2

Hyper Redundant Manipulators

Ivanescu Mircea and Cojocaru Dorian
University of Craiova
Romania

1. Introduction

Modern industrial robots are mostly (human) arm-inspired mechanisms with serially arranged discrete links. When it comes to industrial environment where the workspace is structured and predefined this kind of structure is fine. This type of robots are placed in carefully controlled environments and kept away from human and their world.

When it comes to robots that must interact with the natural world, it needs to be able to solve the same problems that animals do. The rigid structure of traditional robots limit their ability to maneuver and in small spaces and congested environments, and to adapt to variations in their environmental contact conditions. For improving the adaptability and versatility of robots, recently there has been interest and research in "soft" robots. In particular, several research groups are investigating robots based on continuous body "continuum" structure. If a robot's body is soft and/or continuously bendable it might emulate a snake or an eel with an undulating locomotion (Walker & Carreras, 2006).

An ideal tentacle manipulator is a non-conventional robotic arm with an infinite mobility. It has the capability of takeing sophisticated shapes and of achieving any position and orientation in a 3D space. Behavior similar to biological trunks, tentacles, or snakes may be exhibited by continuum or hyper-redundant robot manipulators (Walker et al., 2005). Hence these manipulators are extremely dexterous, compliant, and are capable of dynamic adaptive manipulation in unstructured environments, continuum robot manipulators do not have rigid joints unlike traditional rigid-link robot manipulators. The movement of the continuum robot mechanisms is generated by bending continuously along their length to produce a sequence of smooth curves. This contrasts with discrete robot devices, which generate movement at independent joints separated by supporting links.

The snake-arm robots and elephant's trunk robots are also described as continuum robots, although these descriptions are restrictive in their definitions and cannot be applied to all snake-arm robots (Hirose, 1993). A continuum robot is a continuously curving manipulator, much like the arm of an octopus (Cowan & Walker, 2008). An elephant's trunk robot is a good descriptor of a continuum robot (Hutchinson, S.; Hager et al., 1996). The elephant's trunk robot has been generally associated with an arm manipulation – an entire arm used to grasp and manipulate objects, the same way that an elephant would pick up a ball. As the best term for this class of robots has not been agreed upon, this is still an emerging issue. Snake-arm robots are often used in association with another device meant to introduce the snake-arm into the confined space.

However, the development of high-performance control algorithms for these manipulators is quite a challenge, due to their unique design and the high degree of uncertainty in their dynamic models. The great number of parameters, theoretically an infinite one, makes very difficult the use of classical control methods and the conventional transducers for position and orientation." must be moved after the paragraph "An ideal tentacle manipulator is a non-conventional robotic arm with an infinite mobility. It has the capability of takeing sophisticated shapes and of achieving any position and orientation in a 3D space. These systems are also known as hyper redundant manipulators and, over the past several years, there has been a rapid expanding interest in their study and construction.

An ideal tentacle manipulator is a non-conventional robotic arm with an infinite mobility. It has the capability of takeing sophisticated shapes and of achieving any position and orientation in a 3D space. These systems are also known as hyper redundant manipulators and, over the past several years, there has been a rapid expanding interest in their study and construction.

The control of these systems is very complicated and a great number of researchers tried to offer solutions for this difficult problem. In (Hemami, 1984); (Suzumori et al., 1991) it analyses the control by cables or tendons meant to transmit forces to the elements of the arm in order to closely approximate the arm as a truly continuous backbone. Also, Mochiyama has investigated the problem of controlling the shape of an HDOF rigid-link robot with two-degree-of-freedom joints using spatial curves (Mochiyama & Kobayashi, 1999). Important results were obtained by Chirikjian (Chirikjian, 1993) who laid the foundations for the kinematic theory of hyper redundant robots. His results are based on a "backbone curve" that captures the robot's macroscopic geometric features.

The inverse kinematic problem is reduced to determining the time varying backbone curve behaviour (Takegaki & Arimoto, 1981). New methods for determining "optimal" hyper-redundant manipulator configurations based on a continuous formulation of kinematics are developed. In (Gravagne & Walker, 2001), Gravagne analysed the kinematic model of "hyper-redundant" robots, known as "continuum" robots. Robinson and Davies (Robinson & Davies, 1999) present the "state of art" of continuum robots, outline their areas of application and introduce some control issues. The great number of parameters, theoretically an infinite one, makes very difficult the use of classical control methods and the conventional transducers for position and orientation.

The lack of no discrete joints is a serious and difficult issue in the determination of the robot's shape. A solution for this problem is the vision based control of the robot, kinematics and dynamics.

The research group from the Faculty of Automation, Computers and Electronics, University of Craiova, Romania, started working in research field of hyper redundant robots over 25 years ago. The experiments started on a family of TEROB robots which used cables and DC motors. The kinematics and dynamics models, as well as the different control methods developed by the research group were tested on these robots. Starting with 2008, the research group designed a new experimental platform for hyper redundant robots. This new robot is actuated by stepper motors. The rotation of these motors rotates the cables which by correlated screwing and unscrewing of their ends determine their shortening or prolonging, and by consequence, the tentacle curvature (Blessing & Walker, 2004). Segments were cylindrical in the initial prototype, and cone-shaped in actual prototype. The backbone of the tentacle is an elastic cable made out of steel, which sustains the entire structure and allows the bending. Depending on which cable shortens or prolongs, the tentacle bends in different planes, each one making different angles (rotations) respective to the initial coordinate frame attached to the manipulator segment – i.e. allowing the movement in 3D. Due to the mechanical design it can be assumed that the individual cable torsion, respectively entire manipulator torsion can be neglected. Even if these phenomena would appear, the structure control is not based on the stepper motors angles, but on the information given by the robotic vision system which is able to offer the real spatial positions and orientations of the tentacle segments.

