Robotica: (2019) volume 37, pp. 2073–2086. © Cambridge University Press 2019
doi:10.1017/S0263574719001085

Kinematic Design Optimization of an Eight Degree-of-Freedom Upper-Limb Exoskeleton

Amin Zeiaee†*, Rana Soltani-Zarrin†, Reza Langari† and Reza Tafreshi‡

† *Department of Mechanical Engineering, Texas A&M University, College Station, TX, USA.*
E-mails: rana.soltani@tamu.edu, rlangari@tamu.edu
‡ *Department of Mechanical Engineering, Texas A&M University at Qatar, Doha, Qatar.*
E-mail: reza.tafreshi@qatar.tamu.edu

(Accepted June 1, 2019. First published online: June 25, 2019)

SUMMARY
This paper studies the problem of optimizing the kinematic structure of an eight degree-of-freedom upper-limb rehabilitation exoskeleton. The objective of optimization is achieving minimum volume and maximum dexterity in the workspace of daily activities specified by a set of upper-arm configurations. To formulate the problem, a new index is proposed for effective characterization of kinematic dexterity for wearable robots. Additionally, a set of constraints are defined to ensure that the optimal design can cover the desired workspace of the exoskeleton, while singular configurations and physical interferences are avoided. The formulated multi-objective optimization problem is solved using an evolutionary algorithm (Non-dominated Sorting Genetic Algorithm II) and the weighted sum approach. Among the resulted optimal points, the point with least sensitivity with respect to the variations of design variables is chosen as the final design.

KEYWORDS: Exoskeletons; Design optimization; Kinematic design; Kinematic dexterity; Workspace maximization.

1. Introduction

Producing repeatable and precisely controllable motions are key attributes of robotic systems, making them advantageous for rehabilitation purposes.[1–3] Capabilities of exoskeletons in providing controlled support for individual joints of the limb make them more desirable than other robotic-based rehabilitation candidates.[4] Despite their advantages over other robot-based solutions, kinematic design of exoskeletons is very challenging.[5–7] This is mainly due to the complexities involved in developing a mechanical structure that can comply with the natural motion of the human body[2,6] and the very large range of motion required in certain joints such as shoulder.[5,6]

Kinematics of a robotic system directly affects its functional capabilities, and therefore, an optimally designed kinematic structure is essential for achieving a good performance. Considering the large number of variables involved in the kinematic design and their complex and interconnected effects on the behavior of the system, employing a systematic optimization technique can be helpful in achieving a well-performing system. This has motivated many research studies to focus on the kinematic optimization problem. Achieving enhanced dexterity,[8–11] expanding reachable workspace,[9,10,12] and minimizing system size[13,14] have been some of the optimization goals studied in the literature. Kinematic optimization of robotic systems is not a trivial task. First, there

* Corresponding author. E-mail: amin.zeiaee@tamu.edu

is no uniformly accepted index for quantifying the kinematic performance. While manipulability and condition number of the Jacobian matrix are the two extensively used dexterity measures in the literature,[8–10] various other Jacobian-based indices such as global conditioning number,[11,15,16] relative manipulability,[17–19] and measure of isotropy[18,20,21] are also used by researchers. Second, since Jacobian-based indices require configuration data associated with the points of interest in the robot's workspace, inverse kinematics is an inseparable step in the kinematic optimization. However, it is well known that solving the inverse kinematics problem is computationally challenging.[22] Additionally, nonuniqueness and nonexistence[23,24] of solutions add to the complexity of solving the inverse kinematics.

In addition to the significance of formulating the optimization problem appropriately, the choice of optimization technique is also equally important. For kinematic optimization problems, non-gradient-based techniques have been commonly used due to their capability in searching for the global minimum in complex problems with the prevalence of local minima. Zhang and Nelson have used Genetic Algorithm (GA) to design a spherical serial mechanism with maximum workspace and manipulability and minimum size.[25] Similarly, Zhang and Gao have conducted a multi-objective optimization using particle swarm algorithm to search for the optimal dexterous stiffness and reachable workspace in a bioinspired parallel manipulator.[26] Similar efforts have been made to optimize orthotic devices to maximize manipulator dexterity[5,27,28] and achieve the required workspace while preserving compactness.[29–31] In addition to the common challenges associated with the problem formulation and solving kinematic optimization, optimizing exoskeletons require imposing additional constraints to consider the possibility of physical interference between the device and the body.[32–34] Integrating the possibility of physical collision in the analysis is not a trivial task, since it requires careful consideration of the positioning of the human body relative to the device. Lo and Xie have used GA to optimize the design of a 4R-mechanism shoulder exoskeleton to maximize the achievable shoulder workspace and Jacobian condition number. The constrained optimization is formulated by defining a forbidden region for the structure when inverse kinematics has no solution, or the structure approaches singular configurations, or physical interference could occur.[35] Similarly, an optimization problem to minimize the global conditioning number of a parallel ankle rehabilitation robot is formulated and solved in Ref. [36] using a variation of GA.

The goal of this paper is to optimize the kinematic design of CLEVERarm, an eight Degree-of-Freedom (DoF) upper-limb rehabilitation exoskeleton, to achieve maximum dexterity and a compact device. All the design variables in the problem of interest are located in the device shoulder. Therefore, the objective of optimization is finding a compact shoulder design with maximum dexterity in the desired workspace defined as a set of upper-arm configurations associated with daily activities. This can be mathematically formulated as a constrained multi-objective optimization problem, where the volume and dexterity constitute the 2×1 vector of cost functions. This paper proposes a novel index, named adjusted condition number, to quantify kinematic dexterity. For evaluation of the proposed dexterity cost in each iteration of the optimization, the inverse kinematics problem is solved by a recently developed method named geometric equivalence for anthropomorphic arms (GEAA). This method significantly reduces the computational load of the cost function evaluation and eliminates the need for solving the set of coupled nonlinear equations arising in the solution of inverse kinematics. Workspace requirements and device/body collision considerations, as well as the conditions imposed by the geometry of the CLEVERarm links, define the constraints for the optimization problem. The formulated multi-objective problem is solved in two phases. First, pareto front is found using a variant of the Non-dominated Sorting Genetic Algorithm (NSGA-II). Alternatively, weighted sum approach is used to identify a set of semi-optimal points (in the sense of pareto optimality). The scalar cost function formed by weighted sum method is optimized using a hybrid strategy comprised of GAs followed by Sequential Quadratic Programming (SQP) for localized search. From a practical point of view, dimensional tolerances and inaccuracies are unavoidable in manufacturing process of robot body parts. In this paper, post-optimization sensitivity analysis is used to quantify the effect of such dimensional discrepancies on the cost function. Among the set of points found by solving the multi-objective optimization problem, the design with lowest sensitivity is chosen.

