Proceeding of the 2006 IEEE
International Conference on Automation Science and Engineering
Shanghai, China, October 7-10, 2006

# Redundant Manipulator Control with Constraints for Subgoals

Changwu Qiu, Qixin Cao and Yijun Sun

*Research Institute of Robotics*
*Shanghai Jiao Tong University*
*Shanghai, P.R.China*
*{ cw_qiu, qxcao,sunyj}@sjtu.edu.cn*

***Abstract* – This work analysed the general form of solution for Inverse Redundant Kinematics (IRK) problem in geometry view, and proposed new methods to formulate the obstacle and singularity avoidance as constraints, which were incorporated into standard non-linear Quadratic Programming (QP) method to resolve the IRK problem. With this method, the compromise or conflict between multi optimization criteria and the complicated weighted adjustment process in weighted sum method for IRK were removed. Obstacle avoidance and physical limits, such as joint range, velocity and acceleration, were formulated as dynamic inequality constraints. For on-line inescapable singularity avoidance, the vector, according to the minimum singular value of the Jacobian matrix, was used to revise the desired tracking velocity when manipulators move near to singularity. Numerical examples show the validity and effectiveness of the proposed methods.**

***Index Terms* – Inverse redundant kinematics, obstacle avoidance, on-line singularity avoidance, and dynamic constraints, quadratic programming.**

## I. INTRODUCTION

For the remarkable ability to reconfigure the manipulator without changing the end-effector pose, redundant manipulator can perform specific tasks and overcome the restrictions simultaneously, such as singularities [1], obstacle, joints' range and velocity limits [2]-[6], and multi optimization criteria [2]-[11]. Redundant manipulators have more degrees of freedom (DOF) than that required by task, so infinite solutions exist for the Inverse Redundant Kinematics (IRK). This feature gives the chance that many methods or strategies were studied and realized for taking full advantage of the capabilities of redundant manipulators, such as pseudoinverse method [12], Extended Jacobian Method (EJM) [13], Gradient Projection Method (GPM) [14], compact Quadratic Programming (QP) method [4]-[7] and Direct Search Method (DSM) [8]-[11].

Extra DOF of redundant manipulator can be utilized to avoid obstacles while fulfilling the end-effector motion. Many researchers adopted GPM to control redundant manipulator configuration so as to realize obstacle avoidance [2][3]; GPM extends the pseudoinverse solution by projecting the gradient of distance measure function of robot link to obstacle in null space; the high computation burden, smoothness and physical limits in implementation are its main concerns. EJM as a general expression of Lagrangian multiplier method [15] augments task space with subtask, such as obstacle avoidance; strong limitation on manipulator configuration, limited ability for realizing multi-subtasks and algorithmic singularity are it's main drawback [8][11].

With compact QP method by Cheng et al [4]-[7], Obstacle avoidance, singularity avoidance and drift-free were formulated as quadratic definite objective function together with solution feasibility objective function, such as minimum joint velocity. Despite the popularity of the weighted-sum method, it has some serious drawbacks that are noteworthy. Except the complicated weight adjust mechanism and the compromise between multi optimization criteria, the most serious drawback is optimizing the weighted sum does not necessarily mean that all or most of the criteria are acceptable [8].

DSM [8]-[11] searches the new solution vector by introducing small perturbation to the base solution vector. Then the one, which reduces tracking error of end-effector and optimizes performance criteria, was chosen as new base solution vector. This process continues until the tracking error and performance criteria satisfy the required ones. Other than compact QP method that was realized at joint velocity level, DSM was realized on configuration manifold directly. DSM is more suitable and comprehensive than QP method to evaluate the configuration control quality in view of operation task fulfillment, but less efficiency in view of real time implementation.

In above-mentioned methods, inverse redundancy kinematics solution with obstacle and singularity avoidance is resolved by complicated computation process based on inverse of Jacobian minor or pseudoinverse of Jacobian matrix, some need the distance-function derivatives with respect to joint velocity. The heavy computation load may rule out real-time applications [16]. Neural network approach was recently used to resolve the IRK and obstacle avoidance applications [17][18]. The high quality of nonlinear mapping for neural network makes the IRK solution effectively, however the infinite solution within IRK makes the selection of training data be crucial for the quality of redundancy control algorithm.