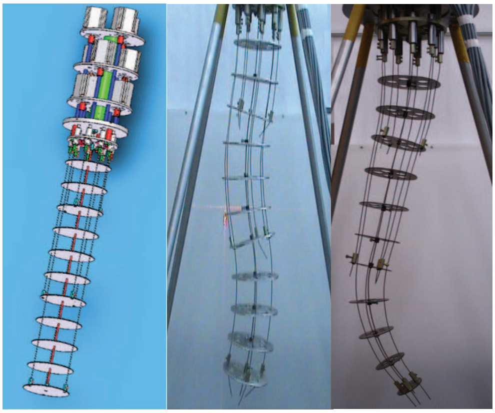


Fig. 1. A tentacle arm prototype

2. Kinematics

In order to control a hyper-redundant robot it has to develop a method to compute the positions for each one of his segments (Immega & Antonelli, 1995). By consequence, given a desired curvature $S^*(x, t_f)$ as sequence of semi circles, identify how to move the structure, to obtain $s(x, t)$ such that

$$\lim_{t\to t_f} s(x,t) = S^*(x,t_f) \tag{1}$$

where x is the column vector of the shape description and t_f is the final time (see Fig. 2).

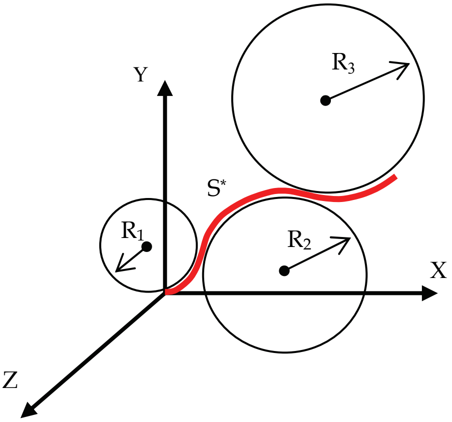


Fig. 2. The description of the desired shape

To describe the tentacle's shape we will consider two angles *(α, θ)* for each segment, where θ is the rotation angle around Z-axis and α is the rotation angle around the Y-axis (see Fig. 2). To describe the movement we can use the roto-translation matrix considering $\theta = 2\beta$ as shown in Fig. 3.

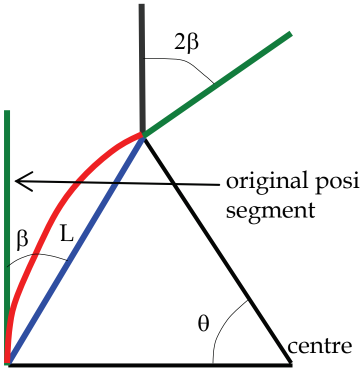


Fig. 3. Curvature and relation between θ and β

The generic matrix in 2D that expresses the coordinate of the next segment related to the previous reference system can be written as follow:

$$\begin{bmatrix} cos(2\cdot\beta) & sin(2\cdot\beta) & L\cdot sin(\beta) \\ -sin(2\cdot\beta) & cos(2\cdot\beta) & L\cdot cos(\beta) \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

In 3D space we cannot write immediately the dependence that exists between two segments. This relation can be obtained through the pre-multiplication of generic roto-translation matrix. One of the possible combinations to express the coordinate of the next segment related to the frame coordinate of the previous segment is the following:

$$R^{i}_{\ generic} := R^{i}_{z}(\theta^{i})\cdot Tr_{y}(V^{i})\cdot R^{i}_{y}(\alpha^{i})\cdot R^{i}_{z}(\theta^{i}) \tag{3}$$

where $R^{i}_{z}(\theta^{i})$ and $R^{i}_{y}(\alpha^{i})$ are the fundamental roto-translation matrix having 4x4 elements in 3-D space, and $Tr_y(V^i)$ is a 4x4 elements matrix of pure translation in 3-D space and where V^i is the vector describing the translation between two segments expressed in coordinate of i-th reference system.e main problem remains to obtain an imposed shape for the tentacle arm. In order to control the robot, we need to obtain the relation between the position of the wires and the position of the segment.

Here, a decoupled approach is used for the robot control scheme. Thus the segments are controlled separately, without considering the interaction between them. Considering the segments of the tentacle separately, then $(\alpha, \theta)_i$ is the asigned coordinate of i-th segment. Having as purpose to command the robot to reach the position $(\alpha, \theta)_i$ the following relation is useful:

$$R = \frac{\overline{L}_{CB}}{\theta} \quad \forall\, \theta \neq 0 \tag{4}$$

where R represents the curvature's radius of the central bone and $\overline{L}_{CB}$ is a constant, equal to the length of the central bone.

Once we have θ and α together as parameters of the desired shape, and after we obtained R, we can compute the corresponding lengths of the wires. Depending on the types of wires and on the structure of the tentacle, we must choose the way to compute the length of each wire.