In summary, the main contributions of this paper can be summarized as follows: proposing a novel index for quantifying kinematic dexterity, using a computationally efficient method for solving inverse kinematics problem, and using sensitivity analysis in choosing the final design. These novel techniques have been utilized to achieve a dexterous and compact upper-limb exoskeleton. This manuscript is organized as follows. Section 2 describes the kinematic design and specifications

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press

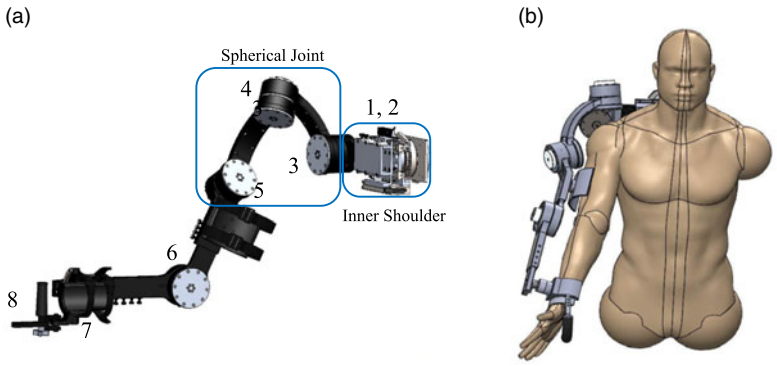


Fig. 1. (a) Kinematic model of CLEVERarm, (b) design concept on human model.

of CLEVERarm, while the mathematical formulation of the optimization problem is outlined in Section 3. The GEAA inverse kinematics technique along with the other methods used to solve the optimization problem are briefly reviewed in Section 4. Finally, the optimization results are presented in Section 5, and the main contributions of the paper are summarized in Section 6.

2. CLEVERarm

CLEVERarm is an eight DoF upper-limb rehabilitation exoskeleton supporting the motions of shoulder, elbow, and wrist.[37] Figure 1(a) shows the kinematic model of CLEVERarm in a right-arm configuration, while Fig. 1(b) demonstrates the system on a manikin. As shown in this figure, human arm can be connected to the device in three interface points, two semi-circular cuffs to be strapped around the upper-arm and wrist, and a handle to be gripped by the user.

Kinematic compatibility between an orthotic device and the human body is necessary to ensure that the natural motion of the body is not restricted and the interaction is comfortable. This can be achieved by considering adequate DoF to allow replicating human movements as close as possible.[38] Designing a mechanism that matches the kinematics of the body and supports the complex motions of the shoulder joint has been the main emphases in the kinematic design of CLEVERarm. Three revolute joints denoted by joints 3, 4, and 5 in Fig. 1(a) are used in the kinematic structure of CLEVERarm to model the spherical shoulder joint. This group of joints is connected to the device base through one revolute and one prismatic joint which support the motion of human inner shoulder (joints 1 and 2). According to the literature, the glenohumeral (GH) center's movement is coupled to the elevation of the arm.[39–41] This movement is modeled in Refs. [40, 42] with two rotations (φ_{ed} and φ_{pr}) and one translation (Δd_{SG}) of the vector connecting GH to a fixed frame (d_{SG}), where the rotations represent the elevation/depression and pronation/supination of this vector on the frontal and transverse plane of the body, respectively. Equation (1) represents the model mathematically where γ denotes the elevation of the arm, and d_0 is the length of the vector connecting GH center to origin when $\gamma = 0$.[42] The value of d_0 in this model can be updated for different individuals based on their measured body sizes. Figure 2 shows the inner shoulder and the parameters used in the inner shoulder model.

$$\varphi_{\text{ed}} = 1.49 \times 10^{-9}\gamma^5 - 4.28\times10^{-7}\gamma^4 + 1.44 \times 10^{-5}\gamma^3 + 5.2\times10^{-3}\gamma^2 - 0.1357\gamma + 0.7078 \quad (1)$$

$$\varphi_{\text{pr}} = 1.82\gamma^3 - 8.073\gamma^2 - 3.99\gamma$$

$$d_{\text{SG}} - d_0 = \left(-1.6\times10^{-5}\gamma^2 + 3\times10^{-4}\gamma\right) d_0$$

Using this model, joints 1 and 2 position the point of intersection of the three shoulder joint axes on the body frontal plane such that the exoskeleton shoulder center matches with the human GH joint center. This allows better alignment for shoulder joint center compared to the self-aligning or linkage-based mechanisms.[43,44] In addition to the shoulder girdle motion and GH rotations in CLEVERarm, flexion and extension of the elbow joint are realized by a one DoF revolute joint denoted by joint 6 in Fig. 1(a), whereas the curved rail connected to the forearm and the revolute joint hinged to the

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press

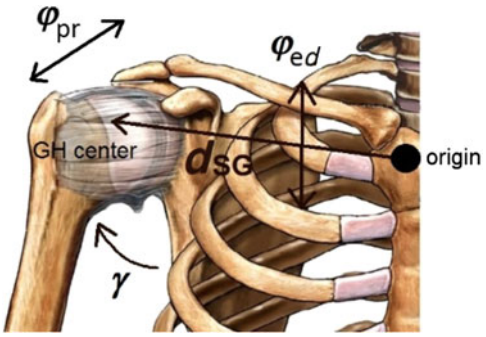


Fig. 2. Inner shoulder schematics.

Table I. Design variable bounds.

	v_1	v_2	v_3	v_4	v_5	v_6
Lower bound	0	0.1 m	0.1 m	10°	10°	0
Upper bound	45°	0.3 m	0.3 m	150°	150°	90°

carrier of the curved rail support the wrist pronation and flexion, respectively. The described wrist degrees-of-freedom are denoted by joints 7 and 8 in Fig. 1(a).

3. Optimization Problem Formulation

To maximize the usability of the device in multiple settings such as clinics and home and to improve portability, a compact device is desired.[45] On the other hand, due to the close interaction of the exoskeleton with the human body and its expected functionality of closely mimicking human upper-limb, designing a dexterous exoskeleton is required. Therefore, the goal of this paper is to optimize the kinematic design of CLEVERarm to achieve minimized volume and maximized dexterity in the desired workspace of the robot. As it will be discussed in the following section, all the design variables are gathered in the device shoulder. Resultantly, the optimization problem is focused on improving the kinematic indices of the shoulder design. This section provides a detailed discussion on the mathematical formulation of the optimization problem.

