Preprints of the Fourth IFAC
Symposium on Robot Control
September 19–21, 1994, Capri, Italy

# ON CONTROLLING ROBOTS WITH REDUNDANCY IN AN ENVIRONMENT WITH OBSTACLES

H. DING [1] and W. SCHIEHLEN

Institute B of Mechanics,University of Stuttgart, Pfaffenwaldring 9,
D-70550 Stuttgart ,Germany

**Abstract.** This paper presents a method to solve the problem of obstacle avoidance for a redundant manipulator so that it closely tracks the desired end-effector trajectory and simultaneously avoids workspace obstacles. The proposed method introduces the describing function $J$ to indicate whether collision occurs between the robot and obstacles. The obstacle avoidance task is achieved by formulating the performance criterion $J > J_{min}$ ( $J_{min}$ represents the minimal distance between the redundant robot and obstacles), and by using configuration control to ensure that this criterion is satisfied. The calculation of $J$ function is only related to some vertices which are used to generate and model the robot and obstacles, and the computational times are nearly linear with the total number of vertices. The configuration control algorithm which achieves end-effector motion and obstacle avoidance does not require either a complex dynamic model or complicated inverse kinematics transformations of the robot. Several simulation cases for a four-link planar manipulator are given to prove that the proposed collision-free trajectory planning scheme is efficient and practical.

**Key Words.** Robots; redundancy; obstacle avoidance; configuration control; simulation

## 1. INTRODUCTION

Redundancy is a key element of designing more versatile robots. The redundancy of manipulators can be effectively used to increase dexterity, avoid singularities, and achieve collision-free motion while performing the desired end-effector task. Collision avoidance is an absolutely essential requirement for a robot to complete a task in an environment with obstacles. It is well recognized that the problem of collision avoidance with obstacles is an important issue in current robotics research. Khatib (1986) has considered the complete problem of controlling a redundant robot in the presence of obstacles based on the concept of artificial potential, which assumes the existence of repelling forces between links and obstacles. In this approach, the redundant robot is controlled directly in Cartesian space using a model-based control law, and obstacle avoidance is achieved using the artificial potential field. Galicki (1992) gave the global planning method of collision-free trajectory based on the use of potential fields and minimization of integral criterion. Guo and Hsia (1993) presented "joint space command" to carry out the inverse kinematic solution and applied this command generator scheme to optimize the distance function between robot links and obstacles. They adopted object modelling and distance computation derived by Gilbert *et al.*(1989). The above methods are generally not convenient to use due to time-consuming calculations and too much input information. Colbaugh *et al.*(1989) presented an approach to obstacle avoidance for redundant robots based on configuration control formulation originally proposed by Seraji (1989), which does not require either the complex manipulator dynamic model or the complicated inverse kinematic transformation. But, the general procedure for calculating distance function and locating critical points were not given (Colbaugh *et al.* 1989).

Xiong and Ding (1989) constructed the describing function J, which provides a concept of pseudo-metric on the moving object related to the obstacle in Cartesian space. They applied the describing function $J$ to find collision-free

[1] On leave from Huazhong University of Science and Technology, 430074 Wuhan, P.R. China

movement for a non-redundant manipulator. In this paper, the describing function is extended to solve the obstacle avoidance for a redundant manipulator. The obstacle avoidance performance criterion is presented as $J > J_{min}$ which can be effectively solved by linear programming and the configuration control scheme ensures that this condition is satisfied simultaneously with the desired end-effector motion. According to the proposed control algorithm, the collision-free motion trajectory for a redundant manipulator in an environment with obstacles can be easily reached. Its simplicity and computational efficiency scheme allow on-line implementation with a high sampling rate for real-time obstacle avoidance in a dynamically varying environment.

