



An optimal design of a flexible piping inspection robot

Swaminath Venkateswaran, Damien Chablat, Pol Hamon

► To cite this version:

Swaminath Venkateswaran, Damien Chablat, Pol Hamon. An optimal design of a flexible piping inspection robot. Journal of Mechanisms and Robotics, 2021, 13 (3), pp.035002. 10.1115/1.4049948 . hal-03126584

HAL Id: hal-03126584

<https://hal.science/hal-03126584>

Submitted on 31 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An optimal design of a flexible piping inspection robot*

Swaminath Venkateswaran[†]

Ecole Centrale de Nantes
Laboratoire des Sciences du
Numérique de Nantes(LS2N),
UMR CNRS 6004,
1 rue de la Noë, 44321,
Nantes, France

Email: swaminath.venkateswaran@ls2n.fr

Damien Chablat[‡]

CNRS
Laboratoire des Sciences du
Numérique de Nantes(LS2N),
UMR CNRS 6004,
1 rue de la Noë, 44321,
Nantes, France

Email: damien.chablat@cnr.fr

Pol Hamon

Ecole Centrale de Nantes
Laboratoire des Sciences du
Numérique de Nantes(LS2N),
UMR CNRS 6004,
1 rue de la Noë, 44321,
Nantes, France
Email: pol.hamon@ls2n.fr

ABSTRACT

This article presents an optimization approach for the design of a piping inspection robot. A rigid bio-inspired piping inspection robot that moves like a caterpillar was designed and developed at LS2N, France. By the addition of tensegrity mechanisms between the motor modules, the mobile robot becomes flexible to pass through the bends. However, the existing motor units prove to be oversized for passing through pipe bends at 90°. Thus, three cascading optimization problems are presented in this article to determine the sizing of robot assembly that can overcome such pipe bends. The first problem deals with the

*The original version of this paper has been accepted for presentation at the ASME-IDETC 2020 under the title “Design of a piping inspection robot by optimization approach”, DETC2020/22021, August 16–19, 2020, Saint-Louis, USA

[†]Address all correspondence for other issues to this author.

[‡]Address all correspondence related to ASME style format and figures to this author.

identification of design parameters of the tensegrity mechanism based on static stability. Followed by that, in the second problem, the optimum design parameters of the robot modules are determined for the robot assembly without the presence of leg mechanisms. The third problem deals with the determination of the size of the leg mechanism for the results of the two previous optimization problems. A digital model of the optimized robot assembly is then realized using CAD software.

NOMENCLATURE

IKP	Inverse Kinematic Problem
CG	Center of gravity
CLR	Center line radius
$\mathbf{R}_x$	Rotation matrix in $\mathbb{R}^3$ about x-axis
$\mathbf{R}_y$	Rotation matrix in $\mathbb{R}^3$ about y-axis
η	The 'x' axis Euler rotation angle of universal joint in $\mathbb{R}^3$
ϕ	The 'y' axis Euler rotation angle of universal joint in $\mathbb{R}^3$
$\mathbf{R}$	Resultant Euler angles transformation matrix
L_i	Distance between base and end-effector of the i^{th} coordinate
L_0	Free length of the spring
$\mathbf{H}$	Hessian matrix
δ_i	Rotation angle of i^{th} module
l_1	Length of motor unit (EC motor+Spindle drive)
w_1	Width/diameter of motor unit (EC motor+Spindle drive)
l_2	Length of screw of spindle drive
w_2	Width/diameter of screw of spindle drive
f_{un}	Objective function
g	Inequality constraint
h	Equality constraint
lb, ub	Lower and upper bounds for design variables

l_{s1}	Length of slot-follower leg mechanism
l_{s2}	Design parameter of leg mechanism
$\Delta x_1, \Delta x_2$	Horizontal space occupied by leg during declamping & clamping
$\Delta y_1, \Delta y_2$	Vertical space occupied by leg during declamping & clamping
ρ_1, ρ_2	Leg stroke lengths during declamping & clamping

INTRODUCTION

Manual intervention of pipelines in Nuclear or Chemical industries is cumbersome and risky as it not only leads to the loss of human life but also causes long term radiation effects. Piping inspection robots play an important role in such situations as they can perform a given task with better accuracy and also ensures safety to human health. Generally, in-pipe inspection robots can be classified [1] viz: Pig type [2], Wheel type [3], Caterpillar type [4], Wall-press type [5], Walking type [6], Inchworm type [7] and Screw type [8]. A bio-inspired piping inspection robot was designed and developed at LS2N, France [9]. This robot accomplishes the locomotion of a caterpillar in six-steps to move inside a pipeline. Using a slot-follower leg mechanism, the robot attains static postures to establish contact with pipeline walls. The leg mechanisms are capable of working between 40–94 mm diameter straight pipelines [10]. However, the robot is a rigid model and is thus limited to straight pipelines. Also, the spindle drive used in the prototype allows the robot to move at a very low speed of 0.43 mm/s [11]. By the addition of tensegrity mechanisms between the motor modules, the robot was made flexible [12]. The dimensions of the existing motor units are oversized and might restrict the flexible robot to overcome pipe bends or junctions. In this article, an optimization approach is followed to determine the optimum sizing of the entire robot assembly inside a pipeline having 90° pipe bend where the tensegrity mechanism operates in a passive mode. Optimization algorithms can be classified into deterministic and heuristic approaches [13]. The deterministic optimization approach solves problems in a structured manner that leads to efficient solutions with respect to design variables and constraints. On the other hand, heuristic/evolutionary algorithms operate on randomness [14]. Based on the complexity of the de-

sign problem, the heuristic approach requires lesser computational times over the deterministic approach [15]. However, there are possibilities where both the algorithms can lead to solutions trapped in a local minimum. While applying such algorithms to real-world engineering problems, the design engineer can select a local minima through careful definition of constraints or bounds for the optimization problem. In the field of robotics, optimization techniques are generally employed for solving problems such as path planning. An interesting example is the research work of Zhang et al. [16], where a heuristic approach using the Particle Swarm Optimization (PSO) was employed to determine a collision-free path for the robot. Also, optimization techniques find their application in the design of parallel robots for determining a robust design with maximum operating workspace [17] as well as in determining the optimum position for placement of the robot [18].

This article focuses on the optimum design of the flexible piping inspection robot proposed in [12] through three different problems. The robot assembly is comprised of multiple sub-systems such as tensegrity mechanisms, motor units and leg mechanisms. Each sub-system is subject to separate set of constraints and objectives under a given working condition. By the incorporation of a multi-objective optimization problem for the entire assembly, the complexity of the system increases and the selection of the solution candidate from the Pareto front is more subjective [17]. So, a cascading approach is being implemented through three different sub-problems to identify an optimal design of the robot assembly. The first optimization problem deals with the determination of optimal design parameters of the 4-SPS-U tensegrity mechanism [19] by a metaheuristic approach. Using Genetic Algorithm in MATLAB, the total length of the mechanism and the free length of the spring are determined by ensuring the static stability of the tensegrity mechanism. Followed by that the second optimization problem is carried out using the *fmincon* function in MATLAB. This problem helps in determining the sizes of motor unit of each module without the presence of leg mechanisms. From the results of the second optimization problem, the motors and gear units are identified from catalogue products of Maxon based on dimensions, velocity and force factors. Also, the second optimization problem permits to set the diameter range of pipelines for the robot. For studying the area occupied by the leg mechanisms, the third optimization problem is then carried out which helps in determining the geometry of the slot-follower leg mechanism.

The area occupied by the leg mechanism is identified by accommodating the mechanism on the motor sizing determined from the results of the second optimization problem. By assembling the tensegrity mechanism, motor modules and the leg mechanisms, the digital model of the optimized robot design is realized using CATIA software.

The outline of the article is as follows. Initially, the architecture of the bio-inspired robot and the setup of the optimization test bench are presented. Followed by that, the first optimization problem, which is associated with the design of tensegrity mechanism is defined and solved. The subsequent sections deals with the deterministic optimization problems for identifying the optimal size of motor units and the slot-follower leg mechanism. The article then ends with conclusions.

DEFINITION OF THE CASE STUDY

This section focuses on the presentation of the architecture of the robot assembly with the presence of tensegrity mechanisms. The dimensions and construction of the optimization test bench, which will be employed for the second and third optimization problems, are then presented.

Architecture of the robot

The rigid bio-inspired robot developed at LS2N, France comprises three modules that ensure clamping and elongation phases for accomplishing the locomotion of a caterpillar. Each module consists of a DC brushless motor coupled with a spindle drive to convert rotary motion to prismatic movement [9, 10]. The front and rear modules have 3 sets of slot-follower leg mechanisms each to have contact with the inner walls of pipelines at all instances of locomotion. The central module takes care of the elongation and retraction phases¹. By the study of design issues viz: Passive compliance, Active compliance and Tilt limits [12, 20], a tensegrity mechanism was proposed and introduced as an articulation unit for the bio-inspired robot. Through singularity and workspace analysis, two types of tensegrity mechanisms viz: 3-SPS-U and 4-SPS-U architectures were analyzed in [19]. These architectures are differentiated by the number of tensions springs employed where the former employs three springs and the latter employs four springs. Based on the results of [19], the 3-SPS-U architecture is compatible with addressing the issue of active compliance

¹Working principle of the existing rigid prototype : [Click here](#).

while the 4-SPS-U architecture addresses both compliance issues. As the 4-SPS-U tensegrity mechanism provides higher tilt limits under passive modes [19], this architecture is coupled along with the bio-inspired robot to make it flexible. The digital model of the existing bio-inspired robot and the 4-SPS-U tensegrity mechanisms are assembled using the CATIA software to have a flexible robot assembly and it is represented in Fig. 1. In Fig. 1, T_1 and T_2 represent the tensegrity

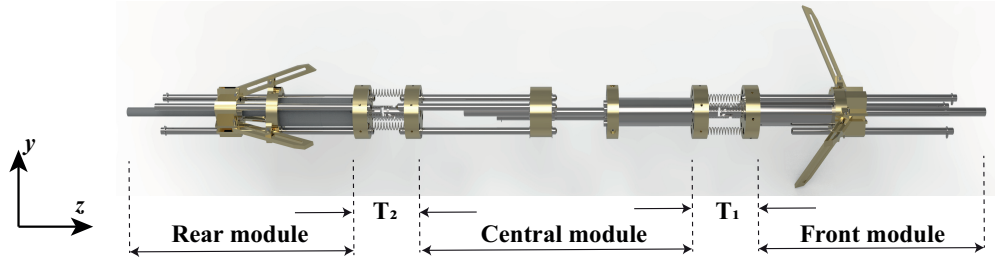


Fig. 1. 3D model of the existing bio-inspired robot coupled with the tensegrity mechanisms

mechanisms. The CAD model depicted in Fig. 1 is constructed using the motors employed in the rigid prototype. If this flexible robot is employed within a pipeline having diameters less than 100 mm, the modules can get trapped in the bends or elbows as the dimensions of the motor units prove to be oversized. For determining the dimensions of tensegrity mechanisms, motors and leg mechanisms that can overcome such bends, an optimization approach is being followed in this article.