For the hard wire, made from the same material as the central bone, and by consequence having the same elasticity, referring to Fig. 4, we can write:

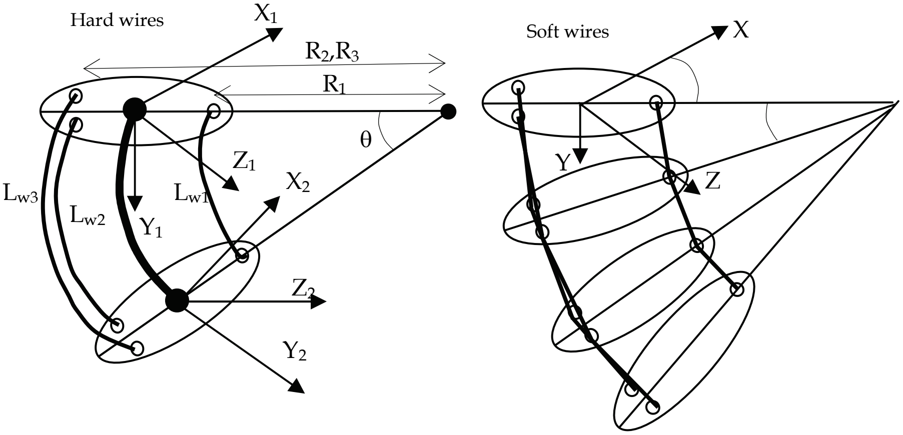


Fig. 4. Different types of wires.

$$\begin{cases} L_{w1} & = & R_1 \cdot \theta \\ L_{w2} & = & R_2 \cdot \theta \\ L_{w3} & = & R_3 \cdot \theta \end{cases} \tag{5}$$

For the soft wires, we can write:

$$\begin{cases} L_{w1} & = & [R_1 \cdot \theta] \cdot \dfrac{sin(\theta / i)}{\theta / i} \\ L_{w2} & = & [R_2 \cdot \theta] \cdot \dfrac{sin(\theta / i)}{\theta / i} \\ L_{w3} & = & [R_3 \cdot \theta] \cdot \dfrac{sin(\theta / i)}{\theta / i} \end{cases} \tag{6}$$

where L_{wn} is the length of the n-th wire and R_i is the radius of the curvature of the real i-th wire.

Farther it can be written:

$$R_n = (R - \Delta R) \cdot \cos(\alpha_n) \tag{7}$$

where ΔR is constant equal to the distance between the center and the wires and α_n is:

$$\begin{cases} \alpha_1 & = & -\alpha \\ \alpha_2 & = & 120° - \alpha \\ \alpha_3 & = & 240° - \alpha \end{cases} \tag{8}$$

Obviously the equations (5) and (6), become the same for $i \to \infty$.

In order to reach the desired shape in a finite time t_f, we should choose the appropriate law for the time variation of the displacements and speed for the three wires, going from the home position to the final position. For each instant, the wires must be moved in order to avoid elongation or compression of it self.

The reference systems for each segment are oriented with the X-axes passing through the first wire. That means that the angles considered between the wires and the desired directions are as in the equation (8).

We can obtain the correlation between these angles and the bending direction of the segment. E.g. if the direction is $\alpha = 2/3\pi$, that means we intend to bend the tentacle in the direction of the second wire with the imposed value of θ degrees. In this case, if we will move the second wire of ΔL_{w2}, we should move the first and third wires with $\Delta L_{w2}/2$ and with the apropiate speed in order to maintain this relation during the movement.

Once we know the angle α, we can obtain the value $\Delta R_i = \Delta R \cdot \cos(\alpha_i)$, defining the displacements of the wires.

The algorithm that we are using, assigns the speed of the wires proportional to ΔR_i in order to go from the home position ($\theta = 0$, $\alpha = 0$) to the position $(\alpha, \theta)_i$ with a constant speed of the motors.

In fact, given the final time t_f and the starting time t_i, after we obtained the displacement of the wires we impose the speed in order to reach the desired position in $(t_f - t_i)$ seconds.

So the speed is:

$$\dot{L}_{wi} = \frac{L_{wi}(t_f) - \overline{L}_{CB}}{(t_f - t_i)} \tag{9}$$

Our structure does not have encoders. Counting the impulses given to the motors, we can evaluate the lengths [L_{w1}, L_{w2}, L_{w3}]. We use these values in order to obtain $(\alpha, \theta)_i$. The algorithm's steps are the following.

For the n-th rigid wire:

$$L_{wn} = \overline{L}_{CB} - \theta \cdot \Delta R \cdot \cos(\alpha_n) \tag{10}$$

Considering the equation (8) and (10), evaluating these for all the wires we can obtain:

$$\begin{cases} \sum_{i=1}^{3} \cos(\alpha_i) = 0 \\ \frac{1}{3}\sum_{i=1}^{3} R_i = R \\ \frac{1}{3}\sum_{i=1}^{3} L_{wi} = L \end{cases} \tag{11}$$

Considering again the equation (10) for the first and second wires, we can write:

$$L_{w1} + \Delta R \cdot \theta \cdot \cos(\alpha_1) = L_{w2} + \Delta R \cdot \theta \cdot \cos(\alpha_2) \tag{12}$$

Replacing the (8) we obtain θ in function of α:

$$\theta = \frac{2}{\Delta R} \cdot \frac{L_{w1} - L_{w2}}{3\cos(\alpha) - \sqrt{3}\sin(\alpha)} \tag{13}$$

And considering the eq. (10) for the third wire:

$$L_{w3} = L_{w1} + \frac{2 \cdot (L_{w1} - L_{w2}) \cdot \left(3\cos(\alpha) - \sqrt{3}\sin(\alpha)\right)}{3\cos(\alpha) - \sqrt{3}\sin(\alpha)} \tag{14}$$

Finally the α angle can be obtained using the function atan2.

$$\alpha = \text{atan2}\left(\sqrt{3}\,(L_{w2} - L_{w3}), 2L_{w1} - L_{w2} - L_{w3}\right) \tag{15}$$

where atan2 is an extension of arctan(y/x) on more quadrant having the following form:

$$\begin{cases} atan(y/x) + \pi & if \quad x < 0, y \geq 0 \\ atan(y/x) - \pi & if \quad x < 0, y < 0 \\ atan(y/x) & if \quad x > 0 \\ \dfrac{\pi}{2} & if \quad x = 0, y > 0 \\ \dfrac{-\pi}{2} & if \quad x = 0, y < 0 \end{cases} \tag{16}$$

The same methodology can be applied for a tronconical robot. The following paragraphs will show how the equations change. The geometry of one segment for the 2D case is described in Fig. 6. The curvature's angle θ of the segment is considered as the input parameter, while the lengths L_1 and L_2 of the control wires are the outputs.

