N90-29002

A NEW APPROACH TO GLOBAL CONTROL OF REDUNDANT MANIPULATORS

Homayoun Seraji
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109

Abstract

A new and simple approach to configuration control of redundant manipulators is presented in this paper. In this approach, the redundancy is utilized to control the manipulator configuration directly in task space, where the task will be performed. A number of kinematic functions are defined to reflect the desirable configuration that will be achieved for a given end-effector position. The user-defined kinematic functions and the end-effector Cartesian coordinates are combined to form a set of task-related configuration variables as generalized coordinates for the manipulator. An adaptive scheme is then utilized to globally control the configuration variables so as to achieve tracking of some desired reference trajectories. This accomplishes the basic task of desired end-effector motion, while utilizing the redundancy to achieve any additional task through the desired time variation of the kinematic functions. The control law is simple and computationally very fast, and does not require the complex manipulator dynamic model.

1. Introduction

The remarkable dexterity and versatility that the human arm exhibits in performing various tasks can be attributed largely to the kinematic redundancy of the arm, which provides the capability of reconfiguring the arm without affecting the hand position. A robotic manipulator is called (kinematically) "redundant" if it possesses more degrees-of-freedom than is necessary for performing a specified task. For instance, in the three-dimensional space, a manipulator with seven or more joints is redundant since six degrees-of-freedom are sufficient to position and orient the end-effector in any desired configuration. Redundancy of a robotic manipulator is determined relative to the particular task to be performed. For example, in the two-dimensional space, a planar robot with three joints is redundant for achieving any end-effector position, whereas the robot is non-redundant for tasks involving both position and orientation of the end-effector. In a non-redundant manipulator, a given position and orientation of the end-effector corresponds to a single set of joint angles and an associated unique robot configuration (with distinct poses such as elbow up or down). Therefore, for a prescribed end-effector motion, the evolution of the robot configuration is uniquely determined. When this evolution is undesirable due to collision with obstacles, approaching kinematic singularities or reaching joint limits, there is no freedom to reconfigure the robot so as to reach around the obstacles, or avoid the singularities and joint limits.

Redundancy in the manipulator structure yields increased dexterity and versatility for performing a task due to the infinite number of joint motions which result in the same end-effector trajectory. However, this richness in choice of joint motions complicates the manipulator control problem considerably. In order to take full advantage of the capabilities of redundant manipulators, effective control schemes should be developed to utilize the redundancy in some useful manner. During recent years, redundant manipulators have been the subject of considerable research, and several methods have been suggested to resolve the redundancy. In 1969, Whitney [1] suggested the use of Jacobian pseudoinverse to resolve the redundancy. Over the past two decades, most of the research on redundant manipulators has been explicitly or implicitly based on the pseudoinverse approach for the utilization of redundancy through *local* optimization of some criterion functional. Furthermore, most proposed methods resolve the redundancy in joint space and are concerned solely with solving the inverse kinematic problem for redundant manipulators.

In this paper, a new and conceptually simple approach for *configuration control* of redundant manipulators is presented, which takes a complete departure from the conventional pseudoinverse methods. In this approach, the redundancy is utilized for *global* control of the manipulator configuration directly in *task space*, where the task will be performed, thus avoiding the complicated inverse kinematic transformation. A set of kinematic functions is chosen to reflect the desired additional task that will be performed due to the redundancy. The kinematic functions succinctly characterize the "self-motion" of the manipulator, in which the internal movement of the links does not move the end-effector. In other words, the kinematic functions are used to "shape" the manipulator configuration, given the end-effector position and orientation. The end-effector Cartesian coordinates and the kinematic functions are combined to form a set of "configuration variables" which describe the physical configuration of the entire manipulator in a task-related coordinate system. The control scheme then ensures that the configuration variables track some desired trajectories as closely as possible, so that the evolution of the manipulator configuration meets the task requirements. The control law is adaptive and does not require knowledge of the complex dynamic model or parameter values of the manipulator or payload. The scheme can be implemented either in a centralized or a decentralized control structure, and is computationally very fast as a real-time algorithm for on-line control of redundant manipulators.

