

# TOWARDS FUZZY CONTROL OF A FLEXIBLE MANIPULATOR

Jürgen Van Gorp

V.U.B. Vrije Universiteit Brussel, Pleinlaan 2, B-1050 Brussels, Belgium

e-mail: jvgorp@vnet3.vub.ac.b

**Abstract** - This paper gives an overview of work that has been done on the modelling and control of robot arms. A search has been done on the control of rigid manipulators, single-link flexible manipulators, and combinations of these two types. Further is the work discussed that has already been done on the concatenation of multiple-link rigid and flexible manipulators. This will lead to the problem of hyper-redundancy of manipulators. After this large introduction this paper will discuss the modelling of a robot arm based on a long nonlinear flexible spring. The dynamic equations of the model include gravity forces and external forces on the spring. The model is based on a local linearization of the spring parameters. The technique is illustrated with simulations that illustrate the seemingly natural movement of the spring model.

## I. INTRODUCTION

A specific problem of flexible robot arms is the difficulty of designing a controller. In the past robot arms were considered to be rigid, so that classical linear controllers could be used for the control. In recent years many studies have been done on light, flexible robot arms and robot arms that had unknown parameters in them, such as an unknown payload.

The first goal of this paper is to give a small overview of work that has been done on the modelling and control of robotic manipulators. First single and multiple link rigid manipulators, and some work on single link flexible links is handled. It is possible to combine rigid and flexible links, which will lead us to the two-link rigid-flexible manipulators. Two other cases of nonlinear manipulators are treated: pneumatic manipulators, where the nonlinearities and hysteresis lay mostly in the pneumatic actuators, and flexible joint manipulators, where the joints are considered as flexible. Finally it is possible to combine large numbers of flexible links into one large chain. The control of such chains is currently impossible on an analytical base, moreover because the chain tends to hyper-redundancy.

In the next chapter a model for a flexible structure is introduced, based on a spring. The model includes the influence of gravity and the dynamic behaviour. A possible way of controlling the spring using pulling cables is suggested and included in the model. Finally some simulations based on the model, are shown and concluding remarks are given on a future design of a controller.

## II. STUDY OF LITERATURE ON THE MODELING AND CONTROL OF MANIPULATORS

### A. Single and Multiple Segment Rigid Manipulators

Models of rigid manipulators are usually linearized to overcome the non-linear movements. After linearization standard linear feedback control schemes can be used.

To obtain a robust control of a rigid manipulator, in [32] the rigid robotic manipulator is treated as a

partially known system. Known dynamics are separated out and used for linearization. For an  $n$ -joint rigid robotic manipulator, the dynamics are given in [32] by the equation

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{u}(t) \quad (1)$$

with  $\mathbf{q}(t)$  the  $n \times 1$  vector of angular positions of the joints.  $\mathbf{M}(\mathbf{q})$  is the  $n \times n$  symmetric positive definite inertia matrix.  $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})$  is an  $n \times 1$  vector containing the coriolis and centrifugal forces and gravity torques, and  $\mathbf{u}(t)$  is an  $n \times 1$  vector containing the joint control inputs, regarded as torques.

For this system a linearized, asymptotically convergent error system is chosen, based on a feedback control law  $\mathbf{u}(t)$ . The proof of stability for the feedback controller was given using a Lyapunov function and sliding modes.

If the robot manipulator has bounded parameter variations, [26] presents an adaptive control scheme. For this scheme the signal boundedness of the closed-loop system can be proven. It is also proven that the tracking error  $e(t)$  can be finite and small.

[10] starts from the presumptions that the joint displacements can be measured precisely, whereas velocity measurements are more noise-prone. An approach by which the state vectors are estimated is suggested, using nonlinear sliding mode observers. Observers for both the frictionless as the friction case are designed.

A step further is taken in [29] where the motion and force control problem is introduced for multiple manipulators that cooperatively handle one single object. The problem was analysed and multiple algorithms for control were treated. Arms and grasps are considered to be rigid, as well as an optional payload carried by the manipulator.

Other control strategies are given in [28], [23], [8] and [1]. In all cases rigid manipulator structures were assumed. [30] presents a unified formulation for the control scheme of a constrained motion control. The assumptions are both a rigid manipulator and a frictionless rigid surface. The proposed general target model can then be used for motion control, as well as hybrid position and force control and constrained motion control.

In case that the compliant surface is elastic, [6] suggests a combination of a PD action on the position loop and a PI action on the force loop. The work also includes the effects of gravity. Asymptotic stability was proven using the Lyapunov method.

An alternative way of controlling the robot movement is by using neural networks, as presented in [24]. A neural controller was implemented to control the nonlinear action of a rigid two-link robot. The goal of the controller was the achievement of a good trajectory control. The training of the controller could be done without any knowledge of the model parameters, but the major drawback was the long time needed to train the controller, and the large number of nodes in the neural network.

## B. Single-Link Flexible Manipulators

Light-weight robotic manipulators have the advantage that the manipulators can have higher accelerations and control can be done faster. The prime drawback is the flexibility of the arm, causing vibrations that can lead to an unstable control.

Many papers discuss the modelling and control of flexible single-link manipulators (see [12], [13], [15], [16], [17], [20] and [25]). [18] uses an end mass connected to a flexible beam. The bend is measured using strain gauges.  $I_b$  is called the rotational inertia,  $R_o A$  the unit length mass of the manipulator and  $I_h$  is the inertia of the driving motor system. The end mass,  $M_e$ , has a rotational inertia  $J_e$ .

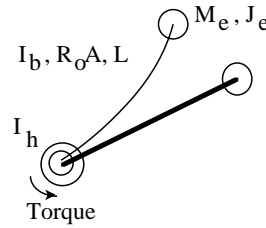


Fig. 1: single-link flexible manipulator

From the dynamic equations, a torque profile was calculated that did not take in account the high-frequency modes of the flexible manipulator. This led to an acceleration profile for the controller that went close to a bang-bang control. Nevertheless the controller was able to follow the trajectory path exactly, without over- or undershoot, or noticeable residual oscillations.

As a comparison, the control was also done using a PD controller for a step-input command. This resulted in a high peak torque, excited the system's natural frequencies and therefore needed a long settling time.

In [14] the manipulator is regarded as a flexible beam with an unknown mass and as such, whole system as considered a second-order spring-mass system. [14] measures the bend of the arm using strain gauges on an arm that was designed to be rigid in the vertical plan and flexible in the horizontal plane.