Setting up the optimization test bench

The optimization problems are solved inside a test pipeline which is constructed using MATLAB software. The first optimization problem is solved without the test bench. However, the results coming from this problem are taken as fixed design variables for the two successive problems which are solved within the test bench. The second problem is solved for diameters of the test bench ranging from 70 mm to 160 mm. The test bench for the optimization problem consists of a horizontal pipe section, a 90° elbow and a vertical pipe section. The optimal sizing of the robot modules is determined especially at the bend section wherein collision of modules against the walls of the pipelines is verified. The test bench for the optimization problem is represented below

in Fig. 2.

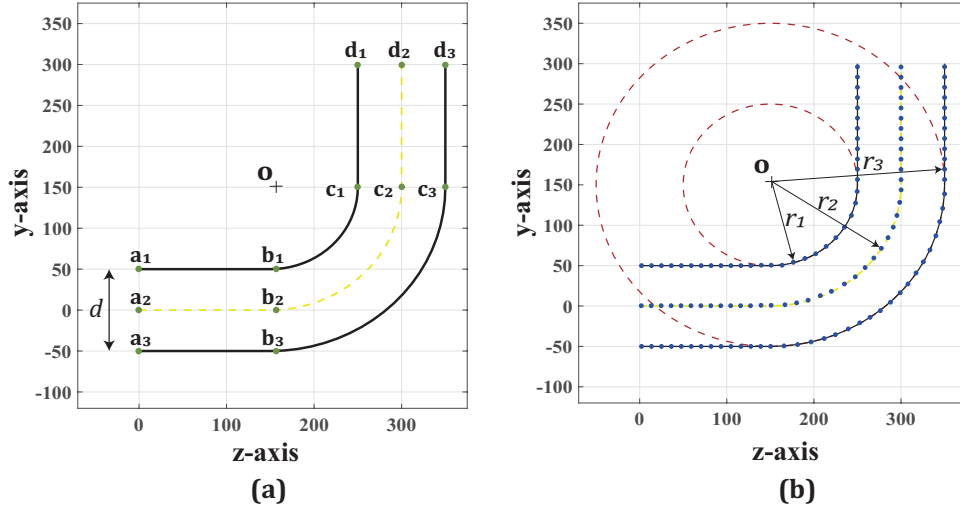


Fig. 2. Overview of the (a) test bench for optimization and (b) parametrization of the test bench for $d = 100$ mm

The test pipe represented in Fig. 2a is generally manufactured by Numerical Control (NC) bending processes using a roller with its origin at $\mathbf{o}$ and flexible mandrels [21]. The coordinates of $\mathbf{o}$ are given by $[1.5d, 1.5d]$, where d is the diameter of the pipeline. With respect to $\mathbf{o}$, the trajectories of the test bench are classified as the inner portion, centerline radius (CLR) and the outer portion. The bending radius of these trajectories represented in Fig. 2b is given by $[r_1, r_2, r_3] = [d, 1.5d, 2d]$. The optimization problems are solved for a fixed diameter value at any instant. The values considered for d are $[70, 80, 90, 100, 110, 120, 130, 140, 150, 160]$ mm. The pipeline geometry is constructed in MATLAB by sub-dividing into straight and bent cross-sections. The coordinate system for the test bench with respect to $\mathbf{o}$ for Fig. 2a is given in Table 1. A discretization is carried

Table 1. Coordinates system of pipe geometry for Fig. 2a

Coordinate	Inner (t=1)	Center (t=2)	Outer (t=3)
$\mathbf{a}_t$	$r_1[0,0.5]$	$r_2[0,0]$	$r_3[0,-0.25]$
$\mathbf{b}_t$	$r_1[1.5,0.5]$	$r_2[1,0]$	$r_3[0.75,-0.25]$
$\mathbf{c}_t$	$r_1[2.5,1.5]$	$r_2[2,1]$	$r_3[1.75,0.75]$
$\mathbf{d}_t$	$r_1[2.5,3]$	$r_2[2,2]$	$r_3[1.75,1.5]$

out between each coordinate wherein the continuous trajectory is being subdivided into discrete counterparts. This discretization is being applied to the walls of the pipelines as well as the CLR and they are represented by blue dotted lines in Fig. 2b. The inner and outer portion discretization is employed to check for collision against modules during the locomotion of the robot. The discretization of the CLR is mainly used to move the robot as well as to rotate each module. The discretization equations for the straight sections are given by:

$$\mathbf{m} = \mathbf{a}_t + disc(\mathbf{b}_t - \mathbf{a}_t) \quad (1)$$

$$\mathbf{m} = \mathbf{c}_t + disc(\mathbf{d}_t - \mathbf{c}_t) \quad (2)$$

where $disc = 0 : 0.01 : 1$ and $t = 1, 2, 3$

In Eqn. (1) and Eqn. (2), $\mathbf{m}$ indicates the coordinates of discretized points between the start and end points of the straight sections. The bend section of pipe with the discretized points between $(\mathbf{b}_t - \mathbf{c}_t)$ is constructed using the equation:

$$\mathbf{m}_j = \left[r_t \sin\left(\frac{j\pi}{40}\right), -r_t \cos\left(\frac{j\pi}{40}\right) \right], \text{ where } j = 0 \text{ to } 20 \text{ and } t = 1, 2, 3 \quad (3)$$

FIRST OPTIMIZATION PROBLEM: DESIGN OF THE TENSEGRITY MECHANISM

During locomotion inside the pipeline, the inspection robot can encounter either an elbow at 90° or a junction. In the event of a junction, the tensegrity mechanism is actuated through cables to tilt the entire assembly along a certain direction. In other words, the mechanism will be controlled from external motor units to follow a given direction. However, in the case of an elbow at 90°, the mechanism operates passively without the influence of external forces to overcome the bend. This article mainly focuses on the determination of optimal design parameters of the robot assembly where the tensegrity mechanism operates in a passive mode. From the results of [20], an inverse pendulum configuration was chosen for the tensegrity mechanism which is by nature an unstable

configuration. The first optimization problem aims to determine the dimensions of the 4-SPS-U tensegrity mechanism by ensuring static stability.

Architecture of the tensegrity mechanism and geometric equations

The tensegrity mechanism comprises of a passive universal joint and four tensions springs. The equations for the mechanism can be generated by correlating to a parallel manipulator of type 4-SPS-U [19] where S indicates spherical joint, P indicates actuated prismatic joint and U indicates universal joint. The representation of the tensegrity mechanism and its correlation to a parallel manipulator at the home-pose is shown in Fig. 3. The tilt angles of the mechanism about

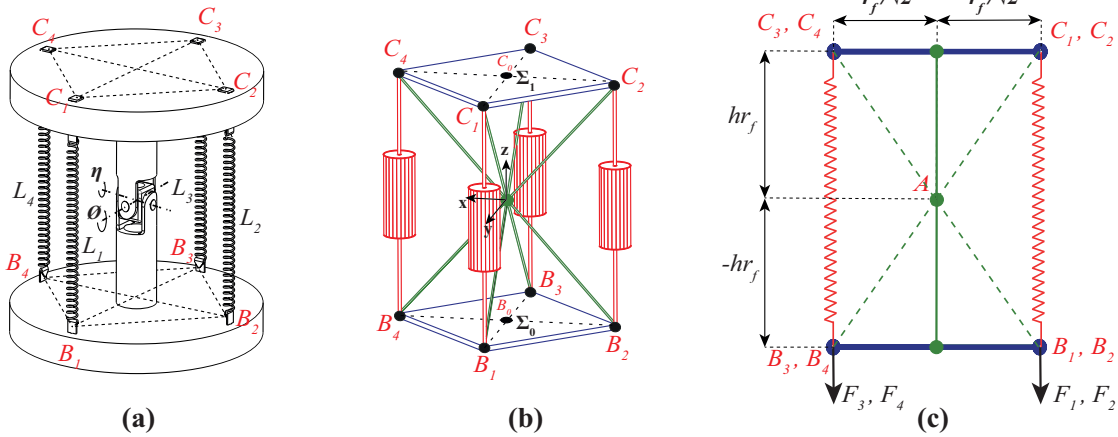


Fig. 3. Representation of the (a) Tensegrity mechanism at home-pose, (b) 3D view of the correlation to a 4-SPS-U manipulator and (c) 2D view of the manipulator

the universal joint are given by η and ϕ . At the home-pose shown in Fig. 3a, the tilt angles are zero. The coordinate frame of the fixed base has its origin at B_0 . The mounting points for the springs are given by B_i ($i = 1, 2, 3, 4$) and they form an imaginary square whose diagonal length is given by $2r_f$. The vector coordinates for the base mounting points are given by:

$$\mathbf{b}_i = \left[r_f \cos \left(\frac{i\pi}{2} \right), r_f \sin \left(\frac{i\pi}{2} \right), -r_f h \right]^T, \text{ with } i = 1, 2, 3, 4 \quad (4)$$

In Eqn. (4), h is a constant that can vary between 0.1 to 1. The coordinate frame of the end-effector has its origin at C_0 . The spring mounting points on the end-effector are given by C_i ($i = 1, 2, 3, 4$). The vector coordinates of the end-effector mounting points can be calculated using the XY Euler rotation angles about the point A of the universal joint. The coordinates are given by:

$$\mathbf{R} = \mathbf{R}_x(\eta)\mathbf{R}_y(\phi) = \begin{bmatrix} \cos(\phi) & 0 & \sin(\phi) \\ \sin(\eta)\sin(\phi) & \cos(\eta) - \sin(\eta)\cos(\phi) \\ -\cos(\eta)\sin(\phi) & \sin(\eta) & \cos(\eta)\cos(\phi) \end{bmatrix} \quad (5)$$

$$\mathbf{c}_i = \mathbf{R} \left[r_f \cos\left(\frac{i\pi}{2}\right), r_f \sin\left(\frac{i\pi}{2}\right), r_f h \right]^T, \text{ with } i = 1, 2, 3, 4 \quad (6)$$

In Eqn. (5), $\mathbf{R} \in SE(3)$ represents the spatial transformation matrix obtained from the Euler angles of universal joint and it is employed to determine the end-effector coordinates. The Inverse Kinematic Problem (IKP) which provides the distance between the base and end-effector of the mechanism is calculated by the equation:

$$L_i = \sqrt{(b_{ix} - c_{ix})^2 + (b_{iy} - c_{iy})^2 + (b_{iz} - c_{iz})^2}, \text{ with } i = 1, 2, 3, 4 \quad (7)$$