2. Configuration Control Scheme

The mechanical manipulator under consideration consists of a linkage of rigid bodies with n revolute or prismatic joints. Let T be the $n \times 1$ vector of torques or forces applied at the joints and θ be the $n \times 1$ vector of the resulting relative joint rotations or translations. The dynamic equation of motion of the manipulator which relates T to θ can be represented in the general form [2]

$$M(\theta)\ddot{\theta} + N(\theta, \dot{\theta}) = T \tag{1}$$

where the matrices M and N are highly complex nonlinear functions of θ, $\dot{\theta}$, and the payload. Let the $m \times 1$ vector Y (with $m < n$) represent the position and orientation of the end-effector (last link) with respect to a fixed Cartesian coordinate system in the m-dimensional task space where the task is to be performed. The $m \times 1$ end-effector coordinate vector Y is related to the $n \times 1$ joint angle vector θ by the forward kinematic model

$$Y = Y(\theta) \tag{2}$$

where $Y(\theta)$ is an $m \times 1$ vector whose elements are nonlinear functions of the joint angles and link parameters and embodies the geometry of the manipulator. For a redundant manipulator with $m < n$, a Cartesian coordinate vector (such as Y) that specifies the end-effector position and orientation does not constitute a set of generalized coordinates to completely describe the manipulator dynamics. Nonetheless, equations (1) and (2) form a valid dynamic model to describe the *end-effector* motion itself in the task space. The desired motion of the end-effector is represented by the reference position and orientation trajectories denoted by the $m \times 1$ vector $Y_d(t)$, where the elements of $Y_d(t)$ are continuous twice-differentiable functions of time. The vector $Y_d(t)$ embodies the information on the "*basic task*" to be accomplished by the end-effector in the task space.

We shall now discuss the definition of configuration variables and the adaptive control of redundant manipulators in the subsequent sections.

2.1 Definition of Configuration Variables

Let $r = n - m$ be the "degree-of-redundancy" of the manipulator, i.e. the number of "extra" joints. Let us define a set of r kinematic functions $\{\phi_1(\theta), \phi_2(\theta), \ldots, \phi_r(\theta)\}$ to reflect the *"additional task"* that will be performed due to the manipulator redundancy. Each ϕ_i can be a function of the joint angles $\{\theta_1, \ldots, \theta_n\}$ and the link geometric parameters. The choice of the kinematic functions can be made in several ways to represent, for instance, the coordinates of any point on the manipulator, or any combination of the joint angles. The kinematic functions succinctly characterize the "self-motion" of the manipulator, in which the internal movement of the links does not move the end-effector.

For the sake of illustration, let us consider a planar three-link arm as shown in Figure 1(i). The basic task is to control the end-effector position coordinates $[x, y]$ in the base frame. Suppose that we fix the end-effector position and allow internal motion of the links so that the arm takes all possible configurations. It is found that the locus of point A is an arc of a circle with center O and radius ℓ_1 which satisfies the distance constraint $AC \leq (\ell_2 + \ell_3)$. Likewise, the locus of point B is an arc of a circle with center C and radius ℓ_3 which satisfies $OB \leq (\ell_1 + \ell_2)$. The loci of A and B are shown as hatched arcs in Figure 1(i), and represent the self-motion of the arm. Now, in order to characterize the self-motion, we can select a kinematic function $\phi(\theta)$ to represent, for instance, the terminal angle $\phi = \theta_1 + \theta_2 + \theta_3$, or alternatively we can designate the wrist height y_B as the kinematic function $\phi = \ell_1 \sin\theta_1 + \ell_2 \sin(\theta_1 + \theta_2)$. The choice of ϕ clearly depends on the particular task that we wish to perform by the utilization of redundancy, in addition to the end-effector motion. Let us now consider a spatial 7 dof arm [3] as shown in Figure 1(ii), in which the end-effector position and orientation are of concern. The self-motion of this arm corresponds to rotation of the elbow point A about the line OB joining the shoulder to the wrist. We can now define the kinematic function $\phi(\theta) = \alpha$, where α is the angle between a normal line from A to OB and a line perpendicular to OB in the vertical plane, as shown in Figure 1(ii). The kinematic function ϕ then succinctly describes the redundancy and gives a simple characterization of the self-motion.