## 2. CARTESIAN-SPACE CONTROL SCHEME

Let $X \in R^m$ define the position and orientation of the end-effector in the task space. The relationship between the end-effector coordinate $X$ and joint coordinate $q$ can be written as:

$$X(t) = f(q) \qquad (1)$$

where $q \in R^n (n > m)$ is the vector of joint rotations/translations, $f : R^n \rightarrow R^m$ represents the forward kinematic transformation.

$$\dot{X} = J_e(q)\dot{q} \qquad (2)$$

$$\ddot{X} = J_e(q)\ddot{q} + \dot{J}_e(q)\dot{q} \qquad (3)$$

where $J_e = \partial f / \partial q \in R^{m \times n}$ is the end-effector Jacobian of the manipulator. The equations of motion for the manipulator in joint space can be written as:

$$H(q)\ddot{q} + l(q, \dot{q}) + g(q) = T \qquad (4)$$

where $H \in R^{n \times n}$ is the symmetric positive-definite inertia matrix, $l(q, \dot{q}), g(q)$ represent centrifugal and Coriolis force and gravity force, respectively, and $T \in R^n$ is the joint torque vector, e. g. Schiehlen (1992).

Similarly, the dynamic model in Cartesian space can be expressed as

$$F = H_x \ddot{X} + V_x \qquad (5)$$

where $F \in R^m$ is the generalized force vector corresponding to the Cartesian coordinate vector $X$. The $F \rightarrow T$ mapping is given by

$$T = J_e^T(q)F \qquad (6)$$

$H_x \in R^{m \times m}$ and $V_x \in R^m$ are defined as follows:

$$H_x = (J_e H^{-1} J_e^T)^{-1} \qquad (7)$$

$$V_x = (J_e H^{-1} J_e^T)^{-1} J_e H^{-1}(l + g) - (J_e H^{-1} J_e^T)^{-1} \dot{J}_e \dot{q} \qquad (8)$$

Different control strategies can be improvised to compute the control force $F$ that ensures that manipulator configuration vector $X$ in Eq.(5) tracks the desired trajectory $X_d(t)$, asymptotically. Here the adaptive control scheme developed by Seraji (1989) is adopted to accomplish this tracking.

This algorithm computes the ith component of the Cartesian control input $F$ as:

$$F_i(t) = f_i(t) + K_{pi}(t)e_i + K_{vi}(t)\dot{e}_i + K_{fi}(t)\ddot{X}_{di} \qquad (9)$$

where $i = 1, 2, \ldots, m$ and $e = X_d - X$ denotes the Cartesian tracking error. $e_i$ and $X_{di}$ are the ith components of $e$ and $X_d$, respectively. $f_i, K_{pi}, K_{vi}$ and $K_{fi}$ are controller gains which can be generated in real-time according to the proportional-plus-integral adaptation laws.

For kinematically redundant robots, the $F \rightarrow T$ mapping problem in Eq.[6] is underdetermined, so that there exists an infinite number of joint torque vectors corresponding to a given control input $F$. By properly selecting a particular joint torque from the possible candidates the robot redundancy can be utilized to avoid obstacles.

Let $Y = [X^T : Z^T] \in R^n$ define the augmented task space coordinate vector. $Z \in R^r, r = n - m$ is the degree of redundancy of the manipulator. The vector $Z$ can be chosen so that the robot redundancy is resolved in some useful manner. $J_a = \partial Y / \partial q \in R^{n \times n}$ stands for augmented Jacobian matrix. In order to avoid obstacles, the additional task is expressed as follows:

$$Z = D(q) > 0 \qquad (10)$$

$D$ is Cartesian distance function between the robot and obstacles. Let $F_c \in R^r$ denote the generalized force vector corresponding to the obstacle avoidance

$$T = J_a^T F = [J_e^T : J_c^T] \begin{bmatrix} F \\ F_c \end{bmatrix} \qquad (11)$$

$$T = J_e^T F + J_c^T F_c \qquad (12)$$

where $J_c \in R^{r \times n}$ is the additional task Jacobian matrix. $F$ and $F_c$ are the $m \times 1$ and $r \times 1$ control force vectors corresponding to the basic task and additional task, respectively. The total control torque is the sum of two components : $J_e^T F_e$ is related to the end-effector motion (basic task) and $J_c^T F_c$ to the obstacle avoidance (additional task). Based on the obstacle avoidance performance index, the appropriate $F \rightarrow T$ map can be chosen effectively to avoid obstacles.