The mathematical representation of the system was done straightforward, as only a second order system was assumed. The controller was designed as a single-input, single-output, parametric observer that was on-line adapted by changing the parameters that represent the natural frequency of the arm.

The control was tested on an experimental setup, and was near to optimal, provided that all parameters were known. Still a tracking error was observed and to improve control performance [14] concluded that the observer-controller should be redesigned using a distributed parameter modelling.

[16] did not measure the torque, but only the angle of the shaft that carried a flexible beam. In the paper a non-reduced model of beam-shaft system is used, making use of fixed, polar coordinates. The

shear force was considered normal to the beam.

With  $r$  the radius from the shaft to the end point,  $\theta$  the angle and  $\rho$  the mass per unit of length, the following equation of motion was obtained:

$$\rho r \ddot{\theta} + (EI) \frac{\partial^4 (\theta - \theta_0) r}{\partial r^4} = 0 \quad (2)$$

in which  $EI$  is the rigidity.

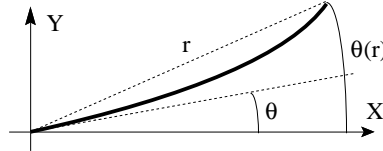


Fig. 2: simple bending of a flexible beam

This representation of the beam was rewritten and put in a closed loop with the model of the actuating motor and a controller. It was stated that only a spatially distributed feedback loop could bring the entire plant under control, but that such is not implementable in practice.

The design problem of the controller was based on the design of a prefilter  $F$  and a compensator  $G$ , such that all poles were in the left half of the complex plane, bounded away from the imaginary axis and as such leading to exponential stability.

This pole placement method left only a very small tolerance on the flexural rigidity and placement of an end mass on the beam. Moreover this design methodology leads to the need for an extremely high gain feedback in the closed loop system. This was solved by satisfying the design criteria only in a very small bandwidth range:  $0 \leq \omega \leq 1$ .

Despite this very small frequency range, the behaviour of the system was satisfying. As was the case in [18], an initial oscillation and exponential stability was experienced. The control was not possible for frequencies that went beyond  $\omega = 2$ .

A  $H^\infty$  control scheme to control a flexible beam, is proposed in [19]. The feedback controller is designed for a highly flexible Euler-Bernoulli beam, considered as an FDLTI (Finite Dimensional, Linear, Time-Invariant) system.

The considered Euler-Bernoulli beam has free ends and a Kelvin-Voigt damping coefficient  $\varepsilon > 0$ . The mass density  $\rho$  and the stiffness parameter  $EI$  are both chosen as being 1. The deflection of the beam at time  $t$  and location  $x$  is denoted by  $w(x, t)$ , under influence of a force  $-u(t)$  at location  $x = 1$ . In that case the dynamic equations of the beam are given as:

$$\frac{\partial^4 w}{\partial x^4} + \varepsilon \frac{\partial^5 w}{\partial x^4 \partial t} + \frac{\partial^2 w}{\partial t^2} = 0 \quad (3)$$

with the following boundary conditions:

$$\frac{\partial^2 w}{\partial x^2}(0, t) + \varepsilon \frac{\partial^3 w}{\partial x^2 \partial t}(0, t) = 0 \quad (4)$$

$$\frac{\partial^2 w}{\partial x^2}(1, t) + \varepsilon \frac{\partial^3 w}{\partial x^2 \partial t}(1, t) = 0 \quad (5)$$

$$\frac{\partial^3 w}{\partial x^3}(0, t) + \varepsilon \frac{\partial^4 w}{\partial x^3 \partial t}(0, t) = 0 \quad (6)$$

$$\frac{\partial^3 w}{\partial x^3}(1, t) + \varepsilon \frac{\partial^4 w}{\partial x^3 \partial t}(1, t) = u(t) \quad (7)$$

From these assumptions, a  $H^\infty$ -optimal controller was designed that was robust to a small, unmodelled delay.

In the particular case of a flexible beam rotating with a constant angular velocity, and moving in a plane, a control strategy is proposed in reference [22]. The beam is clamped on one side and has a bounded free motion on the other side. By applying a suitable boundary control force and torque at the free end of the beam, the asymptotic stability of the controller is proven, but only for sufficiently small angular frequencies and disturbance frequencies.

In the particular case that the flexible beam has a nonlinear triangular structure, the canonical forms were determined in [25]. Systems are divided into two forms  $\tau_1$  and  $\tau_2$ , which respectively correspond to a set of first- and second-order differential equations. First the stabilization of  $\tau_1$  systems was achieved, whereafter  $\tau_2$  systems were characterized. Then a way was proposed to develop adaptive controllers to control tracking problems of nonlinear triangular systems.

### C. Two-Link Rigid-Flexible Manipulators

Systems can have both flexible and rigid manipulator properties. For rigid-flexible systems the equations of motion can be described by hybrid ordinary-partial differential equations. Spatial discretization techniques (e.g. finite elements, assumed modes, or Galerkin's method) are typically employed to obtain a finite dimensional system of ordinary differential equations [13].

In [13] the equations for a constrained rigid-flexible robot arm are derived. The robot arm is considered as a set of nonlinear hybrid ordinary-partial differential equations, with an algebraic constraint equation. The latter equation is derived from the property that the robot arm can only be used in a constrained space. To derive the equations of motion, Hamilton's Principle was used, starting from the energy equation:

$$\int_{t_1}^{t_2} \delta(T - V) dt + \int_{t_1}^{t_2} \delta W dt = 0 \quad (8)$$

where  $V$  is the potential energy,  $T$  is the kinetic energy and  $W$  the virtual work.

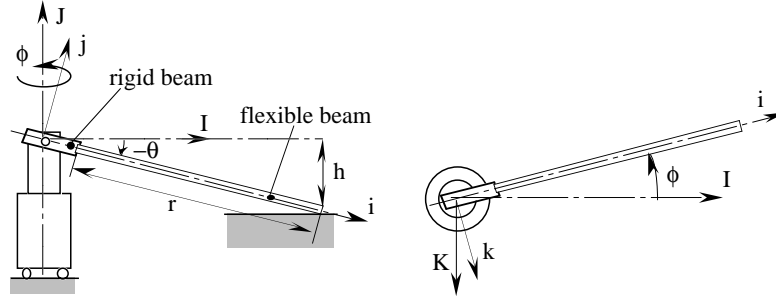


Fig. 3: Flexible robot manipulator with constraint surface

$(I, J, K)$  is an inertial frame, whereas  $(i, j, k)$  is an rotating reference frame, attached to the rigid beam. When the flexible beam is moved, it will bend, leading to vibrations in the flexible beam. The end point of the arm is driven to follow a trajectory  $S(t)$ .

