Arabian Journal for Science and Engineering (2023) 48:3351–3366
https://doi.org/10.1007/s13369-022-07065-0

A Reliable Algorithm for Obtaining All-Inclusive Inverse Kinematics' Solutions and Redundancy Resolution of Continuum Robots

Ammar Merrad[1,2] · Ammar Amouri[1] · Abdelhakim Cherfia[1,2] · Selman Djeffal[3]

Received: 17 February 2021 / Accepted: 14 June 2022 / Published online: 10 August 2022

Abstract
This paper presents a reliable algorithm for solving the inverse kinematic (IK) problem of continuum robots. Admittedly, there are an infinite number of configurations to get the robot's end point (EP) pose. Therefore, the proposed algorithm introduces a new solution that helps narrow the infinite number of the IK solutions down to a finite one by taking into account the first $(n-1)$ sections' sets of reachable EP poses, which are selected and created one-by-one from the first to $(n-1)$th section, in which the Particle Swarm Optimization method is adopted to randomly pick up the first IK solution. However, a specific criterion has been accorded to select the appropriate IK solution (i.e., redundancy resolution) among the existing varieties that perfectly fits within the targeted trajectory. Furthermore, numerical experiments are performed on a Cable-Driven Continuum Robot for tracking a trajectory in a free and confined environment, pointing up on IK, multiple IK solutions and redundancy resolution. The obtained results demonstrated that the proposed algorithm is computationally accurate and efficient in obtaining all-inclusive IK solutions and redundancy resolution of continuum robots in comparison to previously proposed approaches. It is noteworthy to say that the proposed algorithm can be applied to any kind of continuum robots and most importantly regardless of its number of sections, which makes the developed algorithm a sovereign remedy for the IK problem of these robots.

Keywords Continuum robot · Inverse kinematic · Redundancy resolution · Particle swarm optimization · Three-dimensional simulation

1 Introduction

Nowadays, the design, modeling, and control research around continuum robots has received a lot of attention due to their appealing characteristics such as low inertia, high dexterity, flexibility, and, most importantly, their safety near humans when compared to robots with rigid links [1–3]. Additionally, their continuous structure and inherent compliance enable them to work in extremely congested environments and provide them with the ability to dexterously avoid obstacles. These features make them more suitable for a variety of applications including: medical and surgical applications [4, 5], inspection and manipulation [6–8] and so on.

A continuum robot has a complex structure that can be either soft or hard consisting of single section or a number of sections serially connected. The entire structure of continuum robot can be assimilated as piecewise curves, which can be either real or virtual, depending on the type of the robot. Currently, a variety of designs and prototypes of continuum robots have been developed with planar and spatial functionality such as Cable-Driven Continuum Robot (CDCR) [9], Tendon-Driven Continuum Robot (TDCR) [10], pneumatic continuum manipulator as Bionic Handling Assistant (BHA) [11] and concentric-tube continuum robot [12]. Nevertheless, due to their inherent complex behavior, they require extensive modeling and development in order to extract their full physical potential for practical applications.

From a modeling point of view, the Forward Kinematic (FK) problem of continuum robots is relatively easy to solve. As a result, many research projects have been carried out

✉ Ammar Amouri
ammar_amouri@yahoo.fr

[1] Department of Mechanical Engineering, Frères Mentouri Constantine 1 University, Constantine, Algeria

[2] Laboratory of Mechanics, Frères Mentouri Constantine 1 University, Constantine, Algeria

[3] Department of Mechanical Engineering, University of L'Arbi Ben M'Hidi, Oum El Bouaghi, Algeria

to solve it using different methods and theories based on some simplifications and hypotheses, such as Elliptic Integral Approach [13], Constant Curvature Kinematic Approach [14], Modal Approach [15] and Pseudo-Rigid Body Representation [16]. On the other hand, the Inverse Kinematic (IK) problem of continuum robots remains a major challenge due to highly coupled nonlinear equations of the forward kinematic. In fact, it is hard to solve this problem in a purely analytical way except in specific circumstances where the robot meets particular conditions, such as in [17–20]. In [17], the IK of multi-section continuum robot was analytically solved by considering it as a rigid-linkage system (i.e., each rigid-link connects the section's EPs) and assuming that the EP coordinates of all links are known and without considering their orientations. In [18], the IK of continuum structures was solved by using parametric curves approximated by mode shape functions assigned to restricted shapes and motions. The authors in [19] proposed a solution to the IK for inextensible bending segments under specific segment arrangement, and similarly, for a specified EP pose in [20]. In [21], the IK problem of continuum robots has been solved using Pythagorean Hodograph curves, in which the robot's shape is firstly reconstructed, and then the IKs are derived. However, these approaches do not tackle the redundancy problem; hence they cannot be used to find a general solution to the redundancy problem.

Alternatively, some researchers have resorted to use machine learning tools to solve the IKs of continuum robots. In [22], a neural network approach via a distal supervised learning architecture has been used to solve the IK problem of Compact Bionic Handling Assistant (CBHA) manipulator. In [23], a hybrid approach was used to solve the IK problem of the CBHA manipulator in which the robot was considered as a series of vertebrae, where a flexible link joined two successive vertebrae. In the same context, a goal babbling approach was used to solve IK problem of Bionic Handling Assistant (BHA) manipulator [24]. However, only one IK solution was provided among the redundancy varieties in these approaches, and which, in turn, cannot be used in general cases.

Additionally, other researchers have used numerical and optimization methods to solve the IKs problem and the redundancy resolution of continuum robots. In [25, 26], the IK problem for a planar and spatial CDCR has been formulated as an optimization issue dealing with a constant curvature assumption and using PSO method. In [27], a Sequential Quadratic Program (SQP)-based approach is used to approximate the IK of CBHA manipulator. These approaches, however, can only provide one optimal or non-optimal IK solution. Similarly, numerical methods based on the Jacobian matrix have been implemented to solve the IK problem of continuum robots [11, 28, 29], in which the FK equations are differentiated to find a linear transformation between actuator velocities and robot pose velocities and yet, these methods have only solved redundancy in one way, where changing the resolution criterion when computing the IK solution is difficult.

Based on the previously proposed approaches for solving the IK problem of continuum robots, we noticed that the majority of researchers have not thoroughly addressed the redundancy resolution, which is undeniably important. In this paper, a reliable algorithm is established and developed to address the shortcomings of previously proposed algorithms for solving the IK problem and redundancy resolution of continuum robots. It is a two-sub-algorithm: the first sub-algorithm collects and forms a tree of the reachable EP poses of the first $(n-1)$ sections for each robot's EP pose. The second sub-algorithm selects the optimal posture of the robot among the redundant varieties provided by the first one.

The following points summarize the advantages of the proposed algorithms: (1) unlike previous works [11, 28, 29], in which the ease of the model's derivation is strongly related to the number of robot's sections (i.e., increasing number of robot's sections has a negative impact on the process of the IK problem's resolution). On the other hand, these methods suffer generally from several shortcomings such as high calculation cost, the presence of local minima and instabilities close to the singularities; while the proposed algorithm has no links with the robot's degree of redundancy and its characteristics, and does not require computing the Jacobian and even the gradient of the cost function; thus the common problem of singularities can be easily avoided. Besides, the re-generation of the initial population on the PSO method has been introduced to address the occurrence probability of local minima. Thus, the complexity of the continuum robot's mechanism is no longer an obstacle to processing multi-sections which was previously considered as an unavoidable problem. (2) unlike [22–24] that require a massive database just to provide access to a significant number of possible solutions, the proposed algorithm is simple, complete and does not require any data to be applied, except for the treatment of the problem's inputs. Essentially, it is a database builder, which allows researchers to easily build their own databases. (3) with the proposed algorithm, all IK solutions are obtained analytically and thus easily accessible, in contrast to works [25–27], where all these solutions are inaccessible and rely entirely on runtime.

The structure of this paper is as follows: the next section describes the structure of the considered continuum robot, namely CDCR, and presents briefly the kinematics models of its single section. Section 3 focuses on the development of the proposed algorithm and details the pseudo-codes of the first sub-algorithm. After that, the second sub-algorithm and its pseudo-codes related to the resolution of redundancy are presented in Sect. 4. Section 5 presents a series of numerical experiments on a class of continuum robots, namely cable-Driven Continuum Robot (CDCR). Finally, Sect. 6 outlines the main conclusions and future perspectives.

2 Kinematics of a Single Continuum Robot Section

This section provides a summary on the IK solution of a continuum robot with a single section. Commonly, this class of continuum robot consists of n sections; each section is actuated by three cables, namely CDCR, as depicted in Fig. 1.

From a structural point of view, the movable part of a single CDCR section is composed of a flexible backbone on which several equidistant spacer disks are rigidly mounted, each two disks form a unit. The whole section is actuated by three driving cables placed at 120 degree from each other. This arrangement of cables leads to performing two DoFs of bending and orientation motions.

In order to describe the robot's configuration, references coordinate systems $O_k X_k Y_k Z_k$, where $k = 1, 2, \ldots, n$ is the section number, are attached at the end of each section. Additionally, one fixed base reference coordinate system $O_0 X_0 Y_0 Z_0$ (see Fig. 1a, b). Furthermore, we introduce local references coordinate systems $o_{j,k} x_{j,k} y_{j,k} z_{j,k}$, where $j = 1, 2, \ldots, m$ is the unit number, attached to the ends of each intermediate unit (see Fig. 1c).