In this work, we analyzed and constructed the general form of solution for first order IRK problem based on the row vectors of the Jacobian matrix, which span the subspace of the tangent space of configuration manifold. In considering the real time implementation requirement and the deficiency of weighted sum method, we adopted the QP method, but obstacle and inescapable singularity avoidance were formulated as inequality and equality constraints respectively. By this way, the compromise or conflict and complicated weight adjustment in weighed sum of multi goal programming method [5][8] were removed. The new formulation of obstacle avoidance inequality revises the velocity of the critical point on manipulator to obstacle subtly to realize obstacle avoidance. The inescapable singularity was avoided successfully by revising the object end-effector velocity. Examples show the validity of our methods.

In following sections, section 2 shows the general formulation of IRK in geometry view. Section 3 proposes new formulations to set up the inequality constraints to realize obstacle avoidance with QP method. Section 4 presented the on-line singularity avoidance method. Section 5 shows the obstacle and singularity avoidance simulation result with a planar manipulator. Conclusion was summarized in section 6.

## II. GENERALIZED FORMULATION OF IRK PROBLEM

An analytical solution to the inverse kinematics is usually impossible for redundant manipulator due to the nonlinearity, so the IRK solution is often resolved at velocity level due to the linear attribute. We will find the solution for IRK as generalized form based on analysis of the relationship between the tangent space of configuration manifold and the Jacobian matrix.

### *A. Analysis and formulation*

The forward kinematics and differential kinematics of the redundant manipulator can be represented as

$$X = f(q) \tag{1}$$

$$\dot{X} = J(q)\cdot\dot{q} \tag{2}$$

Where $X = [x_1, x_2, \ldots x_n]^T \in R^n$ represents the position and orientation of end-effector in Cartesian space, $q = [q_1, q_2, \ldots q_m] \in R^m$ ( $m > n$ ) represents the joint variables in configuration space, $f(q) \in R^n$ is vector function depicted the forward kinematics relation, $J(q) \in R^{n\times m}$ is the end-effector Jacobian matrix which is consisted of joint twist respect to Cartesian space.

Let $N$ be the n-dimensional manifold, which defines the task in Cartesian space, and $M$ be the corresponding m-dimensional manifold in configuration space. We can find $q \in M$, and suppose that $(U,\phi)$ is a chart around point $q$ and $(V,\varphi)$ is a chart around point $X = f(q)$ [19]. Equation (2) depicts the mapping $f_*$ between the tangent space $TM_q$ located at point $q$ on manifold $M$ in configuration space and the tangent space $TN_X$ located at point $X = f(q)$ on manifold $N$ in Cartesian space. Considering the end-effector velocity vector $\dot{X} \in TN_X$ and joint velocity vector $\dot{q} \in TM_q$

$$\dot{X} = x^1 \frac{\partial}{\partial x_1} + \ldots x^n \frac{\partial}{\partial x_n} \tag{3}$$

$$\dot{q} = q^1 \frac{\partial}{\partial q_1} + \ldots + q^m \frac{\partial}{\partial q_m} \tag{4}$$

Where $x^i$ is the component of $\dot{X}$ in natural basis $\frac{\partial}{\partial x_i}$, $i = 1,2,\ldots n$ ; $q^j$ is the component of $\dot{q}$ in natural basis $\frac{\partial}{\partial q_j}$, $j = 1,2,\ldots m$ . We can find from (2) that the $ith$ component of $\dot{X}$ is the projection of velocity vector $\dot{q}$ on the map of natural coordinates axis $\frac{\partial}{\partial x_i}$ at point $q$ in configuration space, and the $ith\ row(J(q))$ is the map of natural coordinates axis $\frac{\partial}{\partial x_i}$ in configuration space at point $q$ , which has the form $row(J)_i = \sum_{j=1}^{n} (J)_{i,j} \frac{\partial}{\partial q_j}$. Proof can be found in appendix.