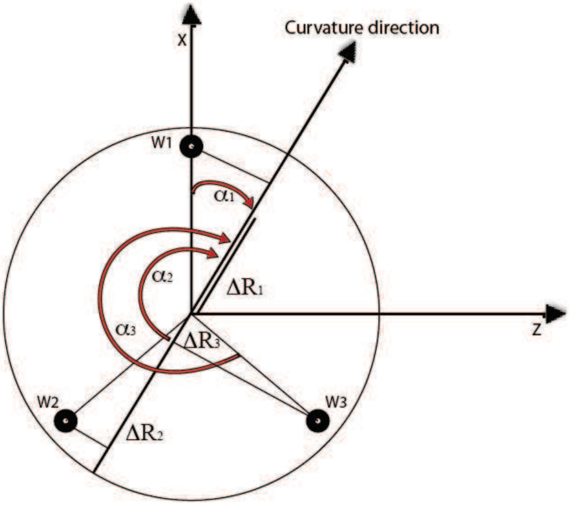


Fig. 5. Projection of the wire to get the α direction

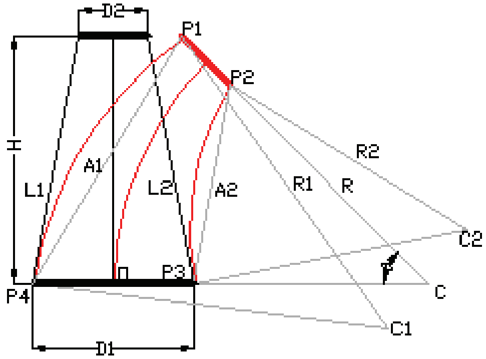


Fig. 6. The geometry of one segment.

The radius R of the segment curvature is obtained using equation (17):

$$R = \frac{H}{\theta} \tag{17}$$

where H is the height of the segment. The following lengths are obtained from Fig. 5, based on the segment curvature:

$$\begin{aligned} L_{11} = \overline{CP_4} = R + D_1/2 \quad L_{12} = \overline{CP_1} = R + D_2/2 \\ L_{21} = \overline{CP_3} = R - D_1/2 \quad L_{22} = \overline{CP_2} = R - D_2/2 \end{aligned} \tag{18}$$

where D_1 and D_2 are the diameters of the segment end discs. Based on the Carnot theorem, the lengths A_1 and A_2 are then obtained:

$$\begin{aligned} A_1 = \sqrt{L_{11}^2 + L_{12}^2 - 2 \cdot L_{11}^2 \cdot L_{12}^2 \cdot \cos\theta} \\ A_2 = \sqrt{L_{21}^2 + L_{22}^2 - 2 \cdot L_{21}^2 \cdot L_{22}^2 \cdot \cos\theta} \end{aligned} \tag{19}$$

The control wires curvature radius R_1 and R_2 are given by the relations (20):

$$R_1 = A_1/2 \cdot \sin\theta/2 \quad R_2 = A_2/2 \cdot \sin\theta/2 \tag{20}$$

Finally, the lengths of the control wires are obtained as in (21):

$$\begin{aligned} L_{w1} = R_1 \cdot \theta = A_1 \cdot \theta / 2 \cdot \sin\tfrac{\theta}{2} \\ L_{w2} = R_2 \cdot \theta = A_2 \cdot \theta / 2 \cdot \sin\tfrac{\theta}{2} \end{aligned} \tag{21}$$

For the 3D case, a virtual wire is considered, which gives the α direction of the curvature. Considering one virtual wire in the direction of the desired curvature having length calculated as follows. Firstly the following lengths are computed:

$$\begin{aligned} L_{11} = R + D_1/2 \cdot \cos(\alpha_1) \qquad L_{12} = R + D_2/2 \cdot \cos(\alpha_1) \\ L_{21} = R + D_1/2 \cdot \cos(\alpha_2) \qquad L_{22} = R + D_2/2 \cdot \cos(\alpha_2) \\ L_{31} = R + D_1/2 \cdot \cos(\alpha_3) \qquad L_{22} = R + D_2/2 \cdot \cos(\alpha_3) \end{aligned} \tag{22}$$

where α_n is according to Fig. 5:

$$\begin{cases} \alpha_1 & = & -\alpha \\ \alpha_2 & = & 120° - \alpha \\ \alpha_3 & = & 240° - \alpha \end{cases} \tag{23}$$

Based on (19) and (20) the curvature radiuses R_1, R_2 and R_3 of the three control wires are then obtained. Finally the lengths of the control wires are computed with (24):

$$\begin{aligned} L_{w1} &= R_1 \cdot \theta \\ L_{w2} &= R_2 \cdot \theta \\ L_{w3} &= R_3 \cdot \theta \end{aligned} \tag{24}$$

Apart from the system presented we can obtain two useful relations:

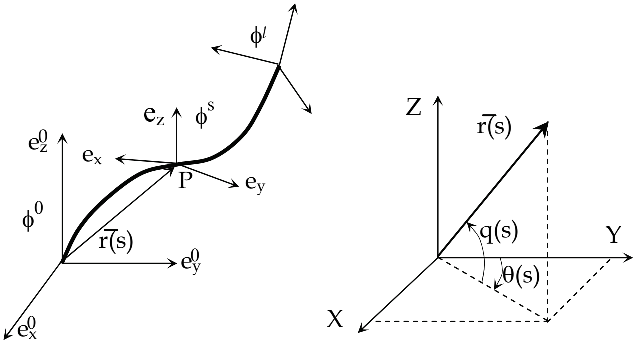
$$\begin{cases} \sum_{i=1}^{3} \cos(\alpha_i) & = & 0 \\ \frac{1}{3}\sum_{i=1}^{3} L_{wi} & = & L \end{cases} \tag{25}$$

The second equation of (25), can be utilized to estimate the virtual compression or the extension of the central bone. We call that virtual compression because before we compress the central bone, the robot will twist to find the shape to guaranty the wrong length of the wires.

