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.

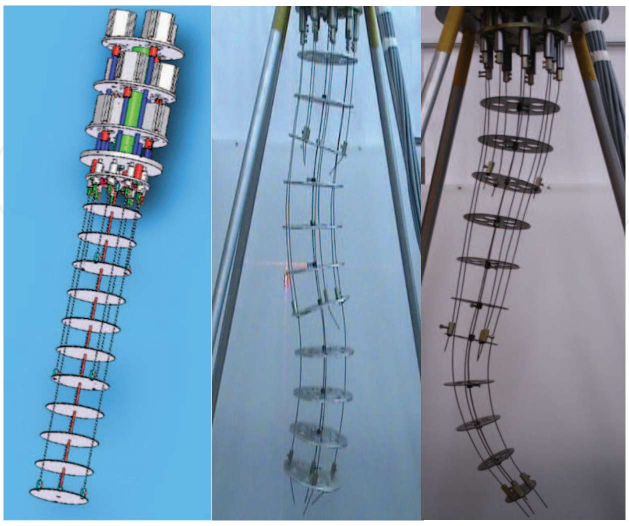


Fig. 1. A tentacle arm prototype

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).

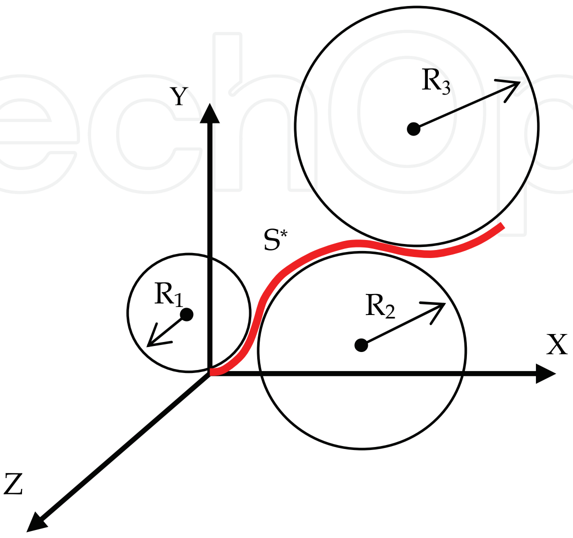


Fig. 2. The description of the desired shape

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.

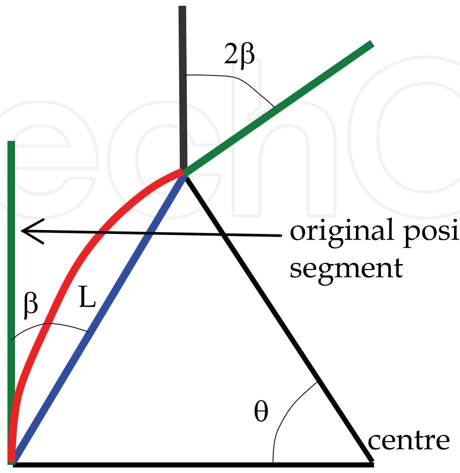


Fig. 3. Curvature and relation between θ and β

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:

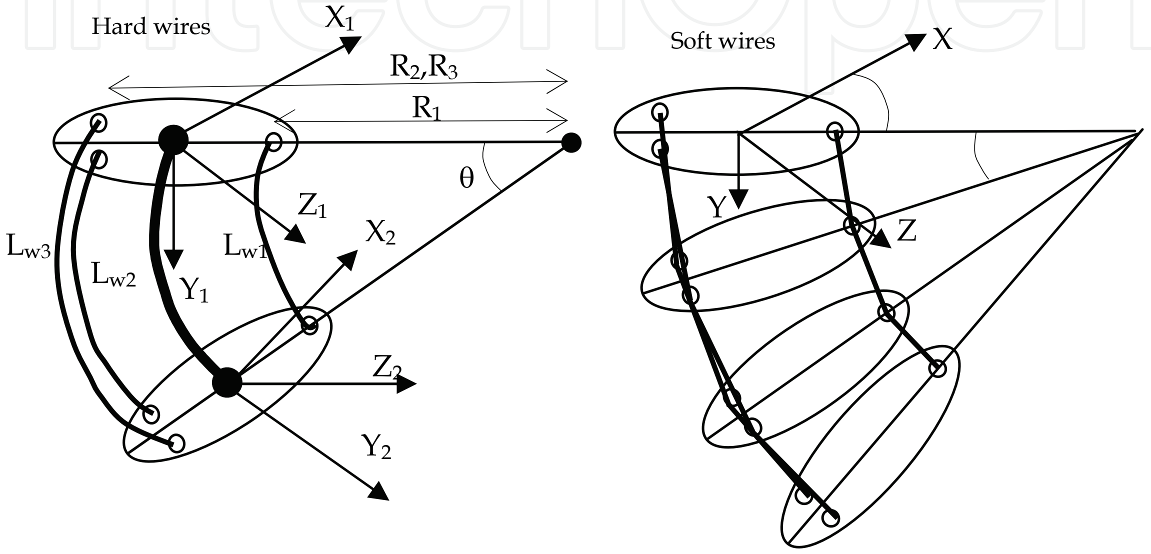


Fig. 4. Different types of wires.

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

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 $i \rightarrow \infty$.

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.

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

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.

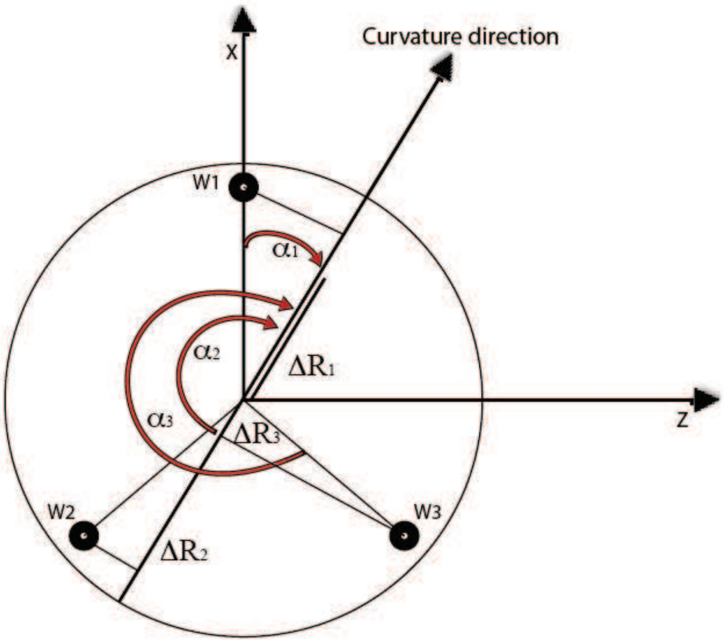


Fig. 5. Projection of the wire to get the α direction

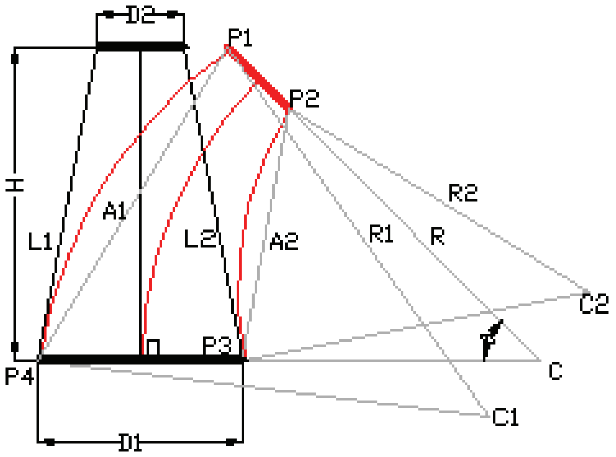


Fig. 6. The geometry of one segment.

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:

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

where α_n is according to Fig. 5:

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

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 $\varphi(s) \in \mathbf{R}^{3\times 3}$ whose columns create the frame bases (Fig. 7a) (Ivănescu et al., 2006).

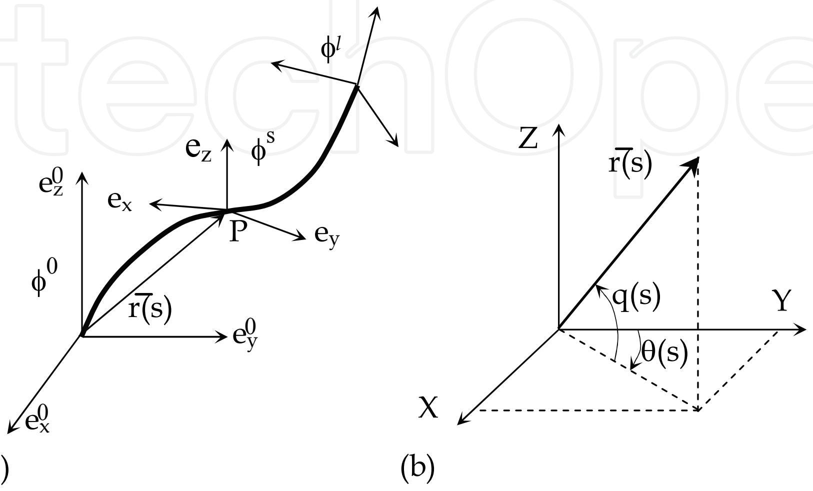


Fig. 7. Tentacle system parameters.