$J(q)$ is a $n\times m$ rectangular matrix, the vector space spanned by rows of $J(q)$ is only of a subspace of $TM_q$. Suppose that the rank of $J(q)$ is $r_J$, we can find the other $m - r_J$ independent m-dimensional vector fields that construct a basis group of vector fields for space $TM_q$ together with linear independent $r_J$ rows vector fields of $J(q)$. Let matrix $N_q$ represent the $m - r_J$ independent m-dimensional vector fields as

$$N_q = [n_1, n_2, \ldots n_{m-r_J}]_{m\times(m-r_J)} \tag{5}$$

Here $n_i$ is the m-dimensional column vector. It is worthy of note that $n_i$ is unnecessary orthogonal to space spanned by row vectors of $J(q)$. Now the mission of the inverse kinematics for redundant robot is finding the velocity vector $\dot{q}$ in space $TM_q$ under the condition that the projection of $\dot{q}$ on the space $R(J(q)^T)$ is invariant, namely satisfies the end-effector velocity $\dot{X}$.

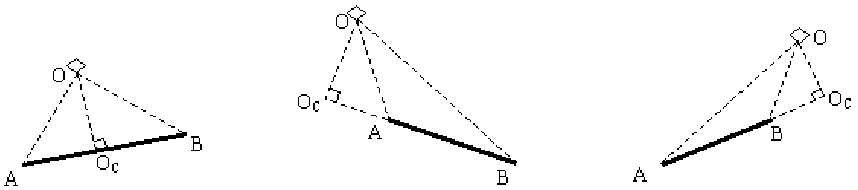
From above analyses, we can construct the basis of linear space $TM_q$ as the collection of vectors

$$\{p_1, p_2, \ldots p_m\} = \left\{\sum_{j=1}^{m}(J)_{1,j}\frac{\partial}{\partial q_j},\ldots,\sum_{j=1}^{m}(J)_{r_J,j}\frac{\partial}{\partial q_j},\sum_{j=1}^{m}(N_q^T)_{1,j}\frac{\partial}{\partial q_j},\ldots,\sum_{j=1}^{m}(N_q^T)_{m-r_J,j}\frac{\partial}{\partial q_j}\right\} \tag{6}$$

Here $p_i \in R^{m\times 1}$ is column vector. The point $q$ and $p_i, i \in (1,2,\ldots m)$ construct an affine coordinates frame. Joint velocity $\dot{q}$ can be represented in this frame as

$$\dot{q} = [p_1, p_2, \ldots p_m]\cdot\lambda = [J_{r_J}^T \quad N_q]\cdot\begin{bmatrix}\lambda_a \\ \lambda_b\end{bmatrix} \tag{7}$$

Here $\lambda \in R^m$, $\lambda_a$ is the components of $\dot{q}$ on the $r_J$ linear independent row vectors of $J(q)$ ; $\lambda_b$ is the components of $\dot{q}$ on the column vectors of $N_q$. It is worth noting that particular choice of $p_i (i > r_J)$ is viable,




(a) C: $=O_C$ (b) C: $=A$ (c) C: $=B$

Fig.1 location of critical point C according to three possible situations for obstacle point and the link $L_{AB}$

and the condition that $p_i(i > r_J)$ is linear independent to each other and to row vector of Jacobian matrix, is the premise. If we take the derivative of optimization criteria $g_j$ as $p_j = \frac{\partial g_j}{\partial q}, j > r_J$, and limit the corresponding components $\lambda_j > 0$, the corresponding solution $\dot{q}$ will make the criteria $g_j$ increase.

## B. Solution of IRK

Equation (7) gives the generalized expression of joint velocity. Kinds of IRK solutions at velocity level are the particular form of (7). Suppose $Rank(J) = n$ and $\lambda_b = 0_{(m-n)\times 1}$, we can get the pseudoinverse or weighted pseudoinverse solution by setting $\lambda_a = (J(J)^T)^{-1}\dot{X}$ or $\lambda_a = W^{-1}(J)^T(JW^{-1}(J)^T)^{-1}\dot{X}$, here $W \in R^{m\times m}$ is the weighted matrix. Similarly any other solution variants in velocity level can be formed in (7) by selecting properly the column vectors of $N_q$ and $\lambda$, such as solution of EJM [13], GPM [14] and QP [4]-[7].