The beam spins with an angle  $\theta$  around the vertical axis and tilts with an angle  $\phi$ , driven by two torques  $\tau_\theta$  and  $\tau_\phi$ .  $r$  is assumed to be constant. In [13] the modes of vibration and the deflections of the beam, due to a force at the end, are calculated and a model of the complete arm is proposed.

[31] also considered a rigid-flexible robot arm. In his case the flexible and the rigid parts were separated by use of an elbow.

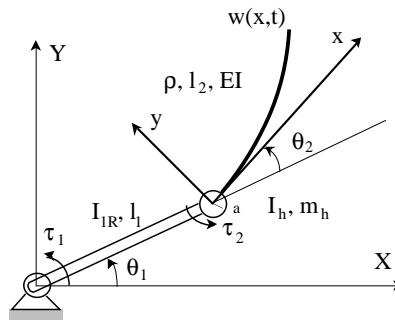


Fig. 4: Rigid-flexible manipulator with elbow

The aim of the work was the analysis of the stability of a PD controller for this problem. [31] starts from the full equations of motion of the manipulator, but concludes that an analytical solution can not be found. Simulations were done with approximate results using Galerkin's method.

Just as was the case in [18], in [31] was discovered that the PD controller had a significant overshoot.

The elastic link of the actuator had a significant damped oscillation. Nevertheless he could prove the global stability of the controller-plant system.

In most examples the gravity is not taken into account. Either the influence of gravity was negligible, or the given examples were pure academic. [9] introduced gravity in a robot arm that had a rigid and a flexible part.

The payload of the arm was unknown, from which was concluded that a model-based compensation would not give a good solution to the control problem. When applied on flexible structures, PID was not a good solution either, as no global convergence proof exists. Therefore a simple iterative scheme for the gravity compensation was suggested.

De Luca and Siciliano proved in 1992 that it is possible to asymptotically stabilize any arm configuration with a simple PD feedback controller on every joint position. In [9] the first iteration of the controller design was thus the development of a PD control on every joint, while the gravity term was only estimated. During further iterations an additional feedforward term was updated. The global stability of this method was proven and an example was given for a two-link rigid-flexible robot-arm.

#### D. Pneumatic Non-Linear Manipulators

In [12] a 200 neurons Manifold Representing Neural Network was used to control a robot arm. Two video cameras provided a four-dimensional system of coordinates. The robot arm was pneumatically driven, leading to a nonlinear and nonseparable control with hysteresis. After 200 learning steps a position control of about one pixel was achieved.

The goal of the study was twofold: a position control and the more advanced task of grasping an object. The used robot was a four-link manipulator with five degrees of freedom. All links were considered to be stiff, but due to the pneumatic actuators a model could not easily be constructed.

This lead to the necessity of an adaptive algorithm for the calculation of the desired vector of pressures,  $\mathbf{P}(\mathbf{u})$  :

$$\mathbf{P}(\mathbf{u}) = \mathbf{P}_k + \mathbf{A}_k(\mathbf{u} - \mathbf{w}_k) \quad (9)$$

where  $k$  is the label of the neuron closest to  $\mathbf{u}$ , the target point of the actuator.  $\mathbf{A}_k$  is a  $3 \times 4$  matrix giving a linear correction of pressure according to the deviation  $\mathbf{u} - \mathbf{w}_k$  where  $\mathbf{w}_k$  is the location matrix assigned to neuron  $k$ . The pressure vector  $\mathbf{P}_k$  is chosen independently in such a way that it moves the end effector to the point  $\mathbf{w}_k$ . [12] starts from this equation to update the pressure matrices:

$$\mathbf{P}_k^{\text{new}} = \mathbf{P}_k^{\text{old}} + \gamma(r, t)\mathbf{A}_k(\mathbf{u} - \mathbf{v}_i) \quad (10)$$

in which  $\gamma(r, t)$  is an arbitrary chosen value that decreases with time  $t$ .  $\gamma$  also decreases for neurons that only very little match the end position, so that these neurons are less updated.  $\mathbf{v}_i$  is the position of

the arm based on an average pressure  $\bar{P}(\mathbf{u})$  calculated from several neurons that are close to neuron  $k$ .

About the same setup was used in [5] to test a Fuzzy controller. The pneumatic system consisted of a single servo valve with the inherent nonlinearities and hysteresis.

Although a fuzzy controller was used in [5], the initial rules were based on fuzzy PID control. With  $e$  the error between the output of the control system and the reference input,  $E_i$ ,  $IE_i$  and  $D_i$  the fuzzy sets corresponding to  $x$ ,  $\dot{x}$  and  $\int x dt$  respectively, an example fuzzy rule looks like:

$$\begin{aligned} & \text{IF}(x \text{ is } E_i \text{ and } \int x dt \text{ is } IE_i \text{ and } \dot{x} \text{ is } D_i) \\ & \text{THEN } U(i) = K_p \mu_{E_i}(x)e + K_i \mu_{IE_i}(\int x dt) \int e dt + K_d \mu_{D_i}(\dot{x}) \dot{e} \\ & \text{AND } \mu(U(i)) = \mu_{E_i}(x) \wedge \mu_{IE_i}(\int x dt) \wedge \mu_{D_i}(\dot{x}) \end{aligned} \quad (11)$$

$\mu_{E_i}(x)$ ,  $\mu_{IE_i}(\int x dt)$ ,  $\mu_{D_i}(\dot{x})$  and  $\mu(U(i))$  are the membership values corresponding to  $x$ ,  $\int x dt$ ,  $\dot{x}$  and  $U(i)$  respectively.  $U(i)$  is the output of the  $i$ -th rule.  $K_p$ ,  $K_i$  and  $K_d$  correspond to the constants of a classical PID controller.

The membership value  $\mu(U(i))$  is defined as:

$$\mu(U(i)) = \min \left\{ \mu_{E_i}(x), \mu_{IE_i}(\int x dt), \mu_{D_i}(\dot{x}) \right\} \quad (12)$$

while the defuzzification was done using the following formula:

$$U = \frac{\sum_{i=1}^n U(i)}{\sum_{i=1}^n \mu(U(i))} \quad (13)$$

with  $U$  the defuzzificated control input.  $n = n_e \times n_{ie} \times n_d$  with  $n_e$ ,  $n_{ie}$  and  $n_d$  the number of fuzzy sets corresponding to  $x$ ,  $\int x dt$  and  $\dot{x}$  respectively.