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,

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

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

where ω and v 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 $(\omega(t,s), v(t,s))$, where $\omega = [\theta \quad q \quad u]^T$. The set of all functions of $s \in \Omega$ that ω, v 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 (ω_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_θ, 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 $(\omega_d(s),\ \nu_d(s)),\ s \in [0,l]$ 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:

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

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

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

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

. . . .

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

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

$$\rho\int_0^s\int_0^s\Big(\ddot{q}'\big(sin\,q'\,sin\,q''\,cos(q'-q'')+cos\,q'\,cos\,q''\big)-\ \ddot{\theta}'\,cos\,q'\,sin\,q''\,sin(\theta''-\theta')+$$

$$+\dot{q}'^2\big(cos\,q'\,sin\,q''\,cos(\theta'-\theta'')-sin\,q'\,cos\,q''\big)+\ \dot{\theta}'^2\,cos\,q'\,sin\,q''\,cos(\theta'-\theta'')\ -\dot{q}'\dot{q}''\,sin(q''-q')\Big)ds'ds''+$$

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

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

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

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

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.

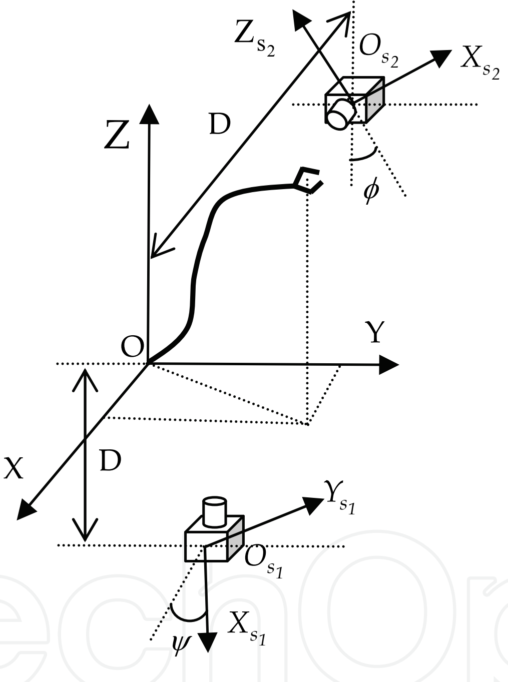


Fig. 8. Camera system

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

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

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

and the orientation angles for each plane will be

$$tg\theta_s = \frac{\Delta x_{s_1}}{\Delta y_{s_1}} = \frac{\Delta x}{\Delta y} = tg\theta \tag{77}$$

hence

$$\theta_s(s') = \theta(s)\,,\, s \in [0,l]\,,\, s' \in [0,l'] \tag{78}$$

for the plane $Z_{S_1}O_{S_1}Y_{S_1}$ and $tgq_s = \frac{\Delta z_{s_2}}{\Delta y_{s2}} = \frac{\Delta z}{\Delta y}$

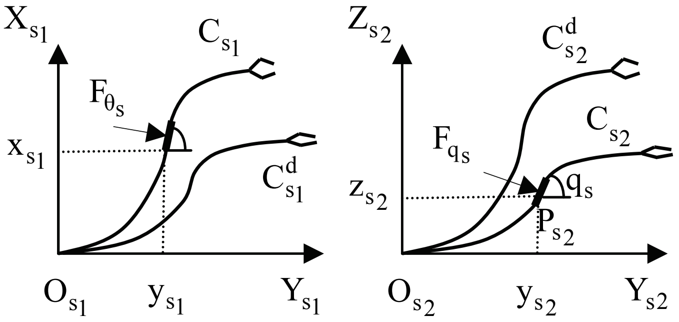


Fig. 9. Image frames

This relation allows the computation of the orientation angle q_s in the plane $Z_{S_2}O_{S_2}Y_{S_2}$

$$tgq_s(s'') = tgq(s) \cdot \frac{1}{cos\,\theta(s)}\,,\; s \in [0,l]\,,\; s'' \in [0,l''] \tag{79}$$

where, s', s'' and l', l'' represent the projections of the variable s and the length l in the two planes, respectively. The projection of the forces on the two planes can be easily inferred and the relations (77)-(79),

$$F_{\theta_s} = F_\theta \tag{80}$$

$$F_{q_s} = F_q \cdot \sqrt{cos^2 q + sin^2 q \cdot cos^2 \theta} \tag{81}$$

4.2 Servoing system

The control system is an image – based visual servo control where the error control signal is defined directly in terms of image feature parameters. The desired position of the arm in the robot space is defined by the curve C_d,

$$C : (\theta^d(s), q^d(s))\;,\; s \in [o,l] \tag{82}$$

or, in the two image coordinate frames $Z_{S_1}O_{S_1}Y_{S_1}$ and $Z_{S_2}O_{S_2}Y_{S_2}$, by the projection of the curve C,

$$C^{d}_{s_1} : (\theta^{d}_{s}(s'))\,,\ s' \in [0,l'] \tag{83}$$

$$C^{d}_{s_2} : (q^{d}_{s}(s''))\,,\ s'' \in [0,l''] \tag{84}$$

Define the motion errors as

$$e_{\theta}(t,s) = \theta(t,s) - \theta_{d}(s)\,,\ s \in [0,l] \tag{85}$$

$$e_{q}(t,s) = q(t,s) - q_{d}(s)\,,\ s \in [0,l] \tag{86}$$

or, in the image coordinate frames, by $s' \in [0,l']$, $s'' \in [0,l'']$

$$e_{\theta_s}(t,s') = \theta_{s}(t,s') - \theta^{d}_{s}(s') \tag{87}$$

$$e_{q_s}(t,s'') = q_{s}(t,s'') - q^{d}_{s}(s'') \tag{88}$$

The global control system is presented in Fig. 10. The control problem of this system is a direct visual servocontrol but we do not use the clasical concept of the position control where the error between the robot end-effector and target is minimized (Grosso et all., 1996).

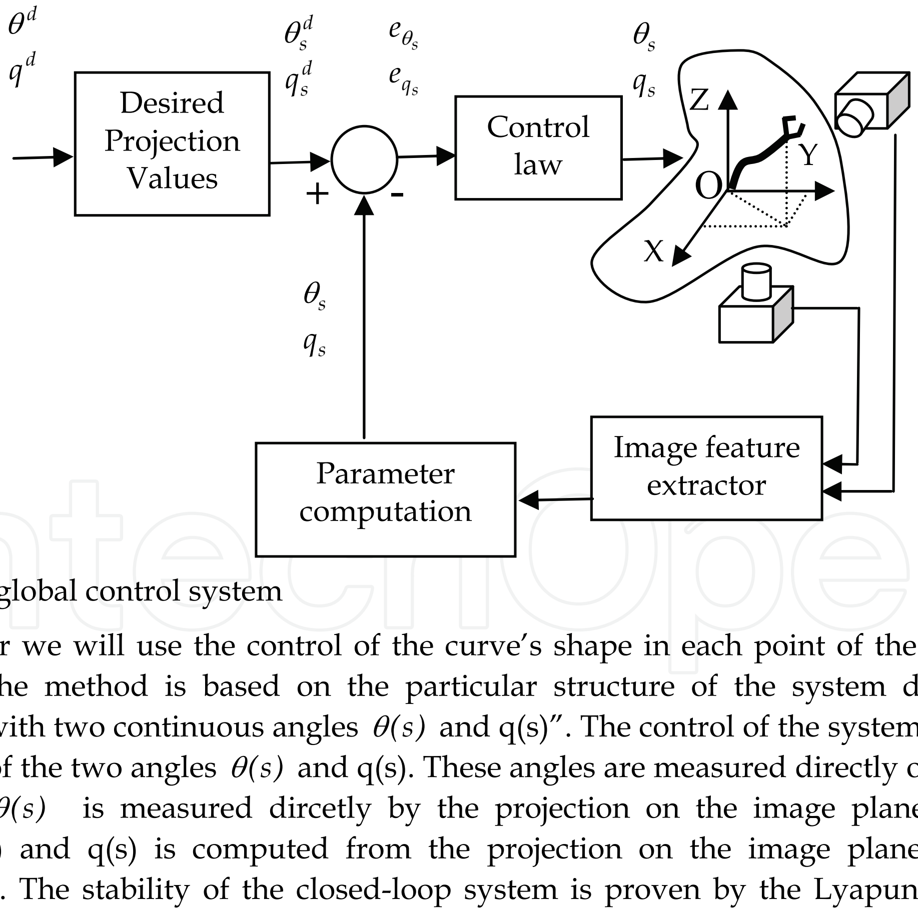


Fig. 10. The global control system

In this paper we will use the control of the curve's shape in each point of the mechanical structure. The method is based on the particular structure of the system defined as a "backbone with two continuous angles $\theta(s)$ and q(s)". The control of the system is based on the control of the two angles $\theta(s)$ and q(s). These angles are measured directly or indirectly. The angle $\theta(s)$ is measured dircetly by the projection on the image plane $Z_{S_1}O_{S_1}Y_{S_1}$ (relation 78) and q(s) is computed from the projection on the image plane $Z_{S_2}O_{S_2}Y_{S_2}$ (relation 79). The stability of the closed-loop system is proven by the Lyapunov's second method but, in order to avoid the complex problems derived from using the nonlinear derivation integral model, a method based on the energy-work relationship (Ge at al.,1996) was be developed (see Appendix 2).