As mentioned above, the objective is to calculate the actuator cable lengths as a function of $\{X_k, Y_k, Z_k\}$, the Cartesian coordinates of the section's endpoint k, defined in its base reference coordinate system, $O_{k-1} X_{k-1} Y_{k-1} Z_{k-1}$, supposed fixed in this case. Various methods can be applied to accomplish this purpose, but most of them require quite complicated and mathematically lengthy steps [14, 30]. In this section, the IK of a single CDCR section is derived by assuming that each of its units is assimilated to a parallel 3UPS-1PU type robot [23, 26]. This model is a crucial part of the algorithm involved in this paper, which will be developed and explained through numerical experiments in the following sections.

For simplicity reasons, the bending shape of each section k is assumed to be a perfect arc of circle and variable in time. Thus, the length vector of the active prismatic actuator for each UPS ($l_{i,j,k} = A_{i,j,k} B_{i,j,k}$), where $i = 1, 2, 3$ is the actuator cable number, which corresponds to the cable length variable vector in our case (see Fig. 1), it can be calculated using the Cartesian coordinates of the endpoint of each section k by the following relationship

$$l_{i,j,k} = o_{j-1,k} o_{j,k} + \mathbf{R_P} \cdot o_{j,k} A_{i,j,k} - o_{j-1,k} B_{i,j,k}, \quad (1)$$

where $\mathbf{R_P}$ is the matrix defines the orientation of the moving platform relative to fixed platform unit (j, k), it can be expressed by.

$$\mathbf{R_P} = \mathbf{rot}\left(x_{j-1,k}, \Theta_{j,k}\right) \cdot \mathbf{rot}\left(y_{j-1,k}, \Psi_{j,k}\right), \quad (2)$$

where **rot** represents the rotation transformation, and $\Theta_{j,k}$ and $\Psi_{j,k}$ indicate pitch and roll angles, which correspond to the rotation around $y_{j,k}$-axis and $x_{j,k}$-axis, respectively, can be obtained by.

$$\Theta_{j,k} = \tan^{-1}\left(\frac{-\mathrm{r}_{31}}{\mathrm{r}_{11}}\right), \quad (3)$$

$$\Psi_{j,k} = \tan^{-1}\left(\frac{-\mathrm{r}_{23}}{\mathrm{r}_{22}}\right), \quad (4)$$

where $\mathrm{r}_{\mathfrak{mn}}$ is the element $(\mathfrak{m}, \mathfrak{n})$ of the matrix $\mathbf{R}$ given by [14]

$$\mathbf{R} = \mathbf{rot}\left(z_{j-1,k}, \varphi_k\right) \cdot \mathbf{rot}\left(y_{j-1,k}, \theta_{j,k}\right) \cdot \mathbf{rot}\left(z_{j-1,k}, -\varphi_k\right), \quad (5)$$

where $\theta_{j,k}$ and φ_k indicate bending and orientation angles. They can be obtained as a function of Cartesian coordinates as follows:

$$\theta_{j,k} = \frac{2}{m}\tan^{-1}\left(\frac{\sqrt{X_k^2 + Y_k^2}}{Z_k}\right), \quad (6)$$

$$\varphi_k = \tan^{-1}\left(\frac{Y_k}{X_k}\right), \quad (7)$$

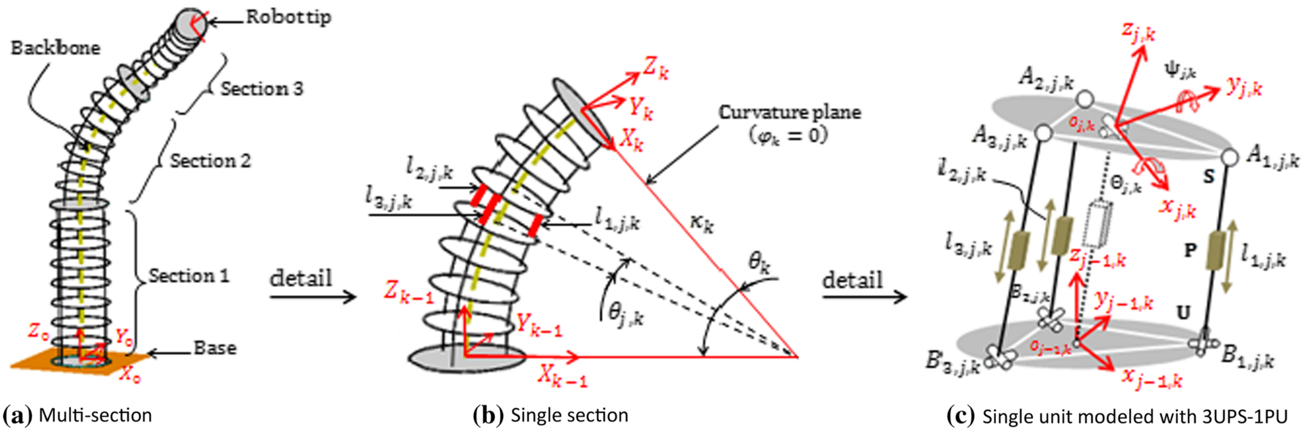
Finally, the total actuator cables' length of the section k (i.e., the actuator space vector), which corresponds to the sum of vectors norm of Eq. (1), can be written as follows:

$$l_{i,k} = \sum_{j=1}^{m} \sqrt{\left(l_{i,j,k}{}^{T} \cdot l_{i,j,k}\right)}, \quad (8)$$

For further use, the endpoint coordinates of each section relative to its base reference as a function of a bending and orientation angle are needed, which presents the forward kinematic model of a single continuum robot section. According to [14], this model can be expressed as

$$\begin{cases} X_k = \frac{\ell_k}{\theta_k}(1 - \cos(\theta_k))\cos(\varphi_k), \\ Y_k = \frac{\ell_k}{\theta_k}(1 - \cos(\theta_k))\sin(\varphi_k), \\ Z_k = \frac{\ell_k}{\theta_k}\sin(\theta_k), \end{cases} \quad (9)$$

in which $\theta_k = \sum_{j=1}^{n}\left(\theta_{j,k}\right)$, and ℓ_k is the length of the continuum robot section.



(a) Multi-section **(b)** Single section **(c)** Single unit modeled with 3UPS-1PU

Fig. 1 Multi-section continuum robot schematics. Panel **a** shows a class of a continuum robot with three sections, namely CDCR. Panel **b** shows a single section of CDCR with three driving cables. It consists of a number of units. Panel **c** shows a single unit modeled as a parallel 3UPS-1PU type robot. The Prismatic actuators for the kinematic chain UPS are considered active, whereas the central one of PU is considered as a passive

3 IKs of Continuum Robots with All-Inclusive IK Solutions

As mentioned before, for a given operational configuration of the robot's EP, there is an infinite number of IK solutions that the robot can reach and this will generally raise two problems: obtaining all-inclusive IK solutions for a given robot's EP pose and the redundancy resolution (i.e., selecting an appropriate IK solution that satisfies a specific criterion for the robot, such as minimum variation of the actuator space vector, traveled distance, obstacle avoidance and so forth). Certainly, from a computational time point of view, the ideal is to use the analytical methods since they provide all IK solutions for a given robot's EP pose. However, these methods generally remain unable to solve the IK problem of this class of redundant robots, which is undoubtedly a non-trivial problem to be solved.

As for the numerical methods, although they work independently of the robot's redundancy, they still suffer from several shortcomings, such as high computational costs, running times, existence of local minima and instabilities close to singularities [11, 28, 29]. Moreover, redundancy resolution criteria for these methods are generally applied to derive a specific IK solution. However, to figure out this commonly encountered problem, a reliable algorithm has been developed, in this section, to provide all-inclusive IK solutions, for a given robot's EP pose.

Since the proposed algorithm is strongly based on an optimization method, many optimization algorithms can be used, such as Genetic Algorithm (GA) [26], Particle Swarm Optimization (PSO) [31], Simulated Annealing (SA) [32], and so on. Although these algorithms have been successfully for solving a variety of real-world problems, each of them has some limitations. However, since PSO method is superior in terms of model complexity, accuracy, iteration and program simplicity in finding the optimal solution compared to other algorithms [33], it will be used to pick up the first randomly IK solution for the proposed algorithm.

In the following, we will introduce the preliminaries used in developing the proposed algorithm, namely PSO method and the cost function, boundary conditions and the physical limitations as well as the proposed algorithm itself.

3.1 PSO Method and the Cost Function

Standard PSO method is a population-based search algorithm, developed by Kennedy and Eberhart in 1995 [31]. This method is inspired by the social behavior of swarming animals such as flocks of birds, schools of fish, and even human social behavior. Generally, PSO method is quite simple and has a fast convergence with few parameters tuned compared to other optimization algorithms. However, it has not been shown to be effective in some challenging problems, especially the type of problems where the algorithm has to be applied in small increments in order to keep the algorithm's stability. Therefore, to increase the PSO performance, Shi and Eberhart [34] added another factor called the inertia weight concept, which will be used in this work.