For the complete control, a reference model was used and the controller was adapted based on the following scheme:

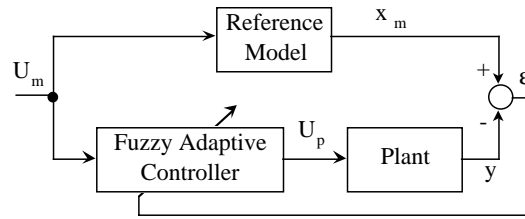


Fig. 5: Fuzzy adaptive control system

The adaption implied the variation of the  $K_p$ ,  $K_i$  and  $K_d$  constants. The gain of using a fuzzy controller lies in the insensitivity of the controller to an external load, and the ability to control a nonlinear system.

## E. Flexible Joint Manipulators

A special type of flexible manipulators are those based on n-degree-of-freedom direct drive revolute flexible joint robots. In [4] research was done on flexible joint robots, with the goal of designing a stable controller for direct drive robots. The links were assumed to be rigid, so that all of the flexibility was concentrated in the joints. This led to the assumption that the system consisted of a slow subsystem, controllable by a slow controller  $u_s$ , and a fast subsystem that can be asymptotically stabilized by means of a fast control  $u_f$ .

The simulations showed that the control strategy was unstable when the flexibility of the joints was ignored. With the controller they proposed, a reasonably good control was possible. The major weak point was that natural frequencies of the manipulator must be known in advance.

[17] noted further that the use of PD regulators yields a performance degradation due to noise, and that the numerical differentiation was inaccurate for both low and high speeds. Starting from the flexible joint robot model, they established the global asymptotic stability, using approximate differentiation, i.e. by replacing the derivative operator  $p$  by the high pass filter  $\frac{bp}{p+a}$ .

The simulation results showed that velocity measurements could be replaced by the approximate differentiation and that no restrictions were necessary on the bandwidth of the filters.

## F. Chaining Flexible Links

Many flexible links can be concatenated to form large chains. [2] gives a more detailed description of how to do this using Hamilton's principle.  $N$  links with length  $l_n$ , mass  $m_n$  and mass density  $\rho_n(x)$  are described by the position vectors  $p_{A_n}(t)$ .  $n$  is the number of the link,  $t$  the time variable and  $\theta_n$  the angle of the link with the horizontal axis. From each link the dynamic equations are formulated and then combined to form the equations of the chain.

Hamilton's principle states that in a given time interval  $[t_1, t_2]$ , a functional  $\mathfrak{S}$  can be found, with:

$$\mathfrak{S}(\theta_1, \dots, \theta_N, w_1, \dots, w_N) = \int_{t_2}^{t_1} L dt \quad (14)$$

In this equation  $w_n$  is the deviation of the middle of a link with respect to a straight line connecting its begin and end point.  $L$  is the Lagrangian of the chain, with

$$L = L^0 + \tilde{L} \quad (15)$$

where

$$L^0 = \sum_{n=1}^N L_n^0 \quad \tilde{L} = \sum_{n=1}^N \tilde{L}_n \quad (16)$$

$L_n^0$  is the Lagrangian of the undeformed link (case  $w_n \equiv 0$ ) and  $\tilde{L}_n$  the Lagrangian density due to deformation.

Starting from these equations, in [2] the lengthy mathematical representation of the dynamic chain behaviour were derived.

[20] used  $m$  actuated chains with one single common reference member. Each chain can have an arbitrary number of links and degrees of freedom.

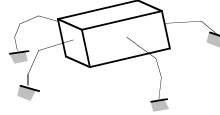


Fig. 6:  $n$  chained links with one common reference

The links connect the reference member with its environment. The method can be used for the modelisation of moving objects. The chains are given by their dynamic equations of motion and have a  $6 \times 1$  spatial force vector imposed by the link on the reference member. From these equations a method for the calculation of the motion of a body was developed.

## G. Hyper-Redundant Manipulators

Hyper-redundant structures have a large or infinite degree of kinematic and mathematical redundancy. An analytical solution for obstacle avoidance or locomotion of hyper-redundant structures is therefore impossible.

As a solution to hyper-redundancy, [7] proposes the use of so called backbone curves. A backbone

curve is “a piecewise continuous curve that captures the important macroscopic geometric features of a hyper-redundant robot” [7]. The mode and formal structure of a curve can be calculated from given constraints that define the task. The manipulator can switch between different modes and each section of the manipulator can be driven to follow the calculated formal structure. The use of backbones is explained with an example of a 30 degrees-of-freedom snake-like robot.

In the three-dimensional space, an extra rotation function  $\mathfrak{R}(s, t)$  is introduced, where  $s$  is a distance parameter and  $t$  denotes time.  $\mathfrak{R}$  describes how a structure curls around the backbone curve, and can be used as an extra constraint. In the two dimensional-space it is sufficient to give the end-point coordinates  $x_{ee}$  and  $y_{ee}$ , and the angle  $\theta_{ee}$  under which the backbone curve should reach the end point.

### III. MODELING OF A HIGHLY FLEXIBLE SPRING MANIPULATOR

#### A. Goal Of The Modelling

In [7] rigid sections were used to obtain a planar 30 degree-of-freedom snake-like robot. We tend to introduce an alike multi-section robot arm, based on a spring. Each section is flexible by nature.

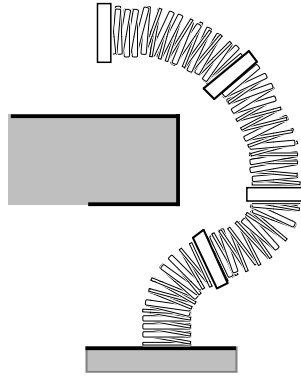


Fig. 7: hyper-redundant flexible structure

Springs are commonly modelled as linear structures or torsion bars [3] [11] [15] [21] [27]. We will model the spring as a macroscopic structure to avoid the lengthy calculations that come with the use of finite element methods. Since the spring is used in its far nonlinear state, a global linearization is not possible.

#### B. Dynamic Equations Of The Spring

In the proposed model, the spring is divided into an arbitrary number of subsections. The spring parameters are locally linearized, without extensive linearization of the dynamic equations themselves. Due to an external system of forces  $X_{N+1}$  and  $Y_{N+1}$  and a Couple  $K_{N+1}$  the spring is bent into a non-linear state.