Once a set of r task-related kinematic functions $\phi = \{\phi_1, \phi_2, \ldots, \phi_r\}$ is defined, we have partial information on the manipulator configuration. The set of m end-effector position and orientation coordinates $Y = \{y_1, y_2, \ldots, y_m\}$ provides the remaining information on the configuration. Let us now combine the two sets ϕ and Y to obtain a complete set of n configuration variables as

$$\begin{aligned} X = \{Y, \phi\} &= \{y_1, y_2, \ldots, y_m \vdots \phi_1, \phi_2, \ldots, \phi_r\} \\ &= \{x_1, x_2, \ldots, x_n\} \end{aligned} \tag{3}$$

The $n \times 1$ vector X is referred to as the *"configuration vector"* of the redundant manipulator and the elements of X, namely $\{x_1, \ldots, x_n\}$, are called the *"configuration variables."* The configuration variables $\{x_1, \ldots, x_n\}$ constitute a set of generalized coordinates for the redundant manipulator. Using the configuration vector X, the manipulator is fully specified and is no longer redundant in this representation. It is noted that in some applications, certain end-effector coordinates are not relevant to the task, for instance, in a spot welding task the orientation of the end-effector is not important. In such cases, the present approach allows the designer to replace the insignificant end-effector coordinates with additional kinematic functions which are more relevant to that particular application. In fact, if $m'(< m)$ end-effector coordinates are specified, then $n - m' = r'(> r)$ kinematic functions can be defined.

The augmented forward kinematic model which relates the configuration vector X to the joint angle vector θ is now given by

$$X = \begin{pmatrix} Y(\theta) \\ \ldots\ldots \\ \phi(\theta) \end{pmatrix} = X(\theta) \tag{4}$$

From equation (4), the differential model which relates the rates of change of X and θ is obtained as

$$\dot{X}(t) = J(\theta)\dot{\theta}(t) \tag{5}$$

where

$$J(\theta) = \begin{pmatrix} J_e(\theta) \\ \ldots\ldots \\ J_c(\theta) \end{pmatrix} = \begin{pmatrix} \frac{\partial Y}{\partial \theta} \\ \ldots \\ \frac{\partial \phi}{\partial \theta} \end{pmatrix} \tag{6}$$

is the $n \times n$ ***augmented Jacobian matrix***. The $m \times n$ submatrix $J_e(\theta) = \frac{\partial Y}{\partial \theta}$ is associated with the end-effector, while the $r \times n$ submatrix $J_c(\theta) = \frac{\partial \phi}{\partial \theta}$ is related to the kinematic functions. The two submatrices J_e and J_c combine to form the square Jacobian matrix J.

The augmented Jacobian matrix J can be used to test the functional independence of the kinematic functions $\{\phi_1, \ldots, \phi_r\}$ and the end-effector coordinates $\{y_1, \ldots, y_m\}$. For the set of configuration variables $X = \{x_1, \ldots, x_n\}$ to be functionally independent throughout the workspace, it suffices to check that det $[J(\theta)]$ is not identically zero for *all* θ, [4]. In other words, when the augmented Jacobian matrix J is rank-deficient for *all* values of θ, the kinematic functions chosen are functionally dependent on the end-effector coordinates and a different choice of ϕ is necessary. When det $[J(\theta)]$ is not identically zero, the configuration variables $\{x_1, \ldots, x_n\}$ are not functionally dependent for all θ. Nonetheless, there can be certain joint configurations $\theta = \theta_o$ at which det $[J(\theta_o)] = 0$, i.e., the augmented Jacobian matrix J is rank-deficient. This implies that the rows J^i of J satisfy the linear relationship $\sum_{i=1}^{n} c_i J^i = 0$, where c_i are some constants not all zero. Since the changes of the configuration variables and joint angles are related by $\Delta x = J(\theta)\Delta\theta$, we conclude that at $\theta = \theta_o$,

