solutions cannot be derived arbitrarily.

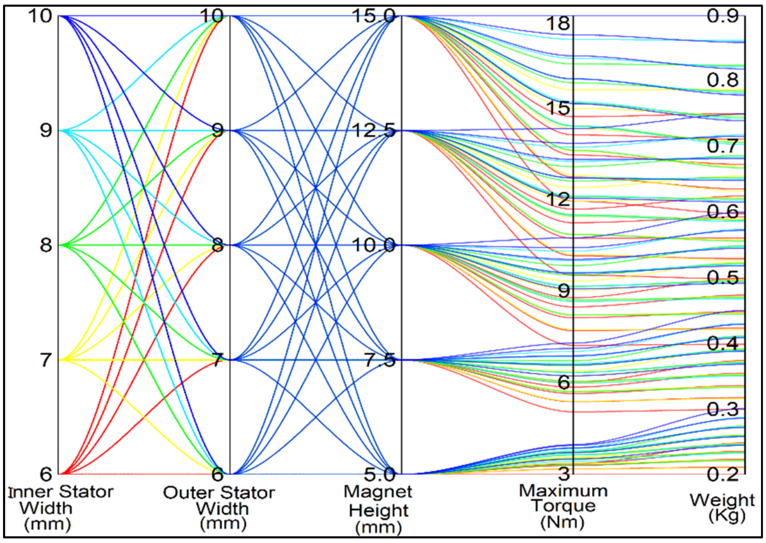

Figure 4. The combined effect of various design parameters on maximum torque and weight.

4.2. Parametric Optimization

The parametric optimization is carried out in NSWOA, which is realized in MATLAB R2019a. All the computational experiments are conducted on a Dell Inspiron 15-3567 series Windows system with Intel(R) Core^{TM} i7-7500U CPU @2.70 GHz, Clock Speed 2.9 Ghz, L2 Cache Size 512, and 8 GB RAM under Microsoft Windows 10 operating environment. For NSWOA, 100 search agents and 500 iterations are considered the termination criteria. The archive size of the non-dominated sorting is fixed at 100. To mitigate any uncertainties in the derived solution due to the stochastic nature of the performance of metaheuristic algorithms, five independent trials of NSWOA are carried out. The best Pareto plot among the five trials is reported in Figure 5. The Pareto plot is observed to be uniform and continuous, indicating that several potential compromise solutions are available for the design problem. The box plots for each of the responses show that the Pareto front is free from skewness in any particular direction. The solutions are evenly present for both low and high torque as well as for low and high weight. This kind of Pareto front-based approach is superior to the single-objective optimization (SOO) methodology adopted by Yoo et al. [6] because, unlike in SOO, several viable solutions are generated in this method, thereby providing the designer with ample choices. It is also important to mention here the computational inexpensiveness of the NSWOA. For a typical NSWOA trial, the average computational time is 21.285 s with a standard deviation of 0.6895. Moreover, the average number of non-dominated solutions generated in the five trials of NSWOA is approximately 52 (±1). Thus, it is evident that NSWOA is stable from both solutions generated and computational time points of view.

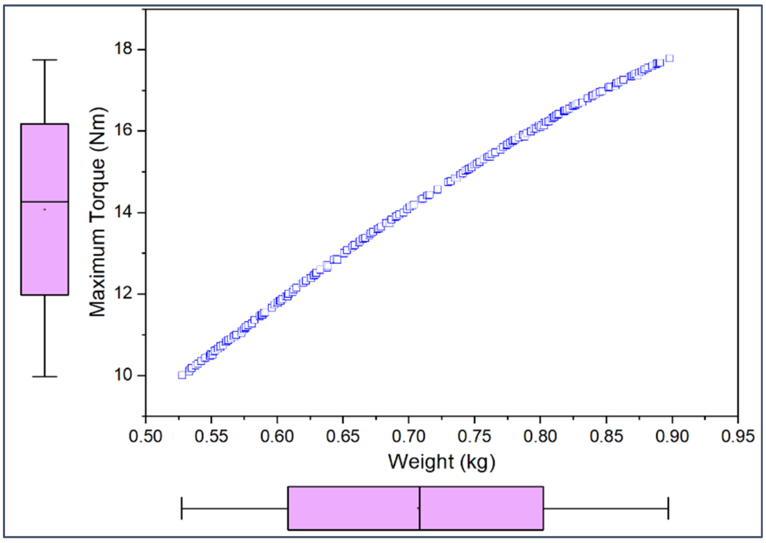

Figure 5. Pareto plot and box plots for the responses.