Objective functions and constraint equations

Singularity and workspace analysis of the 4-SPS-U tensegrity mechanism was carried out in [19] for $h=1$ and the tilt limits were found to be $\pm 5\pi/18$ radians. However, the static stability of the mechanism under this configuration was not studied. According to Lagrange, for a moving system, the equation of motion [22] is given by:

$$\boldsymbol{\tau} = \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\mathbf{q}}} \right) - \frac{\partial T}{\partial \mathbf{q}} + \frac{\partial U}{\partial \mathbf{q}} \text{ where } \mathbf{q} = [\eta, \phi]^T \quad (8)$$

In Eqn. (8), T and U are the kinetic and potential energies of the system. τ represents the generalized torques on the system. Under static modes, the velocity of the system is zero and there exists no kinetic energies. The potential energy of the tensegrity mechanism is contributed mainly by the springs and by the cables that pass through these springs. The total potential energy of the mechanism is given by the equation:

$$U = U_{cable} + U_{spring} = \sum_{i=1}^4 F_i L_i + \sum_{i=1}^4 \frac{k}{2} (L_i - L_0)^2 \quad (9)$$

In Eq. (9), F_i is the magnitude of applied force along the cable passing through spring i . The parameter k represents the spring stiffness and L_0 indicates the free length of each spring. Based on the dimensions of the flange used in the existing prototype, the value of r_f is considered as 11 mm [9] for the spring mounting positions. The spring stiffness k is considered as 0.75 N/mm which is close to the standard sizes available at LS2N. As the optimization problems will be solved for passive modes through a 90° pipe bend, the applied forces (F_i) are taken as zero as there will be no cable actuation. Thus, the total potential energy of the system can be written as:

$$U = (363 h^2 \cos(\phi) - 181.5) \cos(\eta) - 181.5 \cos(\phi) + 1.5 L_0^2 + 363(h^2 + 1) \quad (10)$$

From Eqn. (10), it is observed that the total potential energy of the system has dependencies on the tilt angles, the parameter h and the free length of the spring L_0 . As the tensegrity mechanism has dependencies on both tilt angles, the following steps are carried out for the stability analysis:

- Generation of the Hessian matrix or the mechanism stiffness
- Estimation of the determinant value of the Hessian matrix
- Estimation of the value of the second-order derivative of total potential energy with respect to one of the tilt angles

The Hessian matrix and its determinant are given by:

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 U}{\partial \eta^2} & \frac{\partial^2 U}{\partial \eta \partial \phi} \\ \frac{\partial^2 U}{\partial \eta \partial \phi} & \frac{\partial^2 U}{\partial \phi^2} \end{bmatrix} = \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix} \quad (11)$$

$$\det(\mathbf{H}) = f_{11}f_{22} - f_{12}^2 \quad (12)$$

As the total potential energy is a function of the two pose variables, the following conditions are possible [23]:

1. The total potential energy has a relative maximum when $\det(\mathbf{H}) > 0$ and f_{11} (resp. f_{22}) < 0
2. The total potential energy has a relative minimum when $\det(\mathbf{H}) > 0$ and f_{11} (resp. f_{22}) > 0

Thus, to have a stable configuration the second condition must be satisfied. The optimal design parameters of the tensegrity mechanism are identified at the home-pose where the tilt angles are zero. The objective function of the first optimization will aim to maximize the second-order derivative of the potential energy with respect to one of the tilt angles and it is given by:

$$fun_1 = f_{11} = 77 (L_0 h^2 - 22 h^2 + 11) \quad (13)$$

This function is subject to two inequality constraints, which ensure that the determinant value of the Hessian matrix and the value of the objective function remains a positive value during each iteration. The constraint equations are given by:

$$\det(\mathbf{H}) = (16.5 L_0 h - 363 h^3 + 181.5 h)^2 \quad (14)$$

$$g_1 : \det(\mathbf{H}) \geq 0, \quad g_2 : fun_1 \geq 0 \quad (15)$$

Problem statement and results of optimization

Using the objective function and the constraint equations, the first optimization problem can be stated as:

$$\begin{aligned} &\text{Maximize: } fun_1(\mathbf{x}) \\ &\text{subject to the constraints: } g_1, g_2 \\ &\text{where } \mathbf{x} = [h, L_0]^T \end{aligned}$$

The optimization problem can be solved using a gradient-descent approach. However, this approach leads to only one solution based on the bounds set for the problem. In order to have multiple solutions distributed over a domain, a metaheuristic approach [24] using Genetic Algorithm is carried out in MATLAB with the help of Eqn. (13) and Eqn. (15). The lower and upper bounds for h are set between 0.4 and 1 while the bounds for the free length of the spring are set as 4 mm and 10 mm respectively. The results of solutions obtained by Genetic Algorithm are represented by crossed points on the objective function and it is shown in Fig. 4a. From the re-

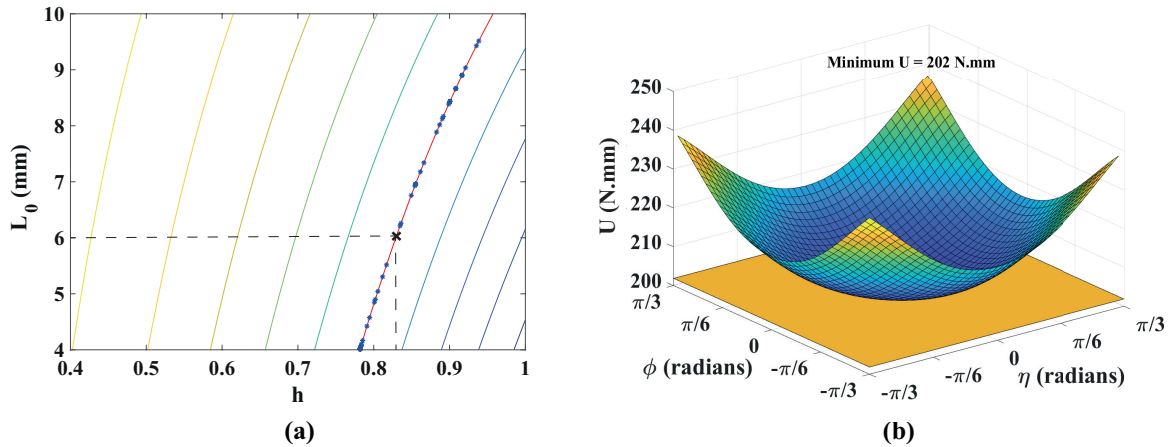


Fig. 4. (a) Solutions for optimization problem obtained by Genetic Algorithm and (b) Plot of total potential energy versus the tilt angles η and ϕ for $[h, L_0] = [0.83, 6 \text{ mm}]$ (flat surface) and $[0.8, 6 \text{ mm}]$ (inverted bell-curve) at the home-pose of the tensegrity mechanism

sults, a value of $L_0 = 6 \text{ mm}$ is chosen as it corresponds to the free length of the spring available at LS2N. For this value of L_0 , the corresponding value of h from Fig. 4a is 0.83. This solution is represented by a black cross in Fig. 4a. At $[h, L_0] = [0.83, 6]$, the value of the objective function f_{11} is close to zero. In other words, it can be said that for these values, a flat plane is obtained for the

total potential energy against the tilt angles and it is represented by the flat surface in Fig. 4b. The mechanism is in a stable configuration if h is less than 0.83, and it is in an unstable configuration if h is greater than 0.83 for $L_0 = 6$ mm. Thus, in order to have a stable configuration, a value of $h = 0.8$ is considered for $L_0 = 6$ mm. Under these design parameters, the total potential energy of the mechanism is plotted against the tilt angles at the home-pose and it is represented in Fig. 4b. It could be observed that under the home-pose, a stable configuration of the mechanism is obtained. Using these results of optimization, the singularity and workspace analysis of the tensegrity mechanism is carried out using the Cylindrical Algebraic Decomposition (CAD) in Maple as investigated in [19]. Under passive modes, tilt limits of $\pm\pi/4$ radians were obtained for the mechanism within the singularity free workspace. As this mechanism will be coupled along with the piping inspection robot to pass through pipe bends, at any instance of locomotion, the orientation of mechanism about the z-axis is unknown. Thus, for further analysis of the robot assembly inside the test bench, the extreme tilt posture of the tensegrity mechanism will be considered. A configuration similar to Fig. 3c is considered where the mounting distance between two springs is given by $2r_f$. In this posture, when one of the tilt angles reaches a maximum of $\pm\pi/4$ radians with the other tilt angle at 0 radians, one of the four prismatic springs reaches a limit of 7.8 mm, while the opposing spring reaches 24.6 mm. The remaining two springs have the same values of 16.3 mm. This orientation of the mechanism will be considered for further analysis to determine the optimal sizing of the robot modules. The 2D representation of the tensegrity mechanism at this condition is shown in Fig. 5.

SECOND OPTIMIZATION PROBLEM: DETERMINATION OF MOTOR SIZING

The second optimization problem aims to determine the sizing of motor modules of the inspection robot inside the test bench without the presence of leg mechanisms. The optimization problem is tested for pipeline diameters ranging from 70 to 160 mm. From the results of optimization, a suitable motor-spindle combination unit is identified from the catalogue of Maxon. Apart from the motor sizing, the existing bio-inspired robot has issues associated with velocities and cable management. Based on results obtained from the second optimization problem, the Maxon motors are

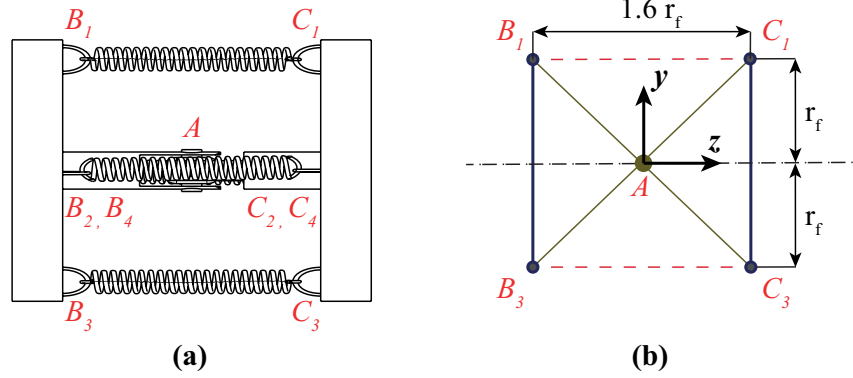


Fig. 5. Representation of the (a) posture considered for the 4-SPS-U mechanism in optimization problem and (b) associated design parameters of the mechanism

identified in such a way that the following factors are also addressed:

1. Velocity: An alternate spindle drive unit with a lower gear reduction ratio compared to the existing prototype (455:1) to accomplish faster displacements during locomotion
2. Hall sensors: A DC-motor coupled with Hall sensor units which will be useful for the control phase of the robot while working inside an unknown environment [25]

Once the optimal motor unit is identified, the third problem is carried out for the results of the second optimization problem to determine the size of the slot-follower leg mechanism.