## 3. COLLISION-AVOIDANCE ALGORITHM

### 3.1. *J Function*

This section considers the problem of controlling the robot to closely track the desired end-effector trajectory while ensuring that none of the robot links collides with workspace obstacles.

For convenience, the manipulator $H$ and obstacles $F$ can be modelled by finite, rigid, possibly overlapping unions of convex $3-D$ polyhedra.

$$H = \bigcup_{i \in I} H_i \qquad F = \bigcup_{j \in I} F_j \qquad (13)$$

where $I_A$ and $I_B$ are disjoint index sets. Both the convex polyhedra $H_i, i = 1, 2, \ldots, n$ and $F_j, j = 1, 2, \ldots, K$ are represented as convex hulls of point sets $h_i = h_{is}, s = 1, 2, \ldots, s_i, i = 1, 2, \ldots, n$ and $f_j = f_{jk}, k = 1, 2, \ldots, k_j, j = 1, 2, \ldots, K$, respectively. Thus

$$H_i = \left\{ h = \sum_{s=1}^{s} \beta_s h_{is}, \beta_s \geq 0, \sum_{s=1}^{s} \beta_s = 1 \right\}$$
$$= Conv[h_i] \qquad (14)$$

$$F_j = \left\{ f = \sum_{k=1}^{k} \mu_k f_{jk}, \mu_k \geq 0, \sum_{k=1}^{k} \mu_k = 1 \right\}$$
$$= Conv[f_j] \qquad (15)$$

where $Conv[h_i]$ and $Conv[f_j]$ represent the convex hulls of point sets. Both $h_i$ and $f_j$ are the sets in Cartesian space; they can generate polyhedra $H_i$ and $F_j$ and are viewed as the extreme points of $H_i$ and $F_j$, respectively. In this way, both the redundant manipulator and obstacles are characterized by point sets. The describing function $J_{ij}$ for the robot link $H_i$ and obstacles $F_j$ is defined as

$$J_{ij} = \min_{\Omega} \sum_{t=1}^{p+2} y_t \qquad (16)$$

where $\Omega_{ij}$ is the feasible set, which is defined as the feasible solutions of the following equality system:

$$\sum_{s=1}^{s} \beta_s h_{is} - \sum_{k=1}^{k} \mu_k f_{jk} + Y = 0 \quad Y = [y_1, \ldots, y_p]^T \qquad (17)$$

$$\sum_{s=1}^{s} \beta_s + y_{p+1} = 1 \qquad \sum_{k=1}^{k} \mu_k + y_{p+2} = 1 \quad (18)$$

where $y_t, \beta_s, \mu_k \geq 0, t = 1, 2, \ldots p, p+1, p+2$, $s = 1, 2, \ldots s_i, k = 1, 2, \ldots k_j$ and $p$ is the dimensionality of the problem ($p = 2$ *or* 3, respectively). The describing function $J$ has the following properties:

1. The range of value for the describing function $J_{ij}$ is as follows

$$0 \leq J_{ij} \leq 2 \qquad (19)$$

2. $J_{ij} = 0$ is equivalent to $H_i \bigcap F_j \neq \phi$, $\phi$ means empty set, i.e.

$$J_{ij} = 0 \qquad \Longleftrightarrow \qquad H_i \bigcap F_j \neq \phi \qquad (20)$$

3. The describing function $J_{ij}$ and Cartesian distance function $D_{ij}$ are pseudo-monotonic, i.e.

$$J_{ij}(X + \Delta X) \geq J_{ij}(X) \quad \Longleftrightarrow$$
$$D_{ij}(X + \Delta X) \geq D_{ij}(X) \qquad (21)$$

and

$$J_{ij}(X) = 0 \qquad \Longleftrightarrow D_{ij} = 0 \qquad (22)$$

where $D_{ij}$ is defined as:

$$D_{ij} = \|H_i - F_j\|$$
$$= \min\{\|h - f\| : h \in H_i, f \in F_j\}. \quad (23)$$

The describing function $J$ [1] for the redundant manipulator $H$ and obstacle $F$ can be defined as:

$$J = \min\{J_{ij}, i = 1, 2, \ldots n; j = 1, 2, \ldots K\} \qquad (24)$$

From Eq.(15), the corresponding algebraic criterion for collision avoidance of the redundant manipulator

$$J > 0 \quad \Longleftrightarrow \quad H_i \bigcap F_j = \phi \qquad (25)$$

can be obtained. The describing function $J$ can measure the distance between the manipulator and obstacles. Therefore, the describing function $J$ can be used to carry out planning of collision-free motion for the redundant manipulators.