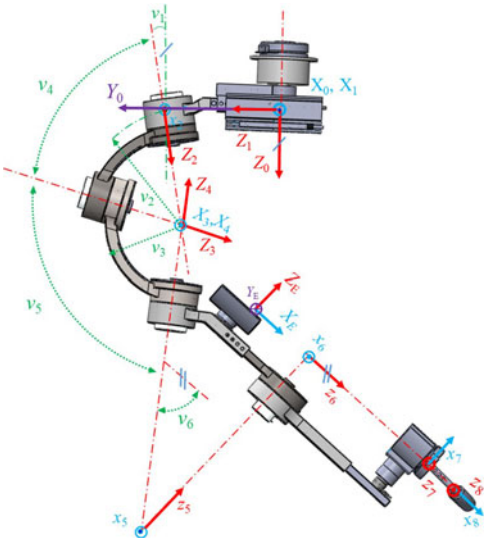
3.1. Design variables

There can be numerous design variations for the chosen model of the spherical serial linkage in CLEVERarm shoulder. These variations can be parameterized and utilized as design variables in the optimization problem. As the first step, Denavit Hartenberg's (DH) convention is used to model the kinematics of the system. The design variables used for the formulation of optimization problem can be represented in terms of the DH variables. Figure 3 shows the assignment of DH coordinate frames and the definition of design variables based on the DH model of CLEVERarm.

Figure 3 shows the design variables on the full kinematic chain of CLEVERarm. While nine coordinate frames are needed to model the full kinematics of CLEVERarm, only the first six are used in the formulation of optimization problem, since all the design variables are located in the device shoulder. The (X_E, Y_E, Z_E) coordinate system attached to the upper-arm is considered to be the last coordinate frame in the optimization problem and is referenced as the "*end-effector frame*" throughout the paper.

Design variables outlined in Fig. 3 cannot have arbitrary values and are bounded. While bounds on the value of v_2 and v_3 are imposed by manufacturing constraints, geometric considerations necessitate other bounds as well. For example, it is clearly visible in Fig. 3 that the value of v_6 cannot exceed 90° since the center of exoskeleton shoulder (point of intersection of the three axes of rotation for the shoulder joints) would not coincide with the GH joint center of the person wearing it. By considering the geometric aspects of the design and expected functionalities from the system, a set of bounds on the designed variables are identified as shown in Table I.

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press



v_1: Tilt about shoulder yaw axis z_1

v_2: Radius of link 3

v_3: Radius of link 4

v_4: Angular length of link 3

v_5: Angular length of link 4

v_6: Angle between the axis of joint 5 and upper-arm

Fig. 3. Kinematic model of CLEVERarm and the defined design variables.

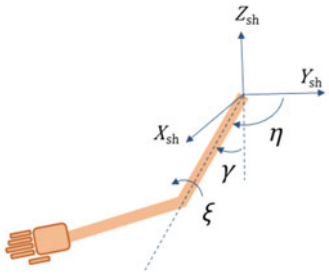


Fig. 4. Angles determining orientation of the upper-arm.

Table II. Shoulder ADL range of motion.

	Horizontal abduction/ adduction ($\eta°$)	**Flexion/extension ($\gamma°$)**	**Internal/external rotation ($\zeta°$)**
Lower bound	0	0	0
Upper bound	135	110	110

3.2. Desired system workspace

CLEVERarm is intended to provide automated therapy for patients suffering from neurological disorders such as stroke, by providing training for Activities of Daily Living (ADL) such as reaching, self-feeding, bathing, combing, etc. Therefore, the robot should be able to cover the workspace required for these ADL motions. A set of upper-arm configurations associated with the aforementioned motions are considered as the desired workspace for CLEVERarm. Motions of the shoulder are categorized as horizontal abduction/adduction, flexion/extension, and internal/external rotation of the upper-arm, and are denoted by three shoulder angles of azimuth (η), elevation (γ), and torsion (ζ), respectively. These shoulder angles, depicted in Fig. 4, can uniquely specify the orientation of the upper-arm.

Based on the Range of Motion (RoM) data for healthy subjects and the RoM required for ADL reported in the literature,[5] the values shown in Table II are chosen as the desired workspace for shoulder motions.

Due to the high computation cost of calculating inverse kinematics, considering a continuous desired workspace is not practical. Resultantly, a discrete subset of upper-arm configurations for ADL is selected as the desired workspace. This subset is created by partitioning the range of motion

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press

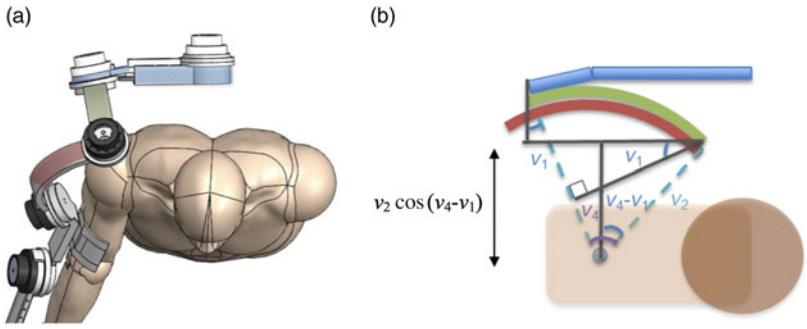


Fig. 5. Geometrical relations for collision prevention between the exoskeleton and the subject's head.

of each DoF of the shoulder in Table II into five equally spaced intervals. The resulting 125 points are assumed to be representatives of the continuous workspace of ADL.[5]

3.3. Constraints

Singularity is a kinematic phenomenon where any infinitesimal motion of the end-effector translates to very large angular velocities in the joint level. Therefore, it is important to avoid singular configurations by placing them outside the desired workspace of the robotic system during the design phase. CLEVERarm does not have any isolated singular configurations within its workspace.[37] However, there are two gimbal lock configurations in the shoulder linkage (i.e., θ_4= 0° and θ_4= 180°) which limit the achievable horizontal abduction and adduction of the upper-arm. By inspecting the geometry of the system, it can be shown that the maximum and minimum achievable horizontal abduction and adduction are related to the design variables as shown in Eq. (2).

$$\begin{aligned} v_1 + v_4 + v_5 + v_6 - \frac{\pi}{2} &= \text{maximum horizontal abduction/adduction} \\ v_1 - v_4 + v_5 + v_6 - \frac{\pi}{2} &= \text{minimum horizontal abduction/adduction} \end{aligned} \tag{2}$$

Using the values in Table II, two inequality constraints can be acquired to ensure that the two gimbal lock configurations occur outside the desired workspace:

$$\begin{aligned} v_1 + v_4 + v_5 + v_6 &\geq 225° \\ v_1 - v_4 + v_5 + v_6 &\leq 90° \end{aligned} \tag{3}$$

The last set of constraints are meant to avoid physical interference between the device and the patient body. Considering the geometry of the system, a collision between the exoskeleton and the user's head can occur during the maximal horizontal abduction where the shoulder links are folded on each other as shown in Fig. 5. To reduce the possibility of such collisions, a safe distance between the device and the head should be preserved at all times. The geometrical relationships between the design variables and the point of the device closest to the body are illustrated in Fig. 5. Through trigonometric calculations, it can be shown that the clearance distance is $v_2 \cos(v_4 - v_1)$.