Proposition: The closed-loop hyper redundant arm system is stable if the control law is

$$F_\theta(s,t) = -k_\theta^1(s) \cdot e_{\theta_s}(s',t) - k_\theta^2(s) \cdot \dot{e}_{\theta_s}(s',t) \tag{89}$$

$$F_q(s,t) = -k_q^1(s) \cdot \left[tg^{-1}(\cos\theta_s(s',t)) \cdot \ tgq_s(s'',t) - q^d(s)\right] \tag{90}$$

where $s' \in [0,l']$, $s'' \in [0,l'']$ and $k_\theta^1(s), k_\theta^2(s), k_q^1(s)$ are positive coefficients of the control law for all $s \in [0,l]$. The parameter of the control law (88), (89), can be inferred from the image feature extraction of the two planes. The parameters e_{θ_s} can be directly calculated from equation (85-88) and $\dot{e}_{\theta_s}$ can be indirectly computed. Also θ_s , qs and q_s^d are evaluated directly from the trajectory projections. We remark that the control law represents a robust control, independent of the camera parameters. No intrinsec camera parameters are assumed known.

4.3 Appendix 2

We will consider a spatial tentacle model, an ideal system, neglecting friction and structural damping. We assume a uniformly distributed mass with linear density $\rho[kg/m]$. We will consider a non-extensible arm with constant length.

We will use the notations:

$$q = q(s,t)\ ,\ q' = q'(s',t)\ ,\ \dot{q} = \frac{\partial q(s,t)}{\partial t}\ ,$$

$$\dot{q}' = \frac{\partial q(s',t)}{\partial t}\ ,\ \ddot{q}' = \frac{\partial^2 q(s',t)}{\partial t^2}\ ,\ \ddot{q}'' = \frac{\partial^2 q(s'',t)}{\partial t^2}\ ,$$

The position of a point *P* is given by (31-33) and the velocity components are given by (34-37). From an element *dm*, kinetic and potential energy are given by will be (38-40). Following (41-42) were computed.

The dynamic model is obtained by using Lagrange equation of motion

$$\frac{d}{dt}\left(\frac{\delta T}{\delta \dot{q}}\right) - \frac{\delta T}{\delta q} + \frac{\delta V}{\delta q} = F \tag{91}$$

where $\delta(.)/\delta(.)$ denotes a functional partial (variational) Gateaux derivate (Wang, 1965), as shown before, that is defined as the variation of the functional Ω with respect to the function θ at a point $s \in [0,l]$. From (41-42) it results,

$$\begin{aligned}
&\rho \cdot \int_0^l \int_0^s \Big(\ddot{q}' \cdot \left(\sin q' \cdot \sin q'' \cdot \cos(q' - q'') + \cos q' \cdot \cos q''\right) - \ddot{\theta}' \cdot \cos q' \cdot \sin q'' \cdot \sin(\theta'' - \theta') + \\
&+ \dot{q}'^2 \cdot \left(\cos q' \cdot \sin q'' \cdot \cos(\theta'' - \theta')\right) - \sin q \cdot \cos q'') + \dot{\theta}'^2 \cdot \cos q' \cdot \sin q'' \cdot \cos(\theta'' - \theta') - \\
&- \dot{q}' \cdot \dot{q}'' \cdot \sin(q'' - q') \Big) \cdot ds' \cdot ds'' + \rho \cdot g \cdot \int_0^s \cos q' \cdot ds' = F_q
\end{aligned} \tag{92}$$

$$\rho \cdot \int_0^l \int_0^s \Bigg(\ddot{q}' sin\, q' \cdot cos'' \cdot sin(q' - q'') + \ddot{\theta}' \cdot cos\, q' \cdot cos'' \cdot cos(\theta'' - \theta') + \\ + \dot{q}'^2 \cdot \left(cos\, q' \cdot cos\, q'' \cdot sin(\theta'' - \theta') \right) + \dot{\theta}' \cdot cos\, q' \cdot cos\, q'' \cdot sin(\theta'' - \theta') - \\ - \dot{\theta}' \cdot \dot{q}' \cdot cos\, q'' \cdot cos(\theta'' - \theta') \Bigg) \cdot ds' \cdot ds'' = F_\theta \tag{93}$$

We consider the following Lyapunov function.

$$V^*(t) = T(t) + V(t) + \frac{1}{2} \cdot \int_0^l k_\theta^1(s) \cdot e_\theta^2(s,t) \cdot ds \tag{94}$$

where *T, V* represent the kinetic and potential energies of the system. *V*(t)* is pozitive definited because the terms that represent the energy *T* and *V* are always $T(t) \geq 0$, $V(t) \geq 0$. For the steady desired position, we have

$$\dot{V}^*(t) = \int_0^l \left(F_\theta(s,t) \cdot \dot{e}_\theta(s,t) + F_q(s,t) \cdot \dot{e}_q(s,t) + k_\theta^1(\theta) \cdot e_\theta(s,t) \right) \cdot ds \tag{95}$$

If we use the control low defined by the relations (89)-(90), where the parameters of motion are evaluated from (78)-(79), (85)-(88), we will have,

$$\dot{V}^*(t) = -\int_0^l k_\theta^1 \cdot \left(\dot{e}_\theta(s,t) \right)^2 \cdot ds \tag{96}$$

$$\dot{V}^*(t) \leq 0 \tag{97}$$

Q.E.D.

The derivative of the error in the control laws (89), (90) can be computed by an iteration procedure. The coordinate x_{s_1} on the projection C_{s_1} can be evaluated by the relation

$$x_{s_1}(i \cdot \Delta s') = \sum_{j=1}^{i} sin\, \theta_{s_1}(j \cdot \Delta s') \cdot \Delta s' \tag{98}$$

$$\Delta x_{s_1}(i \cdot \Delta s') = \sum_{j=1}^{i} cos\, \theta_{s_1}(j \cdot \Delta s') \cdot \Delta\theta(j \cdot \Delta s') \cdot \Delta s' \tag{99}$$

Assuming that $\theta_{s_i}(s_i) \neq \frac{\pi}{2}$, we obtain

$$\begin{aligned} &\text{i=1} \quad \Delta\theta(\Delta s') = \frac{1}{\Delta s'}\left(\Delta x_{s_1}(\Delta s') / cos\, \theta_{s_1}(\Delta s') \right) \\ &\text{i=2} \quad \Delta\theta(2 \cdot \Delta s') = \left(\Delta x_1(2 \cdot \Delta s') / \Delta s' - cos\, \theta_{s_1}(\Delta s') \cdot \Delta\theta(\Delta s') \right) / cos\, \theta_{s_1}(2 \cdot \Delta s') \\ &\vdots \\ &\text{i=m} \quad \Delta\theta(m \cdot \Delta s') = \left(\Delta x_1(m \cdot \Delta s') / \Delta s' -- \sum_{j=1}^{m-1} cos\, \theta_{s_1}(j \cdot \Delta s') \cdot \Delta\theta(j \cdot \Delta s') \right) \cdot \frac{1}{cos(m \cdot \Delta s')} \end{aligned} \tag{100}$$

If $\theta_{s_i}(s_i)=\frac{\pi}{2}$, a similar procedure for $y_{s_1}(i\cdot\Delta s')$ can be used.