$\sum_{i=1}^{n} c_i \Delta x_i = 0$. Therefore at a Jacobian singularity, the changes in the configuration variables $\{\Delta x_1, \ldots, \Delta x_n\}$ must satisfy the constraint relationship $\sum_{i=1}^{n} c_i \Delta x_i = 0$, and hence the configuration vector X cannot be changed arbitrarily.

From expression (6), it is clear that the Jacobian matrix J will be singular at any joint configuration for which the submatrix J_e is rank-deficient; i.e., at any end-effector singular configuration. In addition, the Jacobian J will be singular at those values of θ for which the submatrix J_c loses full rank. The latter singularities of J, which are due to the kinematic functions, are inevitably introduced whenever an additional task is employed to utilize the redundancy. However, by judicious choice of the kinematic functions, some of the singularities due to J_c may be avoided, and the singularities of J may be shifted to the unusable part of the workspace. Note that even when J_e and J_c have full ranks individually, the augmented matrix J may still be rank-deficient.

2.2 Adaptive Configuration Control

Suppose that a user-defined *"additional task"* can be expressed by the following kinematic equality constraint relationships

$$\begin{aligned} \phi_1(\theta) &= \phi_{d1}(t) \\ \phi_2(\theta) &= \phi_{d2}(t) \\ \vdots \quad & \quad \vdots \\ \phi_r(\theta) &= \phi_{dr}(t) \end{aligned} \tag{7}$$

where $\phi_{di}(t)$ denotes the desired time variation of the kinematic function ϕ_i and is a user-specified continuous twice-differentiable function of time. The kinematic relationships (7) can be represented collectively in the vector form

$$\phi(\theta) = \phi_d(t) \tag{8}$$

where ϕ_d is an $r \times 1$ vector. Equation (8) represents a set of "kinematic constraints" on the manipulator and defines the task that will be performed *in addition to* the basic task of desired end-effector motion. The kinematic equality constraints (8) are chosen to have physical interpretations and are used to formulate the desirable characteristics of the manipulator configuration in terms of motion of other members of the manipulator. For instance, in the 7 dof arm of Figure 1(ii), by controlling the elbow height as well as the hand coordinates, we can ensure that the elbow avoids collision with vertical obstacles (such as walls) in the workspace while the hand tracks the desired trajectory. Alternatively, a particular posture of the manipulator which represents a singular configuration can be avoided by an appropriate choice of the kinematic constraints in terms of the joint angles. The proposed formulation appears to be a highly promising approach to the additional task performance in comparison with the previous approaches which attempt to minimize or maximize criterion functionals, since we are now able to make a more specific statement about the evolution of the manipulator configuration. The present approach also covers the intuitive solution to redundant arm control in which certain joint angles are held constant for a portion of the task in order to resolve the redundancy. The functional forms of the kinematic functions ϕ_i and their desired behavior ϕ_{di} may vary widely for different additional tasks, making the approach unrestricted to any particular type of application.

Based on the foregoing formulation, we can now consider the manipulator with the $n \times 1$ configuration vector $X = \binom{Y}{\phi}$ and the $n \times n$ augmented Jacobian matrix $J = \binom{J_e}{J_c}$. Once the desired motion of the end-effector $Y_d(t)$ is specified for the particular basic task and the required evolution of the kinematic functions $\phi_d(t)$ is specified to meet the desired additional task, the $n \times 1$ desired configuration vector $X_d(t) = \binom{Y_d(t)}{\phi_d(t)}$ is fully determined. The configuration control problem for the redundant manipulator is to devise a dynamic control scheme as shown in Figure 2 which ensures that the manipulator configuration vector $X(t)$ tracks the desired trajectory vector $X_d(t)$ as closely as possible. In the control system shown in Figure 2, the actual end-effector position $Y(t)$ and the current value of the kinematic functions $\phi(t)$ are fed back to the controller. The controller uses this feedback information together with the commanded end-effector motion $Y_d(t)$ and the desired time variation $\phi_d(t)$ to compute the driving torques $T(t)$ that are applied at the manipulator joints so as to meet the basic and additional task requirements simultaneously.