While satisfying the end-effector velocity constraints, redundant DOF is usually used to fulfil some subtasks or to optimize some criteria [3]-[7][15]-[17]. Following standard quadratic programming method can be used for the resolution of (2) with equality and inequality constraints [4]-[7].

$$\text{Minimize} \quad \frac{1}{2}\lambda^T H\lambda + \varphi\lambda \tag{8}$$

$$\text{subject to} \quad \mathrm{J}\cdot[J_{r_j}^T \quad N_q]\cdot\lambda = \dot{\mathrm{X}} \tag{9}$$

$$A\cdot\lambda \le B \tag{10}$$

Where $H$ is a positive definite cost matrix with dimension $m\times m$, and $\varphi$ is a linear cost vector with dimension $1\times m$. Equation (9) represents the relationship between the end-effector velocity and the joint velocity.

Physical constraints were formulated as inequality in (10). Reference [4] has introduced the definition of joint range limit and joint velocity limit, we will summarize here for convenience and add acceleration limit into constraints. Let $q_l(q_u)$ represent the lower (upper) limit of joint range, $\dot{q}_l(\dot{q}_u)$ represent the lower (upper) joint velocity limit of joint and $\ddot{q}_l(\ddot{q}_u)$ represent the lower (upper) joint acceleration limit. Then the available joint range, joint velocity and joint acceleration can be converted to velocity constraints as

$$\frac{q_l - q(t)}{\Delta t} \le \dot{q}(t) \le \frac{q_u - q(t)}{\Delta t} \tag{11}$$

$$\dot{q}_l \le \dot{q}(t) \le \dot{q}_u \tag{12}$$

$$\dot{q}(t-\Delta t) + \Delta t\cdot\ddot{q}_l \le \dot{q}(t) \le \dot{q}(t-\Delta t) + \Delta t\cdot\ddot{q}_u \tag{13}$$

We can combine (11), (12) and (13) to matrix form as (10), then $A$ and $B$ have form

$$A = \begin{bmatrix} -I_m \\ I_m \end{bmatrix}_{2m\times m} \cdot \begin{bmatrix} J \\ N_q^T \end{bmatrix}^T \tag{14}$$

$$B = \begin{bmatrix} -\max(\frac{q_l - q(t)}{\Delta t}, \dot{q}_l, \dot{q}(t-\Delta t) + \Delta t\cdot\ddot{q}_l) \\ \min(\frac{q_u - q(t)}{\Delta t}, \dot{q}_u, \dot{q}(t-\Delta t) + \Delta t\cdot\ddot{q}_u) \end{bmatrix}_{2m\times 1} \tag{15}$$

Minimum 2-norm of joint velocity vector is often chosen as optimization criteria in QP scheme [4][7]. However adding quadratic style obstacle and singularity avoidance criteria into objective function will inevitably weaken minimum 2-norm of joint velocity criteria. In following sections, we will realize the obstacle avoidance by formulating it as inequality constraints into (10) and non-escapable singularity [4] avoidance by revising the equality constraints (9) at the situation that the end-effector moves near to the singularity configuration.

## III. Obstacle avoidance constraint considerations

The obstacles in environment must be properly dealt with to prevent possible damage to manipulators. Obstacle point can be extracted by checking the minimum distance information of commonly convex obstacle object to manipulator link [17][18]. We suppose that obstacle points have been detected. Except the situation showed in Fig.1 (a), critical points $C$ on manipulator for possible collision locates only at joint, Fig.1 (b) and (c) show the situation.

Let $P_A$, $P_B$ represent the joint $A$ and $B$, $O$ represents the obstacle point, the vector $\vec{V}_{AB}$ according to link $L_{AB}$ will be

$$\vec{V}_{AB} = [v_1 \quad v_2 \quad v_3]^T = \overrightarrow{P_A P_B} \tag{16}$$

So the point $O_C$ can be easily located as the intersection point between plane and line by resolving the group of equation

$$\begin{cases} v_1(x - O_x) + v_2(y - O_y) + v_z(z - O_z) = 0 \\ \dfrac{x - P_{Ax}}{v_1} = \dfrac{y - P_{Ay}}{v_2} = \dfrac{z - P_{Az}}{v_3} \end{cases} \tag{17}$$