4.4 Camera calibration

The term "camera calibration" in the context of this paper refers to positioning and orienting the two cameras at imposed values (Fig. 11) (Tanasie et al., 2009).). This calibration is performed only at the beginning, after that the cameras remain still. First, a zoom that maximizes the image resolution of the working space used by the manipulator is performed. Second, positioning of the two cameras brings the manipulator in the middle of the two images. Third, a pan / tilt orientation is performed (as descried later in the paper). At this step the manipulator is moved in a test position that allows free of (or minimum) errors calibration. The test images are compared to the images generated by the graphic simulator (ideal images) which represent references for the calibration operation.

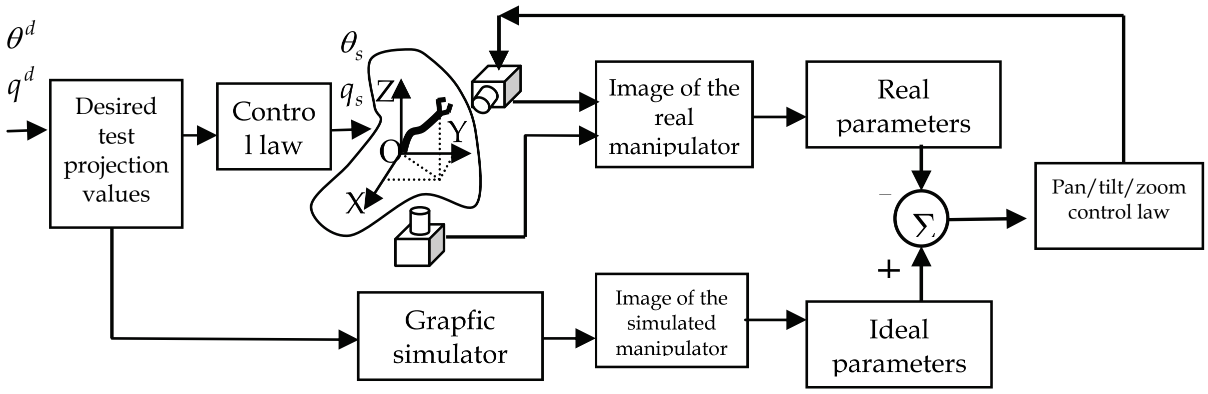

θ^d
q^d
Desired test projection values
Contro l law
θ_s
q_s
Z
Y
O
X
Image of the real manipulator
Real parameters
−
Σ
+
Pan/tilt/zoom control law
Grapfic simulator
Image of the simulated manipulator
Ideal parameters

Fig. 11. Camera calibration system

In order, to ease the fulfil of the cameras calibration, a graphic simulator based on a 2D direct kinematics model was designed, implemented and used. By consequence, during the calibration procedure, the robot was commanded to bend in planes perpendicular to the cameras axes. Thus only the arching angle needs to be computed and a 2D model is sufficient to solve the problem. The next version of the software application introduces also the possibility to calibrate in 3D, the test positions corresponding to unrestricted planes orientation. A very important task in developing this application is to control the camera position and orientation. From this point of view, the calibration operation assures that the two cameras' axes are orthogonal. In the beginning, the tentacle manipulator receives the needed commands in order to stand in a test pose (imposed position and orientation). The same commands are sent to the graphic simulator. Two different sets of images are obtained: real images acquired by the real cameras and simulated images offered by the graphic simulator. From these two sets of images, two sets of parameters are computed: real parameters are computed from real images and, respectively, ideal parameters are computed from synthetic images. Comparing the two sets of parameters and knowing the image/parameters behavior for the camera orientation, the cameras are orientated (pan/tilt/zoom) in order to minimize the error.

A graphical simulator was designed and implemented in order to test the robot behavior under certain circumstances (Cojocaru et al., 2010). The simulator approximates the curved segments of the hyper redundant robot and considers constant the length of the median arc of each segment. To ease the presentation, the term segment will be used in all that follows referring to the median segment (arched or un-arched). For the arched segment, its median arc remains constant. In this paper the term O-X angle will be used to denote the angle that the chord made by an arched element of the robot makes with the O-X axis of a selected reference system.

The inputs for the simulator are: robot configuration; robot initial position; rontrol laws for each of the segments of the hyper redundant robot. The robot configuration consists of the number of segments the hyper redundant robot has, the length of each segment and the angles that the cords make with the O-X axis. The arching angles are computed from these angles. An arching angle is defined as the angle made by the cord (determined by the ends of the arched segment) and the original un-arched segment. For the direct kinematics problem, the control of the robot simulation is accomplished by giving the O-X angles for each of the segments in their final position and the output of the simulation is the hyper redundant robot's end-effector final position in the operation space. In order to compute the final position of the end-effector and the hyper redundant robot's behavior during its motion, a few elements must be computed: the relation between the arching angle and the angle at center determined by the arched segment (this angle determines the length of the arc); the cord length; the relation between an O-X angle and an arching angle; the final arching angles – recurrent set.

The computation of the relation between the arching angle and the angle at center determined by the arched segment is determined by the following axiom: For camera calibration a direct kinematics model was used, thus the rotation angles for each segment are given. For a robot that has only rotation joints, the O-X angle increases (or decreases, depending on the selected positive direction) for each segment with the sum of rotation angles of each of the previous segments (including the current segment). This is true because the orthogonal system attached to the ith segment is obtained from its initial position and applying all the anterior transformations. For a hyper redundant robot the problems are different. The arching angle is double the sum of each previous arching angle plus the current arching angle, because the un-arched segment is a prolongation of the previous segment.

In order to simulate the circular arched segments a series of intermediate points (that are connected by lines) between the segment origins must be determined. The Catmull-Rom interpolation algorithm was used for this simulator because it was need an interpolation algorithm that passes through the control points. Catmull-Rom splines are a family of cubic interpolating splines formulated such that the tangent at each point p_i is calculated using the previous and next point on the splines, $\tau(p_{i+1} - p_{i-1})$.

Camera calibration is the essential procedure for all such applications: positioning and orienting the cameras in order to support the accuracy of the image features extraction. Calibration for a pan/tilt/zoom camera shape is achieved by means of an engineered environment and a graphic simulation module.

Term "camera calibration" in the context of this paper refers to positioning and orienting the two cameras at imposed values. This calibration is performed only at the beginning, after that the cameras remain still. The general control method is an image based visual servoing one instead of position based. Camera calibration based on intrinsic parameters (classic sense, not the one used in this paper) is not necessary. Calibration operation assures that the two cameras' axes are orthogonal.

Taking into account the presented structure of the tentacle - vision system, in order to apply the tested visual servoing algorithm, the two cameras must be positioned and oriented as: both focus on the robot, their axes are orthogonal, both have the same zoom factor.

Two different algorithms were implemented: one uses a cylindrical etalon, other uses the graphical simulator.

For the first algorithm, special starting conditions were imposed in order to support the image processing tasks: white background, dark grey cylinder, red vertical equidistant (90 degrees) axes, friendly initial camera's positions and orientations, zoom x1 (Fig. 12).

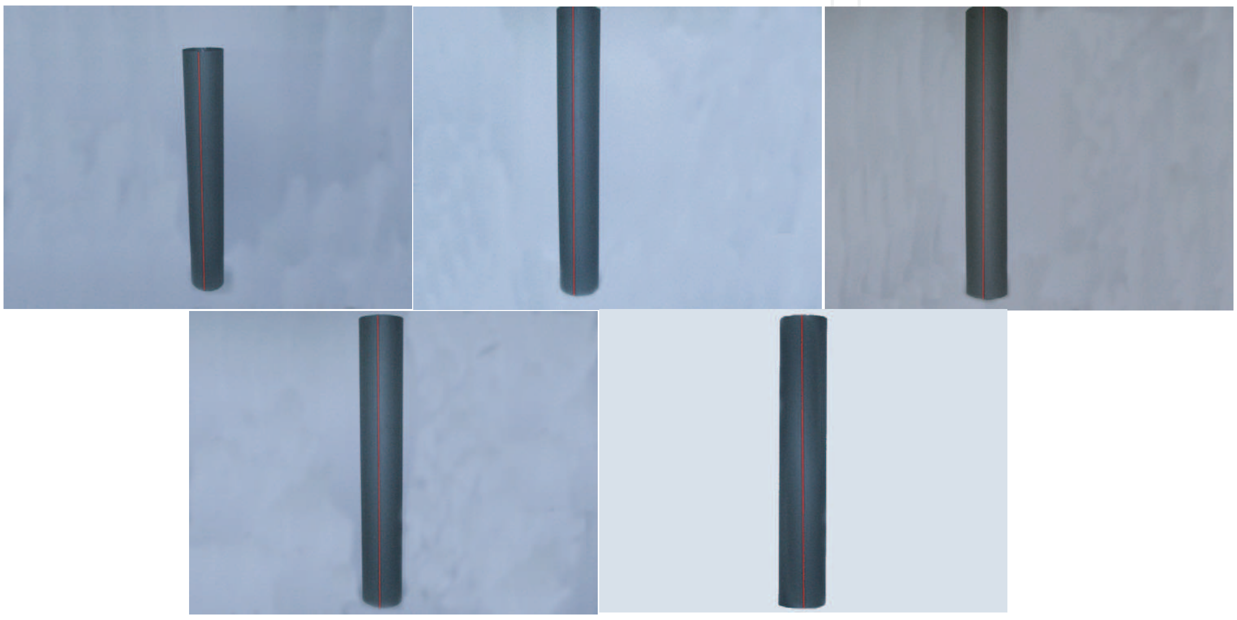


Fig. 12. The cylindrical etalon

Three succesive and dependent calibrations are performed: Horizontal (pan): position and orientation are obtained in two successive, but dependent steps; Vertical (tilt): position and orientation are obtained in two successive, but dependent steps; Zoom: tuning the two cameras as both look to the cylinder from virtual equal distances.