In the PSO algorithm, the potential solution to the optimization problem is called a particle and the population of solutions is called a swarm. Each particle, $\mathcal{P}_{\mathcal{D}}$ is characterized by a position vector $p_{\mathcal{P},\mathcal{D}}(\xi)$ and a velocity vector $v_{\mathcal{P},\mathcal{D}}(\xi)$. The quality of its position is determined by the value of the appropriate cost function. The particle memorizes the best reached position so far and it is denoted by $pbest_{\mathcal{P},\mathcal{D}}$. The best position found by all the particles in the swarm is denoted by $gbest_{\mathcal{D}}$. In the search space $\mathcal{D}$, the velocities and positions of particles keep changing according to the following equations:

$$v_{p,\mathcal{D}}(\xi+1) = \omega v_{p,\mathcal{D}}(\xi) + c_1\lambda_1(pbest_{p_{\mathcal{D}}} - p_{p,\mathcal{D}}(\xi) + c_2\lambda_2\left(gbest_{\mathcal{D}} - p_{p,\mathcal{D}}(\xi)\right), \tag{10}$$

$$p_{p,\mathcal{D}}(\xi+1) = p_{p,\mathcal{D}}(\xi) + v_{p,\mathcal{D}}(\xi+1), \tag{11}$$

where $p_{p,\mathcal{D}}(\xi)$ and $v_{p,\mathcal{D}}(\xi)$ is the current position and velocity of the particle$p_{\mathcal{D}}$, respectively; ξ is the iteration number; $\mathcal{D}$ is the dimension of the particle vector (here, $\mathcal{D} = 3$, i.e., along $X-$, $Y-$ and Z-axis); ω is the inertia weight rate; c_1 and c_2 are two positive constants, λ_1 and λ_2 are two random numbers; $pbest_{p_{\mathcal{D}}}$ and $gbest_{\mathcal{D}}$ represent personal best and swarm best position, respectively.

Algorithm 1. Pseudo-code of PSO method

```
Inputs: PSO parameters, $\xi$ _max.
Initialize randomly the particles and their velocities in the search space $\mathcal{D}$
$\xi \leftarrow 0$
while the stopping criteria is not fulfilled do
    for $p_D \leftarrow 1$ to swarm_size do
        Evaluate the cost function
    end for
    Select $gbest_{\mathcal{D}}$
    for $p_D \leftarrow 1$ to swarm_size do
        Select $pbest_{p,D}$
        Move the particle ($p_D$) according to Eq. 10 and 11
    end for
    $\xi \leftarrow \xi + 1$
    if $\xi == \xi$_max then
        Initialize randomly the particles and their velocities in the search space $\mathcal{D}$
        $\xi \leftarrow 0$
    end if
end while
Output : $p_{p,\mathcal{D}}$
```

Regardless of the time 's implementation issue, a common barrier to most optimization algorithms is the early convergence of solutions to local minima, which makes the algorithms inefficient in practical applications. However, to avoid the algorithm being trapped in local optima, the re-generating of population technique has been added to PSO method [35, 36], see algorithm 1.

As stated in the paper's title, the first goal is to provide all-inclusive IK solutions of continuum robot. Since the proposed algorithm is based on the PSO method, the appropriate cost function which enables the goal to be achieved can be expressed as.

$$C_f = \|\mathbb{X}_{desired} - \mathbb{X}_{generated}\|, \tag{12}$$

where $\mathbb{X}_{desired}$ and $\mathbb{X}_{generated}$ are the desired and generated position vector of the endpoint of the robot. The generated position vector is given by the fourth column of the homogeneous matrix defining the FKM of the continuum robot (for detailed derivation of the FKM refers to ref [37].).

3.2 Parameters Selection of PSO Method

Naturally, the performance and efficiency of the PSO method strongly depends on the choice of its parameters, namely constant parameters c_1 and c_2, inertia weight ω, swarm size, number of iterations. Therefore, these parameters should be carefully determined before starting the optimization process. As the parameters c_1 and c_2 give the tendency of the swarm to converge to local or global best solution, the two parameters are fixed after performing a pre-test of the algorithm by increasing their values with a step of 0.1 in a range [0.1, 3.0] and the fitness value was calculated. As a result, it was found that the value 1.2 is the best for both parameters. For the determination of inertial weight parameter, the method developed in [38] was used for reducing the inertia weight linearly from 0.9 to 0.4 throughout the implementation in order to explore the research space in the early stages and then exploit the optimum solution at the later ones. On the other hand, since the size of the swarm and the number of iterations are related to the complexity of the problem, that is, larger ratios obviously mean better performance with the selected swarm and the number of iterations and therefore, following multiple tests of the algorithm, and it was found that the values 15 and 45 are the lowest values of the convergence of the algorithm in a reasonable computation time.

3.3 Boundary Conditions and PHYSICAL Limitations

The Boundary conditions satisfying the achievement of the central backbone curve to the reference frame's origin must be introduced to correlate the reference frame of the central backbone and the base's reference. Furthermore, another required condition is the association of the desired point, which is located in the prescribed trajectory and the previously mentioned frames. Explicitly, a relationship between the frame of the central backbone, the base, and the prescribed trajectory's specific position (point) that allows the robot to fulfill the task must be considered. Mathematically, these conditions can be expressed as follows:

$$\mathbb{X}_0 = \mathbb{X}_{base},\ \mathbb{X}_n = \mathbb{X}_{desired}, \tag{13}$$

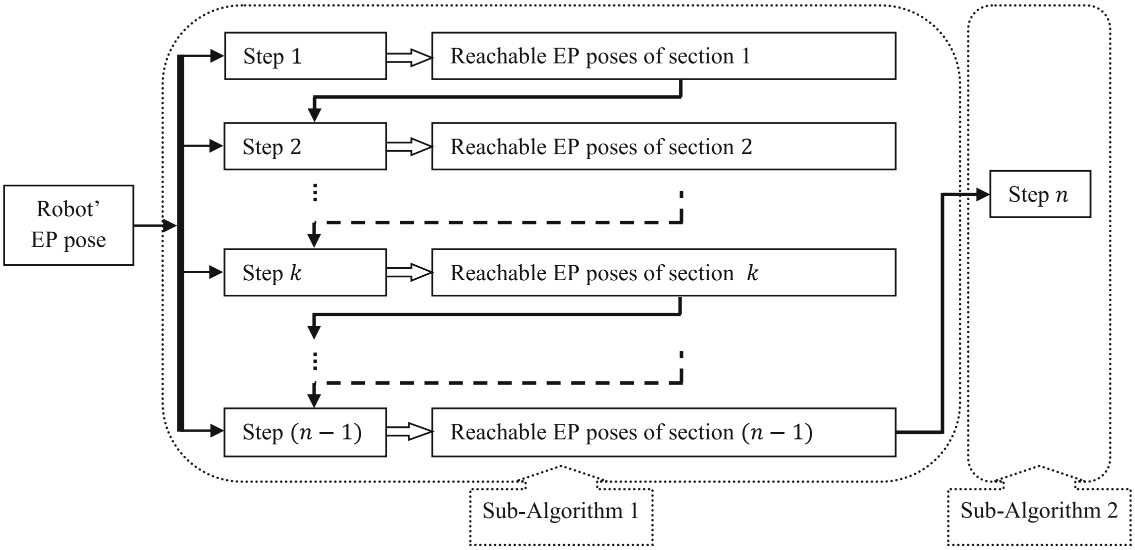

Robot'
EP pose
Step 1
Reachable EP poses of section 1
Step 2
Reachable EP poses of section 2
Step k
Reachable EP poses of section k
Step $(n-1)$
Reachable EP poses of section $(n-1)$
Step n
Sub-Algorithm 1
Sub-Algorithm 2

Fig. 2 Block diagram of algorithm providing with all-inclusive inverse kinematics' solutions and the resolution of its redundancy

where $\mathbb{X}_{base}$ and $\mathbb{X}_{desired}$ are the position of origin of the reference frame associated with the fixed base and the end reference frame of the robot, respectively.

Limitations such as those related to actuator variables, which are typically presented in rigid manipulators, can be considered to confirm the robot's proper operation, especially in the case of a continuum robot with a large number of sections that may overlap. These limitations can be expressed as.

$$l_k^{min} \leq l_{i,k} \leq l_k^{max}, \tag{14}$$

3.4 Proposed Algorithm Step-By-Step

From a computational time point of view, the analytical approaches have been proved to be the ideal way to solve the inverse kinematics of continuum robots. Methodically, it is extremely difficult to solve their inverse kinematics, especially for those who have more than two sections. Solving the inverse kinematics of continuum robots generally leads to an infinite number of inverse kinematics solutions. Therefore, the alternative way is the development and the improvement of numerical and optimization approaches. Thus, this subsection is dedicated to establishing and developing a reliably efficient algorithm capable of providing with all-inclusive IKs solutions of any multi-section continuum robots and; then, its redundancy can be easily resolved. The problem of the infinite number of IK solutions is transformed into a finite number by considering a set of the reachable positions of the EP poses of the first $(n-1)$ sections, which are defined and created one-by-one from the first section to the $(n-1)$-thsection. It should be noted, here, that any numerically selected solution results in small errors.

Figure 2 summarizes the proposed algorithm approach which can be divided into two sub-algorithms with $((n-1)+1)$ steps. Sub-algorithm 1, with $(n-1)$ steps, focuses on defining and creating the reachable positions of the EP poses of the first $(n-1)$. In this sub-algorithm, the robot redundancy is decreased one-by-one by deleting the previous section. While sub-algorithm 2, with one step, is dedicated to the resolution of the redundancy, which will be presented in the next section. This sub-algorithm works with the outputs of sub-algorithm 1 and the robot EP pose.

However, it is known that the difficulty of deriving the inverse kinematic for any continuum robot increases as the number of its sections increases. So, in this paper, the proposed algorithm is applied to a three section continuum robot; hence, sub-algorithm 1 will have two steps. Therefore, according to block diagram depicted in Fig. 2, the first step gives the reachable EP poses of the first section, and the second one for determining the reachable EP poses of the second section. The procedure of each step can be achieved, respectively, by the following pseudo-code.