Next, all the five Pareto fronts obtained from each of the NSWOA simulations are collated together to form the decision matrix for MARCOS. After removing the non-unique solutions, in total, 231 candidate solutions to the problem are obtained. Two different scenarios are assumed—the first scenario where 90% importance is given to weight response whereas in the second scenario where 90% importance is given to torque. Figure 6 shows the K^{-} and K^{+} values for the first scenario with respect to the weight of the variable stiffness joint. It is observed that the K^{-} and K^{+} values will decrease gradually as the weight of the variable stiffness joint increases. On the other hand, for the second scenario as shown in Figure 7, K^{-} and K^{+} values will increase gradually as the weight of the variable stiffness joint increases. This is because in the second scenario, the focus is on torque improvement and as seen from the previous section torque improvement and weight reduction in this case study are conflicting in nature.

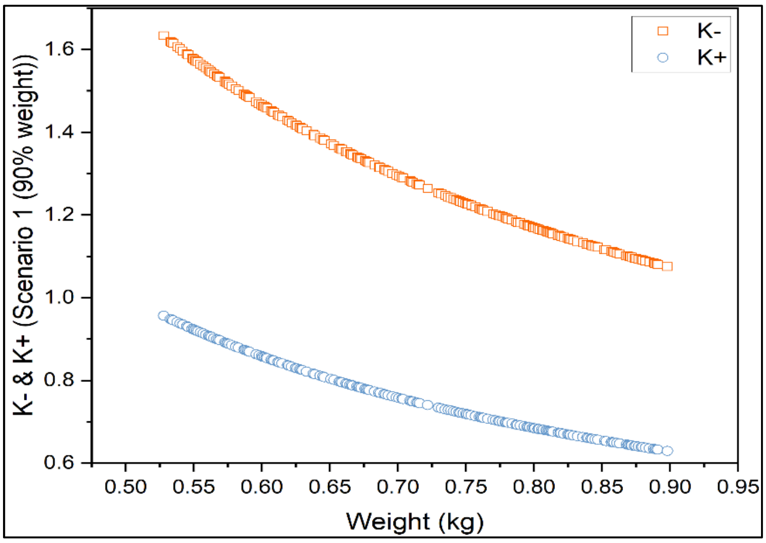

Figure 6. K^{-} and K^{+} values for the first scenario with respect to the weight of the variable stiffness joint.

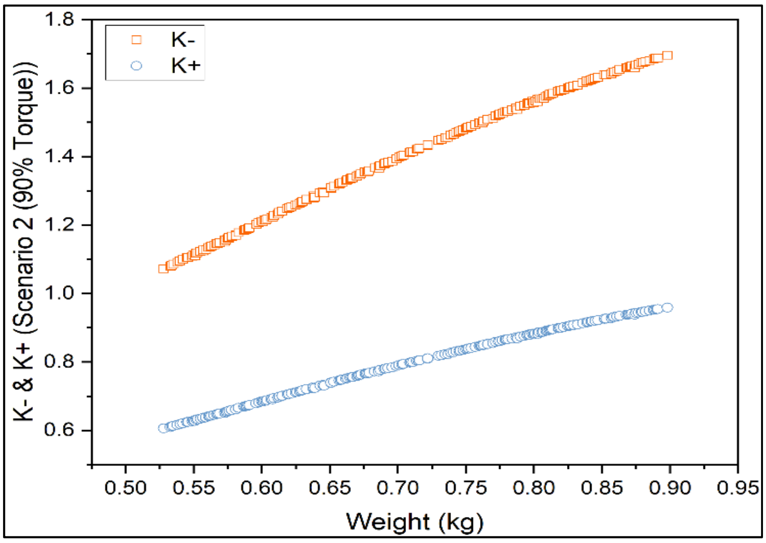

Figure 7. K^- and K^+ values for the second scenario with respect to the weight of the variable stiffness joint.