Modeling of robot and design variables

The robot is modeled as multi-body planar blocks with a tensegrity mechanism between each module in MATLAB. Each module comprises a Maxon brushless DC-motor coupled with a spindle drive. The digital model of a module is shown in Fig. 6a. The robot geometry considered for the optimization problem is represented in Fig. 6b and each module is constructed using the geometrical equations of a rectangle. In Fig. 6b, the design variables for optimization are considered in the following way:

- $[l_{k1}, w_{k1}]$: The length and width of motor and spindle drive unit (without the screw) for module k
- $[l_{k2}, w_{k2}]$: The length and width of output screw drive of spindle unit for module k

Here k indicates a module and it assumes values from 1 to 3. The spindle screw length is one

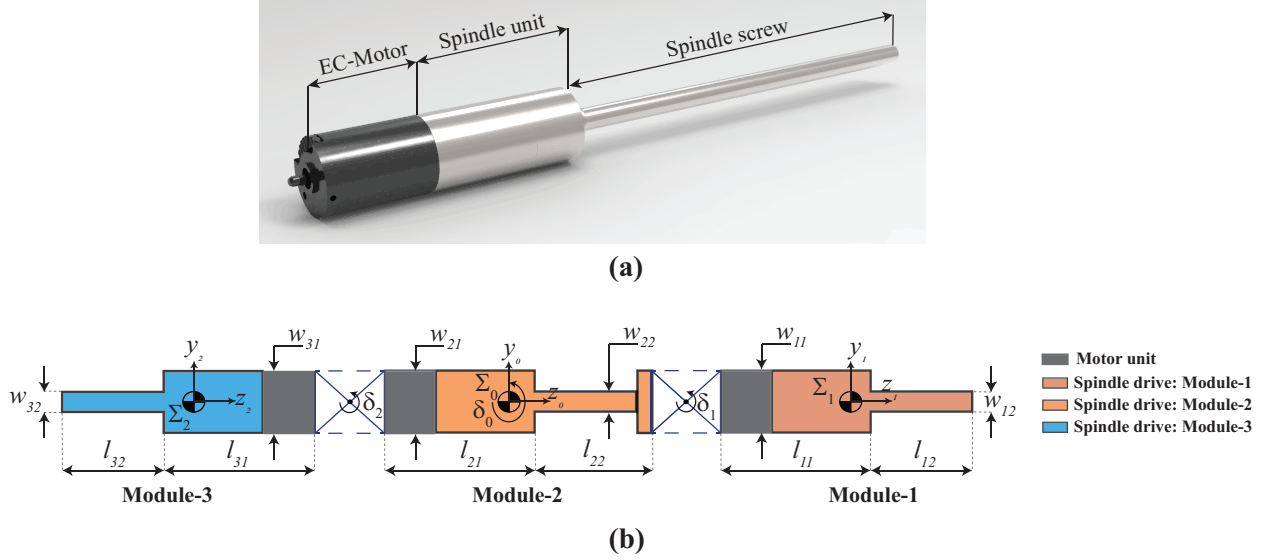


Fig. 6. Representation of the (a) Maxon brushless DC-motor and its spindle drive and (b) robot assembly with various design parameters for the second optimization problem

of the main factors that hinder the existing bio-inspired robot to overcome a pipe bend. This is the reason why the screw unit and motor-spindle unit are considered as separate design variables. Also, it is possible to have spindle drive units with reduced screw length apart from standard sizes at Maxon [26]. The dimensions of the tensegrity mechanism are taken from the results of the first optimization problem. The fixed design variables of the tensegrity mechanism are given by $[1.6r_f, 2r_f]$ as shown in Fig. 5b with $r_f = 11$ mm. In Fig. 6b, the reference frame of the robot Σ_0 is fixed at the center of gravity (CG) position of the central module. The local coordinate frames of Module-1 and Module-3 are given by Σ_1 and Σ_2 . The rotation of modules about the x-axis are given by δ_1 , δ_0 and δ_3 for Modules-1,2,3. The central module is rotated about Σ_0 whereas, for the leg modules, the rotation is carried out about the tensegrity mechanisms. The movement of the robot is simulated in MATLAB by displacing the CG of the reference frame to each discretized point on the CLR. For any position of the robot on a given discretized point on the CLR, the rotation angle is applied to each module as well as to the tensegrity mechanisms to accomplish the rotation while passing through pipe bends. The rotation of modules during locomotion inside the test bench is accomplished in MATLAB as described in Algorithm 1.

Algorithm 1 Rotation of modules at a given position of CLR

Inputs: Discretized CLR coordinates, Robot design parameters

Outputs: Rotation angles of modules ($\delta_0, \delta_1, \delta_2$)

```

1: Initialization
2: for  $i = 1 - n$  do                                 $\triangleright n$  indicates the end point of CLR
3:   Placing reference frame  $\sum_0$  at point  $i$  of CLR
4:   Extract the coordinates ( $z_i$  &  $y_i$ ) of CLR point  $i$  coinciding with frame  $\sum_0$ 
5:   Apply rotation angle  $\delta_0$  to Module-2 by  $\delta_0 = \arctan\left(\frac{y_i}{z_i}\right)$ 
6:   Extract the coordinates of CLR point  $i$  closer to frame  $\sum_1$ .
7:   Calculate rotation angle  $\delta_1$  from the coordinates of point extracted in Step-6
8:   Apply rotation angle  $\delta_1$  to tensegrity mechanism coupled between Modules-1 and 2.
9:   Extract the coordinates of CLR point  $i$  closer to frame  $\sum_2$ .
10:  Calculate rotation angle  $\delta_2$  from the coordinates of point extracted in Step-9
11:  Apply rotation angle  $\delta_2$  to tensegrity mechanism coupled between Modules-2 and 3.
12: end for

```

Objective function

A mono-objective optimization problem subject to constraints is solved using MATLAB. The objective function of the problem aims at maximizing the area of the motor units. Since the dimensions of the tensegrity mechanism are fixed, their areas are not taken into account. As the central module takes care of elongation and retraction phases, it is assumed for the computation that this module remains in a fully extended phase at all instances of the simulation. The area of the robot at a given position of CLR is given by:

$$fun_2 = \sum_{k=1}^3 (l_{k1}w_{k1} + l_{k2}w_{k2}) , \text{ where } k \text{ indicates the module} \quad (16)$$

The area of motor units is estimated as per Eqn. (16). The global sum of fun_2 from Eqn. (16) is calculated throughout the discretized points of the CLR and this will be maximized.

Constraint equations

For each position of the robot on the CLR, the collision of modules is checked against the pipeline walls. Inequality constraints are defined in the optimization problem, which will ensure

that the modules avoid collision during movement inside the test bench, especially at the bends. A complete discretization of each module is performed. The discretization equation is similar to Eqn. (1) and Eqn. (2). The discretized robot modules inside the test bench are represented below in Fig. 7a.

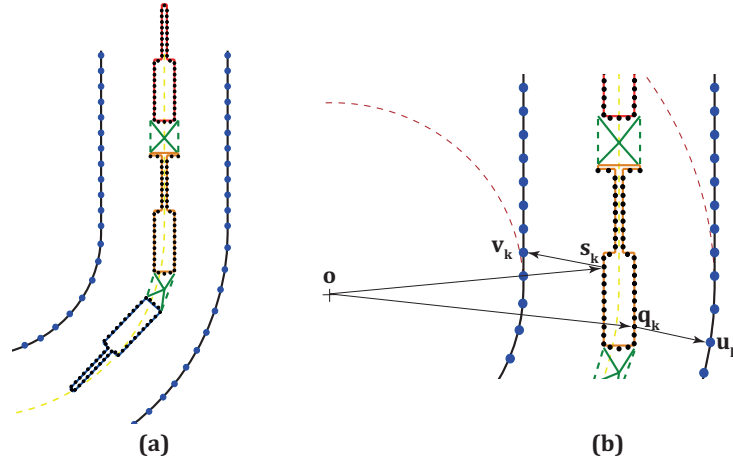


Fig. 7. Representation of the (a) discretized robot assembly and (b) extraction of coordinates from the discretized model for defining constraints

For each position of reference frame $\sum_0$ on the discretized CLR trajectory, the coordinates of points closer and farther on each module from $\mathbf{o}$ are extracted. These points can be extracted using the *min* and *max* function of MATLAB. The representation of the coordinates closer ($\mathbf{s}_k$) and farther ($\mathbf{q}_k$) from $\mathbf{o}$ on a module is shown in Fig. 7b. Followed by that, the coordinate $\mathbf{v}_k$ on the inner portion closer to $\mathbf{s}_k$ and the coordinate $\mathbf{u}_k$ on the outer portion closer to $\mathbf{q}_k$ are extracted using the *min* function. With the extracted set of coordinates, the inequality constraints are defined which ensures that there exist no collisions between the modules and the pipeline walls. The constraint equations for the position of the robot assembly on a discretized CLR point are thus defined by:

$$g_k : \|\mathbf{o} - \mathbf{q}_k\| \leq \|\mathbf{o} - \mathbf{u}_k\| \quad (17)$$

$$g_{k+1} : \|\mathbf{o} - \mathbf{s}_k\| \geq \|\mathbf{o} - \mathbf{v}_k\|, \text{ where } k = 1, 2, 3 \quad (18)$$

Throughout the locomotion sequence, the collision of each module against pipeline walls is checked using Eqn. (17) and Eqn. (18). The pseudo-code for the constraint function and its working in MATLAB is provided in Algorithm 2.