Both offsets must be under the accepted thresholds. Else, the positioning destroyed the orientation and the procedure must be repeated. A similar algorithm is developed for the vertical orientation and positioning.

The second algorithm works together with the graphic simulator. It was proven that the two camera axes are orthogonal if, when both cameras are looking at the tentacle successively bended as circle's arcs in two orthogonal planes, are seeing also two circle's arcs (Fig. 13). The previous condition is fulfilled if each camera looks at the center of the circle containing the arc and the view line is orthogonal on the plane's circle.

Three calibration steps must be performed: Horizontal calibration - positioning and orienting the camera horizontally (pan); Vertical calibration - positioning and orienting the camera vertically (tilt); Zoom calibration - tuning the two cameras as both look at the robot from virtual equal distances.

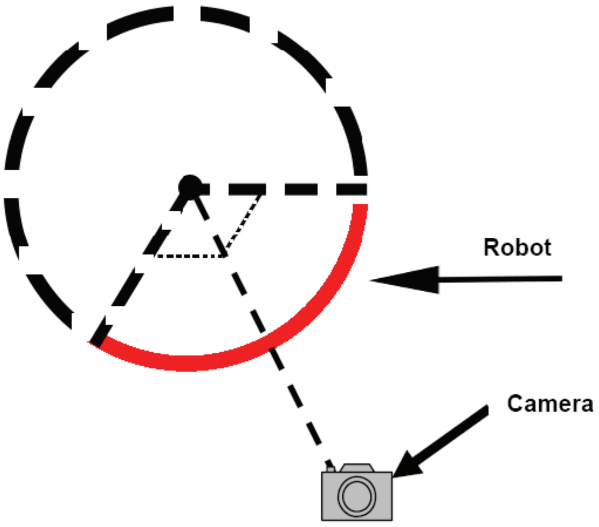


Fig. 13. Camera looks to the center of the circle

How to “move” the camera according to the steps of these algorithms? The image behavior in accordance with camera’s movements was studied. The effect of pan and tilt rotations on two points placed in a quadratic position on a circle was geometrically described. Coordinate transformation matrices corresponding to rotations with pan and tilt angles, respectively for perspective transformation were used. The variation of the distance between the two points, placed in a quadratic position on the circle, and the centre of the circle, depending of the tilt angle X, is plotted bellow in Fig. 14.

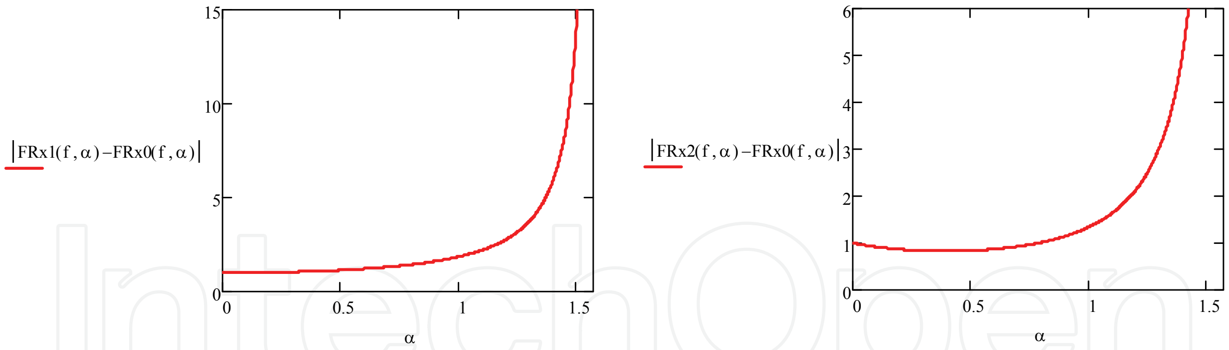


Fig. 14. Distance variation for quadratic positions

The variation of the ratio of the two distances is plotted bellow in Fig. 15a. The plot from Fig. 15b shows how is transformed a rectangle (inscribed in the circle and having the edges parallel with the axes OX and OY) when a tilt rotation is performed. Theoretically, by zooming, the distance between the two points varies in a linear way, as it is shown upper right.

The image’s segmentation is basically a threshold procedure applied to the image’s histogram. All the procedures included in the calibration algorithms were mathematically proven. If the calibration algorithm was successfully applied then the system is ready to perform the visual servoing tasks.

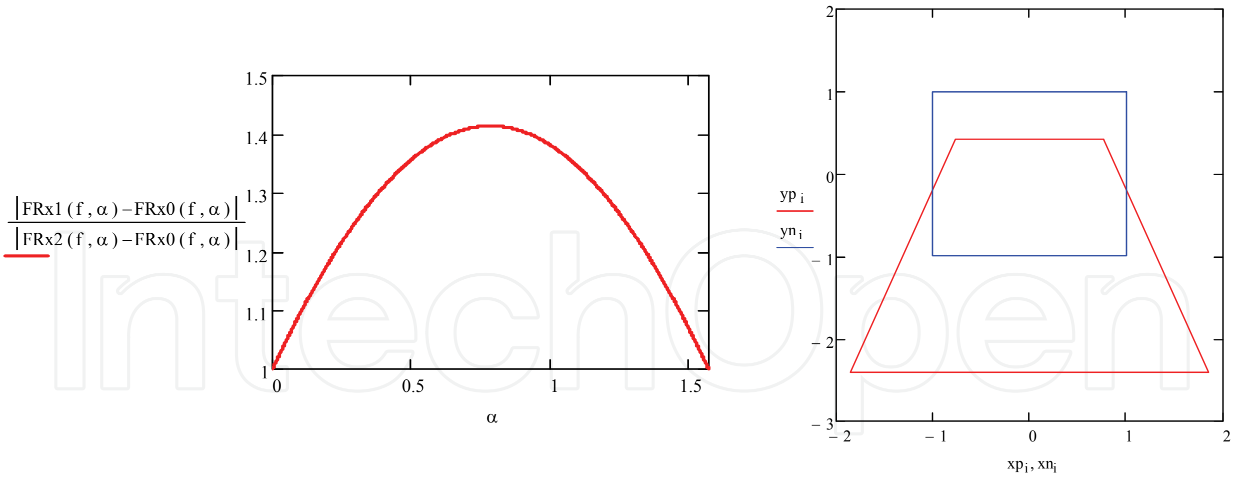


Fig. 15. a. Ratio distances variation b. Rectangle transformation and distance variation under zoom influence

5. A Compliance control of a hyper redundant robot

This section treats a class of hyper redundant arms can achieve any position and orientation in 3D space, and that can perform a coil function for the grasping. The arm is a high degree of freedom structure or a continuum structure, but in this chapter a different technological solution is assumed.

The general form of the arm is shown in Figure 16. It consists of a number (N) of elements, cylinders made of fibre-reinforced rubber.

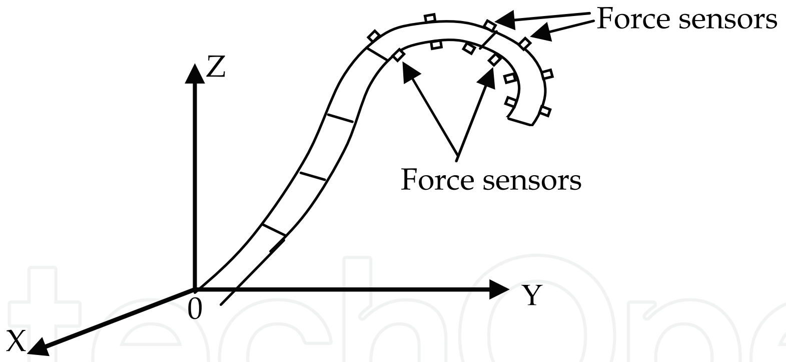


Fig. 16. The force sensors distribution

There are four internal chambers in the cylinder, each of them containing the ER fluid with an individual control circuit. The deformation in each cylinder is controlled by an independent electrohydraulic pressure control system combined with the distributed control of the ER fluid.