3. Dynamics

3.1 Theoretical model

The essence of the tentacle model is a 3-dimensional backbone curve C that is parametrically described by a vector $r(s) \in \mathbf{R}^3$ and an associated frame $\varphi(s) \in \mathbf{R}^{3\times3}$ whose columns create the frame bases (Fig. 7a) (Ivănescu et al., 2006).



(a) (b)

Fig. 7. Tentacle system parameters.

The independent parameter s is related to the arc-length from the origin of the curve C, a variable parameter, where

$$l = \sum_{i=1}^{N} \left(l_{0i} + \Delta l_i \right) \tag{26}$$

or

$$l = l_0 + u \tag{27}$$

where l_0 represents the length of the *N* elements of the arm in the initial position and

$$u = \sum_{i=1}^{N} \Delta l_i \tag{28}$$

determines the control variable of the arm length.
The position of a point s on curve *C* is defined by the position vector,

$$\bar{r} = r(s) \tag{29}$$

when $s \in [0,l]$. For a dynamic motion, the time variable will be introduced, $\bar{r} = \bar{r}(s,t)$.
We used a parameterization of the curve C based upon two "continuous angles" $\theta(s)$ and $q(s)$ and length variable *u* (Fig. 4).
At each point $\bar{r}(s,t)$, the robot's orientation is given by a right-handed orthonormal basis vector $\{\bar{e}_x, \bar{e}_y, \bar{e}_z\}$ and its origin coincides with point $\bar{r}(s,t)$, where the vector e_x is tangent and e_z is orthogonal to the curve C. The position vector on curve C is given by

$$\bar{r}(s,t) = [x(s,t) \;\; y(s,t) \;\; z(s,t)]^T \tag{30}$$

where

$$x(s,t) = \int_0^s \sin\theta(s',t)\cos q(s',t)ds' \tag{31}$$

$$y(s,t) = \int_0^s \cos\theta(s',t)\cos q(s',t)ds' \tag{32}$$

$$z(s,t) = \int_0^s \sin q(s',t)ds' \tag{33}$$

with $s' \in [0,s]$. We can adopt the following interpretation: at any point s the relations (31)-(33) determine the current position and $\Phi(s)$ determines the robot's orientation, and the robot's shape is defined by the behaviour of functions $\theta(s)$ and $q(s)$. The robot "grows" from the origin by integrating to get $\bar{r}(s,t)$, $s \in [0, l_0 + u]$. The velocity components are

$$v_x = \int_0^s \left(-\dot{q}' \sin q' \sin\theta' + \dot{\theta}' \cos q' \cos\theta' \right) ds' \tag{34}$$

$$v_y = \int_0^s \left(-\dot{q}' \, sin \, q' \, cos \, \theta' - \dot{\theta}' \, cos \, q' \, cos \, \theta' \right) ds' \tag{35}$$

$$v_z = \int_0^s \dot{q} \, cos \, q' ds' \tag{36}$$

$$v_u = \dot{u} \tag{37}$$

For an element dm, kinetic and gravitational potential (Douskaia, 1998) energy will be

$$dT = \frac{1}{2} dm \left(v_x^2 + v_y^2 + v_z^2 + v_u^2 \right) \tag{38}$$

$$dV = dm \cdot g \cdot z \tag{39}$$

Where

$$dm = \rho ds \tag{40}$$

From (13)-(15) we obtain

$$T = \frac{1}{2} \rho \int_0^l \left(\left(\int_0^s \left(-\dot{q} \, sin \, q' \, sin \, \theta' + \dot{\theta}' \, cos \, q' \, cos \, \theta' \right) ds' \right)^2 + \right. \tag{41}$$

$$+ \left(\int_0^s \left(-\dot{q}' \, sin \, q' \, cos \, \theta' - \dot{\theta}' \, cos \, q' \, sin \, \theta' \right) ds' \right)^2 +$$

$$\left. \left(\int_0^s \dot{q}' \, cos \, q' ds' \right)^2 \right) ds + \frac{1}{2} \rho \int_0^l \dot{u}^2 ds$$

$$V = \rho g \int_0^l \int_0^s sin \, q' ds' ds \tag{42}$$

The elastic potential energy will be approximated by two components, one determined by the bending of the element

$$V_{eb} = k \frac{d^2}{4} \sum_{i=1}^{N} \left(q_i^2 + \theta_i^2 \right) \tag{43}$$

and the other is given by the axial tension/compression energy component

$$V_{ea} = \frac{1}{2} k u^2 \tag{44}$$

where we assumed that each element has a constant curvature and a uniform equivalent elasticity coefficient k, assumed constant on all the length of the arm.
The total elastic potential energy will be

$$V_e = V_{eb} + V_{ea} \tag{45}$$

We will consider $F_\theta(s,t), F_q(s,t)$ the distributed forces on the length that determine motion and orientation in the θ - plane, q - plane and $F_u(t)$, the force that determines axial motion, assumed constant along the length of the arm.