This distance could also be considered as a cost to be maximized. However, from a practical perspective, having a minimum clearance between the device and the body is sufficient for achieving a safe interaction. Additionally, including a third cost function in the problem formulation will further complicate the optimization problem and selection of the optimal solution among the pareto points. Therefore, it is reasonable to consider a safe distance between the device and the body, and formulate this constraint as an inequality constraint as follows:

$$v_2 \cos(v_4 - v_1) > 0.15 \tag{4}$$

The safe distance of 0.15 meters is selected based on the average chest depth data of healthy subjects in the 90 percentile of population plus a clearance gap of approximately 3 centimeters.[46]

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press

3.4. Optimization cost

Quantifying the kinematic dexterity of a robotic system is not a trivial task in general. According to the literature, this measure can be quantified through various Jacobian-based indices such as manipulability[47] and condition number.[15,48] A high manipulability index is desirable since it shows the capability of a robot's end-effector in achieving higher velocity in more directions.[15] However, in some cases, the designer might not necessarily care about all the components of the end-effector velocity, or no velocity could be generated in certain directions of the end-effector frame. For example, in the case of upper-limb exoskeletons, no linear velocity can be achieved along the axis of upper-arm since the shoulder is a spherical joint incapable of supporting radial velocities. Similarly, the upper-arm link cannot acquire any angular velocities in the Y_E and Z_E directions shown in Fig. 3. Therefore, it is useful to generalize the kinematic dexterity concept to the cases where a subset of end-effector velocities is considered. In other words, formulating the kinematic performance index based on the desirable velocity directions could be a better representation of the dexterity of the system. This is very important from the optimization perspective, since the defined index could be specific to a particular application, and this would ensure that no trade-off is made for achieving a subgoal that is not important to the designer.

Directions that are achievable by human arm can be identified more easily in the local coordinate system of the end-effector. Therefore, a modified Jacobian operator is defined which can map the space of the joint angular velocities to the space of the end-effector linear and angular velocities expressed in the end-effector frame. Let ${R_b}^E$ define the rotation matrix between the base and the end-effector coordinate systems. Using this rotation matrix, the modified Jacobian matrix is defined as:

$$J_m = \begin{bmatrix} R_b^E J_v \\ R_b^E J_\omega \end{bmatrix} \tag{5}$$

To continue, a subset of the modified Jacobian matrix is selected by eliminating the rows corresponding to the directions along which no velocity can be achieved (linear velocity in the X_E direction, as well as the angular velocity along the Y_E and Z_E), all expressed in the end-effector frame:

$$\bar{J}_m = E\, J_m \tag{6}$$

where E is an algebraic operator defined as:

$$E = \begin{bmatrix} 0_{3\times1} & I_{3\times3} & 0_{3\times2} \end{bmatrix} \tag{7}$$

This paper proposes a novel dexterity index, named *adjusted condition number* based on $\bar{J}_m$. The proposed index is defined as follows:

$$\eta_a(q) = \left(\sigma_{\max}\left(\bar{J}_m \bar{J}_m^{\,T}\right) / \sigma_{\min}\left(\bar{J}_m \bar{J}_m^{\,T}\right)\right) \tag{8}$$

As Eq. (8) shows explicitly, the adjusted condition number is a configuration-dependent index. Therefore, to improve the kinematic performance over the entire desired workspace, the cumulative effect of all the desired configurations needs to be considered. It is also important to note that the E matrix defined in Eq. (7) is specific to the problem studied in this paper, that is, optimizing the shoulder kinematic design for an upper-limb exoskeleton. For other problems, other choices of E matrix might be desired.

3.5. Mathematical problem statement

Based on the defined constraints and cost function, the multi-objective optimization problem is formulated as follows:

$$\begin{aligned} &\min \boldsymbol{J}\,(\boldsymbol{v}, \boldsymbol{p}) \\ &\text{s.t. } \boldsymbol{g}\,(\boldsymbol{v}) \leq 0 \end{aligned}$$

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press

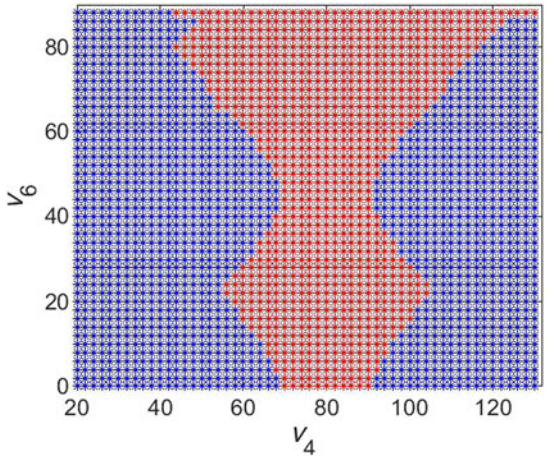


Fig. 6. A two-dimensional cross section of the space of feasible designs.

where

$$\boldsymbol{J} = \begin{bmatrix} J_1 \\ J_2 \end{bmatrix} = \begin{bmatrix} -\max\limits_{\overline{q}} \ \eta_a\,(\overline{q}, v) \\ \frac{2}{3}\pi\left(v_2^3\left(1-\frac{cosv_4}{2}\right)+v_3^3\left(1-\frac{cosv_5}{2}\right)\right) \end{bmatrix} \tag{9}$$

where $\overline{q}$ denotes the set of 125 exoskeleton configurations associated with the workspace of the desired upper-arm directions, v is the vector of the design variables defined in Fig. 3, and the volume cost function (J_2) is defined as the total volume of the two sphere sectors created by two circular links. The inequality and side constraints considered for the optimization problem are defined according to Sections 3.1 and 3.2 as follows:

$$\boldsymbol{g}\,(\boldsymbol{v}) = \begin{bmatrix} -(v_1+v_4+v_5+v_6)+225 \\ v_1-v_4+v_5+v_6-90 \\ v_2\cos(v_4-v_1)-0.15 \\ -v_2+v_3+0.03 \end{bmatrix} \leq \boldsymbol{0},\ \ l_l\,(i) < v_i < u_l\,(i)$$