Different control strategies can be improvised to meet the above tracking requirement, taking into account the dynamics of the manipulator given by equation (1). There are two major techniques for the design of tracking controllers in task space, namely model-based control and adaptive control. For the model-based control [5], the manipulator dynamics is first expressed in task space as

$$M_x(\theta)\ddot{X} + N_x(\theta,\dot{\theta}) = F \tag{9}$$

where F is the $n \times 1$ "virtual" control force vector in the task space, and M_x and N_x are obtained from equations (1)-(6). The control law which achieves tracking through global linearization and decoupling is given by

$$F = M_x(\theta)\left[\ddot{X}_d(t) + K_v\left(\dot{X}_d(t) - \dot{X}(t)\right) + K_p\left(X_d(t) - X(t)\right)\right] + N_x(\theta,\dot{\theta}) \tag{10}$$

where K_p and K_v are constant position and velocity feedback gain matrices. This control formulation requires precise knowledge of the full dynamic model and parameter values of the manipulator and the payload. The alternative approach is the adaptive control technique in which the on-line adaptation of the controller gains eliminates the need for the complex manipulator dynamic model. In this section, we adopt an adaptive control scheme which has been developed recently and validated experimentally on a PUMA industrial robot [6-8]. The adaptive controller produces the control signal based on the observed performance of the manipulator and has therefore the capability to operate with minimal information on the manipulator/payload and to cope with unpredictable gross variations in the payload. The proposed adaptive control scheme is developed within the framework of Model Reference Adaptive Control (MRAC) theory, and the adaptive tracking control law in the task space is given by [6]

$$F(t) = d(t) + [K_p(t)E(t) + K_v(t)\dot{E}(t)] + [C(t)X_d(t) + B(t)\dot{X}_d(t) + A(t)\ddot{X}_d(t)] \tag{11}$$

as shown in Figure 3. This control force is composed of three components, namely:

(i) The *auxiliary signal* $d(t)$ is synthesized by the adaptation scheme and improves transient performance while resulting in better tracking and providing more flexibility in the design.

(ii) The term $[K_p(t)E(t) + K_v(t)\dot{E}(t)]$ is due to the PD *feedback controller* acting on the position tracking-error $E(t) = X_d(t) - X(t)$ and the velocity tracking-error $\dot{E}(t) = \dot{X}_d(t) - \dot{X}(t)$.

(iii) The term $[C(t)X_d(t)+B(t)\dot{X}_d(t)+A(t)\ddot{X}_d(t)]$ is the contribution of the PD^2 *feedforward controller* operating on the desired position $X_d(t)$, the desired velocity $\dot{X}_d(t)$, and the desired acceleration $\ddot{X}_d(t)$.

The required auxiliary signal and feedback/feedforward controller gains are updated based on the $n \times 1$ "weighted" error vector $q(t)$ by the following simple adaptation laws [6]:

$$q(t) = W_p E(t) + W_v \dot{E}(t) \tag{12}$$

$$d(t) = d(0) + \delta_1 \int_0^t q(t)dt + \delta_2 q(t) \tag{13}$$

$$K_p(t) = K_p(0) + \alpha_1 \int_0^t q(t)E'(t)dt + \alpha_2 q(t)E'(t) \tag{14}$$

$$K_v(t) = K_v(0) + \beta_1 \int_0^t q(t)\dot{E}'(t)dt + \beta_2 q(t)\dot{E}'(t) \tag{15}$$

$$C(t) = C(0) + \nu_1 \int_0^t q(t)X_d'(t)dt + \nu_2 q(t)X_d'(t) \tag{16}$$

$$B(t) = B(0) + \gamma_1 \int_0^t q(t)\dot{X}_d'(t)dt + \gamma_2 q(t)\dot{X}_d'(t) \tag{17}$$

$$A(t) = A(0) + \lambda_1 \int_0^t q(t)\ddot{X}_d'(t)dt + \lambda_2 q(t)\ddot{X}_d'(t) \tag{18}$$