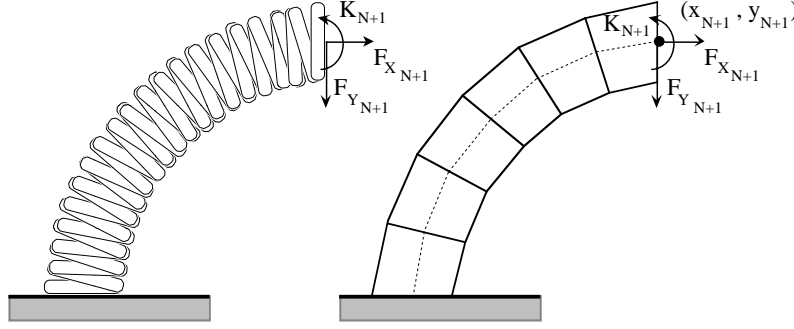


Fig. 8: non-linear bend and piecewise linearization

A two dimensional position control of the spring can be done by mounting two cables on the inner side of the spring, allowing the spring to make a planar move. The cables are attached to each section of the spring in such a way that only normal forces can be applied to the sections. If we consider the section  $n$  with  $0 \leq n \leq N$  and  $N$  the total number of sections of the spring, then the normal forces on the section can be written as:

$$F_k^n = 2F_k \sin\left(\frac{\theta_{n+1}}{2}\right) \quad (17)$$

$$F_l^n = 2F_l \sin\left(\frac{\theta_{n+1}}{2}\right) \quad (18)$$

with  $F_k$  and  $F_l$  the forces from respectively the left and right pulling rope, and  $\theta_{n+1}$  the local bending angle of the subsequent section.

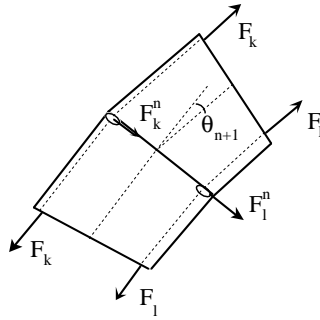


Fig. 9: normal forces induced by two pulling ropes

In this model the friction forces from the cable movements in the spring, were not considered, and are left for a future model. In global coordinates, section  $n$  is bent over an angle  $\xi_n$ , with the vertical axis taken as reference, with:

$$\xi_n = \sum_{i=0}^{n-1} \theta_i \quad (19)$$

The mass of the section is assumed to be concentrated in the centre of gravity, so that the section can be

reduced to the centre line of the spring. The section is contracted under influence of the forces applied by the  $(n + 1)$ -th section, the internal forces, and the resulting forces  $X_n$  and  $Y_n$  that are applied by the  $(n - 1)$ -th section. As a result of the torque  $K_n$  the central line bends over an angle  $\theta_n$ .

The sections that come before section  $n$  will cause the section to translate from location  $(x_1, y_1)$  to a location  $(x_2, y_2)$ , which will introduce the forces  $m\ddot{x}_n$  and  $m\ddot{y}_n$ . The model includes the gravitational force  $mg$  with  $m$  the mass of the section, and the friction forces  $k_a\dot{s}_n$  and  $k_w\dot{\theta}_n$ , with  $k_a$  and  $k_w$  the friction constants of respectively contraction and bending.  $s$  is the length of the central line of the section. The inertia of the bend of the central line is put into the model as an extra couple  $I\ddot{\theta}_n$  with the inertial constant equalling  $I = \frac{1}{3}ms_n^2$ .

For geometry reasons, the sides of the section are assumed to bend over with the same angle  $\theta_n$  as the central line. The section can be drawn in its own local coordinates as:

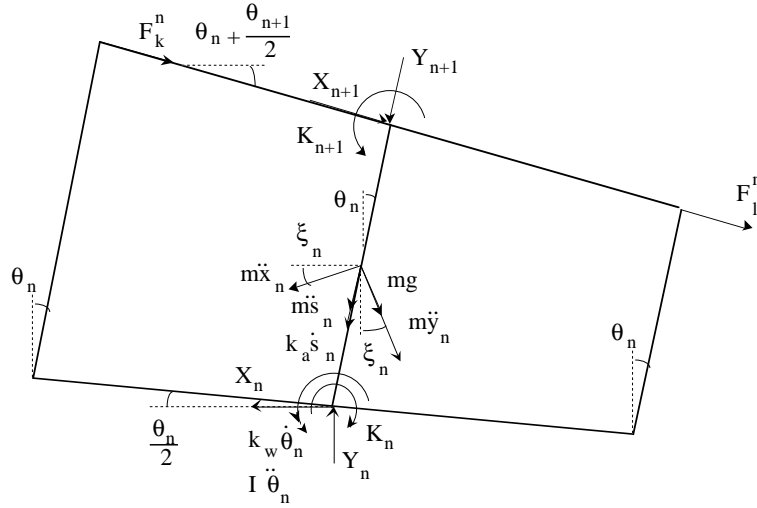


Fig. 10: section of the spring

This will lead us to the following dynamic equations:

$$\begin{aligned} X_n + m\ddot{x}_n \cos(\xi_n) - (mg + m\ddot{y}_n) \sin(\xi_n) + (m\dot{s}_n + k_a\dot{s}_n + Y_{n+1} + F_k + F_l) \sin(\theta_n) \\ - X_{n+1} \cos(\theta_n) - (F_k + F_l) \sin(\theta_n + \theta_{n+1}) = 0 \end{aligned} \quad (20)$$

$$\begin{aligned} -Y_n + m\ddot{x}_n \sin(\xi_n) + (mg + m\ddot{y}_n) \cos(\xi_n) + (m\dot{s}_n + k_a\dot{s}_n + Y_{n+1} + F_k + F_l) \cos(\theta_n) \\ - X_{n+1} \sin(\theta_n) - (F_k + F_l) \cos(\theta_n + \theta_{n+1}) = 0 \end{aligned} \quad (21)$$

$$K_n - K_{n+1} - k_w\dot{\theta}_n - I\ddot{\theta}_n + \frac{a_n}{2}[Y_n \sin(\theta_n) + X_n \cos(\theta_n) + X_{n+1} + (F_k + F_l) \sin(\theta_{n+1})] = 0 \quad (22)$$

The parameter  $b$  is the initial width of the spring and is assumed to be constant. We will therefore not consider the case in which the spring widens due to the longitudinal contraction. As a result of the

parameter linearization, we may assume that the bending torque of the central line,  $K_n$ , is a linear function of  $\theta_n$ , with a bending constant  $k_\theta$ :

$$K_n = -k_\theta \theta_n \quad (23)$$

Similarly we can define a constant  $k_1$  that expresses the fractional contraction of section  $n$  under influence of the resultant forces  $X_n$  and  $Y_n$  as:

$$X_n \sin(\theta_n) - Y_n \cos(\theta_n) = -k_1(s_0 - s_n) \quad (24)$$

with  $s_0$  the initial length of each section. To allow large deviations in the movement of each section, and to preserve a maximal accuracy, these equations were not further simplified. Therefore less sections are needed for the simulation of the spring.