$$l_l = \begin{bmatrix} 0 & 0.1 & 0.1 & 10 & 10 & 0 \end{bmatrix}',\ u_l = \begin{bmatrix} 45 & 0.3 & 0.3 & 150 & 150 & 90 \end{bmatrix}' \tag{10}$$

where the fourth inequality constraint is a practical consideration to account for the thickness of the circular links to enable folding of the two links on each other. It is important to note that the domain of optimization described by Eq. (10) can contain infeasible design vectors for which the solution of inverse kinematics problem does not exist for at least one member of $\overline{q}$. However, it is not possible to include this constraint explicitly in Eq. (10) since the set of feasible designs within the optimization domain cannot be determined analytically. To address this issue, infeasible points for which inverse kinematics has no solution are assigned a large cost value while solving the formulated optimization problem. Figure 6 shows a two-dimensional cross section of the space of feasible designs where variables four and six can acquire values within the range shown in Eq. (10) and the other four variables are held fixed at $(v_1, v_2, v_3, v_5) = (10, 0.2, 0.15, 100)$. In this figure, red and blue regions show the feasible and infeasible design concepts, respectively.

It can be seen that the space of feasible designs is not convex and can shrink and expand in different parts of the optimization domain. The irregular shape of this set adds to the complexity of the optimization problem by making the analysis very sensitive to the starting point. Section 5 discusses the techniques used in this research to overcome this challenge.

4. Methods

This section reviews the methods used for solving the formulated optimization problem. A combination of different techniques from the literature is used to find the optimal solution with computational efficiency.

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press

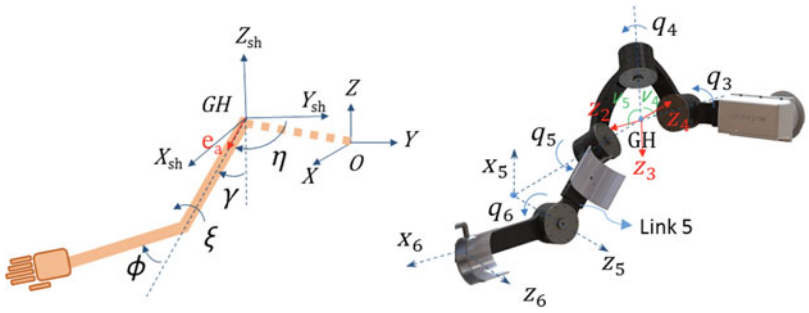


Fig. 7. Configuration space of the exoskeleton and the equivalent system related through transformation T.

4.1. Geometric equivalence for anthropomorphic arms

GEAA is a geometric method for constructing a direct mapping between the configuration spaces of two kinematically equivalent systems. Finding such a transformation via GEAA requires solving a set of algebraic equations and consequently is computationally efficient. This method can be very useful for solving the inverse kinematics problem with low computational cost for geometrically complex systems. This can be achieved by identifying a kinematically equivalent system for which the inverse problem can be easily solved and mapping the solution to the configuration space of the original system.[46] The simplified system considered here for solving the inverse kinematics of CLEVERarm is a stick figure arm model which is shown in Fig. 6, along with the kinematic model of the exoskeleton, and the transformation in a conceptual level.[49]

Solving the inverse kinematics problem is trivial for the equivalent system due to its simple geometry. On the contrary, the complexity of the forward kinematics equations for CLEVERarm makes finding explicit inverse solutions almost impossible. The kinematic link between the two systems is the direction of the upper-arm (i.e., $\boldsymbol{e}_a$ is parallel with the direction of link 5, as shown in Fig. 7). This kinematic constraint along with the geometry of the exoskeleton links can be used to identify the loci of rotation axes for the exoskeleton joints, which can be used to infer the direction of rotation axes (Z_i). Configuration of the exoskeleton can be reconstructed by knowing the direction of rotation axes following the DH convention. Let $\boldsymbol{e}_{a0}$ and $\boldsymbol{n}$ denote the direction of the upper-arm in the base frame and the normal vector of the plane formed by the upper- and lower-arm, respectively. The axis of rotation of joints four and five are the quantities needed for reconstructing the configuration of the exoskeleton (i.e., pose needed in the optimization problem) and can be identified by solving the following set of algebraic equations:

$$\begin{cases} <z_4, n> = 0 \\ <z_4, e_{a0}> = -sin\, v_6 \\ \|z_4\| = 1 \end{cases}, \quad \begin{cases} <z_2, z_3> = \cos v_4 \\ <z_4, z_3> = \cos v_5 \\ \|z_3\| = 1 \end{cases} \tag{11}$$

4.2. Genetic algorithm

GA is a heuristic search technique used to solve complex and nonlinear problems. This method uses probabilistic transition rules to guide the search for global minima. GA's ability to search a population of points in parallel makes the technique a computationally efficient method to find global minimum and prevent getting trapped in local minima.[50] However, it cannot guarantee convergence to the global optimum. GA also enables optimizing systems with continuous as well as discrete, and integer design variables.[51]

An individual design candidate in GA is represented with chromosomes, a fixed-length string of genes. Each chromosome represents a solution, often using strings of ones and zeros. Applying genetic operators including selection, crossover, insertion, and mutation evolves a generation and generates a new generation of the population. A GA algorithm in general works as follows: First a population is initialized and objective function is evaluated for the initial population. Then based on the selection criteria, individuals are selected for mating. Typically, selection is the most important and most computationally expensive step of a GA. Through the crossover operator, selected parents mate and produce children. After mutating, based on the chosen insertion strategy, children are inserted into the population. GA's algorithm is an iteratively cyclic method which continues until

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press

Table III. Design of experiments.

Experiment no.	Factor 1	Factor 2	Factor 3	Factor 4	Factor 5	Factor 6	J_1	J_2
1	10	0.2	0.15	70	100	30	99.90	0.022
2	10	0.16	0.15	70	100	30	97.53	0.023
3	10	0.2	0.18	70	100	30	93.17	0.027
4	10	0.2	0.15	75	100	30	33.45	0.023
5	10	0.2	0.15	70	105	30	33.78	0.022
6	10	0.2	0.15	70	100	20	27.49	0.022

the stopping criteria is satisfied. Average performance of individuals in a population is expected to increase, as good individuals are preserved and breed and less fit individuals die out. However, performance of the algorithm and convergence rate highly depend on tuning of parameters such as selection rate, insertion strategy, mutation rate, crossover fraction, and population size.

4.3. Sequential quadratic programming

SQP is one of the most effective methods for solving general constrained optimization problems.[52,53] This method is a gradient-based optimum search method which uses gradient information to update the search direction. Similar to other gradient-based methods, SQP is fast; however, it is susceptible to local optima.