Algorithm 2 Definition of constraint equations for the second optimization problem

Inputs: Discretized CLR coordinates, Robot design parameters

Outputs: Inequality constraint equations ($g_1, g_2, g_3, g_4, g_5, g_6$)

```

1: Initialization
2: for  $i = 1 - n$  do                                ▷  $n$  indicates the end point of CLR
3:   for  $k = 1 - 3$  do                                ▷  $k$  indicates the module number
4:     for  $j = 0 : 0.01 : 1$  do                        ▷ Discretization
5:       Discretize each face of each module
6:     end for
7:     Extract the coordinate ( $\mathbf{s}_{ik}$ ) on module  $k$  closer to  $\mathbf{o}$  using min
8:     Extract the coordinate ( $\mathbf{q}_{ik}$ ) on module  $k$  farther to  $\mathbf{o}$  using max
9:     for  $l = 1 - n$  do                                ▷ For each discretized point on inner & outer portions
10:      Calculate  $\|\mathbf{u}_l - \mathbf{q}_{ik}\|$ 
11:      Calculate  $\|\mathbf{v}_l - \mathbf{s}_{ik}\|$ 
12:    end for
13:    Extract the coordinate from  $\mathbf{u}_{ik}$  on outer portion closer to  $\mathbf{q}_{ik}$ 
14:    Extract the coordinate from  $\mathbf{v}_{ik}$  on inner portion closer to  $\mathbf{s}_{ik}$ 
15:    Check inequality constraints ( $g_k$ ):  $\|\mathbf{o} - \mathbf{q}_{ik}\| \leq \|\mathbf{o} - \mathbf{u}_{ik}\|$ 
16:    Check inequality constraints ( $g_{k+1}$ ):  $\|\mathbf{o} - \mathbf{s}_{ik}\| \geq \|\mathbf{o} - \mathbf{v}_{ik}\|$ 
17:  end for
18: end for

```

Problem statement

With the objective function and constraint equations being defined, the second optimization problem can be stated as:

$$\begin{aligned}
 &\text{maximize: } \sum_{i=1}^n fun_2(\mathbf{y}) \\
 &\text{subject to the constraints: } g_1, g_2, g_3, g_4, g_5, g_6
 \end{aligned}$$

where $\mathbf{y} = [l_1, w_1, l_2, w_2]^T$, $i = 1..n$ indicates the discretized CLR points

The design variables for each module which were considered as $[l_{k1}, w_{k1}, l_{k2}, w_{k2}]$, resulting in 12 variables are reduced to 4 variables which are given by $[l_1, w_1, l_2, w_2]$. This is because an identical Maxon motor-spindle drive unit will be employed in the three modules. The objective function and

the constraint equations are solved using the $fmincon$ function in MATLAB. The design variables are subject to lower and upper bounds to have a closer interpretation of the catalogue parts of Maxon. The existing prototype uses a brushless DC-motor of diameter 16 mm coupled with a spindle drive GP 16 S. The overall length of the motor-spindle unit is 58 mm with a screw length of 102 mm. The advantage of using brushless motors is that they offer Hall sensors that can be useful in the control phase of the robot to determine pipeline diameters inside an unknown/closed environment [25]. The lower (lb) and upper bounds (ub) for the design variables of the optimization problem are set as follows:

1. Parameter l_1 : This parameter corresponds to the length of the motor-spindle unit employed in each module. The lower bound is set at 40 mm and the upper bound is set at 60 mm.
2. Parameter w_1 : This parameter corresponds to the diameter of the motor-spindle unit. Since Maxon offers DC-Motors with Hall sensors starting from 16 mm, the lower bound is set at 10 mm while the upper bound is fixed at 16 mm. Values higher than 16 mm series are not considered as this may affect the length parameters of the motor-spindle drive or the screw length.
3. Parameter l_2 : This corresponds to the length of the screw coming from the spindle drive. This is an essential parameter as it not only plays a vital role in avoiding collisions but also accommodates the slot-follower leg mechanism. The lower bound is set at 20 mm while the upper bound is set at 102 mm which is the maximum length offered by Maxon spindle drives.
4. Parameter w_2 : This corresponds to the diameter of the screw. The lower bound is set at 2 mm and the upper bound is set at 5 mm inline with the catalogue of Maxon [26].

Thus, the lower and upper bounds for the second optimization problem are given by $lb = [40, 10, 20, 2]$ and $ub = [60, 16, 102, 5]$. Spindle drives of Maxon offer lead screws and ball screws. The existing prototype uses a ball screw type as it has a higher load carrying capacities and this could be useful when the leg mechanisms are coupled along with the drive. The optimization algorithm is executed with a constant value of diameter throughout the section depicted in Fig. 2a. The algorithm will be evaluated one after the other for pipe diameters starting from 70 mm to 160 mm. For faster convergence of the problem and also to satisfy bounds at all iterations, the Sequential

Quadratic Programming (SQP) algorithm [27] is employed within $fmincon$. The tolerance value of the guess variables and the objective function is set at 10^{-9} for the optimization problem.

Results and discussions

As the definition of objective function and constraint equations appear simpler, a faster convergence is obtained. The dimensions of motor units obtained from the optimization algorithm for the various pipe diameters are provided below in Table 2.

Table 2. Results obtained for the second optimization problem in MATLAB

Pipe diameter d	l_1	w_1	l_2	w_2
(mm)	(mm)	(mm)	(mm)	(mm)
70	41.4	12.7	20	2
80	47.44	14	24.7	2
90	50.95	16	32	5
100	52.72	16	45.17	5
110	55.02	16	58.25	5
120	58.21	16	67.33	5
130	60	16	84.4	5
140	60	16	86.54	5
150	60	16	98.8	5
160	60	16	102	5

From the results of the optimization provided in Table 2, a gradual increase in all design parameters could be observed with the increase in pipe diameter. During the simulation of the robot in MATLAB, the minimum and maximum lengths of the tensegrity mechanism were found to be 11.5 mm and 17.6 mm. These limits are well within the estimated numerical values of the tensegrity mechanism at $\pm\pi/4$ radians. Thus, a stacked model is not necessary for performing the simulations with a modified design of the robot assembly. Based on the results obtained in Table 2, a list of possible motors that match with the dimensions are identified from the catalogue of Maxon and it is provided in Table 3. The results obtained for 70–80 mm diameter pipe ranges provide the least possible motor sizing. The nearest catalogue part for this series is the RE 8 DC-Motor

Table 3. Identification of motor-spindle series from catalogue of Maxon [26] based on results of optimization where OV indicates over-sizing, OP indicates optimal sizing and UN indicates under-sizing

Type	Series	l_1 (mm)	w_1 (mm)	l_2 (mm)	w_2 (mm)	D70	D80	D90	D100	D110	D120	D130	D140	D150	D160
Motor	463223	31.9	8	20	3	OP	OP	UN	UN	UN	UN	UN	UN	UN	UN
Spindle	473645														
Motor	320178	50.4	16	32	5	OV	OV	OP	OP	OP	UN	UN	UN	UN	UN
Spindle	424731														
Motor	283828	46.4	16	32	5	OV	OV	OP	OP	OP	UN	UN	UN	UN	UN
Spindle	424731														
Motor	283828	51.5	16	45	5	OV	OV	OV	OP	OP	OP	UN	UN	UN	UN
Spindle	424744														
Motor	283828	62.4	16	70	5	OV	OV	OV	OV	OV	OP	OP	OP	OP	UN
Spindle	424745														

(463223) coupled with an 8 mm diameter spindle drive (473645) with a maximum screw length of 56 mm. This combination is not compatible with the other diametrical ranges as they prove to be undersized. Also, this DC-Motor does not have Hall sensors and they require an additional encoder for having a position control algorithm. Apart from that the efficiencies and feed force factors for this series are significantly low [26] and this might be an issue when the leg mechanisms are accommodated. Thus, the results of 70–80 mm diameter pipelines are not considered for further analysis. For the 90–120 mm diameter range pipelines, two possible motor-spindle combinations are identified. The first solution is the combination of 320178 RE 16 DC-Motor coupled with a 424731 16 mm diameter spindle drive. Another combination uses the 283828 EC-Motor coupled along with the 424731 spindle series. The first solution offers a DC-Motor with a higher velocity however, the disadvantage of employing the 320178 series motor is that it requires an additional encoder if a position control algorithm is incorporated. It has to be noted that the screw length obtained for the 90 mm diameter pipe is 32 mm. This length might not be sufficient enough to accommodate the slot-follower leg mechanism, however, this can only be verified in the third optimization problem. For the 100–120 mm diameter pipeline, the 283828 EC-Motor coupled with 424744 spindle drive is identified. A screw length of 45 mm is chosen for this combination. This combination, however, appears to be undersized for the 130–160 mm diameter pipes. Another combination that uses the 283828 EC-motor along with the 424745 spindle drive with a screw length of 70 mm is identified, which is optimal for the 120–150 mm diameter pipes. From the list provided in Table 3, the last three motor-spindle combinations can be potentially accommodated

for 90–160 mm diameter pipes provided the screw length is modified. For the 160 mm diameter pipe, all the combinations prove to be undersized. However, a motor-spindle combination with 90 mm screw length can be compatible for 150–160 mm diameter pipes. In order to have an optimal result that can be suitable for a smaller diameter pipeline ranges, the results of 100–120 mm range is considered. The screw length is thus fixed at 45 mm. The last three solutions from Table 3 are ideal for this diameter range. The feed force, gear ratio and efficiencies of these motor-spindle combinations are provided below in Table 4.

Table 4. Technical specifications of motors from Table 3 for the results of 100 to 160 mm diameter pipes

Type	Series	Gear ratio	Feed force	Efficiency	Length (Motor+Spindle)
			(N)	(%)	(mm)
Motor	283828	5.4:1	189	87	46.4
Spindle	424731				
Motor	283828	29:1	331	79	51.5
Spindle	424744				
Motor	283828	104:1	403	71	55.1
Spindle	424745				

The EC-motor series 283828 is chosen for all these combinations. The 424731 spindle drive offers the highest velocity. However, the feed force offered by this series is low. This might pose a problem during static phases, especially when the leg mechanisms are coupled with the screw drive. The 424744 and 424745 series, on the other hand, offers better feed forces. The 424745 series offers a feed force similar to the 424749 series which is used in the existing prototype. The 424744 series, on the other hand, offers a lesser feed force but this factor will be sufficient to accommodate the leg mechanisms based on experiments conducted on the existing prototype

[9]. Also, the velocity offered by 424744 series is higher than 424745. This will address the issue of low velocity that is associated with the existing prototype [11]. Taking into account the length, efficiency and velocity factors, the 283282 EC-motor coupled with 424744 spindle drive within a working diameter range of 100–120 mm is considered for further analysis. For ensuring a simpler procedure for the control sequence of the prototype, the same motor-spindle combination is chosen for all three modules.

THIRD OPTIMIZATION PROBLEM: DESIGN OF THE SLOT-FOLLOWER LEG MECHANISM

With the optimal motor sizing determined from the second optimization problem, the next step is to determine the geometry of leg mechanisms that could be accommodated within the screw length. The ratio of screw length to the pipe diameter has a greater influence on the geometry of the leg mechanisms. The geometry of the slot-follower leg mechanism and its assembly on the existing robot is represented in Fig. 8 ². By using the inverse kinematic model, the dimensions

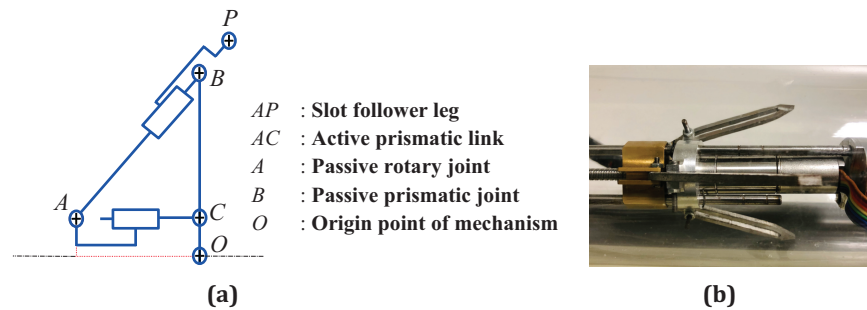


Fig. 8. Representation of the (a) geometry of slot-follower leg mechanism and (b) assembly of the mechanism on the existing prototype [9, 10]

for the leg mechanism were identified in [10] by a multi-objective optimization approach. In this article, a simple mono-objective optimization problem is solved wherein the dimensions of the leg mechanisms are identified through a geometrical approach. The maximization of the area occupied by the space of leg mechanism is carried out for determining the leg length that can work inside 100–120 mm diameter pipelines.