In equations (13)-(18), $\{\delta_1, \alpha_1, \beta_1, \nu_1, \gamma_1, \lambda_1\}$ are any positive scalar integral adaptation gains, and $\{\delta_2, \alpha_2, \beta_2, \nu_2, \gamma_2, \lambda_2\}$ are zero or any positive scalar proportional adaptation gains. In equation (12), $W_p = \mathrm{diag}_i\{w_{pi}\}$ and $W_v = \mathrm{diag}_i\{w_{vi}\}$ are constant $n \times n$ weighting matrices chosen by the designer to reflect the relative significance of the position and velocity errors E and $\dot{E}$ in forming the vector q. The values of the adaptation gains and weighting matrices determine the rate at which the tracking-errors converge to zero.

Since the control actuation is at the manipulator joints, the control force F is implemented as the joint torque T where

$$T(t) = J'(\theta)F(t) \tag{19}$$

The augmented Jacobian matrix $J(\theta)$ is used in equation (19) to map the task forces $F(t)$ to the joint torques $T(t)$. Equation (19) represents the fundamental relationship between the task and joint spaces and is the basis for implementation of any task-based control scheme [5]. Equation (19) can be rewritten as

$$T(t) = \left[J_e'(\theta) \vdots J_c'(\theta) \right] \begin{bmatrix} F_e(t) \\ \cdots \\ F_c(t) \end{bmatrix} = J_e'(\theta)F_e(t) + J_c'(\theta)F_c(t) \tag{20}$$

where F_e and F_c are the $m \times 1$ and $r \times 1$ control force vectors corresponding to the basic task and the additional task, respectively. It is seen that the total control torque is the sum of two components: $T_e = J_e' F_e$, for the end-effector motion (basic task), and $T_c = J_c' F_c$, for the kinematic constraints (additional task). Equation (20) shows distinctly the contributions of the basic and the additional tasks to the overall control torque. Under the joint control law (20), the desired end-effector trajectory $Y_d(t)$ is tracked, and the "extra" degrees-of-freedom are conveniently used to control the evolution of the manipulator configuration through tracking of the desired kinematic functions $\phi_d(t)$. In other words, the self-motion of the manipulator is controlled by first characterizing this motion in terms of a set of kinematic functions and then controlling these functions through trajectory tracking.

The adaptive control scheme presented in this section is extremely simple since the auxiliary signal and controller gains are evaluated from equations (12)-(18) by simple numerical integration by using, for instance, the trapezoidal rule. Thus the computational time required to calculate the adaptive control law (11) is extremely short. As a result, the scheme can be implemented for on-line control of redundant manipulators with high sampling rates, resulting in improved dynamic performance. This is in contrast to most existing approaches which require time-consuming optimization processes unsuitable for fast on-line control implementation. It is important to note that the adaptation laws (12)-(18) are based solely on the observed performance of the manipulator rather than on any knowledge of the complex dynamic model or parameter values of the manipulator and the payload.