The SQP optimization method iteratively approximates the nonlinear programming problem by a quadratic programming subproblem at each iteration, where from the current point, a line direction search is conducted to reach to a new point which reduces the cost function of the original nonlinear optimization function. The achieved point is then used in the new iteration, and this process continues until a termination criterion is satisfied. The challenge in this method is designing the quadratic programming subproblem such that it yields a good step for the nonlinear optimization problem.

5. Results

Initial population for starting the optimization algorithms is selected among the optimal points found through the Design of Experiments (DOEs) along with other random points. Details of the DOE are provided below.

5.1. Design of experiments

DOE is used to better understand the effect of various design variables on the objective functions and to identify good initial guesses for gradient-based and heuristic methods. Table III shows the DOE using the parameter study method where each row shows one experiment which differs from the baseline design, highlighted in blue, in only one variable. Discretization technique is used due to the fact that design variables are continuous. While several levels for each variable were used to study the behavior of the cost functions locally around the baseline design, only two levels for each variable are shown in Table III for the sake of brevity.

5.2. Multi-objective optimization

The multi-objective optimization problem formulated in Section 3.5 is solved using NSGA-II to find the pareto front. Additionally, weighted sum approach is used to calculate a set of semi-optimal points. While solutions achieved from the weighted some approach might not be pareto in the sense of the defined cost functions, they might have lower sensitivity, thus making them favorable. This is important from a practical point of view, since manufacturing parts with the exact dimensions is not guaranteed. Therefore, it is important to consider the sensitivity of the optimal points in addition to their cost value.

The weighted sum cost function is defined as a convex linear combination of the two cost functions as follows:

$$J = \beta W_1 J_1 + (1-\beta)\, W_2 J_2,\ \beta \epsilon (0,\ 1)$$

where W_1 and W_2 are weights used to ensure that the relative magnitude of the two cost functions are close. Based on the absolute value of the cost function in the problem of interest, the following weight values were chosen: $\boldsymbol{W}_1 = 1$ and $\boldsymbol{W}_2 = 100$. The scalar cost function is then optimized with a

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press

Table IV. GA tuning parameters.

Population size	Crossover fraction	Elite count	Max generations	Initial penalty	Function tolerance
300	0.65	0.05 × population size	50	100	10^{-8}

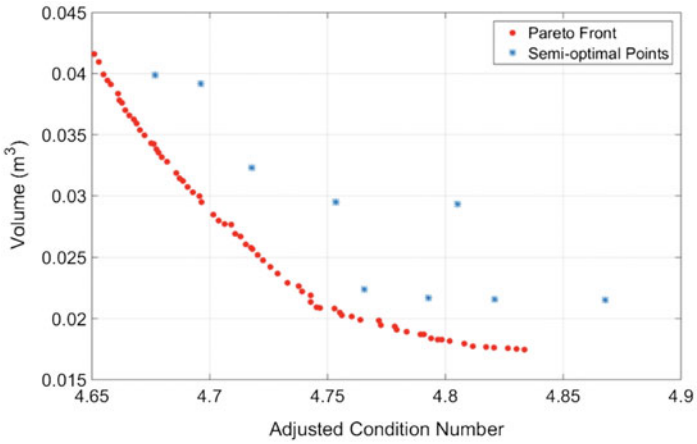


Fig. 8. Pareto and semi-optimal points.

hybrid approach comprised of GA and SQP. The set of parameters presented in Table IV were used for GA analysis.

An important observation about the GA is the convergence issues due to the occurrence of infeasible design vectors for which the inverse kinematics problem does not have a solution at least for one of the desired orientations for the arm. To overcome this issue, a quarter of the initial population for GA were chosen from the known feasible points (DOE) to ensure that the later generations could benefit from the known feasible points in the initial population until better solutions are found. This was significantly helpful in improving the convergence of GA.

For each value of β, GA optimization was repeated several times and the best solutions were retained for finer local optimization using SQP algorithm. Optimality conditions were checked for the output points of SQP to verify that they are indeed a minimum point. Additionally, to understand whether the problem needs scaling, hessian of the cost function was evaluated using the central finite difference approximation.

The pareto front for the multi-objective cost function was derived using MATLAB's implementation of the NSGA-II. Pareto and the semi-optimal points found from the weighted sum approach constitute the final set of points among which the optimal solution needs to be chosen. Figure 8 shows these points where red dots denote the pareto front and the blue dots represent a subset of semi-optimal points.

Choosing a solution for a multi-objective optimization problem is often a subjective task. In this study, sensitivity of the points with respect to the design variables is chosen as a factor to help in the decision process. For this purpose, the normalized sensitivity $\overline{S}_{v_i}$, defined below, is used to find the sensitivity with respect to design variables:

$$\overline{S}_{v_i} = \frac{v_i^*}{J(v^*)} \nabla J_i(v^*)$$

The sum of the absolute values of sensitivity indices was calculated for the optimal and semi-optimal points found through the optimization process. Table V shows the top three design concepts and their corresponding sensitivities.

The highlighted design concept with the lowest cumulative sensitivity was chosen as the optimal design concept. The achieved design is implemented in CAD environment and is verified to be consistent with the natural motions of the human body, confirming that the formulation and final solution of the optimization are consistent with the physics of the problem. Figures 9 and 10 show the CAD model of the system before and after optimization for the same pose of the human arm. As it is

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press

Table V. Cumulative sensitivity of selected points.

x_1	x_2	x_3	x_4	x_5	x_6	**Cumulative sensitivity**
19.81	0.26	0.23	68.33	70.68	45.56	1.38
18.67	0.27	0.24	71.96	79.90	41.55	1.41
16.53	0.28	0.25	63.87	56.29	58.56	1.54

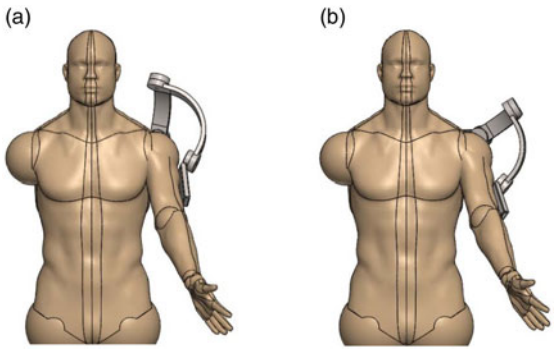


Fig. 9. Exoskeleton before (a) and after (b) optimization – front view.