Finally, the variation of the MARCOS performance metric $f(K_i)$ values are plotted with respect to the weight and maximum torque of the variable stiffness joint and shown in Figure 8. It is seen that for the first scenario, where the focus is on weight reduction, $f(K_i)$ values decrease as weight increases. This is because in MARCOS higher the $f(K_i)$ values better are the alternative. In this scenario, the focus of MARCOS is to reduce weight.

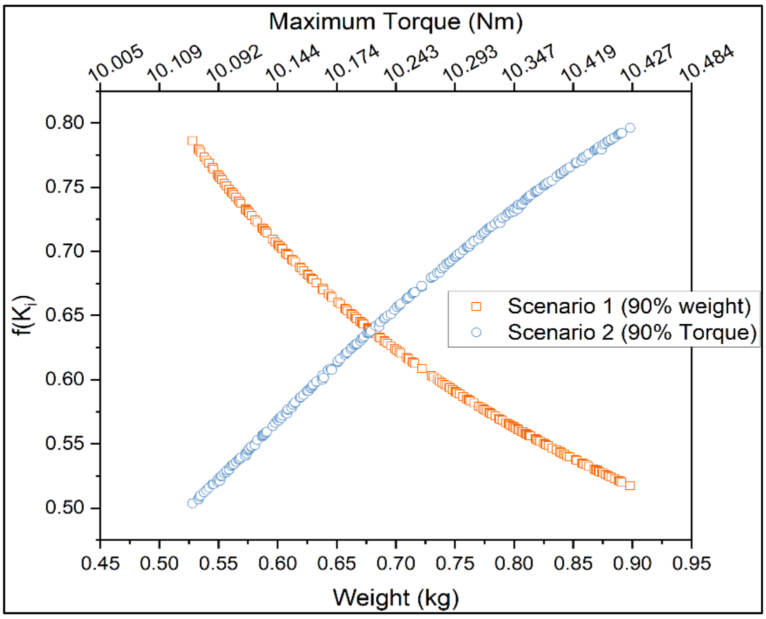

Figure 8. $f(K_i)$ values with respect to the weight and maximum torque of the variable stiffness joint.

However, in the second scenario, where the focus is on increasing the maximum torque, with any increase in weight (accompanied by an increase in maximum torque), $f(K_i)$ values increase. If in the second scenario $f(K_i)$ values are plotted against torque, it is seen that $f(K_i)$ values increase with an increase in torque.

The optimal values of the design parameters and responses for both scenarios are presented in Table 1. It is clear that based on the scenario i.e., the importance accorded to torque maximization or weight minimization, the design parameters drastically change. This depicts the importance of the proposed method that intensively can offer the designer a multitude of solutions depending on the specific application/scenario to be considered. Further, the improvement of the solutions obtained by the proposed method over traditional approaches reported by Yoo et al. [6] is presented in Table 2. It should be noted that the solution presented by Yoo et al. [6] corresponds to scenario 1 and thus, is compared only with it in this paper. Further, as mentioned by Yoo et al. [6], their objective is to minimize the weight with a constraint of torque such that the maximum torque is greater than 10 Nm. The current solution satisfies the torque constraint and is better than the weight minimization of Yoo et al. [6] by 1.79%. Despite being incrementally better than the previous optimized solution, the improvement can be considered significant considering the critical nature of applications robotic systems are used in. However, it should be noted that with respect to the original design the current solution is 41.26% better.

Table 1. Optimal design parameter combinations (coded) and corresponding responses.