Three cases showed in Fig.1 can be distinguished by following pseud code:

*If* $O_{Cx} \in [P_{Ax}, P_{Bx}]$, *then case (a)*

*Else if* $\left\|\overrightarrow{O_C P_A}\right\|_2 \le \left\|\overrightarrow{O_C P_B}\right\|_2$, *then case (b)*

*Else case (c)*

*End*

In [17], the inequality constraint limits stringently the solution space, because only the sign of vector $\overrightarrow{OO_C}$ was used to formulate the inequality constraints for the velocity of critical point on manipulator link. For reducing the nonessential limit on available solution space, projection of critical point velocity on the vector $\overrightarrow{OO_C}$ will be used to set up the inequality constraints in the QP method. Unit vector $\overrightarrow{OO_C}$ can be formulated as

$$r_{oc} = \left[X_{O_C} - X_O \quad Y_{O_C} - Y_O \quad Z_{O_C} - Z_O\right]^T \Big/ \left\|\overrightarrow{OO_C}\right\| \tag{18}$$

When the distance, $d$, from the obstacle to the manipulator link is less than the safety margin, $d_S$, we must adopt obstacle avoidance measure to eliminate the possible damage to the manipulator. Considering the relation between the critical point velocity and the vector $r_{oc}$ for a static obstacle, we can find that the critical point will escape the possible collision zone when the projection value, $P$, of the critical point velocity on $r_{oc}$ is positive. So obstacle avoidance inequality constraints (only static obstacle) can be roughly formulated as

$$P = r_{oc}^T \cdot J_O \cdot \dot{q} \geq 0 \tag{19}$$

Here $J_O$ is the Jacobian matrix with the critical point $O_C$ as velocity reference point. If manipulator works in a dynamic environment, the sum of the projection of obstacle point velocity and the critical link point velocity on $r_{oc}$ must be positive to assure manipulator moving away from the obstacle. Here only the static obstacle situation is formulated, but extension to the dynamic is intuitive.

We can find that a discontinuity in joint velocity will occur when $P < 0$ and constraint (19) was suddenly applied into (10). So the smoothing measure $P \geq t$ must be adopted to avoid discontinuity. Where $t$ has the form

$$t = \begin{cases} P & d > d_S + \sigma \\ \rho(d) \cdot P_C & d_S < d \leq d_S + \sigma \\ \eta & d \leq d_S \end{cases} \tag{20}$$

where $\rho(d) \in [0,1]$ is a bell shape function, $\sigma \in R^+$ is the buffer distance, $\eta$ is a small positive number and $P_C = \{P \mid d(q,t) = d_S + \sigma\}$. $[d_S, d_S + \sigma]$ is a buffer zone. Parameter, $\eta$, will push the vulnerable manipulator link away from the safety margin. Combining (14), (15) and (20), matrix $A$ and $B$ can be extended.

Above process formulates the obstacle avoidance and physical limits as inequality constraints for the QP method. However the constraints are over stringent sometimes, it means the constraints (10) must be broken to fulfil the task. For satisfying the physical limit and safety consideration, high-level task planning algorithm must be applied to construct the task track again.

It is well known that infeasible solution may be obtained when end-effector moves near to the singularity point [4]. Following section will formulate the singularity avoidance into equality constraints by revising (9) when end-effector moves near to the singularity point.

## IV. SINGULARITY AVOIDANCE CONSIDERATIONS

When end-effector moves to the singularity point, infeasible solution will be obtained and the Jacobian matrix will lose rank. Singularity-free motion can be achieved with an off-line path planning for non-redundant manipulator. However, that needs a-priori knowledge of all the singular configurations of the manipulator. For a redundant manipulator, self-motion can realize escapable singularity avoidance but cannot do that for the inescapable singularity [20]. An on-line singularity avoidance method will alleviate the burden to check if the configuration of manipulator is singular. We set up a method for on-line singularity avoidance.