²Digital model simulation of the leg mechanism on the existing rigid prototype in CATIA : [Click here](#)

Modeling of the leg mechanisms and design variables

The front and rear module assembly comprise three legs assembled at 120° with respect to each other [9]. These legs ensure tight contact during static and dynamic phases of the robot. For a given locomotion instance, the orientation of the robot about the z-axis is unknown. As the optimization problem is solved within a planar test bench, two-dimensional projections of the leg mechanisms at two extremities are possible and these projections are represented in Fig. 9. In the

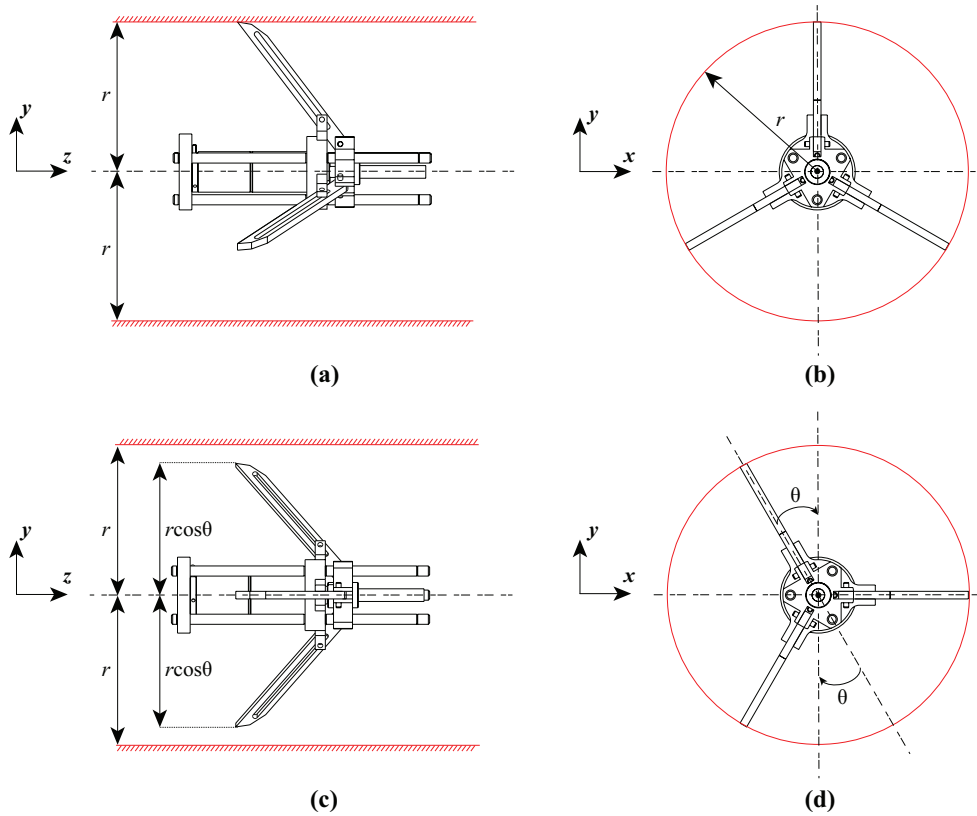


Fig. 9. Representation of the front and side views of the two possible orientations of the leg mechanism at the extremities caused by the rotation of the robot about z-axis

first possible extremity, the 2D projection of the robot yields a view where the distance between the central axis of the robot and the contact point of one of the legs is equal to the radius r of the pipeline. This representation is shown in Fig. 9a and Fig. 9b. The second possible extremity occurs when the 2D projection of the robot yields a view where two sets of legs establish contact

with the walls of the pipeline. In this scenario, the distance between the central axis of the robot and the contact point with the pipeline wall is given by $r \cos(\theta)$, where θ is the angle between the leg and $y - z$ plane. This representation is shown in Fig. 9c and Fig. 9d. For the third optimization problem, a geometry similar to Fig. 9c is considered such that collision will be verified on either side against the pipeline walls. For the computation, it is considered that $r \cos(\theta)$ is equal to r . The geometry of the leg mechanisms that will be assembled on the robot is represented below in Fig. 10. For the third optimization problem, the central module dimensions are carried over

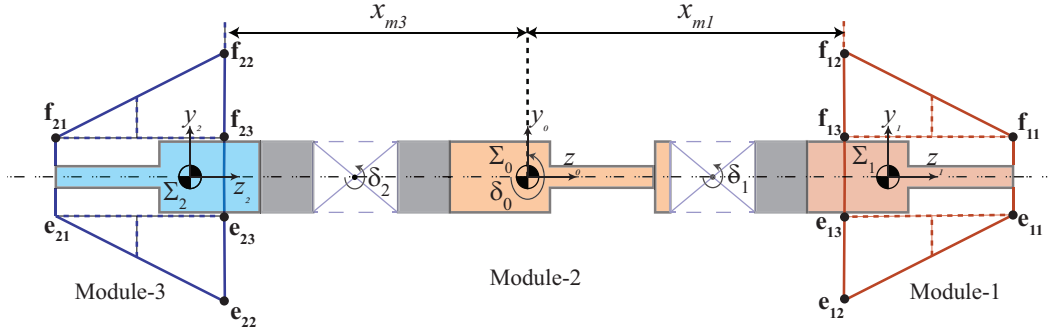


Fig. 10. Representation of the geometry of the slot-follower leg mechanism on the motor modules

from the results of the second optimization problem and are considered fixed. This is because the central module has no leg masses attached to them. The reference frame for the robot is fixed at Σ_0 in the central module. The tip of each leg mechanism is referenced from the frame Σ_0 through a distance x_{m1} for Module-1 and x_{m3} for Module-3. The design parameters of the slot-follower leg mechanism during declamped and clamped phases are represented below in Fig. 11a and Fig. 11b. The parameters of the leg mechanism from Fig. 11a are the lengths l_{s1} , l_{s2} , the offsets o_1 , o_2 , the stroke lengths ρ_1 , ρ_2 and the horizontal and vertical space vectors $\Delta \mathbf{x}$ and $\Delta \mathbf{y}$ [9, 10]. The horizontal and vertical space vectors of the leg mechanism are given by:

$$\Delta \mathbf{x} = [\Delta x_1, \Delta x_2]^T, \Delta \mathbf{y} = [\Delta y_1, \Delta y_2]^T \quad (19)$$

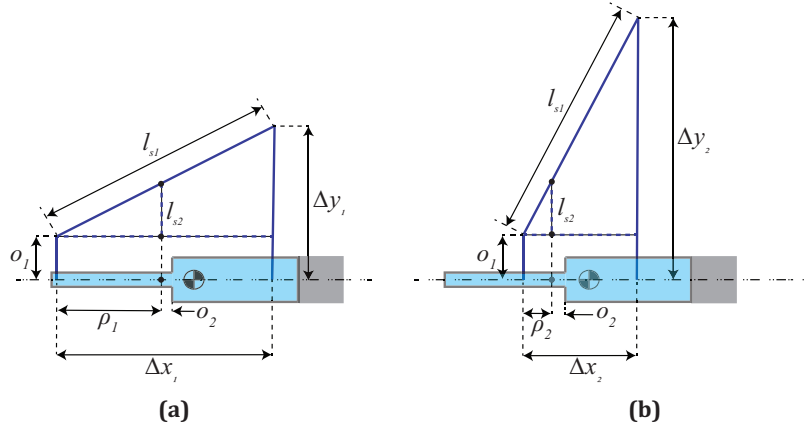


Fig. 11. Representation of the design parameters of the leg mechanism during (a) declamped and (b) clamped phases

During locomotion of robot inside the pipeline, at least one set of leg mechanisms establish contact with pipeline walls. However, this sequence will be time-consuming to construct in MATLAB. In order to have faster convergence and a simpler model, it is assumed that at all positions of CLR, the leg mechanisms in both the modules remain in the declamped phase during the simulation. The design variables for the optimization problem are given by $[\Delta x_1, \Delta y_1, l_{s2}]$. The offsets o_1 and o_2 are retained as 11 mm and 7 mm, from the existing prototype as these values correspond to offsets caused by the EC-Motor unit (o_1) and spacers (o_2) used for assembly of the leg mechanisms. The parameter l_{s2} represents the vertical distance of the hinge point of the leg mechanism. This parameter plays an essential role to determine the leg length l_{s1} for a given diameter range of pipelines.

Objective function

The objective function of the third optimization problem aims to maximize the space occupied by the leg mechanisms. The geometry can be split into triangular and rectangular areas which is represented in Fig. 10. The area of the leg mechanisms used in the robot at a given discretized CLR point can be calculated by:

$$fun_3 = \sum_{k=1}^2 \left(\frac{\|\mathbf{e}_{k1} - \mathbf{e}_{k3}\| \|\mathbf{e}_{k3} - \mathbf{e}_{k2}\|}{2} + \frac{\|\mathbf{f}_{k1} - \mathbf{f}_{k3}\| \|\mathbf{f}_{k3} - \mathbf{f}_{k2}\|}{2} + \|\mathbf{e}_{k1} - \mathbf{e}_{k3}\| \|\mathbf{e}_{k1} - \mathbf{f}_{k1}\| \right) \quad (20)$$

where $k = 1, 2$ indicates Module-1 and Module-3

As only two modules are being considered, the index k goes from 1 to 2, which corresponds to Modules 1 and 3. Similar to Eqn. (16), the global sum of fun_3 for Eqn. (20) is estimated and it will be maximized.

Constraint equations

Similar to the second optimization problem, the collision of the geometry of leg mechanisms is verified against the pipeline walls at each position of the CLR. A complete discretization of the leg mechanism geometry is performed using an equation similar to Eqn. (1) and Eqn. (2). The representation of the discretized leg mechanism assembled along with the optimized robot modules is shown below in Fig. 12a.