3.2 Dynamic model

In this paper, the manipulator model is considered a distributed parameter (Ivanescu, 2002). system defined on a variable spatial domain $\Omega = [0,l]$ and the spatial coordinate is denoted by s.
The dynamic model of this manipulator with hyper-redundant configurations can be obtained, in general form, from Hamilton partial differential equations of the distributed parameter model,

$$\frac{\partial \omega(t,s)}{\partial t} = \frac{\delta H}{\delta \nu(t,s)} \tag{46}$$

$$\frac{\partial \nu(t,s)}{\partial t} = -\frac{\delta H}{\delta \omega(t,s)} + F(t,s) \tag{47}$$

where ω and ν are the generalized coordinates and momentum densities, respectively, and $\delta(\cdot)/\delta(\cdot)$ denotes a functional partial derivative.
The state of this system at any fixed time t is specified by the set $(\omega(t,s), \nu(t,s))$, where $\omega = [\theta \quad q \quad u]^T$. The set of all functions of $s \in \Omega$ that ω, ν can take on at any time is state function space $\Gamma(\Omega)$. We will consider that $\Gamma(\Omega) \subset L_2(\Omega)$.
The control forces have the distributed components along the arm, $F_\theta(t,s), F_q(t,s), s \in [0,l]$ and a lumped component $F_u(t)$.
A practical form of dynamic model expressed only as a function of generalised coordinates is derived by using Lagrange equations developed for infinite dimensional systems,

$$\frac{\partial}{\partial t}\left(\frac{\delta T}{\delta \dot{\theta}(t,s)}\right) - \frac{\delta T}{\delta \theta(t,s)} + \frac{\delta V}{\delta \theta(t,s)} + \frac{\delta V_e}{\delta \theta(t,s)} = F_\theta \tag{48}$$

$$\frac{\partial}{\partial t}\left(\frac{\delta T}{\delta \dot{q}(t,s)}\right) - \frac{\delta T}{\delta q(t,s)} + \frac{\delta V}{\delta q(t,s)} + \frac{\delta V_e}{\delta q(t,s)} = F_q \tag{49}$$

$$\frac{\partial}{\partial t}\left(\frac{\partial T}{\partial \dot{u}}\right) - \frac{\partial T}{\partial u} + \frac{\partial V}{\partial u} + \frac{\partial V_e}{\partial u} = F_u \tag{50}$$

where $\partial / \partial(\cdot), \delta / \delta(\cdot)$ denote classical and functional partial derivatives (in Gateaux sense]), respectively.

In Appendix 1 the dynamic model of this ideal spatial tentacle manipulator will be developed and the difficulties to obtain a control law will be easily inferred.

The great number of parameters - theoretically an infinite number of parameters - the complexity of the dynamic model make the application of the classical algorithms meant to obtain the control law very difficult. In much of the literature concerned with the control of these systems, the complexity of the problem is emphasized and various methods that compensate all nonlinear terms in dynamics in real time are developed in order to reduce the complexity of control systems. Also, simplified procedures are introduced or the difficult components are neglected in order to generate a particular law for position or motion control. In all these cases, these methods require a large amount of complicated calculation so that it is difficult to implement these methods with usual level controllers. In addition, the reliability of these methods may be lost when a small error in computation or a small change in the system's parameters occurs.

3.3 Unconstrained control

The artificial potential is a potential function whose points of minimum are attractors for a dissipative controlled system. It was shown that the control of robot motion to a desired point is possible if the function has a minimum in the desired point. In this section we will extend this result for the infinite dimensional model of the tentacle manipulator with variable length.

We consider that the initial state of the system is given by

$$\omega_0 = \omega(0,s) = [\theta_0, q_0, l_0]^T \tag{51}$$

$$v_0 = v(0,s) = [0, 0, 0]^T \tag{52}$$

$$\theta_0 = \theta(0,s), \ \ q_0 = q(0,s), \ \ s \in [0, l_0] \tag{53}$$

$$l_0 = l(0) \tag{54}$$

corresponding to the initial position of the manipulator defined by the curve C_0

$$C_0 : (\theta_0(s), \ q_0(s), l_0), \ s \in [0, l_0] \tag{55}$$

The desired point in $\Gamma(\Omega)$ is represented by a desired position of the arm, the curve C_d,

$$\begin{gathered} \omega_d = [\theta_d, q_d, l_d]^T \ , \ v_d = [0, 0, 0]^T \\ C_d : (\theta_d(s), \ q_d(s), l_d), \ s \in [0, l_d] \end{gathered} \tag{56}$$

The system motion (48)-(5) corresponding to a given initial state $(\omega_0, \ v_0)$ defines a trajectory in the state function space $\Gamma(\Omega)$. The control problem of the manipulator means the motion control by the forces $F_\theta, \ F_q, F_u$ from the initial position C_0 to the desired position C_d. From the viewpoint of mechanics, the desired position (ω_d, v_d) is asymptotically stable if the potential function of the system has a minimum at $(\omega, v)(s) = (\omega_d, v_d)(s), \ \ s \in [0, l]$ and the

system is completely damped. As a control problem in this paper the results of will be extended for the infinite dynamic systems.
We will consider the control forces,

$$F_{\theta}(t,s)=\frac{\delta V}{\delta\theta(t,s)}+\frac{\delta V_e}{\delta\theta(t,s)}-F_{\theta d}-\frac{\delta \Pi}{\delta\theta(t,s)} \tag{57}$$

$$F_u(t)=\frac{\partial V}{\partial u(t)}+\frac{\partial V_e}{\partial u(t)}-F_{ud}-\frac{\partial \Pi}{\partial u(t)} \tag{58}$$