Different sets of equations are needed for the first and the  $N$ -th section. The first section can be fully based on the general equations of the other sections, with  $\xi_0 = 0$  and  $\dot{x}_n = \dot{y}_0 = 0$ . Section  $N$  in contrary will need a specific set of equations, since both pulling forces  $F_k$  and  $F_1$  were attached to the top of the section.

Furthermore we may assume that the spring is carrying a payload or is perturbed by external forces and torques. All external forces and torques can be referenced in global coordinates, and we will reduce them to two global forces  $X_{N+1}$  and  $Y_{N+1}$  and one global couple  $K_{N+1}$ . This will lead to the following figure for the top section:

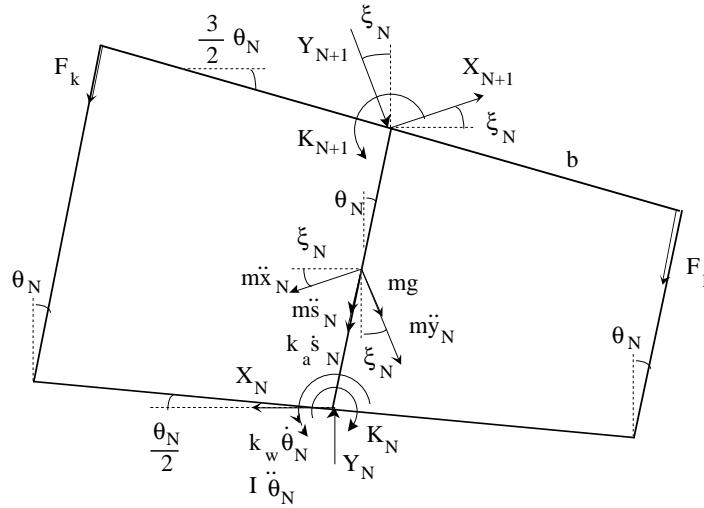


Fig. 11: top section of the spring

The according dynamic equations for the top section can then easily be found:

$$\begin{aligned} X_N + (m\dot{s}_N + k_a \dot{s}_N + F_k + F_l) \sin(\theta_N) + (m\ddot{x}_N - X_{N+1}) \cos(\xi_N) \\ - (mg + m\ddot{y}_N + Y_{N+1}) \sin(\xi_N) = 0 \end{aligned} \quad (25)$$

$$\begin{aligned} Y_N - (m\dot{s}_N + k_a \dot{s}_N + F_k + F_l) \cos(\theta_N) - (m\ddot{x}_N - X_{N+1}) \sin(\xi_N) \\ - (mg + m\ddot{y}_N + Y_{N+1}) \cos(\xi_N) = 0 \end{aligned} \quad (26)$$

$$\begin{aligned} K_N - K_{N+1} - k_w \dot{\theta}_N - \frac{1}{3} m s_N^2 \ddot{\theta}_N + [X_N + X_{N+1} \cos(\xi_N) + Y_{N+1} \sin(\xi_N)] \frac{s_N}{2} \cos(\theta_N) \\ + [Y_N + Y_{N+1} \cos(\xi_N) - X_{N+1} \sin(\xi_N)] \frac{s_N}{2} \sin(\theta_N) + \frac{b}{2} \cos\left(\frac{\theta_N}{2}\right) (F_l - F_k) = 0 \end{aligned} \quad (27)$$

Accordingly to the previous equations:

$$K_N = -k_\theta \theta_N \quad (28)$$

and

$$X_N \sin(\theta_N) - Y_N \cos(\theta_N) = -k_l (s_0 - s_N) \quad (29)$$

This will lead to a total of  $5 \times N$  equations that describe the dynamic motion of the spring.

The end position in the euclidian plan  $(x_n, y_n)$  of each section can than be found with:

$$x_n = \sum_{i=0}^n s_i \sin \left[ \sum_{j=0}^i \theta_j \right] \quad y_n = \sum_{i=0}^n s_i \cos \left[ \sum_{j=0}^i \theta_j \right] \quad (30)$$

The last section points to the end position  $(x_N, y_N)$  under an angle  $\xi_n = \sum \theta_i$ .

### C. Simulation Results

The first simulation shows the spring with no external forces or couples but gravity applied. The spring is put in a vertical unstable position and is initially perturbed with a large and short force applied at the right pulling cable. The goal of this simulation was the study of the impulse response of the spring.

Due to gravity we may expect two stable (on the left and on the right of the spring) and one unstable (standing upright) end position of the spring, when a small bending constant  $k_\theta$  is taken. If on the other hand we take a large bending constant, the spring should be able to overcome gravity force: it will return to its upright position and the two stable end positions will converge to one stable end point.

The following spring parameters were used:  $X_{N+1} = Y_{N+1} = 0$  and  $K_{N+1} = 0$ . 10 sections were taken, with a total spring mass of  $M = 600$ grams, so that  $m = 0.06$ kg. Further the following constants

were taken:  $b = 0.048\text{m}$ ,  $k_w = 0.5$ ,  $k_a = 30$  and  $k_l = 3000$ . The total length of the spring was  $s_0N = 1\text{m}$ . The left pulling force was chosen as zero:  $F_k = 0$ , while the right pulling force was set to  $F_l = 100\text{N}$  for 50 milliseconds.

During the first simulation  $k_\theta = 2$  is chosen, while for the second simulation  $k_\theta = 40$ .

The first and second derivatives of  $x$ ,  $y$  and  $\theta$  were calculated using third-order Simpson estimations:

$$\begin{aligned}\dot{\varepsilon}_n^t &= \frac{1}{\Delta t} \left( \frac{11}{6} \varepsilon_n^t - 3\varepsilon_n^{t-1} + \frac{3}{2} \varepsilon_n^{t-2} - \frac{1}{3} \varepsilon_n^{t-3} \right) \\ \ddot{\varepsilon}_n^t &= \frac{1}{\Delta t^2} (2\varepsilon_n^t - 5\varepsilon_n^{t-1} + 4\varepsilon_n^{t-2} - \varepsilon_n^{t-3})\end{aligned}\quad (31)$$

For these simulations the virtual time slice between two simulations, is chosen as  $\Delta t = 0.01\text{s}$ .