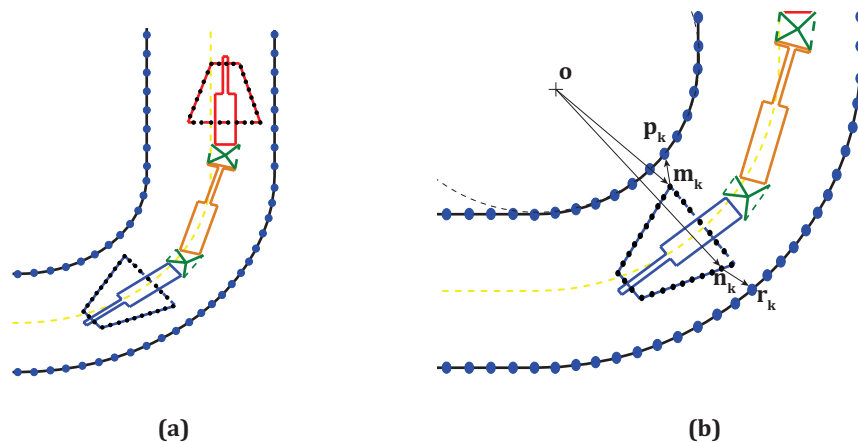


Fig. 12. Representation of the (a) discretized leg mechanism on the motor modules and (b) extraction of coordinates from a discretized leg mechanism for defining the constraints

For a given CLR position, the coordinates of the points closer and farther on each leg mechanism from the point $\mathbf{o}$ are extracted. This is done using the *min* and *max* functions of MATLAB. The representation of the coordinates closer ($\mathbf{m}_k$) and farther ($\mathbf{n}_k$) on the rear leg mechanism from central coordinate $\mathbf{o}$ is shown in Fig. 12b. The coordinates of the point $\mathbf{p}_k$ on the inner portion which is closer to $\mathbf{m}_k$ and the coordinates of the point $\mathbf{r}_k$ on the outer portion which is closer to $\mathbf{n}_k$ are extracted using the *min* function. With these extracted points, four inequality constraints are defined in MATLAB which are given by:

$$g_k : \|\mathbf{o} - \mathbf{n}_k\| \leq \|\mathbf{o} - \mathbf{r}_k\| \quad (21)$$

$$g_{k+1} : \|\mathbf{o} - \mathbf{m}_k\| \geq \|\mathbf{o} - \mathbf{p}_k\|, \text{ where } k = 1, 2 \text{ indicates Module-1 and Module-3} \quad (22)$$

In addition to the inequality constraints, two equality constraints are defined for the third optimization problem. As the simulation is performed for fully declamped phase, the length of leg mechanism l_{s1} can be calculated from Fig. 10 by the Pythagoras theorem and it is given by:

$$l_{s1}(d) = \sqrt{(\|\mathbf{f}_{k1} - \mathbf{f}_{k3}\|)^2 + (\|\mathbf{f}_{k2} - \mathbf{f}_{k3}\|)^2}, \text{ where } k = 1, 2 \text{ indicates Module-1 and Module-3} \quad (23)$$

In Eqn. (23), $l_{s1}(d)$ indicates the leg length during the declamped phase. During the clamped phase, the length l_{s1} can be calculated using Δx_2 , Δy_2 and o_1 by the Pythagoras theorem from Fig. 11b. The equation is given by:

$$l_{s1}(c) = \sqrt{\Delta x_2^2 + (\Delta y_2 - o_1)^2} \quad (24)$$

In Eqn. (24), $l_{s1}(c)$ is the length of the leg during clamped phase. During the initialization sequence of the optimization algorithm, the design variable l_{s2} is calculated and set as a function of the variables Δx_1 , Δy_1 during the declamped phase by similar triangles. From Fig. 11a, the

value of l_{s2} can be calculated by:

$$\frac{l_{s2}}{\rho_1} = \frac{\Delta y_1 - o_1}{\Delta x_1} \quad (25)$$

The dimensions of the motor units from the results of second optimization problem for the 100 mm diameter pipe from Table 3 is given by $[l_1, w_1, l_2, w_2] = [51.5, 16, 45, 5]$ mm. The stroke lengths ρ_1 and ρ_2 are set as 32 mm and 7 mm with reference to the 45 mm screw length of the motor unit. The reduction in stroke length is caused by the flanges and fasteners that will be used in the robot assembly. The two equality constraints for the third optimization problem are thus given by:

$$h_{k1} : \rho_1 = 32 \quad (26)$$

$$h_{k2} : l_{s1}(d) = l_{s1}(c), \text{ where } k = 1, 2 \text{ indicates Module-1 and Module-3} \quad (27)$$

The pseudo-code of the constraint function for the third optimization problem in MATLAB is provided in Algorithm 3.

Problem statement

For the third optimization problem, the diameter d of the pipeline is fixed as 100 mm. The third optimization problem which is used to estimate the size of the leg mechanism is defined by:

$$\begin{aligned} & \text{maximize } \sum_{i=1}^n fun_3(\mathbf{z}) \\ & \text{subject to constraints: } g_1, g_2, g_3, g_4, h_1, h_2 \end{aligned}$$

where $\mathbf{z} = [\Delta x_1, \Delta y_1, l_{s2}]^T$, $i = 1..n$ indicates the discretized CLR positions

The lower and upper bounds of the design parameters for the simulation are given by $lb = [40, 20, 5]$, $ub = [65, 40, 12]$. The upper bounds for the design parameters Δx and Δy are set by considering a factor, which is 60% of the pipeline diameter. A value higher than 60% leads to

Algorithm 3 Definition of constraint equations for the third optimization problem

Inputs: Discretized CLR coordinates, Leg mechanism design parameters, Motor dimensions for 100 mm diameter pipe

Outputs: Inequality constraint equations (g_1, g_2, g_3, g_4) , Equality constraint equations (h_1, h_2)

```

1: Initialization
2: for  $i = 1 - n$  do                                ▷  $n$  indicates the end point of CLR
3:   for  $k = 1 - 2$  do                                ▷  $k$  indicates the module number
4:     for  $j = 0 : 0.01 : 1$  do                        ▷ Discretization
5:       Discretize each face of leg mechanism on each module
6:     end for
7:     Extract the coordinate ( $\mathbf{m}_{ik}$ ) on module  $k$  closer to  $\mathbf{o}$  using min
8:     Extract the coordinate ( $\mathbf{n}_{ik}$ ) on module  $k$  farther to  $\mathbf{o}$  using max
9:     for  $l = 1 - n$  do                                ▷ For each discretized point on inner & outer portions
10:      Calculate  $\|\mathbf{r}_l - \mathbf{n}_{ik}\|$ 
11:      Calculate  $\|\mathbf{p}_l - \mathbf{m}_{ik}\|$ 
12:    end for
13:    Extract discretized coordinate from  $\mathbf{r}_{ik}$  on outer portion closer to  $\mathbf{n}_{ik}$ 
14:    Extract discretized coordinate from  $\mathbf{p}_{ik}$  on inner portion closer to  $\mathbf{m}_{ik}$ 
15:    Check inequality constraints ( $g_k$ ):  $\|\mathbf{o} - \mathbf{n}_{ik}\| \leq \|\mathbf{o} - \mathbf{r}_{ik}\|$ 
16:    Check inequality constraints ( $g_{k+1}$ ):  $\|\mathbf{o} - \mathbf{m}_{ik}\| \geq \|\mathbf{o} - \mathbf{p}_{ik}\|$ 
17:    Calculate  $l_{s1}(d)$  using Eqn. (23) and  $l_{s1}(c)$  using Eqn. (24)
18:    Check equality constraints:  $h_{k1}$  as per Eqn. (26) and  $h_{k2}$  as per Eqn. (27)
19:  end for
20: end for

```

convergence issues for a screw length of 45 mm. The bounds for the design parameter l_{s2} are set in line with the dimension of the existing prototype [11, 20]. Using the *fmincon* function with the SQP algorithm, the third optimization problem is solved to determine the geometry of the leg mechanism.

Results and discussions

The dimensions of the slot-follower leg mechanism obtained after optimization are provided below in Table 5. Compared to the existing prototype, the parameters l_{s1} and l_{s2} are increased from 57 mm and 7 mm to 62.7 mm and 9.7 mm respectively. In the event of a straight pipeline with no bends, the slot-follower leg mechanism is capable of adjusting its size between 58.6 mm ($2\Delta y_1$) to 124 mm ($2\Delta y_2$) diameter range. When the robot is moving inside a pipeline with bends, the modules are capable of passing through such bends, provided the diameter of pipelines is strictly between 100 to 120 mm. The video link for the simulation of the robot inside 100–120 mm

Table 5. Results obtained for the third optimization problem in MATLAB

Parameters	Dimensions (mm)
Δx_1	60
Δy_1	29.3
Δy_2	62
l_{s1}	62.7
l_{s2}	9.7

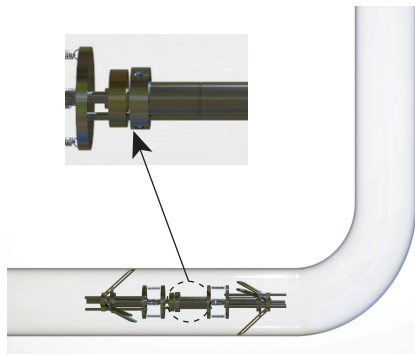
diameter pipelines in MATLAB is provided at the bottom of this page³. By assembling the tensegrity mechanism, the EC-motor with spindle drive unit identified from the catalogue of Maxon, the optimized slot-follower leg mechanism and standard fasteners such as circlips, bolts, etc., the entire robot is realized in CAD software. The positions of the robot inside a 100 mm diameter during its locomotion are rendered in CATIA and the sequences are represented in Fig. 13a to Fig. 13e. The flexible robot resembles an “Elephant trunk” during a passive mode and it is demonstrated in Fig. 13f.

CONCLUSIONS

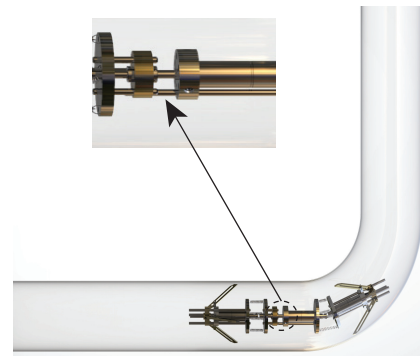
This article mainly focused on the design of a flexible piping inspection robot by an optimization approach. Three optimization problems were defined and solved to arrive at the design of a flexible robot assembly which can work inside pipelines having straight and bent profiles. The first optimization problem dealt with the determination of design parameters of the tensegrity mechanism by ensuring static stability. As the mechanism was proposed to be tested inside a pipeline having 90° pipe bend, the design parameter h and the free length of the spring L_0 were identified under passive mode by a metaheuristic approach in MATLAB. Followed by that, the second optimization problem was defined and solved by a deterministic approach to identify the sizing of motor modules without the presence of leg mechanisms. This problem was carried out inside a test bench with various pipe diameters. From the results of the second optimization problem, a series of motor-spindle combinations were identified from the catalogue of Maxon motors. Based on the

³Simulation of the robot assembly in MATLAB : [Click here](#).