	Coded Parameters			Un-Coded Parameters (in mm)			Maximum Torque	Weight
	W_i	W_o	H	W_i	W_o	H	(Nm)	(kg)
Scenario 1	−1	−1	0.654	6.00	6.00	13.27	10.005	0.528
Scenario 2	1	1	1	10.00	10.00	15.00	17.781	0.898

Table 2. Improvement of the current solution.

Source	Maximum Torque (Nm)	Weight (kg)
Original model [6]	18.4	0.899
Optimized solution [6]	10.393	0.538
Current optimized solution	10.005	0.528
Improvement with respect to the original	−45.62%	41.26%
Improvement with optimized solution of [6]	−3.73%	1.79%

5. Conclusions

In this article, a novel approach to design the optimization of robotic components is presented. Since robotic systems must consider multiple objectives, a Pareto optimal front generation capable, non-dominated sorting whale optimization algorithm (NSWOA) is used. The NSWOA generates a multitude of non-dominated solutions which are then passed through a MARCOS algorithm to select the optimal compromise solution for a pre-specified application. A case study of optimal variable stiffness joint by using permanent magnets is tackled in this paper. The inner stator width, outer stator width, and magnet height are the design parameters considered to minimize the weight and maximize the torque generated. It was found during the design process that weight and torque generated are conflicting objectives in nature i.e., in general, one cannot be improved without deteriorating the other. With respect to the previously reported best solution in the literature, an improvement of about 1.8% in weight reduction was observed in the new design. The improvement is about 41% with respect to the original design. The study is limited by consideration of only one nature-inspired algorithm. In the future, this study can be extended to test the hybridization of other metaheuristics like gray wolf optimizer, multiverse optimizer with MCDMs like MABAC, CoCoSo, etc. The concept of fuzzification can also be introduced to account for uncertainty in design.

Author Contributions: Conceptualization, G.S., M.M. and K.K.; data curation, G.S. and V.F.; formal analysis, G.S. and V.F.; investigation, G.S. and V.F.; methodology, K.K.; software, M.M. and K.K.; supervision, M.M.; Writing—original draft, G.S., V.F. and K.K.; writing—review and editing, M.M. and K.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available through email upon request to the corresponding author.