### 3.2. *The Calculation of the J Function*

From the expression of the $J$ function ($Eq.$(16)–(18)), its calculation can be effectively accomplished by linear programming. Input parameters are only the coordinates of vertices. The

[1] $J$ stands for the describing function, which is different from the Jacobian matrix $J_e$

computational times are related to the total number of vertices. An important feature of function $J$ lies in that the computational complexity increases less when the space dimensionality increases from $2D$ to $3D$. The example of the $J$ function will be given in $2D$ as follows:

Given a rectangle with four vertices $(30,20),(30,50),(60,50)$ and $(60,20)$ and a triangle with three vertices $(10,30),(20,70)$ and $(30,40)$, the $J_{12}$ between two polygons can be solved by $Eq.(11)-(13)$. As it is a planar problem, $p=2$. Substituting the values of vertices, $p_i(s_i=4)$ and $(k_j=3)$ into $Eq.(11)-(13)$ yields

$$J_{12} = \min_{\Omega_{12}}(y_1+y_2+y_3+y_4)$$

$$30\beta_1+30\beta_2+60\beta_3+60\beta_4-10\mu_1-20\mu_2-30\mu_3+y_1 = 0$$

$$20\beta_1+50\beta_2+50\beta_3+20\beta_4-30\mu_1-70\mu_2-40\mu_3+y_2 = 0$$

$$\beta_1+\beta_2+\beta_3+\beta_4+y_3 = 1$$

$$\mu_1+\mu_2+\mu_3+y_4 = 1$$

The result of calculation $J_{12}=0$ shows that there is a collision between two polygons. Let the triangle move along the negative $X$ axis in a direction that is away from the rectangle so that the distance between the two polygons becomes greater. With the increasing distance, the value of function $J$ becomes 0.083, 0.1667, and 2, respectively. Therefore, the $J$ function not only indicates whether overlap occurs between the moving object and the obstacle, but also represents a measure for the distance between the moving object and the obstacle, as shown in Fig. 1.

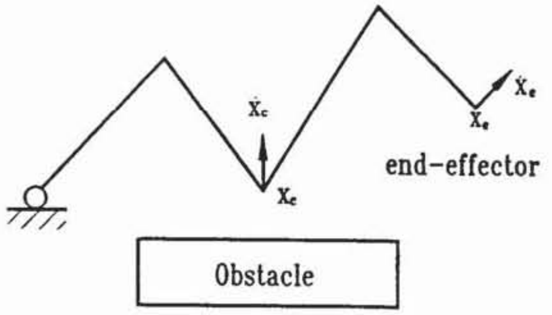


Fig. 1 An example of $J$ function calculation

### 3.3 *Trajectory Planning of Obstacle Avoidance*

The $J$ function is an effective tool used to check the interaction between the moving object and obstacles. If $J$ function between the redundant robot and the obstacles is greater than 0, then the configuration of the robot satisfies the collision-free condition. If $J=0$, the robot collides with obstacles. In this situation, the robot redundancy is utilized to inhibit the motion of links in the direction toward obstacles.