3. Conclusions

A simple formulation for configuration control of redundant manipulators has been developed in this paper. The controller achieves trajectory tracking for the end-effector directly in the Cartesian space to perform some desired basic task. In addition, the redundancy is utilized by imposing a set of kinematic constraints on the manipulator to accomplish an appropriate additional task. The proposed formulation incorporates the kinematic constraints (additional task) and the end-effector motion (basic task) in a conceptually simple and computationally efficient manner to resolve the redundancy. Furthermore, the adaptive controller has a very simple structure and the controller gains are adjusted in a simple manner to compensate for changing dynamic characteristics of the manipulator. The adaptation laws are based on the observed performance of the manipulator rather than on any knowledge of the manipulator dynamic model. Thus, the adaptive controller is capable of ensuring a satisfactory performance when the payload mass is unknown and time-varying. Any approach used to resolve redundancy should be implementable as a real-time algorithm, and therefore the speed of computation is a critical factor. The small amount of computations required by the proposed method offers the possibility of fast real-time control of redundant manipulators.

4. Acknowledgement

The research described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

5. References

[1] D.E. Whitney: "Resolved motion rate control of manipulators and human prostheses," *IEEE Trans. Man-Machine Systems*, 1969, Vol. MMS-10, No. 2, pp. 47-53.

[2] J.J. Craig: *Robotics - Mechanics and Control*, Addison-Wesley Publishing Company, Reading, MA, 1986.

[3] J.M. Hollerbach: "Optimum kinematic design for a seven degree of freedom manipulator," *Proc. 2nd Intern. Symp. on Robotics Research*, Kyoto, August 1984.

[4] R. Courant: *Differential and Integral Calculus*, Vol. II, Interscience Publishers Inc., New York, 1961.

[5] O. Khatib: "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal of Robotics and Automation*, 1987, Vol. RA-3, No. 1, pp. 43-53.

[6] H. Seraji: "Direct adaptive control of manipulators in Cartesian space," *Journal of Robotic Systems*, 1987, Vol. 4, No. 1, pp. 157-178.

[7] H. Seraji: "A new approach to adaptive control of manipulators," *ASME Journ. Dynamic Systems, Measurement and Control*, 1987, Vol. 109, No. 3, pp. 193-202.

[8] H. Seraji: "Adaptive independent joint control of manipulators: Theory and experiment," *Proc. IEEE Intern. Conf. on Robotics and Automation*, Philadelphia, April 1988, pp. 854-861.