The first two terms compensate the gravitational and elastic potential, the third components assure the damping control and the last terms define the new artificial potential introduced in order to assure the motion to the desired position. The minimum points of this potential must be identical with desired positions of the manipulator, as attractors of its motion. For example, the potential Π can be selected as a functional of generalised coordinates,

$$\Pi(\theta,q,u)=\int_0^l\left((\theta-\theta_d(s))^2+(q-q_d(s))^2\right)ds+(l_0+u-l_d)^2 \tag{59}$$

The control law (57)-(59) modifies the system potential and the Lagrange equation (48)-(50) (Masoud & Masoud, 2000) become

$$\frac{\partial}{\partial t}\left(\frac{\delta T}{\delta\dot{\theta}(t,s)}\right)-\frac{\delta T}{\delta\theta(t,s)}+\frac{\delta \Pi}{\delta\theta(t,s)}=F_{\theta_d} \tag{60}$$

$$\frac{\partial}{\partial t}\left(\frac{\delta T}{\delta\dot{q}(t,s)}\right)-\frac{\delta T}{\delta q(t,s)}+\frac{\delta \Pi}{\delta q(t,s)}=F_{q_d} \tag{61}$$

$$\frac{\partial}{\partial t}\left(\frac{\partial T}{\partial\dot{u}}\right)-\frac{\partial T}{\partial u}+\frac{\partial \Pi}{\partial u}=F_{u_d} \tag{62}$$

The force components F_{θ_d}, F_{q_d}, F_{u_d} represent the damping components of the control and have the form

$$F_{\theta_d}(s,t)=-\int_0^l K_{\theta}(s,s')\dot{\theta}(s',t)ds' \tag{63}$$

$$F_{q_d}(s,t)=-\int_0^l K_q(s,s')\dot{q}(s',t)ds' \tag{64}$$

$$F_{u_d}(t)=-K_u\dot{u}(t) \tag{65}$$

where $K_{\theta}(s,s')$, $K_q(s,s')$ are positive definite specified spatial weighting functions on $(\Omega\times\Omega)$ and K_u is a positive constant. For practical reasons, the derivative components of the control have the form

$$K_\theta(s,s')=\delta(s-s')\cdot k_\theta(s) \tag{66}$$

$$K_q(s,s')=\delta(s-s')\cdot k_q(s) \tag{67}$$

3.4 Constrained control

Let B be the region of the state (Ceah & Wang, 2005) space where the mechanical system motion is not admissible, its complement $\overline{B}$ is the region of admissible movements and ∂B is the boundary of B. The control problem is to determine the potential function $\Pi(\theta,q,u)$ which would determine the motion to the desired position $(\omega_d(s),\, v_d(s)),\ s\in[0,l]$ and it does not penetrate the constrained area B. In terms of the artificial potential, this means that this functional should have a single stationary point in $\overline{B}$ and grows without limit when the system penetrates the boundary ∂B.

We will consider the following artificial potential,

$$\Pi(\theta,q,u)=max\{\Pi_1(\theta,q,u),\, \Pi_2(\theta,q,u)\} \tag{68}$$

where $\Pi_1(\theta,q,u)$ is the artificial potential for unconstrained problem and $\Pi_2(\theta,q,u)$ is the potential for constrained control problem.

$\Pi_2(\theta,q,u)$ is a non-negative, continuous functional defined in $\overline{B}$ and

$$\lim_{d\to 0}\Pi_2(\theta,q,u)=\infty \tag{69}$$

where d is the distance between the current state (θ,q,u) and the boundary ∂B.

3.5 Appendix 1

We will consider a spatial tentacle model, an ideal system, neglecting friction and structural damping. We assume a uniformly distributed mass with a linear density ρ [kg/m].

We will use the notations:

$$q=q(s,t),\ \ s\in[0,l],\ \ t\in\lfloor 0,t_f\rfloor \qquad \theta=\theta(s,t),\ \ s\in[0,l],\ \ t\in\lfloor 0,t_f\rfloor$$

$$q'=q(s',t),\ \ s'\in[0,s],\ \ t\in\lfloor 0,t_f\rfloor \qquad \dot{q}=\frac{\partial q(s,t)}{\partial t},\ \ s\in[0,l],\ \ t\in\left[0,t_f\right]$$

$$\dot{q}'=\frac{\partial q(s',t)}{\partial t},\ \ s'\in[0,s],\ \ t\in\left[0,t_f\right] \qquad \ddot{q}'=\frac{\partial^2 q(s',t)}{\partial t^2},\ \ s'\in[0,s],\ \ t\in\left[0,t_f\right]$$

$$\ddot{q}''=\frac{\partial^2 q(s'',t)}{\partial t^2},\ \ s''\in[0,s],\ \ t\in\left[0,t_f\right]$$

. . . .

$$F_q=F_q(s,t),\ \ s\in[0,l],\ \ t\in\lfloor 0,t_f\rfloor \qquad F_u=F_u(t),\ \ t\in\lfloor 0,t_f\rfloor$$