To achieve collision avoidance, it is very important to keep the value of $J$ function at a constant threshold $J_{min}$ which stands for a distance threshold to be kept between robot links and obstacles. The criterion for obstacle avoidance can be expressed as a set of inequality constraints:

$$J_{ij} \geq J_{min}, \quad i=1,\ldots,n, j=1,\ldots,K \tag{26}$$

or

$$J - J_{min} \geq 0 \tag{27}$$

Throughout the end-effector motion, inequality (26) must be satisfied. If $J_{ij}$ is less than $J_{min}$, then the the degrees of redundancy can be used to expand the distance between the critical point and the obstacles (pushing the critical point as far away as possible from obstacles). Define $X_c \in R^3$ to be the position of the critical point on the link relative to the obstacle , where the critical point $X_c$ is that point on the link currently at a minimum distance from the obstacle which is shown in Fig. 2. The location of the critical point $X_c$ can be given in Eq. (14). In order to satisfy the inequality constraints (26), the tracking errors due to the obstacle avoidance can be formed

$$e_{ci}=0, \quad \dot{e}_{ci}=0 \quad if \quad J-J_{min}\geq 0 \tag{28}$$

$$e_{ci}=-J+J_{min}, \dot{e}_{ci}=-J_c\dot{q} \quad if \quad J-J_{min}<0 \tag{29}$$

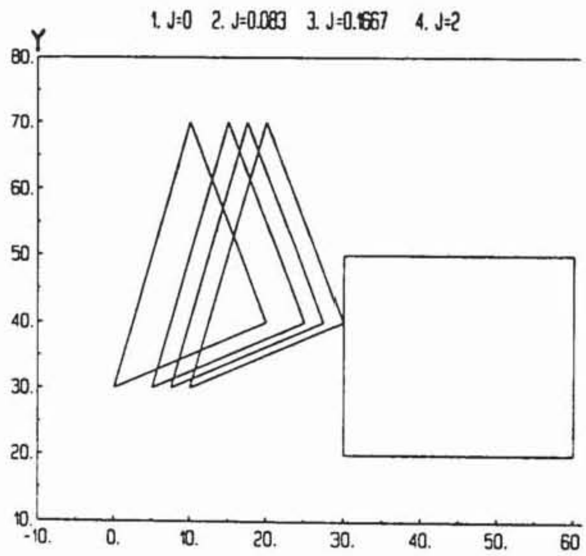


Fig.2. The velocity and location of the critical point

where $J_c\dot{q} = \dot{X}_c$ is specified as the critical point velocity in a direction away from the obstacle surface. The matrix $J_c$ can be given for any $X_c$ as given in section 4.
The control action $F_c$ for avoiding obstacles is computed as

$$F_{ci} = d(t) + K_{pi}(t)e_{ci} + K_{vi}(t)\dot{e_{ci}} \quad i = 1, \ldots, r \tag{30}$$

The collision-free trajectory generation scheme is described by the following steps:

1. Set the initial configuration of the manipulator in joint space coordinates $q_0 = [q_{10}, q_{20}, \ldots, q_{n0}]^T$.

2. Calculate the point sets:

$$h_i = \{h_{i1}, h_{i2}, \ldots, h_{is}\} \quad i = 1, 2, \ldots, n$$

$$f_j = \{f_{j1}, f_{j2}, \ldots, f_{jk}\} \quad j = 1, 2, \ldots, K$$

If the obstacles are immovable, all point sets $f_j$ are fixed. The point sets $h_i$ are dependent on the manipulator's joint variable $q$, they are determined by the direct kinematics of the manipulator.

3. Construct $J_{ij}$ and $J$ for given sets $h_i$ and $f_j$ and calculate the value of $J_{ij}$ and $J$.

4. Judge the obstacle avoidance condition. If $J \geq J_{min}$, then the obstacle has no influence on the manipulator and the homogeneous solution can be used to satisfy another desired criterion. If $J < J_{min}$, the critical point $X_c$ is located and $J_c$ is computed. According to $Eq.$(29), (30) and (12) the obstacle avoidace force $F_c$ and joint driving torque $T$ can be calculated, so that a manipulator can avoid obstacles in the workplace while completing a specific work.