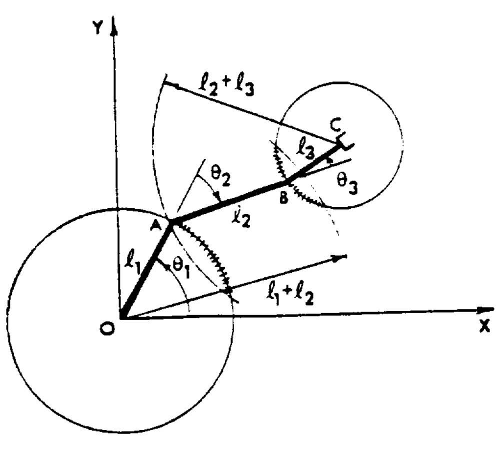

Figure 1(i). Self-motion of Planar 3 dof Arm

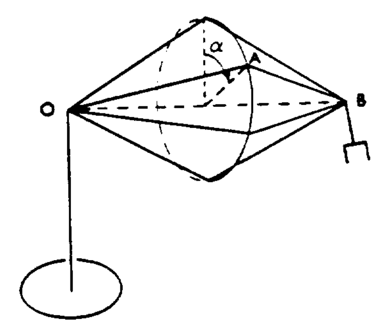

Figure 1(ii). Self-motion of Spatial 7 dof Arm

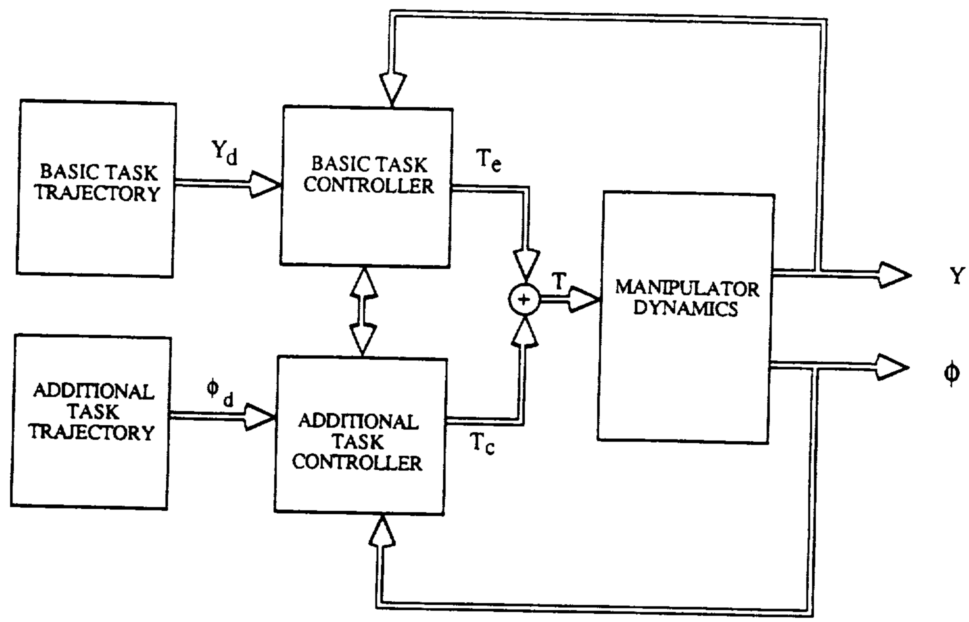

Figure 2. Architecture of Configuration Control Scheme

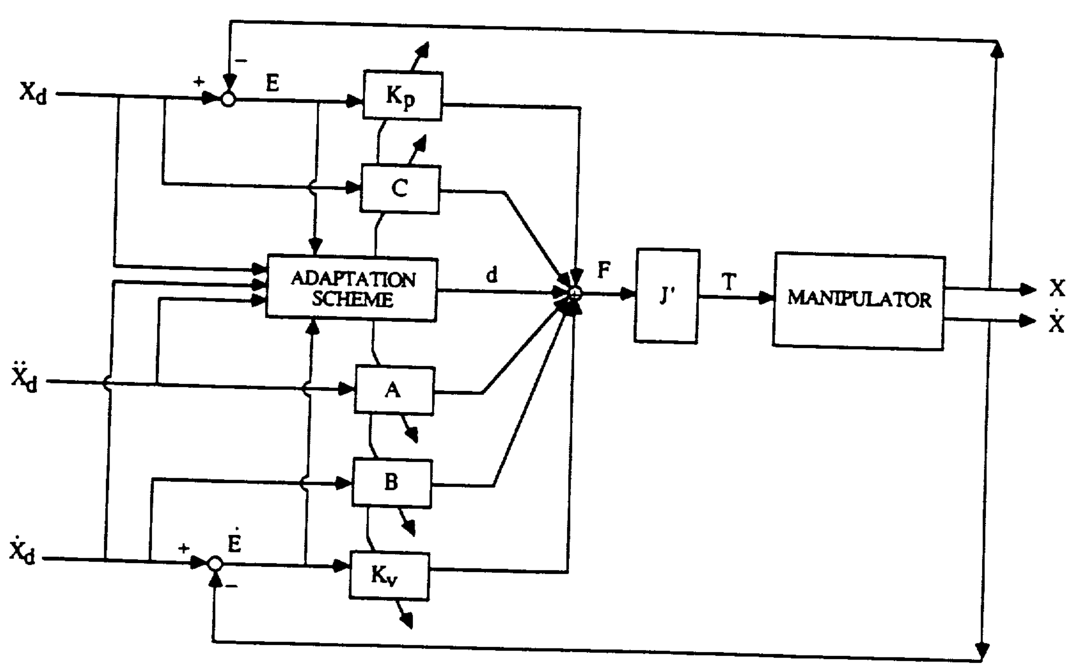

Figure 3. Adaptive Manipulator Control System