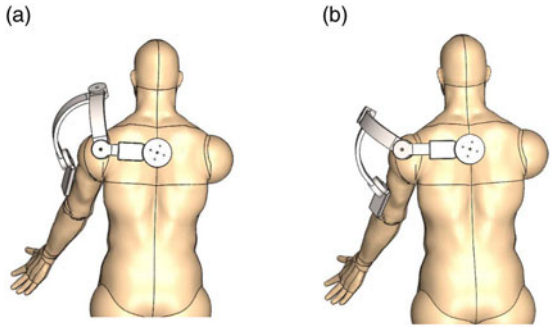


Fig. 10. Exoskeleton before (a) and after (b) optimization – back view.

seen in the figure, since the kinematic parameters of the robot have changed, the optimized and the original exoskeletons acquire different configurations for the same pose of the manikin.

This optimized design is currently used in the development of the second generation of Texas A&M upper-limb exoskeleton, and the results will be published as soon as the development phase is completed.

6. Conclusion

This paper presents the procedure of optimizing the kinematic design of CLEVERarm to achieve high dexterity while preserving compactness. The problem is formulated as a constrained multi-objective optimization using a novel dexterity index. The resulting multi-objective optimization problem is solved using NSGA-II as well as the weighted sum approach. Considering the high computational cost of this optimization problem, a novel method for solving inverse kinematics problem is used. Finally, the optimal solution is chosen by studying the sensitivity of the achieved points with respect to the changes in design variables. The optimization results found through the analysis in this paper are used in the development of the second generation of Texas A&M upper-limb exoskeleton.

References

1. P. Maciejasz, J. Eschweiler, K. Gerlach-Hahn, A. Jansen-Troy and S. Leonhardt, "A survey on robotic devices for upper limb rehabilitation," *J. Neuroeng. Rehab.* **11**, 3 (2014).
2. T. Nef, M. Guidali and R. Riener, "ARMin III-arm therapy exoskeleton with an ergonomic shoulder actuation," *Appl. Bion. Biomechan.* **6**, 127–142 (2009).

https://doi.org/10.1017/S0263574719001085 Published online by Cambridge University Press

3. B. R. Brewer, S. K. McDowell and L. C. Worthen-Chaudhari, "Poststroke upper extremity rehabilitation: A review of robotic systems and clinical results," *Top. Stroke Rehab.* **14**, 22–44 (2007).
4. H. S. Lo and S. Q. Xie, "Exoskeleton robots for upper-limb rehabilitation: State of the art and future prospects," *Med. Eng. Phys.* **34**, 261–268 (2012).
5. J. C. Perry, J. Rosen and S. Burns, "Upper-limb powered exoskeleton design," *IEEE/ASME Trans. Mechatron.* **12**, 408–417 (2007).
6. Y. Mao and S. K. Agrawal, "Design of a cable-driven arm exoskeleton (CAREX) for neural rehabilitation," *IEEE Trans. Robot.* **28**, 922–931 (2012).
7. P. Agarwal, M. S. Narayanan, L. Lee, F. Mendel and V. N. Krovi, "Simulation-based design of exoskeletons using musculoskeletal analysis," *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Montreal, Canada, 1357–1364 (2010).
8. R. Kurtz and V. Hayward, "Multiple-goal kinematic optimization of a parallel spherical mechanism with actuator redundancy," *IEEE Trans. Robot. Automat.* **8**, 644–651 (1992).
9. M. Stock and K. Miller, "Optimal kinematic design of spatial parallel manipulators: Application to linear delta robot," *J. Mech. Design* **125**, 292–301 (2003).
10. C. Gosselin and J. Angeles, "A global performance index for the kinematic optimization of robotic manipulators," *J. Mech. Design* **113**, 220–226 (1991).
11. P. Zhang, Z. Yao and Z. Du, "Global performance index system for kinematic optimization of robotic mechanism," *J. Mech. Design* **136**, 031001 (2014).
12. T. Huang, M. Li, Z. Li, D. G. Chetwynd and D. J. Whitehouse, "Optimal kinematic design of 2-DOF parallel manipulators with well-shaped workspace bounded by a specified conditioning index," *IEEE Trans. Robot. Automat.* **20**, 538–543 (2004).
13. M. J. Lum, J. Rosen, M. N. Sinanan and B. Hannaford, "Kinematic Optimization of a Spherical Mechanism for a Minimally Invasive Surgical Robot," *Proceedings of the IEEE International Conference on Robotics and Automation*, New Orleans, Louisiana, USA (2004) pp. 829–834.
14. A. Faraz and S. Payandeh, "A Robotic Case Study: Optimal Design for Laparoscopic Positioning Stands," *Proceedings of the IEEE International Conference on Robotics and Automation*, Albuquerque, New Mexico, USA (1997) pp. 1553–1560.
15. J.-P. Merlet, "Jacobian, manipulability, condition number, and accuracy of parallel robots," *J. Mech. Design* **128**, 199–206 (2006).
16. K. S. Hong and J. G. Kim, "Manipulability analysis of a parallel machine tool: Application to optimal link length design," *J. Robot. Syst.* **17**, 403–415 (2000).
17. J.-O. Kim and P. K. Khosla, "Design of Space Shuttle Tile Servicing Robot: An Application of Task Based Kinematic Design," *Proceedings of the IEEE International Conference on Robotics and Automation*, Atlanta, Georgia, USA (1993) pp. 867–874.
18. J.-O. Kim and K. Khosla, "Dexterity Measures for Design and Control of Manipulators," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Osaka, Japan (1991) pp. 758–763.
19. R. G. Roberts, S. A. Siddiqui and A. A. Maciejewski, "Designing Equally Fault-Tolerant Configurations for Kinematically Redundant Manipulators," *Proceedings of the 41st Southeastern Symposium on System Theory*, Tullahoma, Tennessee, USA (2009) pp. 335–339.
20. J.-O. Kim and P. Khosla, "A multi-population genetic algorithm and its application to design of manipulators," *Proceedings of the IEEE International Conference of Intelligent Robots and Systems*, Raleigh, North Carolina, USA (1992) pp. 279–286.
21. S. Khatami and F. Sassani, "Isotropic Design Optimization of Robotic Manipulators Using a Genetic Algorithm Method," *Proceedings of the IEEE International Symposium on Intelligent Control*, Vancouver, Canada (2002) pp. 562–567.
22. D. Tolani, A. Goswami and N. I. Badler, "Real-time inverse kinematics techniques for anthropomorphic limbs," *Graph. Models* **62**, 353–388 (2000).
23. B. Durmuş, H. Temurtaş and A. Gün, "An Inverse Kinematics Solution Using Particle Swarm Optimization," *6th International Advanced Technologies Symposium IATS*, Elazig, Turkey (2011).
24. B. Daya, S. Khawandi and M. Akoum, "Applying neural network architecture for inverse kinematics problem in robotics," *J. Software Eng. Appl.* **3**, 230 (2010).
25. X. Zhang and C. A. Nelson, "Multiple-criteria kinematic optimization for the design of spherical serial mechanisms using genetic algorithms," *J. Mech. Design* **133**, 011005 (2011).
26. D. Zhang and Z. Gao, "Forward kinematics, performance analysis, and multi-objective optimization of a bio-inspired parallel manipulator," *Robot. Comput.-Integrated Manuf.* **28**, 484–492 (2012).
27. X. Cui, W. Chen, X. Jin and S. K. Agrawal, "Design of a 7-DOF cable-driven arm exoskeleton (CAREX-7) and a controller for dexterous motion training or assistance," *IEEE/ASME Trans. Mechatron.* **22**, 161–172 (2017).
28. J. Iqbal, N. Tsagarakis and D. Caldwell, "Design Optimization of a Hand Exoskeleton Rehabilitation Device," *Proceedings of RSS Workshop on Understanding the Human Hand for Advancing Robotic Manipulation*, Seattle, Washington, USA (2009) pp. 44–45.
29. S. Nakagawara, H. Kajimoto, N. Kawakami, S. Tachi, and I. Kawabuchi, "An Encounter-Type Multi-Fingered Master Hand Using Circuitous Joints," *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, (2005) pp. 2667–2672.