Sub-Algorithm 1: Step 1. Calculation of reachable EP poses of section 1

```
1:  Inputs: Geometric data of the robot, a desired robot's EP pose ($P_3$)
2:  Find a random IK solution using PSO method        /* $i.e. P_k(\theta_k,\ \varphi_k)$  $with\ \ k = 1, 2$ */
3:  /* Find a specific IK solution, denoted $P_{1s}$ , i.e. let's find $\theta_{1,max}$ in the bending plane $\varphi_1$ (lines: 4-13) */
4:  $J \leftarrow true$
5:  while ($J == true$) do
6:          $P_{1s} \leftarrow P_1$
7:          Fixed $\varphi_1$ and  $\theta_1 \leftarrow \theta_1 + \Delta\tilde{\theta}_1$
8:          Calculate the new EP pose $P_1(X_1, Y_1, Z_1)$ using Eq. 9
9:          Fixed $P_1$ and check the existence of $P_2$ using PSO method
10:         if $P_2$ does not exist then
11:             $J \leftarrow false$
12:         end if
13: end while
14: Calculate coefficients of equation of the plane $(aX + bY + cZ + d = 0)$ passing through the point $P_{1s}$ and perpendicular
    to $\overrightarrow{O_0P_3}$.
15: The reachable EP poses of the first section are all points that verify the inequality: $aX_1 + bY_1 + cZ_1 + d \geq 0$
16: /* Selection of a set of EP poses  $[MAT\_P_1]$, (Lines: 17-25) */
17: $[MAT\_P_1] \leftarrow [\,]$
18: for  $\theta_1 \leftarrow 0 : \Delta\theta_1 :\ \theta_{1max}$ do
19:     for $\varphi_1\ \leftarrow 0 : \Delta\varphi_1 : 2\pi$ do
20:         Calculate the coordinate of the endpoint $P_1(X_1, Y_1, Z_1)$ using Eq. 9
21:         if $(aX_1 + bY_1 + cZ_1 + d \geq 0\ )$ then
22:             $[MAT\_P_1] \leftarrow [MAT\_P_1; P_1]$
23:         end if
24:     end for
25: end for
```

Step $k = 2$. Calculation of reachable EP poses of section 2

```
26: for $q \leftarrow 1 :\ nb\_P_1$ do                   /* $nb\_P_1$:  is the selected number of EP poses of  $[MAT\_P_1]$*/
27:     $[MAT\_P_{2,q}] \longleftarrow [\,]$
28:     Fixed $P_1$ and find a point $P_2$ using PSO method
29:     Calculate the coefficients of equation of the plane $(aX + bY + cZ + d = 0)$ passing through the point $P_2$ and
        perpendicular to $\overrightarrow{O_1P_3}^{(*)}$
30:     The reachable EP poses of the second section are all points that verify the equality: $\mathrm{aX}_2 + \mathrm{bY}_2 + \mathrm{cZ}_2 + \mathrm{d} = 0$
31:     /* selection of a set of EP poses $[MAT\_P_{2,q}]$  */
32:     for  $\theta_2 \leftarrow 0 : \Delta\theta_2 :\ \theta_{2max}$  do
33:         for  $\varphi_2\ \leftarrow 0 : \Delta\varphi_2 : 2\pi$  do
34:             Calculate the coordinate of the endpoint $P_2(X_2, Y_2, Z_2)$ using Eq. 9
35:             if  $|aX_2 + bY_2 + cZ_2 + d| \leq \varsigma)$ then      /* $\varsigma$ is a given precision criterion */
36:                 $[MAT\_P_{2,q}] \leftarrow [MAT\_P_{2,q}; P_2]$
37:             end if
38:         end for
39:     end for
40: Output: $[MAT_{P_1}]_{(nb\_P_1 \times 3)}$, $[MAT\_P_{2,q}]_{(nb\_P_2 \times 3)}$      /* Return $[MAT\_P_2]$  according to each point of  $(MAT\_P_1)$*/
41: end for
```

To summarize, sub-algorithm 1 collects and forms a tree of the reachable EP poses of the first $(n-1)$ sections for each robot's EP pose, while sub-algorithm 2 selects the optimal posture of the robot based on a specific criterion from among the redundant varieties provided by sub-algorithm 1.

4 Resolution of Redundancy

In fact, any selected IK solution results in Cartesian and orientation errors; thus, the most accurate solution criterion is used in redundancy resolution process. This solution can be reliably used in various applications that do not require very high accuracy. In this context, to obtain the optimal posture of the robot in which its endpoint reaches the desired trajectory, the intermediate endpoints of the first $(n - 1)$ sections can be selected among the redundant varieties, provided by sub-algorithm 1, according to a specific criterion, such as.

- *Minimum variation of the actuator space vector*: this criterion can be expressed as

$$l = \min_{i,k} \| l_{sum} - l_{sum}^{opt} \|, \tag{15}$$

where l_{sum} and l_{sum}^{opt} are the sum of current and last continuum robot actuator space, respectively, i.e., the sum of cables length of all sections of the robot.

- *Minimum variation of traveled distances of the endpoints of the first* $(n - 1)$ *sections*: this criterion can be expressed in the general form as

$$dist = \min_{k} \sum_{\mathrm{k}=1}^{\mathrm{n}} \| P_k - P_k^{opt} \|, \tag{16}$$

where P_k and P_k^{opt} are the current and last position of endpoint of the section k, respectively.

To thoroughly describe the application of the two criteria that serve to generate the optimal posture of the robot for a given EP pose, the following pseudo-codes are presented:

Sub-Algorithm 2. Generating an optimal posture of the robot by minimizing the actuator space vector

```
Inputs: initial configuration of the robot ($P_{1,init}$ and $P_{2,init}$), desired robot's EP poses $[MAT\_P_3]_{(nb\_P_3 \times 3)}$,
        $[MAT\_P_2]_{(nb\_P_2 \times 3)}$ and $[MAT\_P_1]_{(nb\_P_1 \times 3)}$.
$[MAT\_P_1^{opt}] \leftarrow [P_{1,init}]$                  /* Creating of optimal matrix of $P_1$ */
$[MAT\_P_2^{opt}] \leftarrow [P_{2,init}]$                  /* Creating of optimal matrix of $P_2$ */
$l_{sum}^{opt} = \sum_{k=1}^{3} \sum_{i=3}^{3} l_{i,k}$
for $r \leftarrow 1 : nb\_P_3$ do
    $P_3 = [MAT\_P_3(r, :)]$
    val_min ← ∞
    for $q \leftarrow 1 : nb\_P_1$ do
        $l_{sum} \leftarrow [\,]$
        $P_1 \leftarrow [MAT\_P_1(q, :)]$
        for $u \leftarrow 1 : nb\_P_2$ do
            $P_2 \leftarrow [MAT\_P_{2,q}(u, :)]$
            $l \leftarrow \sum_{k=1}^{3} \sum_{i=3}^{3} l_{i,k}$     /* Sum of the cable lengths according to points $P_1, P_2, P_3$ using Eq. 8 */
            $l_{sum} \leftarrow [l_{sum}, l]$
        end for
        $diff\_l = |l_{sum}^{opt} - l_{sum}|$
        [val , rang] ← min($diff\_l$)
        $P_2 \leftarrow [MAT\_P_{2,q}(\mathrm{rang}, :)]$
        if val $<$ $val\_min$ then
            val ← val_min
            $P_1^{opt} \leftarrow P_1$
            $P_2^{opt} \leftarrow P_2$
            $l^{opt} = \sum_{k=1}^{3} \sum_{i=3}^{3} l_{i,k}^{opt}$
            $l_{sum}^{opt} \leftarrow l_{sum}$ (rang)
        end if
    end for
    $[MAT\_P_1^{opt}] \leftarrow [MAT\_P_1^{opt};\ P_1^{opt}]$
    $[MAT\_P_2^{opt}] \leftarrow [MAT\_P_2^{opt};\ P_2^{opt}]$
    $l_{sum}^{opt} \leftarrow l^{opt}$
end for
Outputs : $[MAT\_P_1^{opt}]$ and $[MAT\_P_2^{opt}]$
```

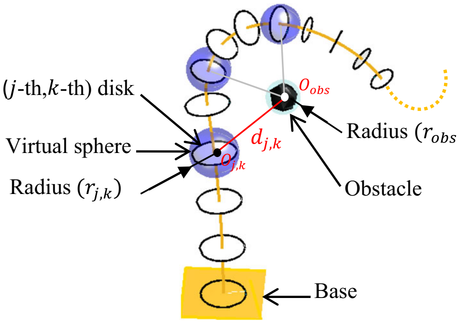


Fig. 3 Obstacle avoidance strategy

```
Sub-Algorithm 2. Generating an optimal posture of the robot by minimizing the traveled distances
Inputs: initial configuration of the robot ($P_{1,init}$ and $P_{2,init}$), desired robot's EP poses $[MAT\_P_3]_{(nb\_P_3 \times 3)}$,
        $[MAT\_P_2]_{(nb\_P_2 \times 3)}$, and $[MAT\_P_1]_{(nb\_P_1 \times 3)}$.
$[MAT\_P_1^{opt}] \leftarrow [P_{1,init}]$
$[MAT\_P_2^{opt}] \leftarrow [P_{2,init}]$
for $r \leftarrow 1 : nb\_P_3$ do
    $P_3 = [MAT\_P_3(r, :)]$
    val_min ← ∞
    for $q \leftarrow 1 : nb\_P_1$ do
        $P_1 \leftarrow [MAT\_P_1(q,:)]$
        $dist_1 \leftarrow \|P_1 - P_1^{opt}\|$            /* Calculate the Euclidean distance between the actual $P_1$ and the last $P_1^{opt}$ */
        $dist_2 \leftarrow \|(MAT\_P_{2,q}) - P_2^{opt}\|$   /* Calculate the Euclidean distance between each actual $P_2$ of the matrix
                                                               $[MAT\_P_{2,q}]$ and the last $P_2^{opt}$ */
        $dist \leftarrow dist_1 + dist_2$
        $[\mathrm{val}, \mathrm{rang}] \leftarrow \min(dist)$
        $P_2 \leftarrow [MAT\_P_{2,q}(\mathrm{rang}, :)]$
        if val $< val\_min$ then
            val ← val_min
            $P_1^{opt} \leftarrow P_1$
            $P_2^{opt} \leftarrow P_2$
        end if
    end for
    $[MAT\_P_1^{opt}] \leftarrow [MAT\_P_1^{opt}; P_1^{opt}]$
    $[MAT\_P_2^{opt}] \leftarrow [MAT\_P_2^{opt}; P_2^{opt}]$
end for
Outputs : $[MAT\_P_1^{opt}]$ and $[MAT\_P_2^{opt}]$
```