The singularities can be directly identified from the singular value of the Jacobian matrix [21]. The singular value decomposition of the Jacobian matrix is

$$J = U \Sigma V^T \tag{21}$$

Where $U = [u_1, u_2, ... u_m]$ and $V = [v_1, v_2, ... v_n]$ are orthogonal matrixes and $\Sigma$ is a $m \times n$ matrix whose diagonal elements are the ordered singular values of $J$

$$\Sigma = [diag(\sigma_1, \sigma_2, ... \sigma_m) \mid 0] \ , \ (\mathrm{m} < \mathrm{n}) \tag{22}$$

A singular direction vector, $u_S$, in workspace is identified with the corresponding singular value is zero; the vector $u_S$ can be used to revise the object end-effector velocity to avoid the inescapable singularity. Revising method can be formulated as

$$\dot{X}_R = \dot{X} - k \cdot \frac{1 - sign(u_S \bullet \dot{X})}{2} (u_S \bullet \dot{X}) \cdot u_S \tag{23}$$

Where $\dot{X}_R$ is the revised end-effector velocity and $k$ is defined as

$$k = \begin{cases} 0 & \sigma_{\min} > \sigma_S \\ 1 & \sigma_{\min} \leq \sigma_S \end{cases} \tag{24}$$

Where $\sigma_{\min}$ is the minimum singular value of the manipulator Jacobian matrix, $\sigma_S$ is the low bounds for allowable singular value. Above formulations confine the singularity by revising the end-effector motion with the vector pointed to the singular configuration. $\dot{X}_R$ will make the end-effector bypass the singularity point in workspace. So this method is more suitable to inescapable singularity avoidance to prevent the infeasible solution for (2).

Above sections have formulated the obstacle avoidance and singularity avoidance as inequality and equality constraints respectively. So the compromise between the minimum 2-norm of joint velocity and the other subtasks, such as obstacle avoidance and singularity avoidance, has been removed.

## V. NUMERICAL EXAMPLES

Two examples are given in this section to demonstrate the effect of above methods for resolving the IRK problem. Matlab was used to realize the algorithm and numerical simulation. A classic three-link planar manipulator shown in Fig.2 will be used in following examples.

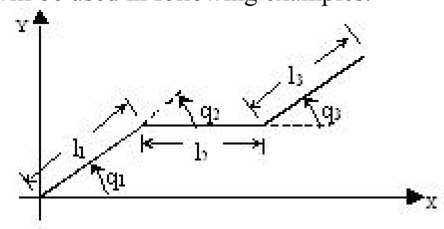



Fig.2 Three-link planar manipulator.

First example is for the point obstacle avoidance only. A circular path-tracking mission and avoiding a static point obstacle were used to exhibit the validity of obstacle avoidance method. The manipulator link parameters and initial states are

$$[l_1, l_2, l_3] = [1, 0.8, 0.6] \text{ meters}$$
$$[q_1, q_2, q_3] = [0.277, -0.5, -1.25] \text{ rad}$$
$$[\dot{q}_1, \dot{q}_2, \dot{q}_3] = [0, 0, 0] \text{ rad/s}$$

The physical limits are defined as

$$q_u = -q_l = [3, 3, 3]^T \text{ rad}$$
$$\dot{q}_u = -\dot{q}_l = [1.2, 1.5, 1.6]^T \text{ rad/s}$$
$$\ddot{q}_u = -\ddot{q}_l = [2.4, 2.4, 3]^T \text{ rad/s}^2$$

The desired end-effector path is a circle with radius 0.6 m and center (1.8,0), which is tracked in counter-clockwise. The Point obstacle locates at (1.1,0.6). Other parameters are $d_S = 0.1m$ , $\sigma = 0.2m$ and $\eta = 0.01$ . Fig.3 shows the simulation results for pseudoinverse solution without obstacle avoidance consideration. Fig.4 shows the simulation results for QP method with point obstacle (hexagram) avoidance as inequality constraints. Fig.4 (b), (c) and (d) show the history of joint angle, velocity and shortest distance from the obstacle to the manipulator. From Fig.3~4, the validity of the obstacle avoidance method is obvious.