30. A. Sledd and M. K. O'Malley, "Performance Enhancement of a Haptic Arm Exoskeleton," *Proceedings of the 14th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Arlington, Virginia, USA (2006) pp. 375–381.
31. H. Bian, Z. Chen, H. Wang and T. Zhao, "Mechanical Design of EFW Exo II: A Hybrid Exoskeleton for Elbow-Forearm-Wrist Rehabilitation," *Proceedings of the IEEE International Conference on Rehabilitation Robotics*, London, England (2017) pp. 689–694.
32. T.-M. Wu, S.-Y. Wang and D.-Z. Chen, "Design of an exoskeleton for strengthening the upper limb muscle for overextension injury prevention," *Mech. Mach. Theory* **46**, 1825–1839 (2011).
33. H. S. Lo and S. S. Xie, "An Upper Limb Exoskeleton with an Optimized 4R Spherical Wrist Mechanism for the Shoulder Joint," *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Besancon, France (2014) pp. 269–274.
34. R. A. R. C. Gopura, K. Kiguchi and Y. Li, "SUEFUL-7: A 7DOF Upper-Limb Exoskeleton Robot with Muscle-Model-Oriented EMG-Based Control," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, USA (2009) pp. 1126–1131.
35. H. S. Lo and S. Xie, "Optimization and analysis of a redundant 4R spherical wrist mechanism for a shoulder exoskeleton," *Robotica* **32**, 1191–1211 (2014).
36. P. K. Jamwal, S. Xie and K. C. Aw, "Kinematic design optimization of a parallel ankle rehabilitation robot using modified genetic algorithm," *Robot. Autonom. Syst.* **57**, 1018–1027 (2009).
37. A. Zeiaee, R. Soltani-Zarrin, R. Langari and R. Tafreshi, "Design and Kinematic Analysis of a Novel Upper Limb Exoskeleton for Rehabilitation of Stroke Patients," *Proceedings of the 2017 International Conference on Rehabilitation Robotics*, London, England (2017) pp. 759–764.
38. J. H. X. T. J. He, "Design and evaluation of the RUPERT wearable upper extremity exoskeleton robot for clinical and in-home therapies," *IEEE Trans. Syst. Man Cybernet.* **46**, 926–935 (2016).
39. E. K. Chadwick, D. Blana, R. F. Kirsch and A. J. Van Den Bogert, "Real-time simulation of three-dimensional shoulder girdle and arm dynamics," *IEEE Trans. Biomed. Eng.* **61**, 1947–1956 (2014).
40. N. Klopčar and J. Lenarčič, "Bilateral and unilateral shoulder girdle kinematics during humeral elevation," *Clin. Biomech.* **21**, S20–S26 (2006).
41. V. T. Inman and L. C. Abbott, "Observations on the function of the shoulder joint," *J. Bone Joint Surg. Am.* **26**, 1–30, (1944).
42. R. Soltani-Zarrin, A. Zeiaee, R. Langari and N. Robson, "Reference Path Generation For Upper-Arm Exoskeletons Considering Scapulohumeral Rhythms," *Proceedings of the 2017 International Conference on Rehabilitation Robotics*, London, England (2017) pp. 753–758.
43. K. Kiguchi, K. Iwami, M. Yasuda, K. Watanabe and T. Fukuda, "An exoskeletal robot for human shoulder joint motion assist," *IEEE/ASME Trans. Mechatron.* **8**, 125–135 (2003).
44. Y. Ren, S. H. Kang, H. Park, Y. Wu and L. Zhang, "Developing a multi-joint upper limb exoskeleton robot for diagnosis, therapy, and outcome evaluation in neurorehabilitation," *IEEE Trans. Neural Syst. Rehab. Eng.* **21**, 490–499 (2013).
45. R. Riener, T. Nef and G. Colombo, "Robot-aided neurorehabilitation of the upper extremities," *Med. Biol. Eng. Comput.* **43**, 2–10 (2005).
46. R. Soltani-Zarrin, A. Zeiaee, R. Langari, "Human-like path generation in upper-limb exoskeleton", *arXiv preprint*.
47. J. Gunasekara, R. Gopura, T. Jayawardena, and G. Mann, "Dexterity Measure of Upper Limb Exoskeleton Robot with Improved Redundancy," *Proceedings of the IEEE International Conference on Industrial and Information Systems*, Sri Lanka (2013) pp. 548–553.
48. S. Kucuk, "Kinematics, Singularity and Dexterity Analysis of Planar Parallel Manipulators based on DH Method," **In**: Robot Manipulators New Achievements, Aleksandar Lazinica and Hiroyuki Kawai (Ed.), ISBN: 978-953-307-090-2, InTech.
49. R. Soltani-Zarrin, A. Zeiaee, R. Langari and R. Tafreshi, "A Computational Approach for Human-like Motion Generation in Upper Limb Exoskeletons Supporting Scapulohumeral Rhythms," *Proceedings of the International Symposium on Wearable Robotics and Rehabilitation*, Houston, Texas, USA (2017) pp. 1–2.
50. C. M. Anderson-Cook, Practical Genetic Algorithms, Taylor & Francis (2005).
51. D. E. Goldberg, *Genetic Algorithms* Pearson Education India, (2006).
52. J. Nocedal and S. J. Wright, "Sequential quadratic programming," Springer, (2006).
53. P. T. Boggs and J. W. Tolle, "Sequential quadratic programming," *Acta Num.* **4**, 1–51 (1995).