## 4. SIMULATION EXAMPLES

Application of the above obstacle avoidance algorithm to a four-link planar manipulator is demonstrated through simulation. The robot parameters are link lengths $l_1 = l_2 = l_3 = l_4 = 1.0$ meter. The forward kinematics $X = f(q)$ and end-effector Jacobian matrix $J_e$ are

$$\begin{aligned} X_1(t) = & \; c_1 + c_{12} + c_{123} + c_{1234} \\ X_2(t) = & \; s_1 + s_{12} + s_{123} + s_{1234} \end{aligned} \tag{31}$$

where $s_1 = \sin(q_1)$, $s_{12} = \sin(q_1+q_2)$, etc. For the four-link robot with two end-effector coordinates controlled, the $J_c$ in $Eq.$(29) can possess at most two rows. The $J_c \in R^{2\times4}$ can be computed for each critical point.
Critical point on link 1

$$J_c = \begin{bmatrix} -\alpha_1 s_1 & 0 & 0 & 0 \\ \alpha_1 c_1 & 0 & 0 & 0 \end{bmatrix} \tag{32}$$

Critical point on link 2

$$J_c = \begin{bmatrix} -s_1 - \alpha_2 s_{12} & -\alpha_2 s_{12} & 0 & 0 \\ c_1 + \alpha_2 c_{12} & \alpha_2 c_{12} & 0 & 0 \end{bmatrix} \tag{33}$$

Critical point on link 3

$$J_c^T = \begin{bmatrix} -s_1 - s_{12} - \alpha_3 s_{123} & c_1 + c_{12} + \alpha_3 c_{123} \\ -s_{12} - \alpha_3 s_{123} & c_{12} + \alpha_3 c_{123} \\ -\alpha_3 s_{123} & \alpha_3 c_{123} \\ 0 & 0 \end{bmatrix} \tag{34}$$

Let $(X_i, Y_i)^T \in R^2$ be the location of joint $i$ relative to the base frame, and $(X_{cj}, Y_{cj})^T \in R^2$ be the location of the critical point, then

$$\alpha_i = \sqrt{(X_{cj} - X_i)^2 + (Y_{cj} - Y_i)^2} i, j = 1, 2, 3, 4 \tag{35}$$

where

$$X_{cj} = \sum_{s=j}^{j+1} \beta_s X_s$$

$$Y_{cj} = \sum_{s=j}^{j+1} \beta_s Y_s \qquad j = 1, 2, 3, 4$$

$\beta_s$ is defined as in Eq.(14).

The computerized generation of equations of motion that relates joint torques $T \in R^4$ and joint angles $q \in R^4$ is given, e.g. by Schiehlen (1984).

The initial configuration of the robot for this simulation is $q(0) = [-\pi/4, \pi/3.7, \pi/4.5, \pi/9]^T$ and the robot is initially at rest. The desired Cartesian trajectory is a straight line parallel to the $Y$ axis, connecting $[250, 15]^T to [250, -65]^T mm$. Fig.3 shows the trajectory when the collision avoidance scheme based on $J$ function and configuration control is implemented. The collision between robot links and the obstacle is sucessfully avoided, while Carterian trajectory tracking is not affected.

Fig.4 indicates robot configuration in another rectangular obstacle simulation. The initial configuration of the robot is $q(0) = [\pi/4, -\pi/2, \pi/2, -\pi/2]^T$ . The desired end-effector trajectory is $X_d(t) = [282, 50t]^T mm$ for $t \subset [0, 1.7]s$. The results of simulation are given in Fig.4 and show that the desired end-effector trajectory is tracked closely and the rectangular obstacle is avoided.