The last m elements ($m < N$) represent the grasping terminal. These elements contain a number of force sensors distributed on the surface of the cylinders. These sensors measure the contact with the load and ensure the distributed force control (Singh & Popa, 2005) during the grasping. The theoretical model is described as in Fig. 7 and equation (26)-(33).

For an element *dm*, kinetic and gravitational potential energy will be:

$$dT = \frac{1}{2} dm\left(v_x^2 + v_y^2 + v_z^2\right),\ dV = dm \cdot g \cdot z \tag{101}$$

where $dm = \rho \cdot ds$, and ρ is the mass density.

The elastic potential energy will be approximated by the bending of the element:

$$V_e = k\frac{d^2}{4}\sum_{i=1}^{N}\left(q_i^2 + \theta_i^2\right) \tag{102}$$

We will consider $F_\theta(s,t)$, $F_q(s,t)$ the distributed forces on the length of the arm that determine motion and orientation in the θ-plane, q-plane. The mechanical work is:

$$L = \int_0^l \int_0^t \left(F_\theta(s,\tau)\dot{\theta}(s,\tau) + F_q(s,\tau)\dot{q}(s,\tau)\right) d\tau ds \tag{103}$$

The energy-work relationship will be

$$L = \int_0^l \int_0^t \left(F_\theta(s,\tau)\dot{\theta}(s,\tau) + F_q(s,\tau)\dot{q}(s,\tau)\right) d\tau ds \tag{104}$$

where $T(t)$ and $T(0)$, $V^*(t)$ and $V^*(0)$ are the total kinetic energy and total potential energy of the system at time *t* and *0*, respectively.

In this chapter, the manipulator model is considered as a distributed parameter system defined on a variable spatial domain $\Omega = [0,\quad L]$ and the spatial coordinate s.

From (101-103), the distributed parameter model becomes,

$$\begin{aligned}
&\rho\int_0^S\int_0^S \left(\ddot{q}'\left(\sin q' \sin q'' \cos\left(q' - q''\right) + \cos q' \cos q''\right) - \ddot{\theta}' \cos q' \sin q'' \sin\left(\theta'' - \theta'\right) + \right.\\
&+ \left(\dot{q}'\right)^2\left(\cos q' \sin q'' \cos\left(\theta' - \theta''\right) - \sin q' \cos q''\right) + \left(\dot{\theta}'\right)^2 \cos q' \sin q'' \cos\left(\theta' - \theta''\right) - \\
&\left. - \dot{q}'\dot{q}'' \sin\left(q'' - q'\right)\right) ds' ds'' + \rho g\int_0^S \cos q' ds' + k^* q = F_q
\end{aligned} \tag{105}$$

$$\begin{aligned}
&\rho\int_0^S\int_0^S \left(\ddot{q}' \sin q' \cos q'' \sin\left(\theta'' - \theta'\right) + \ddot{\theta}' \cos q' \cos q'' \cos\left(\theta'' - \theta'\right) - \left(\dot{q}'\right)^2 \cos q' \cos q'' \sin\left(\theta'' - \theta'\right) + \right.\\
&\left. + \left(\dot{\theta}'\right)^2 \cos q' \cos q'' \sin\left(\theta'' - \theta'\right) - \dot{\theta}'\dot{q}' \sin q' \cos q'' \cos\left(\theta'' - \theta'\right)\right) ds' ds'' + k^* \theta = F_\theta
\end{aligned} \tag{106}$$

The control forces have the distributed components along the arm, $F_\theta(s,t)$, $F_q(s,t)$, $s \in [0,\quad L]$ that are determined by the lumped torques,

$$\begin{cases}
F_\theta(s,t) = \sum_{i=1}^{N} \delta(s - il)\tau_{\theta_i}(t) \\
F_q(s,t) = \sum_{i=1}^{N} \delta(s - il)\tau_{q_i}(t)
\end{cases} \tag{107}$$

where δ is Kronecker delta, $l_1 = l_2 = \ldots = l_N = l$, and

$$\tau_{\theta_i}(t) = \left(p_{\theta_i}^1 - p_{\theta_i}^2\right)S \cdot d/8 \tag{108}$$

$$\tau_{q_i}(t) = \left(p_{q_i}^1 - p_{q_i}^2\right)S \cdot d/8 \text{ , } i = 1,2,\ldots,N \tag{109}$$

In (107)-(108), $p_{\theta_i}^1$, $p_{\theta_i}^2$, $p_{q_i}^1$, $p_{q_i}^2$ represent the fluid pressure in the two chamber pairs, θ, q and S, d are section area and diameter of the cylinder, respectively (Fig. 17).

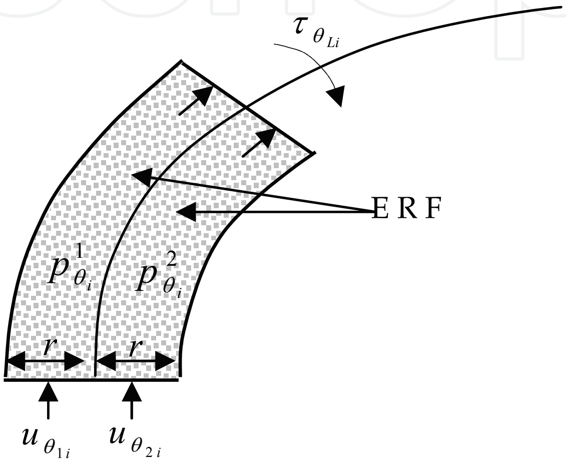


Fig. 17. The cylinder driving

The pressure control of the chambers is described by the equations:

$$a_{ki}(\theta)\frac{dp_{\theta i}^k}{dt} = u_{\theta ki} \tag{110}$$

$$b_{ki}(q)\frac{dp_{qi}^k}{dt} = u_{qki} \text{ , } k = 1,2 \text{ , } i = 1,2,\ldots,N \tag{111}$$

where $a_{ki}(\theta)$, $b_{ki}(q)$ are determined by the fluid parameters and the geometry of the chambers and

$$a_{ki}(0) > 0 \text{ , } b_{ki}(0) > 0 \tag{112}$$

The control problem of a grasping function by coiling is constituted from two subproblems: the position control of the arm around the object-load and the force control of grasping (Chiaverini et al., (1996). We consider that the initial state of the system is given by

$$\omega_0 = \omega(0,s) = \left[\theta_0, \quad q_0\right]^T \tag{113}$$

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

$$C_0 : \left(\theta_0(s), \quad q_0(s)\right) \text{, } s \in \left[0, \quad L\right] \tag{114}$$

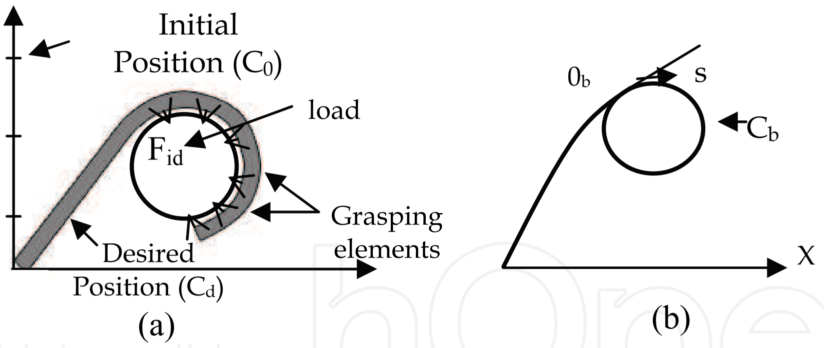


Fig. 18. (a) The grasping position; (b) The grasping parameters

The desired point is represented by a desired position, the curve C_d that coils the load,

$$\omega_d = \left[\theta_d, \quad q_d\right]^T \tag{115}$$

$$C_d : \left(\theta_d(s), \quad q_d(s)\right),\ s \in \left[0, \quad L\right] \tag{116}$$

In a grasping function by coiling, only the last m elements (m < N) are used. Let l_g be the active grasping length, where $l_g = \sum_{i=m}^{n} l_i$. We define by $e_p(t)$ the position error

$$e_p(t) = \int_{L-l_g}^{L} \left(\left(\theta(s,t) - \theta_b(s)\right) + \left(q(s,t) - q_b(s)\right)\right) ds \tag{117}$$

It is difficult to measure practically the angles θ, q for all $s \in \left[0, \quad L\right]$. These angles can be evaluated at the terminal point of each element. In this case, the relation (117) becomes

$$e_p(t) = \sum_{i=m}^{N} \left(\left(\theta_i(t) - \theta_{bi}\right) + \left(q_i(t) - q_{bi}\right)\right) \tag{118}$$