4.1 Obstacle Avoidance

When the robot is operating in a restricted environment, the inverse kinematic involves generating an optimal posture allowing the robot to avoid the obstacles and reach the target. The obstacle avoidance strategy can be performed through two steps. First, a set of appropriate EP poses of the first $(n - 1)$ sections, that the robot does not collide with the obstacles, is selected from the redundancy varieties provided by sub-algorithm 1.

A simple manner to achieve the obstacle avoidance is to place some virtual spheres such that their gravity center is placed on the backbone curve of the robot (see Fig. 3), and then; if the distance between any obstacle and these spheres are greater than a given value compared to the sum of the radii of the obstacle and that of the spheres, the IK solution is retained, otherwise the solution is rejected. The obstacle avoidance strategy can be written in the general form as.

$$d_{j,k} > r_{obs} + r_{j,k}, \tag{17}$$

Then, the optimal posture that allows the robot to avoid the obstacles and achieves the specified target can be selected among the resulting set of EP poses of the first $(n-1)$ sections in which Eq. 16 will be satisfied using a certain criterion.

5 Numerical Experiments

The aim of this research work is to find all-inclusive IK solutions that allow the robot to track a given EP pose in its workspace (sub-algorithm 1), and then to solve the redundancy problem according to a given criterion (sub-algorithm 2).

The proposed algorithm that is dedicated to a CDCR with three sections begins with finding out the dimension of the problem at step 1 in sub-algorithm 1 which is $(nb\_P_1 \times 3)$, where nb_P_1 is the number of EP poses of Sect. 1. While at step 2, the dimension of the problem is $(nb\_P_2 \times 3)$ for each point of $[MAT\_P_1]$, where $[MAT\_P_1]$ is the matrix of all Sect. 1's reachable EP poses that verify the inequality $aX_1 + bY_1 + cZ_1 + d \geq 0$, where X_1, Y_1 and Z_1 are the Cartesian coordinates of the specific IK solution denoted P_{1s}. On the other hand, the reachable EP poses of Sect. 2, denoted $[MAT\_P_2]$, are all points that verify the equality $aX_2+bY_2+cZ_2+d = 0$, for each point of $[MAT\_P_1]$, where X_2, Y_2 and Z_2 are the Cartesian coordinates of the specific point denoted P_2 (see, step 2 in sub-algorithm 1).

Regarding PSO method, the population is the set of possible reachable EP poses and the coordinates of the particle are the variables of the optimization problem, which represent in our case the specific accessible EP poses, denoted $P_{1s}(X_1, Y_1, Z_1)$ and $P_2(X_2, Y_2, Z_2)$ in sub-algorithm 1. Concerning the redundancy problem (i.e., sub-algorithm 2), the optimal posture of the robot, denoted $\left[MAT\_P_1^{opt}\right]$ and $\left[MAT\_P_2^{opt}\right]$ are selected according to a given criterion among redundancy varieties.

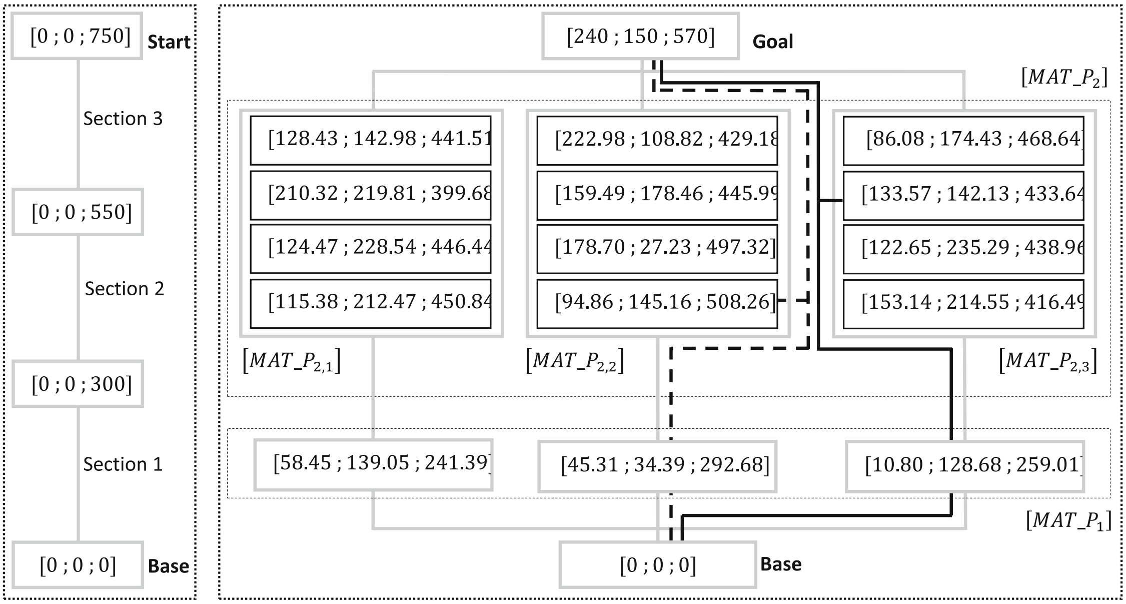
The proposed algorithm is tested through three numerical experiments utilizing a CDCR with three sections (see Fig. 1). The first example is considered to illustrate the block diagram depicted in Fig. 3 that showing the reachable EP poses of the two sections for a given robot's EP pose. The second numerical experiment has been carried out in order to track a line-shaped trajectory in a free environment. The last is performed to reach a desired goal while avoiding a static obstacle in the robot's workspace, in which the collision and free-collision's areas are illustrated. To select the optimal IK solution, in all examples, two criteria have been applied, namely minimum variation of the actuator space vector and traveled distance, and after that, their results have been compared. The numerical experiments have been performed in MATLAB software on a computer with the following specifications: Intel(R) Xeon(R) CPU E5-1620 0 @ 3.60 GHz, 16 Go of RAM, 64-bit operating system. The geometric parameters of CDCR are given in Table 1.

Table 1 Parameters of CDCR for the numerical experiments

Designation	Section's length (mm)	Radial distance between cables and the backbone (mm)	Disk radius (mm)
Section 1	300	40	45
Section 2	250	30	35
Section 3	200	20	25

5.1 Illustration of the Block Diagram of the Proposed Algorithm: A Numerical Example

The purpose of this numerical example is to thoroughly illustrate the block diagram as it is shown in Fig. 3. To emphasize, the three-section continuum robot starts from its initial configuration[0; 0; 750], see Fig. 4a, to reach the desired goal located at[240; 150; 570]. In fact, there is an infinite possible configuration. However, for simplicity reasons, three EP poses are exclusively taken for the first section, denoted$[MAT_P_1]$; and for each point



(a) Initial configuration **(b)** Multiple considered configurations

Fig. 4 Numerical example in the form of a tree illustrating the block diagram of the proposed algorithm

Table 2 Optimal reachable EP poses of Sect. 1 and 2 satisfying both criteria to achieve the desired goal

Criterion	Minimum variation of the actuator space	Traveled distance
Section 1	[10.80; 128.68; 259.01]	[45.31; 34.39; 292.68]
Section 2	[133.57; 142.13; 433.64]	[94.86; 145.16; 508.26]

of$[MAT_P_1]$, four EP poses are chosen for the second section, denoted$\left[MAT_P_{2,q}\right]$, in which the reachable EP poses $\left[MAT\_P_{2,q}\right]$, with $q = 1, 2, 3,$ forms the matrix of all reachable EP poses of the second section, denoted$[MAT_P_2]$, see Fig. 4b. However, to select an optimal configuration, namely to pick up the most appropriate IK solution, among the considered varieties of solutions, criteria of minimum variation of the actuator space and traveled distance are granted. The obtained optimal achievable EP poses of Sects. 1 and 2 are given in Table 2 and which are displayed, respectively, in bold-black line and dashed-line in Fig. 4b.

5.2 IKs for Tracking a Trajectory in Free Environment

This simulation experiment attempts at first to demonstrate the existence of infinite configurations of the robot, i.e., all possible IK solutions, from any starting pose to any destination pose on the trajectory. From a realistic view, it is not possible to reach all IK solutions; however, through the proposed algorithm, the problem can be reduced to a finite number of IK solutions by considering the sets of reachable EP poses of the two first sections. Before starting, it is obvious that when the robot is in a straight line, there is a unique configuration of the robot, which represents generally its initial configuration, where P_1 and P_2 represent the EP poses of the two first sections in this case, see Fig. 5a. Figure 5.b shows such a destination pose of the robot in which the sets of reachable EP poses of the two first sections are shown (red points). Besides, three random configurations of the robot, presented by its backbone, are depicted.