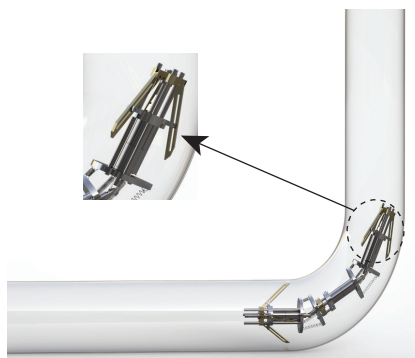
An optimal design of a flexible piping inspection robot



(a) Clamped legs and retracted central module, horizontal position of robot



(b) Clamped legs and extended central module, robot passage into bend section



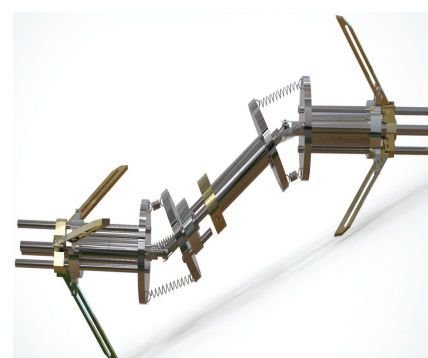
(c) Front leg declamping, rear leg clamping and extended central module, intermediate position in bend



(d) Rear leg declamping, front leg clamping and retracted central module, exit from pipe bend



(e) Clamped legs and retracted central module, vertical position of robot



(f) Resemblance of Elephant trunk posture of the robot

Fig. 13. Locomotion sequence of the optimized bio-inspired robot inside a 100 mm diameter pipe from (a) to (e) and correlation of the robot to an "Elephant trunk" in (f)

results obtained, an intermediate-range that holds good for 100–120 mm diameter pipe was chosen. The selection of the motor-spindle unit was done based on dimensions, velocity, feed force and efficiency factors, which were not considered in the design of the initial prototype [9, 10, 11]. The third optimization problem was then carried out to determine the sizing of the leg mechanisms that could be accommodated for the results of the second optimization problem. The dimensions of the slot-follower leg mechanism were found to be increased when compared to the existing prototype. However, with the modified architecture, the robot is capable of working inside 100–120 mm diameter pipeline range that can have straight sections and bends at 90°. The presence of leg mechanisms and tensegrity mechanism permits to have a flexible “Elephant trunk” model that allows the robot to pass through pipe bends in the event of passive compliance. By assembling the tensegrity mechanisms, motor modules and the leg mechanisms, the 3D model of the flexible robot was realized in CATIA software.

In future works, the optimization algorithm can be modified to accommodate a relation between screw length and pipeline diameter thereby eliminating the need for fixing the pipeline diameters for each computation. This modification might probably lead to the definition of another objective and a metaheuristic approach will be implemented to solve this problem. With the 3D model being assembled using catalog parts, the static force model developed in [9] can be extended to the flexible robot. This model can help in determining the forces on the tensegrity mechanisms and the clamping forces which can be useful for performing experiments. A simulation of the robot will be performed using Adams software to validate the locomotion procedure followed by which prototyping of the flexible robot will be realized. For the real-time application, suitable control units and a camera will be accommodated on the robot to perform an inspection of pipelines. Based on simulation trails, an investigation will also be done to replace the central module with a spindle drive having higher velocity for accomplishing faster displacements.

REFERENCES

- [1] Choi, H. R., and Roh, S. G., 2007. “In-pipe robot with active steering capability for moving inside of pipelines”. In *Bioinspiration and Robotics Walking and Climbing Robots*. IntechOpen.

- [2] Okamoto Jr, J., Adamowski, J. C., Tsuzuki, M. S., Buiochi, F., and Camerini, C. S., 1999. "Autonomous system for oil pipelines inspection". *Mechatronics*, **9**(7), pp. 731–743.
- [3] Okada, T., and Kanade, T., 1987. "A three-wheeled self-adjusting vehicle in a pipe, ferret-1". *The International journal of robotics research*, **6**(4), pp. 60–75.
- [4] Roman, H. T., Pellegrino, B., and Sigrist, W., 1993. "Pipe crawling inspection robots: an overview". *IEEE transactions on energy conversion*, **8**(3), pp. 576–583.
- [5] Ryew, S., Baik, S., Ryu, S., Jung, K. M., Roh, S., and Choi, H. R., 2000. "In-pipe inspection robot system with active steering mechanism". In Proceedings. 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)(Cat. No. 00CH37113), Vol. 3, IEEE, pp. 1652–1657.
- [6] Neubauer, W., 1994. "A spider-like robot that climbs vertically in ducts or pipes". In Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94), Vol. 2, IEEE, pp. 1178–1185.
- [7] Fukuda, T., Hosokai, H., and Uemura, M., 1989. "Rubber gas actuator driven by hydrogen storage alloy for in-pipe inspection mobile robot with flexible structure". In Proceedings, 1989 international conference on robotics and automation, IEEE, pp. 1847–1852.
- [8] Horodinca, M., Doroftei, I., Mignon, E., and Preumont, A., 2002. "A simple architecture for in-pipe inspection robots". In Proc. Int. Colloq. Mobile, Autonomous Systems, pp. 61–64.
- [9] Venkateswaran, S., Chablat, D., and Boyer, F., 2019. "Numerical and experimental validation of the prototype of a bio-inspired piping inspection robot". *Robotics*, **8**(2), p. 32.
- [10] Henry, R., Chablat, D., Porez, M., Boyer, F., and Kanaan, D., 2014. "Multi-objective design optimization of the leg mechanism for a piping inspection robot". In Proceedings of the ASME 2014 IDETC-CIE, Vol. 5A: 38th Mechanisms and Robotics Conference. V05AT08A001.
- [11] Chablat, D., Venkateswaran, S., and Boyer, F., 2019. "Dynamic model of a bio-inspired robot for piping inspection". In *ROMANSY 22—Robot Design, Dynamics and Control*. Springer, pp. 42–51.
- [12] Venkateswaran, S., and Chablat, D., 2019. "A new inspection robot for pipelines with bends and junctions". In IFToMM World Congress on Mechanism and Machine Science, Springer,

pp. 33–42.

- [13] Guessasma, S., and Bassir, H., 2009. “Comparing heuristic and deterministic approaches to optimise mechanical parameters of biopolymer composite materials”. *Mechanics of Advanced Materials and Structures*, **16**(4), pp. 293–299.
- [14] Bennis, F., and Bhattacharjya, R. K., 2020. *Nature-Inspired Methods for Metaheuristics Optimization: Algorithms and Applications in Science and Engineering*, Vol. 16. Springer.
- [15] Rao, R. V., and Waghmare, G., 2017. “A new optimization algorithm for solving complex constrained design optimization problems”. *Engineering Optimization*, **49**(1), pp. 60–83.
- [16] Zhang, Y., Gong, D.-W., and Zhang, J.-H., 2013. “Robot path planning in uncertain environment using multi-objective particle swarm optimization”. *Neurocomputing*, **103**, pp. 172–185.
- [17] Caro, S., Chablat, D., Ur-Rehman, R., and Wenger, P., 2011. “Multiobjective design optimization of 3-prr planar parallel manipulators”. In *Global Product Development*. Springer, pp. 373–383.
- [18] Caro, S., Dumas, C., Garnier, S., and Furet, B., 2013. “Workpiece placement optimization for machining operations with a kuka kr270-2 robot”. In 2013 IEEE International Conference on Robotics and Automation, IEEE, pp. 2921–2926.
- [19] Venkateswaran, S., and Chablat, D., 2020. “Singularity and workspace analysis of 3-sps-u and 4-sps-u tensegrity mechanisms”. In *International Symposium on Advances in Robot Kinematics*, Springer, pp. 226–233.
- [20] Venkateswaran, S., Furet, M., Chablat, D., and Wenger, P., 2019. “Design and analysis of a tensegrity mechanism for a bio-inspired robot”. In *Proceedings of the ASME 2019 IDETC-CIE*, Vol. 5A: 43rd Mechanisms and Robotics Conference. V05AT07A026.
- [21] Li, H., Yang, H., Zhan, M., and Gu, R., 2007. “The interactive effects of wrinkling and other defects in thin-walled tube nc bending process”. *Journal of Materials Processing Technology*, **187**, pp. 502–507.
- [22] Meirovitch, L., 2010. *Fundamentals of vibrations*. Waveland Press.
- [23] Dawkins, P., 2003. “Paul’s online math notes”. URL: <http://tutorial.math.lamar.edu>.
- [24] Yang, X.-S., 2011. “Metaheuristic optimization”. *Scholarpedia*, **6**(8), p. 11472.

- [25] Venkateswaran, S., Chablat, D., and Ramachandran, R., 2019. "Prototyping a piping inspection robot using a beaglebone black board". In 24^{ème} Congrès Français de Mécanique, Brest.
- [26] Maxon Motors, Program 2017/18. High precision Drives and Systems. <http://epaper.maxonmotor.com/>. Accessed: 2019-12-15.
- [27] Choosing the algorithm- MATLAB & Simulink- Mathworks France. <https://fr.mathworks.com/help/optim/ug/choosing-the-algorithm.html>. Accessed: 2020-02-02.

LIST OF FIGURES

1	3D model of the existing bio-inspired robot coupled with the tensegrity mechanisms	6
2	Overview of the (a) test bench for optimization and (b) parametrization of the test bench for $d = 100$ mm	7
3	Representation of the (a) Tensegrity mechanism at home-pose, (b) 3D view of the correlation to a 4-SPS-U manipulator and (c) 2D view of the manipulator	9
4	(a) Solutions for optimization problem obtained by Genetic Algorithm and (b) Plot of total potential energy versus the tilt angles η and ϕ for $[h, L_0] = [0.83, 6 \text{ mm}]$ (flat surface) and $[0.8, 6 \text{ mm}]$ (inverted bell-curve) at the home-pose of the tensegrity mechanism	13
5	Representation of the (a) posture considered for the 4-SPS-U mechanism in optimization problem and (b) associated design parameters of the mechanism	15
6	Representation of the (a) Maxon brushless DC-motor and its spindle drive and (b) robot assembly with various design parameters for the second optimization problem	16
7	Representation of the (a) discretized robot assembly and (b) extraction of coordinates from the discretized model for defining constraints	18
8	Representation of the (a) geometry of slot-follower leg mechanism and (b) assembly of the mechanism on the existing prototype [9, 10]	24
9	Representation of the front and side views of the two possible orientations of the leg mechanism at the extremities caused by the rotation of the robot about z-axis . .	25
10	Representation of the geometry of the slot-follower leg mechanism on the motor modules	26
11	Representation of the design parameters of the leg mechanism during (a) declamped and (b) clamped phases	27
12	Representation of the (a) discretized leg mechanism on the motor modules and (b) extraction of coordinates from a discretized leg mechanism for defining the constraints	28
13	Locomotion sequence of the optimized bio-inspired robot inside a 100 mm diameter pipe from (a) to (e) and correlation of the robot to an “Elephant trunk” in (f)	33

LIST OF TABLES

1	Coordinates system of pipe geometry for Fig. 2a	7
2	Results obtained for the second optimization problem in MATLAB	21
3	Identification of motor-spindle series from catalogue of Maxon [26] based on results of optimization where OV indicates over-sizing, OP indicates optimal sizing and UN indicates under-sizing	22
4	Technical specifications of motors from Table 3 for the results of 100 to 160 mm diameter pipes	23
5	Results obtained for the third optimization problem in MATLAB	32