Fig. 5 is for the inescapable singularity avoidance and obstacle avoidance simultaneously. The link parameters and initial states are same to the above one except the initial joint angle $[q_1, q_2, q_3] = [-0.11, -0.75, -1.97]$ . The circumscribing singular value is $\sigma_S = 0.06$ . The desired tracking path is an elliptical curve. The top point of the elliptical curve, (2.4,0), is on the workspace boundary. Fig.5 shows the simulation result and the minimum singular value history. We can find that the end-effector bypasses the inescapable singularity point successfully. The validity of singularity and obstacle avoidance is obvious in two numerical examples.

## VI. CONCLUSIONS

In this work, the general form of solution to the IRK was analysed with differential geometry tool, and its possible extension has been pointed out. The new obstacle avoidance and singularity avoidance formulation have been proposed and formulated as inequality and equality constraints respectively, which are incorporated into QP method to resolve the IRK problem. The singularity avoidance method suits for the inescapable singularity avoidance. By the similar way, other commonly required subtasks, such as drift-free and joint torque limit, can be formulated as dynamic inequality constraints too.

The compromise or conflict between the optimization criteria in weighted sum method [4]-[7] has been successfully removed. Numerical simulation results for a three-link planar manipulator demonstrate the method's effectiveness.

## APPENDIX

For any point $q \in U$ and $p \in V$ , the map relation between $\frac{\partial}{\partial q_i}$ and $\frac{\partial}{\partial x_j}$ [pp18, 22] is

$$f_*\left(\frac{\partial}{\partial q_i}\right) = \sum_{\beta=1}^{n} \left(\frac{\partial f^\beta}{\partial q_i}\right) \frac{\partial}{\partial x_\beta} \quad (25)$$

So the $ith$ column of Jacobian matrix $J(q)$ represents the components of the map of $\frac{\partial}{\partial q}$ in $TN_X$ on the natural

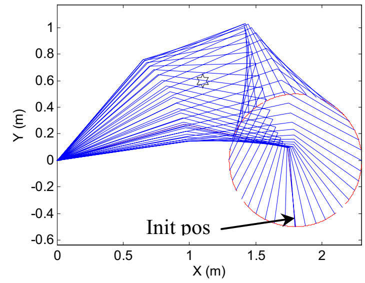
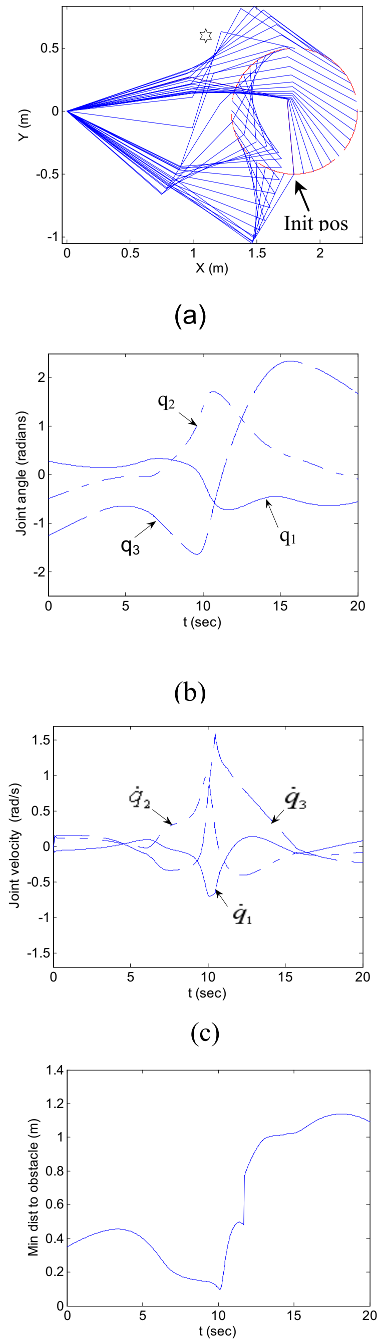



Fig.3 Simulation result with psedudoinverse method




(a)

(b)

(c)

(d)

Fig.4 Point obstacle (hexagram) avoidance as inequality constraints in QP method, (a) simulation result for the tracking process (b) joint angle history;(c) joint velocity history;(d) the history of the shortest distance from obstacle to manipulator.