To illustrate more about Fig. 5b, let $[MAT\_P_1] = 4621$ EP poses for the first section and $\left[MAT\_P_{2,q}\right] = 300$ EP poses for the second section for each point of $[MAT\_P_1]$. The dimension of the problem becomes 4621×300 possible configurations of the robot for such pose on the trajectory, which can be explained graphically in Fig. 5c.

To highlight the redundancy resolution aspect along the desired trajectory, two cases have been considered for a line-shaped trajectory $[40t, 25t, 750 - 30t]^T$(mm). Figure 6a and b illustrates the robot's EP poses and its backbone deflection for tracking a line-shaped trajectory with two criteria, namely minimum variation of the actuator space vector and traveled distance, respectively.

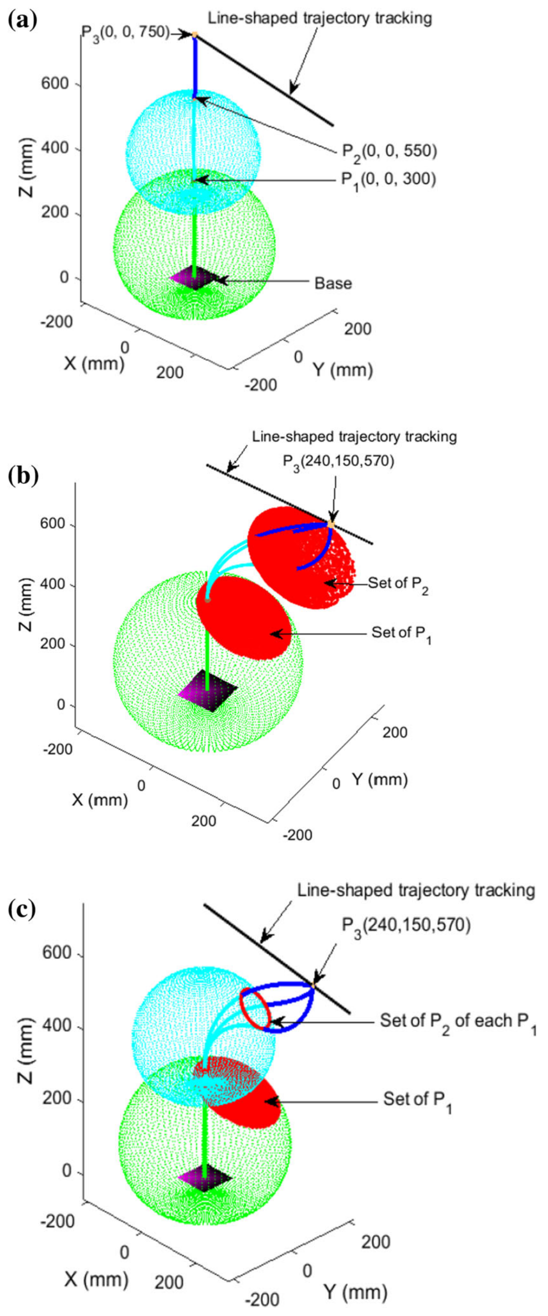


Fig. 5 Representation of reachable EP poses of two first sections for such robot's EP pose. Panel **a** shows the initial configuration of the CDCR presented by its central backbones. The green and cyan set points present the theoretical workspaces of the two first sections of the robot, respectively. Panel **b** shows the reachable EP poses of two first sections presented by the sets red points, i.e., set of P_1 and P_2, for such robot's EP pose. Panel **c** shows three random configurations of the robot reached such pose on the line-shaped trajectory

Figure 7 presents the variation of cable lengths (actuator space vectors) required to track the desired line-shaped trajectory with two considered criteria. From this Figure, it can be seen that the two criteria give the similar solution in finding IK solutions. Of course, the two solutions are very similar because practically and according to the given trajectory, it

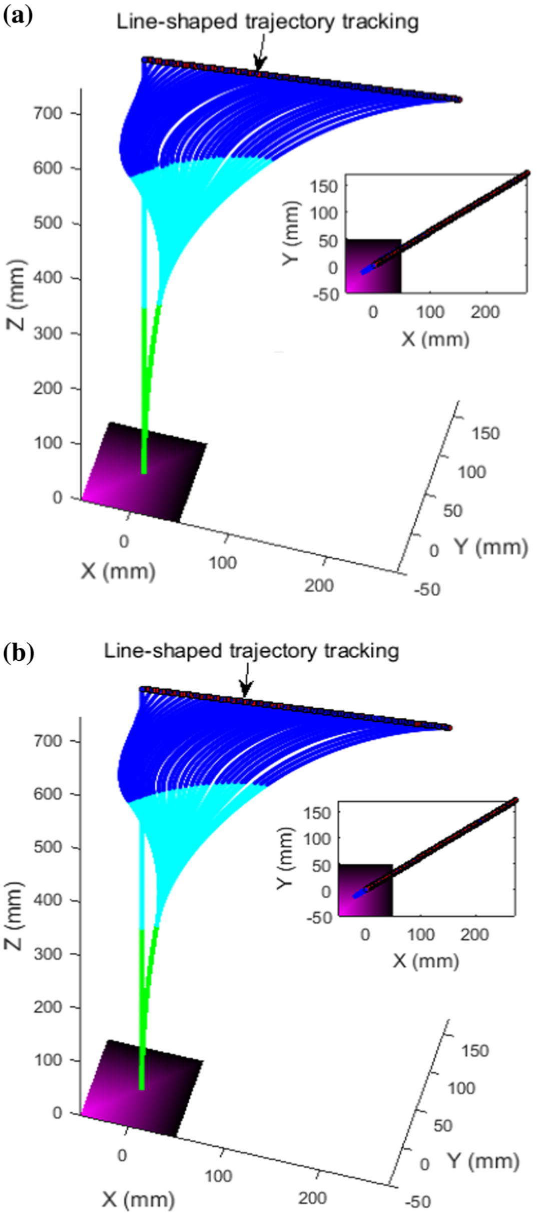


Fig. 6 Inverse position kinematics: **a** and **b** the robot's EP pose tracks a line-shaped trajectory with minimum variation of the actuator space vector and the traveled distance of two first sections' EP poses, respectively

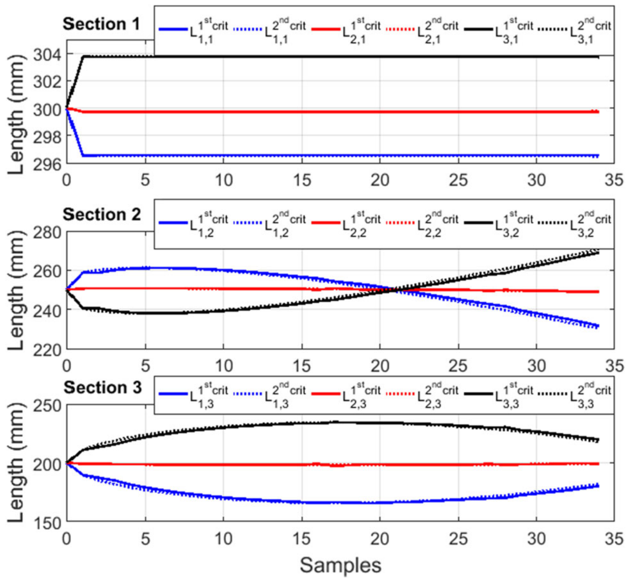


Fig. 7 The required actuator space vectors to track the line-shaped trajectory for two criteria

is the unique and optimal solution that can be obtained; and this indicates the computational efficiency of the proposed algorithm in finding IK solutions and redundancy resolution. However, the slight difference apparent in the lengths variation, especially in the second and third sections, is due to the small number of EP poses considered for the second section.

5.3 IKs for Reaching a Goal in Presence of Obstacle

In this numerical experiment, the proposed algorithm is evaluated against avoiding obstacles which can be encountered in the robot's workspace. According to this numerical experiment, areas of collision and free-collision (obstacle avoidance) of the robot can be identified. Let us consider a static obstacle that is located at$\mathbb{X}_{obs} = [60; 70; 350]$(mm), within the robot's workspace (see Fig. 8), from the fixed reference frame and then, the robot starts from its initial state [0; 0; 750](mm) to reach the desired goal$\mathbb{X}_{desired} =$ [240; 150; 570](mm). Interestingly, to ensure that obstacles along the trajectory are avoided, the path must be subdivided into a large number of points. To achieve this purpose, the obstacle avoidance is provided at the point of destination. For visual representation, the red and green points, depicted in Fig. 8, present the collision and free-collision's areas, respectively, of the robot with obstacle.

To avoid that obstacle, an additional constraint has been introduced to the optimization problem aiming at retaining the Euclidean distance between the obstacle's position and some points on the robot's backbone beyond a safety distance equal to 40 mm. However, to select the optimal configuration through which the robot does not collide with the obstacle and properly tracks the desired trajectory, the previously criteria, namely minimum variation of the actuator space vector and traveled distance are accorded. The reachable EP poses of the two first sections of the robot that satisfy the two aforementioned criteria are presented in Table 3. Besides, the actuator space vectors required to reach the desired goal for both criteria, namely minimum variation of the actuator space vector and traveled distance are given in Table 4.