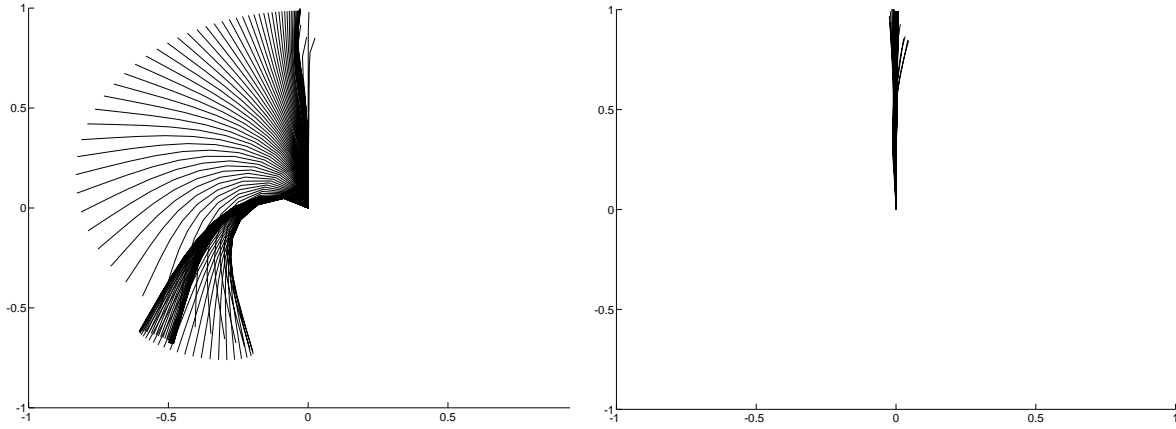


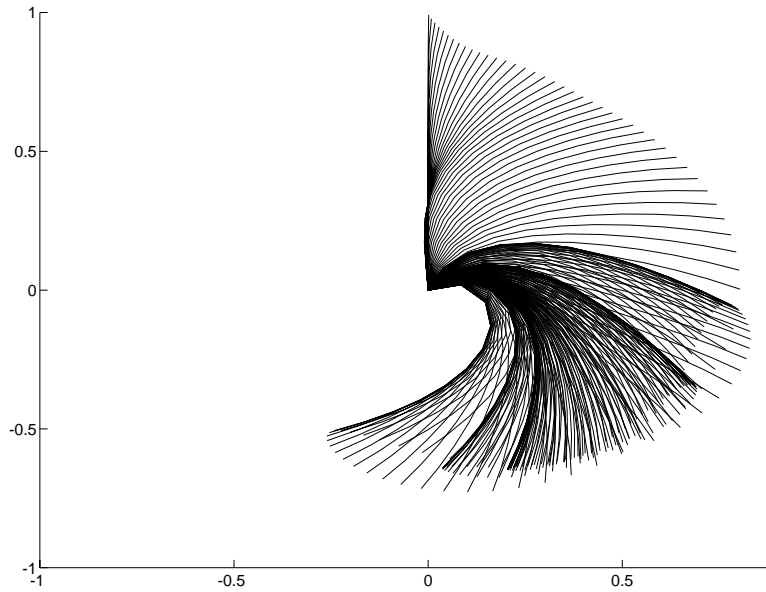
Fig. 12: spring in vertical position a) small bending constant b) large bending constant

In figure 12 a) we can see that for a small bending constant, i.e. the spring has only very little resistance against lateral deformations, gravity will cause the spring to fall down. In figure 12 b) a larger bending constant is used and, as expected, the spring will regain its original state after a perturbation.

A third simulation shows the dynamic movement of the spring with following two pulling forces applied:

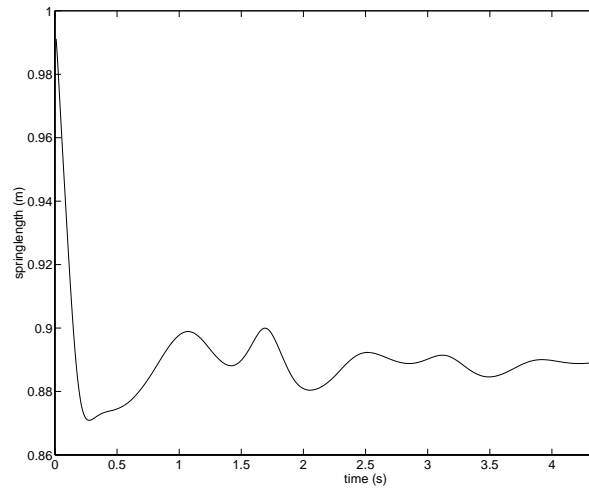
$$F_k = 15\text{N} \quad F_l = 20\text{N} \quad (32)$$

The forces are applied at time  $t = 0$  and the initial state of the spring is upright in a vertical unstable position. The friction constant against bending is lowered to a value  $k_w = 0.2$  to show more periods, and  $k_\theta$  is set to a value of 2.



*Fig. 13:* dynamic behaviour of the sideways bended spring

From figure 13 we learn that the spring will first contract and bend in an S-formed shape, i.e. the inertia of the spring mass will keep the spring from a pure sideways bend and force the spring to contract rather than bend. Due to friction the spring will end in a stable state for both position and length. This motion was also observed on a small practical setup. The variation of the length of the spring is shown in figure 14.



*Fig. 14:* variation of the length of the spring

## IV. CONCLUSIONS

This paper gives a contribution to the tendency of changing from heavy and rigid robot manipulators to more lightweight and flexible arms. A model for severe nonlinear constructions was introduced, based on a spring form. Furthermore a suggestion was given to control the movement using two pulling ropes inside the spring following its contour.

The simulations using this model, show that the model comes remarkably close to the intuitive natural behaviour of a long spring. Our first goal now is to do further research on the effectiveness of this model. For this a study on the real movements of springs will be performed, and the parameters of each spring will be mapped with the model.

Other research will be done on the control of the spring with the purpose of following a given trajectory, handling an unknown payload and applying given forces on a vaguely known surface. Because of the complexity and nonlinearities in the model of the spring, a typical nonlinear controller, such as fuzzy logic, appears to be the obvious solution. A practical setup with computer controlled pulling motors is planned for the near future.

On the long term the cooperation of multiple spring manipulators can be examined.

## ACKNOWLEDGMENT

Support for this research was provided by the department of Development and Research for robotics of Siemens A. G., 81730 München, Germany.