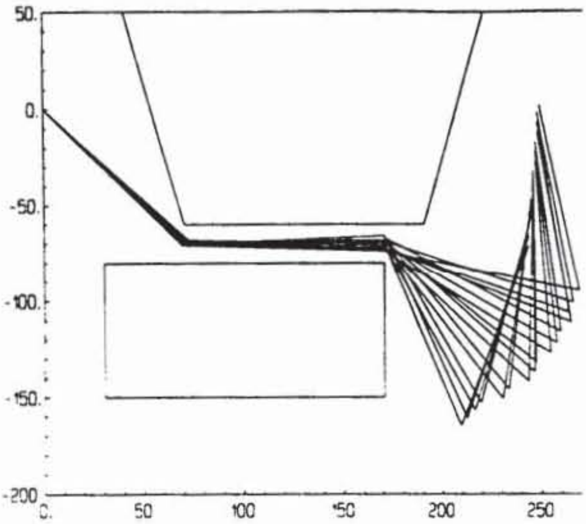


Fig. 3 Obstacle avoidance

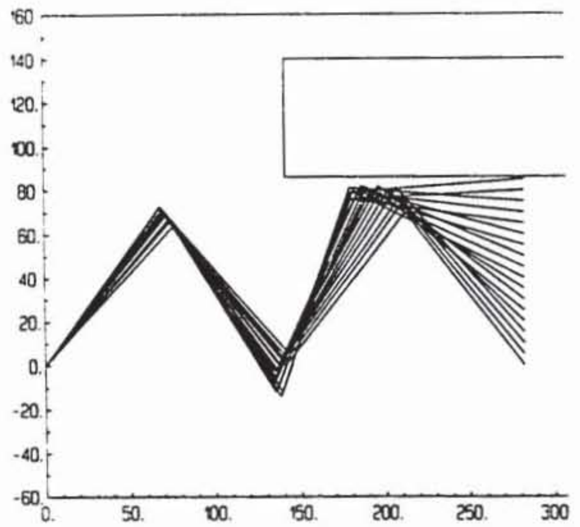


Fig. 4 Rectangular obstacle simulation

## 5. CONCLUSION

This article proposes a formulation of the obstacle avoidance for a redundant manipulator. The describing function $J$ is introduced into the trajectory planning algorithm to judge the collision and locate critical points between the redundant manipulator and the obstacles so that the algorithm can be effectively implemented. First, set the initial configuration of the manipulator in joint space coordinates and input the coordinate values of vertices whose convex hulls generate polyhedra. The redundant manipulator and obstacles are modelled as polyhedra. Then, construct the $J$ function between the redundant manipulator and obstacles and define the obstacle avoidance performance criterion as $J > J_{min}$. Throughout the end-effector motion, this algebraic criterion must be satisfied. Finally, use configuration control algorithm to reach the collision-free movement configuration of the redundant manipulator. In contrast to the previous methods, the proposed algorithm which can achieve the desired end-effector motion and avoid obstacles does not require the complex distance function and dynamic model calculations. The method is illustrated by simulation results of a four-link robot example.

Acknowledgments The first author thanks Alexander von Humboldt foundation, Germany, for supporting this research.

## 6. REFERENCES


Colbaugh R. et al (1989). Obstacle avoidance for redundant robots using configuration control, *Journal of Robotic Systems*, Vol. 6, No. 6, pp.721-744.

Galicki M.(1992). Optimal planning of collision-free trajectory of redundant manipulators, *The Intern. J. of Robotics Research*, Vol. 11, pp.549-559.

Gilbert E. G. and Hong S. M.(1989). A new algorithm for detecting the collision of moving objects, *Proc. of 1989 IEEE Conf. on Robotics and Automation*,Scottsdale, AZ, pp.8-14.

Guo Z.Y. and Hsia T.C.(1993). Joint trajectory generation for redundant robots in an environment with obstacles, *J. of Robotics Systems*, Vol.10, pp.199-215.

Khatib O.(1986). Real-time obstacle avoidance for manipulators and mobile robots, *The Intern. of Robotics Research*, Vol. 5, pp. 90-98.

Schiehlen W.(1984): Computer generation of equation of motion, *Computer aided analysis and optimization of mechanical sysytem dynamics*, Springer-Verlag, pp. 183-215.

Schiehlen W.(1992): Multibody system and robot dynamics, *Proc. of RoManSy 8*, Cracow, Poland, pp. 13-21.

Seraji H.(1989): Configuration control of redundant manipulators: Theory and implementation, *IEEE Trans. on Robotics and Automation*, Vol. 5, No. 4, pp. 472-490.

Xiong Y. L. and Ding H.(1989). General criterion and control strategy of collision-free movement for manipulators, *Intern. J. of Robotics and Automation*,Vol. 4, No. 2, pp.75-90.