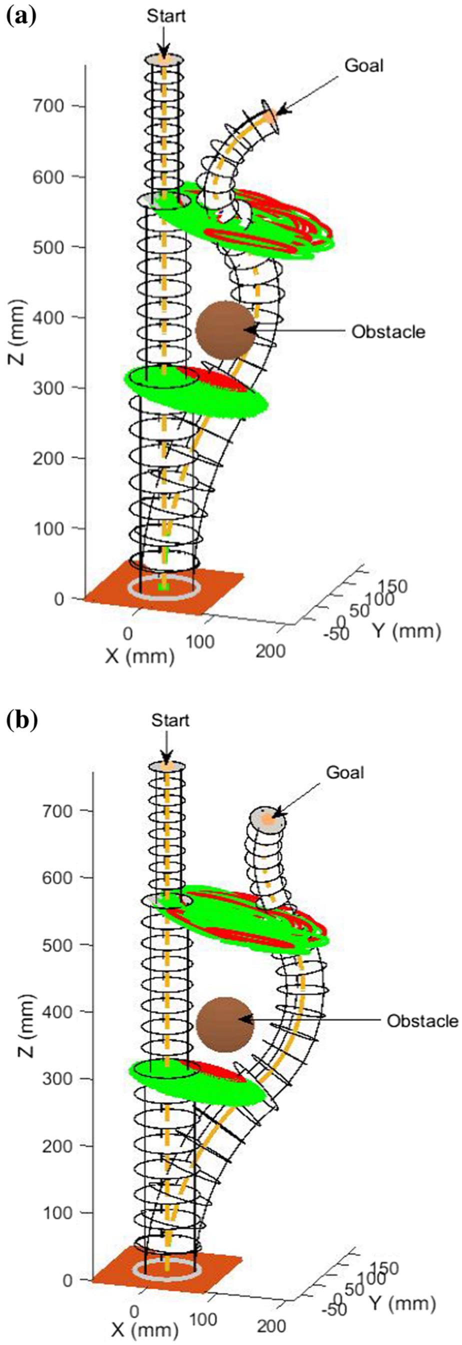


Fig. 8 **a** and **b** Collision and free-collision's areas for the robot (Red and green points present collision and free-collision's areas, respectively)

Table 3 Reachable EP poses of Sects. 1 and 2 satisfying both criteria to reach the desired goal

Criterion	Minimum variation of the actuator space	Traveled distance
Section 1	[145.70; 115.48; 199.38]	[0; 0; 300]
Section 2	[258.80; 188.08; 374.68]	[74.49; 46.72; 529.53]

6 Summary

Three numerical experiments are conducted on the IK problem and redundancy resolution for a class of continuum robots, namely Cable-Driven Continuum Robot (CDCR), to test and validate the proposed algorithm. The aim of the first example is to provide with an explicit illustration of the block diagram as it is shown in Fig. 3. The second one is dedicated to figuring out the redundant IK solutions and the redundancy resolution for tracking a trajectory without obstacle avoidance, while third one is considered for reaching a desired goal in the presence of obstacle. According to the achieved results, the proposed algorithm achieves good performance in terms of computational accuracy and efficiency in obtaining global IK solutions and redundancy resolution criteria, unlike the previously proposed approaches.

Regardless of the runtime, the main advantages of the proposed algorithm can be summarized as follows: (1) easy parameterization and simple build, (2) does not require any data to apply it except the treatment of the problem's inputs, (3) no links with the degree of redundancy of the robot and its characteristics, (4) a database builder, and (5) it can be applied to any continuum robot regardless of its number of sections.

Comparatively, the so-called continuum robots are composed of hard or soft structure, which makes their characteristics and properties different, such as sizes, material, continuous/discontinuous shape, etc. On the other hand, these robots can be controlled by different inputs including pneumatic actuation, hybrid actuation (pneumatic + electric), cable/tendon actuation, and so on. Based on the aforementioned differences between the existing continuum robots' proprieties, a comprehensive comparison cannot be fully considered by merely considering the results obtained in this work. Table 5 lists some contributions dealing with IK problem and redundancy resolution of continuum robots.

Table 4 Actuator space vectors required to reach the desired goal for both criteria

Criterion	Minimum variation of the actuator space	Traveled distance
Cable lengths of Sect. 1 $(l_{i,1})$	(252.57, 290.80, 355.30)	(300, 300, 300)
Cable lengths of Sect. 2 $(l_{i,2})$	(286.79, 261.00, 200.79)	(231.32, 249.10, 269.30)
Cable lengths of Sect. 3 $(l_{i,3})$	(199.48, 200.36, 200.14)	(178.25, 198.96, 222.15)

Table 5 Some contributions dealing with IK problem and redundancy resolution of continuum robots

Robot	Characteristics	Modeling method	IK problem	Redundancy problem
CDCR	Continuous shape, rigid structure, multi-sections, controlled by cables	Proposed algorithm in this paper	All-Inclusive IK Solutions can be provided	Can deal with redundancy using different criteria
		PSO metaheuristic approach [25, 26]	Only one IK solution can be provided	
CBHA	Continuous shape, soft structure, two sections, pneumatic actuation	Pythagorean Hodograph curves-based approach [21]	One optimal or non-optimal IK solution can be provided	Cannot deal with redundancy problem
		Neural network approach [22]		
		Hybrid approach: Geometry and Neural Network [23]		
		PSO metaheuristic approach [26]	Only one IK solution can be provided	Can deal with redundancy using different criteria
		SQP-based approach [27]		
		Approach-based Jacobian matrix [29]	Only one IK solution can be provided	Difficult dealing with redundancy problem
BHA	Continuous shape, soft structure, two sections, pneumatic actuation	Approach-based Jacobian matrix [11]		
		Goal babbling learning [24]	Only one IK solution can be provided	Cannot deal with redundancy problem
Redundant manipulator	Discontinuous shape, rigid structure, multi-sections	Modal approach [18]		
Multi-section continuum arm	Continuous shape, soft structure, three sections, Pneumatic muscle actuation	Modal approach [28]		
		Geometrical-based approach [17]		

Comparatively speaking, an overview of the previously developed continuum robot's IK solutions is given in Table 5, which proved that the proposed algorithm is the best and can be used as an exhaustive method to solve the IK problem and its redundancy for continuum robots (despite the differences in characteristics and properties of the studied robots), and thus, it can be considered as an alternative and a sovereign remedy to compensate for the shortcoming experienced by the previously proposed algorithms.

7 Conclusion

In this paper, a novel reliable algorithm is proposed. It accurately provides with all-inclusive inverse kinematics' solutions of continuum robots as well as the resolution of redundancy. Certainly, there are an infinite number of configurations through which the continuum robot's EP pose can attain a specific pose within its workspace, in other words, an endless number of IK solutions are inevitably encountered for a given pose in the robot's workspace. Through the proposed algorithm, the infinite number of the IK solutions can be reduced to a finite number by considering sets of reachable EP poses of the first $(n-1)$ sections. It consists of a two-sub-algorithm: the first sub-algorithm collects and forms a tree of the reachable EP poses of the first $(n-1)$ sections for each robot's EP pose. In this way, the robot's sets EP poses are selected and built up one-by-one starting from the first section to $(n-1)$th section. The second sub-algorithm selects the optimal posture of the robot, among the redundant varieties solutions provided by the first sub-algorithm's outputs, according to a specific criterion. However, since the proposed algorithm is strongly related to an optimization method, the Particle Swarm Optimization (PSO) method is adopted, to pick up the first specific IK solution, due to its particularly considerable advantages compared to other algorithms, particular: its implementation simplicity and its fast convergence that provides a low computation time.

To test and validate the efficiency of the proposed algorithm, three numerical experiments are carried out on a cable-Driven Continuum Robot (CDCR). The first example is explicitly provided to illustrate the functionality of the two sub-algorithms. While the second and third numerical experiments are performed for tracking a trajectory in a free environment as well as to reach a desired goal in the presence of obstacle within the robot's workspace. In all simulations, IK, multiple IK solutions and redundancy resolution are thoroughly discussed. The obtained results showed that the proposed algorithm is computationally accurate and efficient in obtaining all-inclusive IK solutions and redundancy resolution of continuum robots.

In comparison with existing works about IK, that directly address the problem in this paper, and despite the differences in the above-mentioned properties and characteristics, the proposed algorithm can be considered as a comprehensive method for perfectly addressing the IK problem and its redundancy for continuum robots. Furthermore, the proposed algorithm can be applied to any kind of continuum robot regardless of the number of sections, which makes it a sovereign remedy for the IK problem and redundancy of continuum robots. One of the many advantages regarding the developed algorithm in this paper is that it can be adopted by researchers to directly build their database without using the forward kinematic which significantly helps to narrow down the unwanted singularities triggered by the forward kinematic during the build-up of the needed database, for instance, neural networks and growing gas techniques strongly depend on the database, thus a well-collected data helps to achieve accurate results.

In the future, we will address the issue of mobile continuum manipulators.