Acknowledgments: The authors thank M Ramachandran, Lead Research Scientist, REST Labs, India for his help in the data curation and some of the analyses.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Choi, J.; Hong, S.; Lee, W.; Kang, S.; Kim, M. A robot joint with variable stiffness using leaf springs. *IEEE Trans.Robot.* **2011**, *27*, 229–238. [CrossRef]
2. Shetty, B.R.; Ang, M.H. Active compliance control of a PUMA 560 robot. In Proceedings of the IEEE International Conference on Robotics and Automation, Minneapolis, MN, USA, 22–28 April 1996.
3. Zinn, M.; Roth, B.; Khatib, O.; Salisbury, J.K. A new actuation approach for human friendly robot design. *Int. J. Robot.Res.* **2004**, *23*, 379–398. [CrossRef]
4. Tonietti, G.; Schiavi, R.; Bicchi, A. Design and control of a variable stiffness actuator for safe and fast physical human/robot interaction. In Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 18–22 April 2005.
5. Yun, S.; Kang, S.; Kim, M.; Hyun, M.; Yoo, J.; Kim, S. A novel design of high responsive variable stiffness joints for dependable manipulator. In Proceedings of the ACMD, Tokyo, Japan; 2006.
6. Yoo, J.; Hyun, M.W.; Choi, J.H.; Kang, S.; Kim, S.-J. Optimal design of a variable stiffness joint in a robot manipulator using the response surface method. *J. Mech. Sci. Technol.* **2009**, *23*, 2236–2243. [CrossRef]
7. Akdas, D.; Medrano-Cerda, G.A. Design of a stabilizing controller for a ten-degree-of-freedom bipedal robot using linear quadratic regulator theory. *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.* **2001**, *215*, 27–43. [CrossRef]
8. Hyun, M.W.; Yoo, J.; Hwang, S.T.; Choi, J.H.; Kang, S.; Kim, S.-J. Optimal design of a variable stiffness joint using permanent magnets. *IEEE Trans. Magn.* **2007**, *43*, 2710–2712. [CrossRef]
9. Choi, J.-S.; Yoo, J. Design of a Halbach magnet array based on optimization techniques. *IEEE Trans. Magn.* **2008**, *44*, 2361–2366. [CrossRef]
10. Song, S.; She, Y.; Wang, J.; Su, H.-J. Toward tradeoff between impact force reduction and maximum safe speed: Dynamic parameter optimization of variable stiffness robots. *J. Mech. Robot.* **2020**, *12*, 054503. [CrossRef]
11. Hsiao, J.C.; Shivam, K.; Chou, C.L.; Kam, T.Y. Shape design optimization of a robot arm using a surrogate-based evolutionary approach. *Appl. Sci.* **2020**, *10*, 2223. [CrossRef]
12. Chau, N.L.; Le, H.G.; Dao, T.-P.; Dang, V.A. Design and optimization for a new compliant planar spring of upper limb assistive device using hybrid approach of RSM-FEM and MOGA. *Arab. J. Sci. Eng.* **2019**, *44*, 7441–7456. [CrossRef]
13. Zhao, Z.; Tao, Y.; Wang, J.; Hu, J. The multi-objective optimization design for the magnetic adsorption unit of wall-climbing robot. *J. Mech. Sci. Technol.* **2022**, *36*, 305–316. [CrossRef]
14. Fong, S.; Deb, S.; Chaudhary, A. A review of metaheuristics in robotics. *Comput. Electr. Eng.* **2015**, *43*, 278–291. [CrossRef]
15. Cruz-Bernal, A. Meta-Heuristic Optimization Techniques and Its Applications in Robotics. In *Recent Advances on Meta-Heuristics and Their Application to Real Scenarios*; IntechOpen: London, UK, 2013.
16. Jangir, P.; Jangir, N. Non-dominated sorting whale optimization algorithm (NSWOA): A multi-objective optimization algorithm for solving engineering design problems. *Glob. J. Res. Eng.* **2017**, *17*, 1–29.
17. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]
18. Gharehchopogh, F.S.; Gholizadeh, H. A comprehensive survey: Whale Optimization Algorithm and its applications. *Swarm Evol. Comput.* **2019**, *48*, 1–24. [CrossRef]
19. Aljarah, I.; Faris, H.; Mirjalili, S. Optimizing connection weights in neural networks using the whale optimization algorithm. *Soft Comput.* **2018**, *22*, 1–15. [CrossRef]
20. Dao, T.-K.; Pan, T.-S.; Pan, J.-S. A multi-objective optimal mobile robot path planning based on whale optimization algorithm. In Proceedings of the 2016 IEEE 13th International Conference on Signal Processing (ICSP), Chengdu, China, 6–10 November 2016. [CrossRef]
21. Tanvir, M.H.; Hussain, A.; Rahman, M.M.; Ishraq, S.; Zishan, K.; Rahul, S.K.; Habib, M.A. Multi-objective optimization of turning operation of stainless steel using a hybrid whale optimization algorithm. *J. Manuf. Mater. Process.* **2020**, *4*, 64. [CrossRef]
22. Got, A.; Moussaoui, A.; Zouache, D. A guided population archive whale optimization algorithm for solving multiobjective optimization problems. *Expert Syst. Appl.* **2020**, *141*, 112972. [CrossRef]
23. Kumawat, I.R.; Nanda, S.J.; Maddila, R.K. Multi-objective whale optimization. In Proceedings of the Tencon 2017–2017 IEEE Region 10 Conference, Penang, Malaysia, 5–8 November 2017. [CrossRef]
24. Wang, J.; Du, P.; Niu, T.; Yang, W. A novel hybrid system based on a new proposed algorithm-Multi-Objective Whale Optimization Algorithm for wind speed forecasting. *Appl. Energy* **2017**, *208*, 344–360. [CrossRef]