The error can also be expressed with respect to the global desired position C_d

$$e_p(t) = \sum_{i=1}^{N} \left(\left(\theta_i(t) - \theta_{di}\right) + \left(q_i(t) - q_{di}\right)\right) \tag{119}$$

The position control of the arm means the motion control from the initial position C_0 to the desired position C_d in order to minimize the error. An area reaching control problem is discussed. The desired area is specified by the inequality function:

$$f(\delta r) \leq 0 \tag{120}$$

where f is a scalar function with continuous first partial derivates, $\delta r = r_F - r_0$, $r_0 \in R^3$ is a reference point of the desired area and r_F is the position vector of the terminal point.
The potential energy function for the area reaching control has the form:

$$\tau_{\theta_i}(t) = -k_{\theta_i} e_{\theta_i}(t) - k^*_{\theta_i} \dot{e}^2_{\theta_i}(t) - max\left(0, \frac{\partial V_P^T}{\partial r} \cdot k_{P_{\theta_i}} a^*\left(\theta_i, q_i\right)\right) \tag{121}$$

Theorem 1. The closed-loop control system for the desired reaching area problem is stable if the control forces are

$$\tau_{\theta_i}(t) = -k_{\theta_i} e_{\theta_i}(t) - k^*_{\theta_i} \dot{e}^2_{\theta_i}(t) - max\left(0, \partial V_P^T / \partial r \cdot k_{P_{\theta_i}} a^*(\theta_i, q_i)\right) \tag{122}$$

$$\tau_{\theta_i}(t) = -k_{q_i} e_{q_i}(t) - k^*_{q_i} \dot{e}^2_{q_i}(t) - max\left(0, \partial V_P^T / \partial r \cdot k_{P_{q_i}} a^*(\theta_i, q_i)\right) \tag{123}$$

Theorem 2. The closed-loop control system of the position (107)-(108), (110)-(111) is stable if the fluid pressures control law in the chambers of the elements given by:

$$u_{\theta ji}(t) = -a_{ji}(\theta)\left(k^{j1}_{\theta i} \dot{e}_{\theta i}(t) + k^{j2}_{\theta i} \ddot{e}_{\theta i}(t)\right) \tag{124}$$

$$u_{qji}(t) = -b_{ji}(\theta)\left(k^{j1}_{qi} \dot{e}_{qi}(t) + k^{j2}_{qi} \ddot{e}_{qi}(t)\right) \tag{125}$$

where $j = 1,2$; $i = 1,2,\ldots,N$, with initial conditions:

$$p^1_{\theta i}(0) - p^2_{\theta i}(0) = \left(k^{11}_{\theta i} - k^{21}_{\theta i}\right) e_{\theta i}(0) \tag{126}$$

$$p^1_{qi}(0) - p^2_{qi}(0) = \left(k^{11}_{qi} - k^{21}_{qi}\right) e_{qi}(0) \tag{127}$$

$$\dot{e}_{\theta i}(0) = 0 \tag{128}$$

$$\dot{e}_{qi}(0) = 0 \text{ , } i = 1,2,\ldots,N \tag{129}$$

and the coefficients $k_{\theta i}$, k_{qi}, $k^{mn}_{\theta i}$, k^{mn}_{qi} are positive and verify the conditions

$$k^{11}_{\theta i} > k^{21}_{\theta i} \text{ ; } k^{12}_{\theta i} > k^{22}_{\theta i} \tag{130}$$

$$k^{11}_{qi} > k^{21}_{qi} \text{ ; } k^{12}_{qi} > k^{22}_{qi} \text{ , } i = 1,2,\ldots,N \tag{131}$$

The grasping by coiling of the continuum terminal elements offers a very good solution in the fore of uncertainty on the geometry of the contact surface. The contact between an element and the load is presented in Fig. 19. It is assumed that the grasping is determined by the chambers in θ-plane. The relation between the fluid pressure and the grasping forces can be inferred for a steady state from,

$$\int_0^l k \frac{\partial^2 \theta(s)}{\partial s^2} ds + \int_0^l f(s) \tilde{T} \tilde{\theta}(s) \int_0^s \tilde{T}^T \tilde{\theta}(s) ds = (p_1 - p_2) S \frac{d}{8} \tag{132}$$

where $f(s)$ is the orthogonal force on C_b, $f(s)$ is $F_\theta(s)$ in θ-plane and $F_q(s)$ in q-plane.
For small variation $\Delta\theta_i$ around the desired position θ_{id}, in θ-plane, the dynamic model (118) can be approximated by the following discrete model,

$$m_i \Delta\ddot{\theta}_i + c_i \Delta\dot{\theta}_i + H_i(\theta_{id} + \Delta\theta_i, \quad \theta_{id}, \quad q_d) - H(\theta_{id}, \quad q_d) = d_i(f_i - F_{ei}) \tag{133}$$

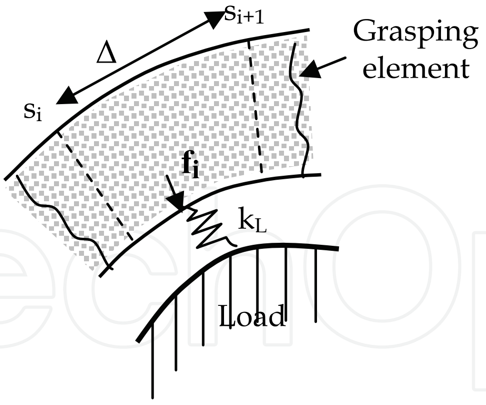


Fig. 19. The grasping force

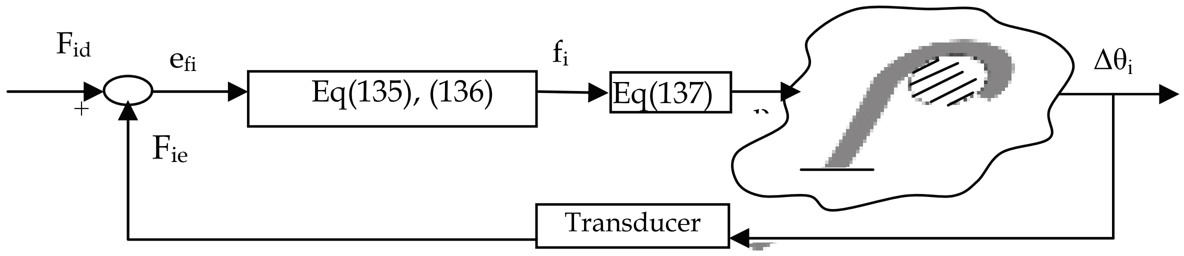


Fig. 20. The block scheme of the control system

where $m_i = \rho S\varDelta$, $i = 1,2,\ldots,l_1$. $H(\theta_{id},\ \ q_d)$ is a nonlinear function defined on the desired position $(\theta_{id},\ \ q_d)$, $c_i = c_i(\nu,\ \ \theta_i,\ \ q_d)$, $c_i > 0$, $\theta, q \in \Gamma(\Omega)$, where ν is the viscosity of the fluid in the chambers. The equation (133) becomes:

$$m_i \varDelta\ddot{\theta}_i + c_i(\nu,\ \ \theta_i,\ \ q_d)\varDelta\dot{\theta}_i + h_i(\theta_{id},\ \ q_d)\cdot \varDelta\theta_i = d_i(f_i - F_{ei}) \tag{134}$$

The aim of explicit force control is to exert a desired force F_{id} . If the contact with load is modelled as a linear spring with constant stiffness k_L , the environment force can be modelled as $F_{ei} = k_L \varDelta\theta_i$. The error of the force control may be introduced as

$$e_{fi} = F_{ie} - F_{id} \tag{135}$$

It may be easily shown that the equation (134) becomes

$$\frac{m_i}{k_L}\ddot{e}_{fi} + \frac{c_i}{k_L}\dot{e}_{fi} + \left(\frac{h_i}{k} + d_i\right)e_{fi} = d_i f_i - \left(\frac{h_i}{k} + d_i\right)F_{id} \tag{136}$$

Theorem 3. The closed force control system is asymptotic stable if the control law is

$$f_i = \frac{1}{k_L d_i}\left(\left(h_i + k_L d_i + m_i\sigma^2\right)e_{fi} - \left(h_i - k_L d_i\right)F_{id}\right),\ c_i > m_i\sigma \tag{137}$$

6. Conclusion

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 used cables and DC motors or 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.

The inverse kinematics problem is reduced to determining the time varying backbone curve behaviour. New methods for determining "optimal" hyper-redundant manipulator configurations based on a continuous formulation of kinematics are developed.