References

1. Seleem, I.A.; El-Hussieny, H.; Assal, S.F.M.; Ishii, H.: Development and stability analysis of an imitation learning-based pose planning approach for multi-section continuum robot. IEEE Access. **8**, 99366–99379 (2020). https://doi.org/10.1109/ACCESS.2020.2997636
2. Li, H.; Yao, J., et al.: A bioinspired soft swallowing robot based on compliant guiding structure. Soft Robot. **7**, 491–499 (2020). https://doi.org/10.1089/soro.2018.0154
3. Ma, X.; Song, C.; Chiu, W.Y.P.; Li, Z.: Autonomous flexible endoscope for minimally invasive surgery with enhanced safety. IEEE Robot. Autom. Lett. **4**, 2607–2613 (2019). https://doi.org/10.1109/LRA.2019.2895273
4. Alfalahi, H.; Renda, F.; Stefanini, C.: Concentric tube robots for minimally invasive surgery: current applications and future opportunities. IEEE Trans. Med. Robot. Bionics. **2**, 410–424 (2020). https://doi.org/10.1109/TMRB.2020.3000899
5. Zhang, Y.; Lu, M.: A review of recent advancements in soft and flexible robots for medical applications. Int. J. Med. Robot. Comput. Assist. Surg. (2020). https://doi.org/10.1002/rcs.2096
6. Mingfeng, W.; Xin, D.; Weiming, B.; Abdelkhalick, M.; Dragos, A.; Andy, N.: Design, modelling and validation of a novel extra slender continuum robot for in-situ inspection and repair in aeroengine. Robot. Comput. –Integr. Manuf. (2021). https://doi.org/10.1016/j.rcim.2020.102054
7. Kolachalama, S.; Lakshmanan, S.: Continuum robots for manipulation applications: a survey. J. Robot. **2020**(4187048), 19 (2020). https://doi.org/10.1155/2020/4187048
8. Niu, G.; Zhang, Y.; Li, W.: Path planning of continuum robot based on path fitting. J. Control Sci. Eng. **2020**(8826749), 11 (2020). https://doi.org/10.1155/2020/8826749
9. Bousbia, L.; Amouri, A.; Cherfia, A.: Dynamics modeling of a 2-DoFs cable-driven continuum robot. World J. Eng. (2022). https://doi.org/10.1108/WJE-01-2021-0028
10. Rao, P.; Peyron, Q.; Lilge, S.; Burgner-Kahrs, J.: How to model tendon-driven continuum robots and benchmark modelling performance. Front. Robot. AI (2021). https://doi.org/10.3389/frobt.2020.630245
11. Mahl, T.; Hildebrandt, A.; Sawodny, O.: A variable curvature continuum kinematics for kinematic control of the bionic handling assistant. IEEE Trans. Rob. **30**(4), 935–949 (2014). https://doi.org/10.1109/TRO.2014.2314777
12. Abdel-Nasser, M.; Salah, O.: New continuum surgical robot based on hybrid concentric tube-tendon driven mechanism. Proc. Inst. Mech. Eng. C J. Mech. Eng. Sci. **235**(24), 7550–7568 (2021). https://doi.org/10.1177/09544062211042407
13. Mishra, M.K.; Samantaray, A.K.; Chakraborty, G.; Pachouri, V.; Pathak, P.M.; Merzouki, R.: Kinematics model of bionic manipulator by using elliptic integral approach. In: Kumar, R.; Chauhan, V.S.; Talha, M.; Pathak, H. (Eds.) Machines, Mechanism and Robotics. Lecture Notes in Mechanical Engineering. Springer, Singapore (2022). https://doi.org/10.1007/978-981-16-0550-5_30
14. Webster, R.J.; Jones, B.A.: Design and kinematic modeling of constant curvature continuum robots: a review. Int. J. Robot. Res. **29**(13), 1661–1683 (2010). https://doi.org/10.1177/0278364910368147
15. Santina, C.D.; Rus, D.: Control oriented modeling of soft robots: the polynomial curvature case. IEEE Robot. Autom. Lett. **5**, 290–298 (2020). https://doi.org/10.1109/LRA.2019.2955936
16. Huang, S.; Meng, D.; Wang, X.; Liang, B.; Lu, W.: A 3D static modeling method and experimental verification of continuum robots based on pseudo-rigid body theory. In: IEEE International Conference on Intelligent Robots and Systems, Macau, China, (2019). https://doi.org/10.1109/IROS40897.2019.8968526.
17. Neppalli, S.; Csencsits, M.A.; Jones, B.A., et al.: Closed-form inverse kinematics for continuum manipulators. Adv. Robot. **23**, 2077–2091 (2012). https://doi.org/10.1163/016918609X12529299964101
18. Chirikjian, G.S.; Burdick, J.W.: A modal approach to hyper-redundant manipulator kinematics. IEEE Trans. Robot. Autom. **10**(3), 343–354 (1994). https://doi.org/10.1109/70.294209
19. Zhao, B.; Zeng, L.; Wu, B.; Xu, K.: A continuum manipulator with closed-form inverse kinematics and independently tunable stiffness. In: IEEE International Conference on Robotics and Automation (ICRA). 1847–1853, (2020). https://doi.org/10.1109/ICRA40945.2020.9196688.
20. Garriga-Casanovas, A.; Rodriguez y Baena, F.: Kinematics of continuum robots with constant curvature bending and extension capabilities. J. Mech. Robot. (2019). https://doi.org/10.1115/1.4041739
21. Singh, I.; Amara, Y.; Melingui, A.; Mani Pathak, P.; Merzouki, R.: Modeling of continuum manipulators using pythagorean hodograph curves. Soft Robot. (2018). https://doi.org/10.1089/soro.2017.0111
22. Melingui, A.; Lakhal, O.; Daachi, B., et al.: Adaptive neural network control of a compact bionic handling arm. IEEE/ASME Trans. Mechatron. **20**(6), 2862–2875 (2015). https://doi.org/10.1109/TMECH.2015.2396114
23. Lakhal, O.; Melingui, A.; Merzouki, R.: Hybrid approach for modeling and solving of kinematics of a compact bionic handling assistant manipulator. IEEE/ASME Trans. Mechatron. **21**(3), 1326–1335 (2016). https://doi.org/10.1109/TMECH.2015.2490180

24. Rolf, M.; Steil, J.J.: Efficient exploratory learning of inverse kinematics on a bionic elephant trunk. IEEE Trans. Neural Netw. Learn. Syst. **25**(6), 1147–1160 (2014). https://doi.org/10.1109/TNNLS.2013.2287890
25. Djeffal, S.; Amouri, A.; Mahfoudi, C.: Kinematics modeling and simulation analysis of variable curvature kinematics continuum robots. UPB Sci. Bull., Ser. D: Mech. Eng. **83**(1), 28–42 (2021)
26. Amouri, A.; Mahfoudi, C.; Zaatri, A., et al.: A metaheuristic approach to solve inverse kinematics of continuum manipulators. J. Syst. Control Eng. **231**(5), 380–394 (2017). https://doi.org/10.1177/0959651817700779
27. Lakhal, O.; Melingui, A.; Chibani, et al.: Inverse kinematic modeling of a class of continuum bionic handling arm. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Besacon. 1337–1342, (2014). doi: https://doi.org/10.1109/AIM.2014.6878268.
28. Godage, I.S.; Guglielmino, E.; Branson, D.T.; et al.: Novel modal approach for kinematics of multisection continuum arms. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA. 1093–1098, (2011). https://doi.org/10.1109/IROS.2011.6094477.
29. Singh, I.; Lakhal, O.; Amara, Y.; Coelen, V.; et al.: Performances evaluation of inverse kinematic models of a compact bionic handling assistant. In: IEEE International Conference on Robotics and Biomimetics (ROBIO), Macau. 264–269, (2017). https://doi.org/10.1109/ROBIO.2017.8324428.
30. Jones, B.A.; Walker, I.D.: Kinematics for multisession continuum robots. IEEE Trans. Robot. **22**(1), 43–55 (2006). https://doi.org/10.1109/TRO.2005.861458
31. Kennedy Eberhart, R.: Particle swarm optimization. In: Proceedings of International Conference on Neural Networks, Perth, WA, Australia. 1942–1948, (1995). https://doi.org/10.1109/ICNN.1995.488968.
32. Triki, E.; Collette, Y.; Siarry, P.: A theoretical study on the behavior of simulated annealing leading to a new cooling schedule. Eur. J. Oper. Res. **166**, 77–92 (2005). https://doi.org/10.1016/j.ejor.2004.03.035
33. Rezaee Jordehi, A.; Jasni, J.: Parameter selection in particle swarm optimization: a survey. J. Exp. Theor. Artif. Intell. **25**(4), 527–542 (2013). https://doi.org/10.1080/0952813X.2013.782348
34. Shi, Y.; Eberhart, R.: A modified particle swarm optimizer. In: IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360), Anchorage, AK, USA.69–73, (1998). https://doi.org/10.1109/ICEC.1998.699146.
35. Ram, R.V.; Pathak, P.M.; Junco, S.J.: Inverse kinematics of mobile manipulator using bidirectional particle swarm optimization by manipulator decoupling. Mech. Mach. Theory **131**, 385–405 (2019). https://doi.org/10.1016/j.mechmachtheory.2018.09.022
36. Noel, M.M.: A new gradient based particle swarm optimization algorithm for accurate computation of global minimum. Appl. Soft Comput. **12**(1), 353–359 (2012). https://doi.org/10.1016/j.asoc.2011.08.037
37. Amouri, A.: Contribution à la modélisation dynamique d'un robot flexible bionique. Ph.D. dissertation, University of Constantine 1, Algeria (2017). http://depot.umc.edu.dz/handle/123456789/6489
38. Eberhart, R.C.; Shi, Y.: Comparing inertia weights and constriction factors in particle swarm optimization. In: Congress of Evolutionary Computation, San Diego, CA. **1**, 84–88, (2000). https://doi.org/10.1109/CEC.2000.870279.