From (60)-(62), it results,

$$\rho\int_0^s\int_0^s\left(\ddot{q}'\left(\sin q'\sin q''\cos\left(q'-q''\right)+\cos q'\cos q''\right)-\ \ddot{\theta}'\cos q'\sin q''\sin\left(\theta''-\theta'\right)+\right.$$

$$\left.+\dot{q}'^2\left(\cos q'\sin q''\cos\left(\theta'-\theta''\right)-\sin q'\cos q''\right)+\ \dot{\theta}'^2\cos q'\sin q''\cos\left(\theta'-\theta''\right)\ -\dot{q}'\dot{q}''\sin\left(q''-q'\right)\right)ds'ds''+$$

$$+\rho g\int_0^s\cos q'ds'+\frac{1}{2}kd^2q=F_q$$

$$\rho\int_0^s\int_0^s\left(\ddot{q}'\sin q'\cos q''\sin\left(\theta''-\theta'\right)+\ \ddot{\theta}'\cos q'\cos q''\cos\left(\theta''-\theta'\right)-\ \dot{q}'^2\cos q'\cos q''\sin\left(\theta''-\theta'\right)+\right.$$

$$\left.+\dot{\theta}'\cos q'\cos q''\sin\left(\theta''-\theta'\right)-\ \dot{\theta}'\dot{q}'\sin q'\cos q''\cos\left(\theta''-\theta'\right)\right)ds'ds''+\frac{1}{2}kd^2\theta=F_\theta$$

$$\rho\ddot{u}+\frac{1}{2}\rho\dot{u}^2+ku=F_u$$

4. Visual servoing system

4.1 Camera system

In the Appendix 2 the dynamic model of the 3D spatial hyper redundant arm is determinated. Two video cameras provide two images of the whole robot workspace. The two images planes are parallel with XOY and ZOY planes from robot coordinate frame, respectively (Fig. 8). The cameras provide the images of the scene stored in the frame grabber's video memory being displayed on the computer screens (Hannan & Walker, 2005); (Kelly, 1996). Related to the image planes, two dimensional coordinate frames, called screen coordinate frames or image coordinate systems are defined. Denote X_{S_1}, Y_{S_1} and Z_{S_2}, Y_{S_2}, respectively, the axes of the two screen coordinate frames provided by the two cameras. The spatial centers for each camera are located at the distances D_1 and D_2, with respect to the XOY and ZOY planes, respectively. The orientation of the cameras arround the optical axes with respect to the robot coordinate frame, are noted with ψ and ϕ, respectively. A point P in the coordinate frame is

$$P=[\,x,\,y,\,z]^T \tag{70}$$

The description of a point P in the two screen coordinate frames are denoted by

$$P_{S2}=[\,x_{S1}\,,y_{S_2}\,] \tag{71}$$

$$P_{S2}=[\,z_{S_2}\,,y_{S_2}\,] \tag{72}$$

Geometric optics are used to model the mapping between the robot Cartesian space and the screen coordinate systems. We assume that the quantization and the lens distortion effects are negligible. The description of the point $P=[\,x, y, z]^T$ in the robot coordinate frame is given in terms of screen coordinate frames as

$$\begin{bmatrix} x_{s_1} \\ y_{s_1} \end{bmatrix} = \alpha_1 \cdot \frac{\lambda_1}{\lambda_1 - (D_1 + x)} \cdot R(\phi) \cdot \left\{ \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} o_{11} \\ o_{12} \end{bmatrix} \right\} + \begin{bmatrix} c_{x_1} \\ c_{y_1} \end{bmatrix} \tag{73}$$

for the $Z_{S_1} O_{S_1} Y_{S_1}$ frame and

$$\begin{bmatrix} z_{s_2} \\ y_{s_2} \end{bmatrix} = \alpha_2 \cdot \frac{\lambda_2}{\lambda_2 - (D_2 + x)} \cdot R(\phi) \cdot \left\{ \begin{bmatrix} z \\ y \end{bmatrix} - \begin{bmatrix} o_{21} \\ o_{22} \end{bmatrix} \right\} + \begin{bmatrix} c_{z_2} \\ c_{y2} \end{bmatrix} \tag{74}$$

for the $Z_{S_2} O_{S_2} Y_{S_2}$ frame, where $[\,c_{x_1}, c_{y_1}\,]^T$ and $[\,c_{z_2}, c_{y_2}\,]^T$ the image centers, α_1 and α_2 are the scale factors of the length units in the front image planes given in pixel/m, $R(\psi)$ and $R(\phi)$ are the rotation matrices generated by clockwise rotating the cameras about their optical axes by ψ and ϕ radians, respectively, and $[O_{11}, O_{12}]^T$ and $[O_{21}, O_{22}]^T$ represent the distances between the optical axes and the XOY and ZOY planes, respectively.

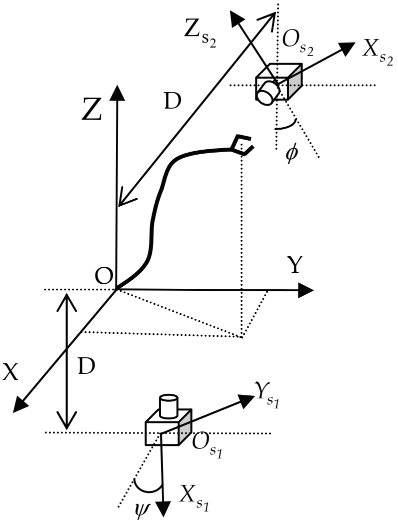


Fig. 8. Camera system

In Fig. 9 the frames corresponding to the screen images of the two cameras are presented. From the relations (73), (74), we obtain

$$\begin{bmatrix} \Delta x_{s_1} \\ \Delta y_{s_1} \end{bmatrix} = \alpha_1 \cdot \frac{\lambda_1}{\lambda_1 - (D_1 + x)} \cdot \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \tag{75}$$

$$\begin{bmatrix} \Delta z_{s_2} \\ \Delta y_{s_2} \end{bmatrix} = \alpha_2 \cdot \frac{\lambda_2}{\lambda_2 - (D_2 + x)} \cdot \begin{bmatrix} \Delta z \\ \Delta y \end{bmatrix} \tag{76}$$