The difficulty of the dynamic control is determined by integral-partial-differential models with high nonlinearities that characterize the dynamic of these systems. First, the dynamic model of the system was inferred. The method of artificial potential was used for these infinite dimensional systems. In order to avoid the difficulties associated with the dynamic model, the control law was based only on the gravitational potential and a new artificial potential.

The control system is an image – based visual servo control. Servoing was based on binocular vision, a continuous measure of the arm parameters derived from the real-time computation of the binocular optical flow over the two images, and is compared with the desired position of the arm. The method is based on the particular structure of the system defined as a "backbone with two continuous angles". The control of the system is based on the control of the two angles. The error angle was used to calculate the spatial error and a control law was synthesized. The general control method is an image based visual servoing one instead of position based. By consequence, camera calibration based on intrinsic parameters is not necessary („calibration" in the classic sense of the term, not the one used in this paper). The term "camera calibration" in the context of this paper refers to positioning and orienting the two cameras at imposed values. This calibration is performed only at the beginning, after that the cameras remain still.

A new application investigates the control problem of a class of hyper-redundant arms with continuum elements that performs the grasping function by coiling. The control problem of a grasping function by coiling is constituted from two subproblems: the position control of the arm around the object-load and the force control of grasping.

7. Acknowledgement

The research presented in this paper was supported by the Romanian National University Research Council CNCSIS through the IDEI Research Grant ID93 and by FP6 MARTN through FREESUBNET Project no. 36186.

8. References

Blessing, B.; & Walker, I.D. (2004). Novel Continuum Robots with Variable- Length Sections, *Proc. 3rd IFAC Symp. on Mechatronic Systems*, Sydney, Australia, pp. 55-60.

Boccolato, G.; Dinulescu, I.; Predescu, A.; Manta, F.; Dumitru, S.; & Cojocaru, D.; (2010). 3D Control for a Tronconic Tentacle, *12th International Conference on Computer Modelling and Simulation,* p380-386, ISBN 978-0-7695-4016-0, Cambridge University, England.

Ceah, C.C. & Wang, D.Q. (2005). Region Reaching Control of Robots: Theory and Experiments, *Proceedings of IEEE Intl Conf on Rob. and Aut.*, Barcelona, pp. 986-991.

Chiaverini, C.; Siciliano, B. & Villani, L. (1996). Force and Position Tracking: Parallel Control with Stiffness Adaptation, *IEEE Control Systems*, Vol. 18, No 1, pp. 27-33.

Chirikjian, G.S. (1993). A continuum approach to hyper-redundant manipulator dynamics, *Proc. 1993 Int. Conf. on Intelligent Robots and Syst.*, Yokohama, Japan, pp. 1059 - 1066.

Cojocaru, D.; Ivanescu, M.; Tanasie, R.T.; Dumitru, S.; Manta, F. (2010), Vision Control for Hyperredundant Robots, *International Journal Automation Austria (IJAA)*, ISSN 1562-2703, IFAC-Beirat Österreich, Vol. 1, 18(2010), p52-66.

Cowan, L. S. & Walker, I.D., 2008. "Soft" Continuum Robots: the Interaction of Continuous and Discrete Elements, *Artificial Life X.*

Douskaia, N.V. (1998). Artificial potential method for control of constrained robot motion, *IEEE Trans. on Systems, Man and Cybernetics*, part B, vol. 28, pp. 447-453.

Ge, S.S.; Lee, T.H. & Zhu, G. (1996). Energy-Based Robust Controller Design for Multi-Link Flexible Robots, *Mechatronics*, No 7,Vol. 6, pp. 779-798.

Gravagne, I. D. & Walker, I.D. (2001). Manipulability, force, compliance analysis for planar continuum manip, *Proc. IEEE/RSI Intl. Conf. o Intell. Rob. and Syst.*, pp. 1846-1867.

Grosso, E.; Metta, G. & a.o. (1996). Robust Visual Servoing in 3D Reaching Tasks, *IEEE Transactions on Robotics and Automation*, vol. 12, no. 15, pp. 732-742.

Hannan, M.W. & Walker, I.D. (2005). Real-time shape estimation for continuum robots using vision, *Robotica*, volume 23, pp. 645–651.

Hemami, A, (1984). Design of light weight flexible robot arm, *Robots 8 Conference Proceedings*, Detroit, USA, pp. 1623-1640.

Hirose, S. (1993). *Biologically Inspired Robots*, Oxford University Press.

Hutchinson, S.; Hager, G. D. & Corke, P. F. (1996). A Tutorial on Visual Servor Control, *IEEE Transactions on Robotics and Automation*, vol. 12, no. 15, pp. 651-670.

Immega, G. & Antonelli, K. (1995). The KSI Tentacle Manipulator. *Proc. 1995 IEEE Conf. on Robotics and Automation*, pp. 3149-3154.

Ivănescu, M.; Cojocaru, & a.o. (2006). Hyperredundant Robot Control by Visual Servoing, *Studies in Informatics and Control Journal*, Vol. 15, No. 1, ISSN 1220-1766, p93-102.

Ivanescu, M. (2002). Position dynamic control for a tentacle manipulator, *Proc. IEEE Int. Conf. on Robotics and Automation*, Washington, A1-15, pp. 1531-1539.

Kelly, R. (1996), Robust Asymptotically State Visual Servoing, *Proceedings IEEE Inernational Conference on Robotics and Automation*, vol. 22, no. 15, pp. 759-765.

Masoud, S. A. & Masoud, A.A. (2000). Constrained motion control using vector potential fields, *IEEE Trans. on Systems, Man and Cybernetics*, part A, vol. 30, pp. 251-272.

Mochiyama, H. & Kobayashi, (1999). H. The shape Jacobian of a manip with hyper degrees of freedom, *Proc. 1999 IEEE Intl. Conf. on Rob. and Autom.*, Detroit, pp. 2837- 2842.

Robinson, G. & Davies, J. B. C. (1999). Continuum robots—A state of the art. *In IEEE International Conference on Robotics and Automation*, pages 2849–2854. Detroit, MI.

Singh, S.K. & Popa, D.O. (2005) An Analysis and Some Fundamental Problems in Adaptive Control of Force, *IEEE Trans. on Robotics and Automation*, Vol. 11 No 6, pp 912-922.

Suzumori, K.; Iikura, S.; & Tanaka, H. (1991). Develop. of flexible microactuator and its appl. to robot mech, *IEEE Intl. Conf. on Rob. and Autom.*, Sacramento, pp. 1564 - 1573.

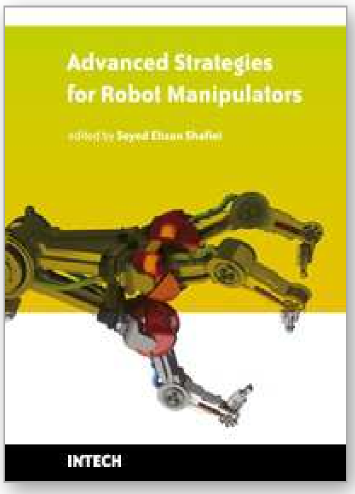
Takegaki, T.; & Arimoto, S. (1981). A new feedback methods for dynamic control of manipulators, *Journal of Dynamic Systems, Measurement and Control*, pp. 119-125.

Tanasie, R.T.; Ivănescu, M. & Cojocaru, D. (2009). Camera Positioning and Orienting for Hyperredundant Robots Visual Servoing Applications, *Journal of Control Engineering and Applied Informatics*, ISSN 1454-8658, Vol 11, No 1, p19-26.

Walker, I.D., Dawsona, D.M. & a.o. (2005), Continuum Robot Arms Inspired by Cephalopods, DARPA Contr. N66001-C-8043, http://www.ces.clemson.edu.

Walker, I.D. & Carreras, C. (2006) Extension versus Bending for Continuum Robots, *Internl. Journal of Advanced Robotic Systems*, Vol. 3, No.2, ISSN 1729-8806, pp. 171-178.

Wang, P.C. (1965). Control of distributed parameter systems, *Advance in Control Systems*, Academic Press.


Advanced Strategies for Robot Manipulators
edited by Seyed Ehsan Shafiei
INTECH

Advanced Strategies for Robot Manipulators
Edited by S. Ehsan Shafiei

ISBN 978-953-307-099-5
Hard cover, 428 pages
Publisher Sciyo
Published online 12, August, 2010
Published in print edition August, 2010

Amongst the robotic systems, robot manipulators have proven themselves to be of increasing importance and are widely adopted to substitute for human in repetitive and/or hazardous tasks. Modern manipulators are designed complicatedly and need to do more precise, crucial and critical tasks. So, the simple traditional control methods cannot be efficient, and advanced control strategies with considering special constraints are needed to establish. In spite of the fact that groundbreaking researches have been carried out in this realm until now, there are still many novel aspects which have to be explored.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821