## REFERENCES

- [1] H. Asada and J. J. E. Slotine, "*Robot Analysis and Control*", Wiley, New York, 1986.
- [2] M. Benati and A. Morro, "Formulation of Equations of Motion for a Chain of Flexible Links Using Hamilton's Principle", *ASME Journal of Dynamic Systems, Measurement and Control*, Vol. 116, pp. 81-88, March 1994.
- [3] J. M. Baxter Brown, "*Introductory solid mechanics*", Wiley, London, 1973.
- [4] Y. Z. Chang and R. W. Daniel, "On the Adaptive Control of Flexible Joint Robots", *Automatica*, Vol. 28, No. 5, pp. 969-974, 1992.
- [5] C. L. Chen, P. C. Chen and C. K. Chen, "a Pneumatic Model-following Control System Using a Fuzzy Adaptive Controller", *Automatica*, Vol. 29, No. 4, pp. 1101-1105, 1993.
- [6] S. Chiaverini, B. Siciliano and L. Villani, "Force/Position Regulation of Compliant Robot Manipulators", *IEEE Transactions on Automatic Control*, Vol. 39, No. 3, pp. 647-652, March 1994.
- [7] G. S. Chirikjian and J. W. Burdick, "A Modal Approach to Hyper-Redundant Manipulator Kinematics", *IEEE Transactions on Robotics and Automation*, Vol. 10, No. 3, pp. 343-354, June 1994.
- [8] J. J. Craig, "*Introduction to Robotics Mechanics & Control*", Addison-Wesley Pub. Co., Reading, 1986.
- [9] A. De Luca and S. Panzieri, "An Iterative Scheme for Learning Gravity Compensation in Flexible Robot Arms", *Automatica*, Vol. 30, No. 6, pp. 993-1002, 1994.
- [10] C. C. De Wit and J. J. E. Slotine, "Sliding Observers for Robot Manipulators", *Automatica*, Vol. 27, No. 5, pp. 859-864, 1991.
- [11] J. H. Faupel, "*Engineering design: a synthesis of stress analysis and materials engineering*", Wiley, London, 1964.

- [12] T. Hesselroth, K. Sarkar, P. P. van der Smagt and K. Schulten, "Neural Network Control of a Pneumatic Robot Arm", *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 24, No. 1, pp. 28-38, January 1994.
- [13] F. L. Hu and A. G. Ulsoy, "Dynamic Modelling of Constrained Flexible Robot Arms for Controller Design", *ASME Journal of Dynamic Systems, Measurement and Control*, Vol. 116, pp. 56-65, March 1994.
- [14] Y. Iwata, H. Hirai and A. Watanabe, "Adaptive Servo Control of Flexible Arm", *Memoirs of the Faculty of Tech., Tokyo Metropolitan University*, No. 39, pp. 41-46, 1989.
- [15] R. C. Juvinall and K. M. Marshek, "*Fundamentals of machine component design*", Wiley, Chichester, 1991.
- [16] M. Kelemen and A. Bagchi, "Modeling and Feedback Control of a Flexible Arm of a Robot for Prescribed Frequency-domain Tolerances", *Automatica*, Vol. 29, No. 4, pp. 899-909, 1993.
- [17] R. Kelly, R. Ortega, A. Ailon and A. Loria, "Global Regulation of Flexible Joint Robots Using Approximate Differentiation", *IEEE Transactions on Automatic Control*, Vol. 39, No. 6, pp. 1222-1224, June 1994.
- [18] D. S. Kwon and W. J. Book, "A Time-Domain Inverse Dynamic Tracking Control of a Single-Link Flexible Manipulator", *ASME Journal of Dynamic Systems, Measurement and Control*, Vol. 116, pp. 193-200, June 1994.
- [19] K. Lenz, H. Özbay, A. Tannenbaum, J. Turi and B. Morton, "Frequency Domain Analysis and Robust Control Design for an Ideal Flexible Beam", *Automatica*, Vol. 27, No. 6, pp. 947-961, 1991.
- [20] K. W. Lilly and D. E. Orin, "Efficient Dynamic Simulation of Multiple Chain Robotic Mechanisms", *ASME Journal of Dynamic Systems, Measurement and Control*, pp. 223-231, June 1994.
- [21] W. Matek, D. Muhs, H. Wittel, M. Becker, "*Roloff/Matek Machine-onderdelen*", Academic Service, Schoonhoven, 1993, Dutch version.
- [22] Ö. Morgül, "Control and Stabilization of a Rotating Flexible Structure", *Automatica*, Vol. 30, No. 2, pp. 351-356, 1994.
- [23] P. P. Richard, "*Robot Manipulators: Mathematics, Programming and Control*", MIT Press, Cambridge, 1983.
- [24] M. Saad, P. Bigras, L. A. Dessaint and K. Al-Haddad, "Adaptive Robot Control using Neural Networks", *IEEE Transactions on Industrial Electronics*, Vol. 41, No. 2, pp. 173-181, April 1994.
- [25] D. Seto, A. M. Annaswamy and J. Baillieul, "Adaptive Control of Nonlinear Systems with a Triangular Structure", *IEEE Transactions on Automatic Control*, Vol. 39, No. 7, pp. 1411-1428, July 1994.
- [26] G. Tao, "On Robust Adaptive Control of Robot Manipulators", *Automatica*, Vol. 28, No. 4, pp. 803-807, 1992.
- [27] S. P. Timoshenko and J. M. Gere, "*Mechanics of materials*", Van Nostrand Reinhold, London, 1990.
- [28] J. T. Wen, "Position and Force Control of Robot Arms", *Proc. IEEE International Symposium on Intelligent Control*, Albany, New York, pp. 251-256, September 1989.
- [29] J. T. Wen and K. Kreutz-Delgado, "Motion and Force Control of Multiple Robotic Manipulators", *Automatica*, Vol. 28, No. 4, 1992.
- [30] B. Yao, S. P. Chan and D. Wang, "Unified Formulation of Variable Structure Control Schemes for Robot Manipulators", *IEEE Transactions on Automatic Control*, Vol. 39, No. 2, pp. 371-376, February 1994.
- [31] A. S. Yigit, "On the Stability of PD Control for a Two-Link Rigid-Flexible Manipulator", *ASME Journal of Dynamic Systems, Measurement and Control*, Vol. 116, pp. 208-215, June 1994.
- [32] M. Zhihong and M. Palaniswami, "Robust Tracking Control for Rigid Robotic Manipulators", *IEEE Transactions on Automatic Control*, Vol. 39, No. 1, pp. 154-159, January 1994.