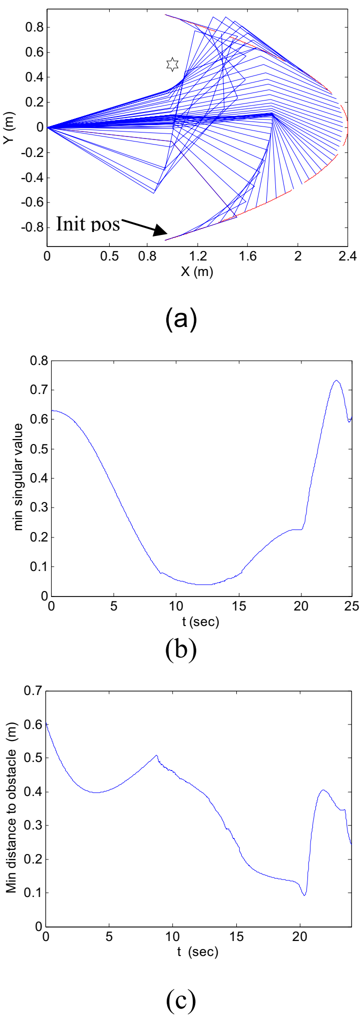



Fig.5 Singularity and obstacle avoidance as equality and inequality constraints in QP method, (a) simulation result for the tracking process; (b) the history of minimum singular value; (c) the history of shortest distance from obstacle to manipulator.

basis of $TN_X$. Considering the cotangent space $TM_q^*$ and $TN_X^*$, which are the dual space of $TM_q$ and $TN_X$ respectively, $dq_i, i \in (1,2,...m)$ and $dx_j, j \in (1,2,...l)$ are the natural basis of $TM_q^*$ and $TN_X^*$ respectively. Considering bilinear map $\langle \cdot, \cdot \rangle : TN \times TN^* \to R$, we denote

$$\begin{aligned}
\left\langle \frac{\partial}{\partial x_\beta}, \mathrm{dx}_\beta \right\rangle &= \left\langle \frac{\partial}{\partial x_\beta}, d(x_\beta \circ f) \right\rangle \\
&= \left\langle \frac{\partial}{\partial x_\beta}, \sum_{i=1}^{n} \frac{\partial f^\beta}{\partial q_i} dq_i \right\rangle \\
&= \sum_{i=1}^{n} \left\langle \frac{\partial}{\partial x_\beta}, dq_i \right\rangle \frac{\partial f^\beta}{\partial q_i} \qquad (26) \\
&= \sum_{i=1}^{n} \left\langle \frac{\partial}{\partial q_i}, dx_\beta \right\rangle \frac{\partial f^\beta}{\partial q^i} \\
&= \left\langle \sum_{i=1}^{n} \frac{\partial f^\beta}{\partial q_i} \frac{\partial}{\partial q_i}, dx_\beta \right\rangle
\end{aligned}$$

So we have $f_*(\frac{\partial}{\partial x_\beta}) = \sum_{i=1}^{n} \frac{\partial f^\beta}{\partial q_i} \frac{\partial}{\partial q_i}$, this equation manifests that the $\beta th$ row of Jacobian matrix $J(q)$ represents the components of the map of $\frac{\partial}{\partial x_\beta}$ in $TM_q$ on the natural basis of $TM_q$. Therefore we can get the $\beta th$ component of $\dot{X} \in TN_X$ as

$$x^\beta = \left\langle \sum_{i=1}^{n} \frac{\partial f^\beta}{\partial q_i} \frac{\partial}{\partial q_i}, \dot{q} \right\rangle = \langle row_\beta(J(q)), \dot{q} \rangle \qquad (27)$$

## REFERENCES


[1] R.V. Mayorga and A.K. Wong: A Singularities Prevention Approach for Redundant Robot Manipulators. IEEE Int. Conf. on Robotics and Automation (1990) 812-817.

[2] A.A. Maciejewski and C.A. Klein: Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments. Int. J. of Robotics Res. 4 (1985) 109-117.

[3] Y. Nakamura, H. Hanafusa, T. Yoshikawa: Task Priority based redundancy control of robot manipulators. Int. J. of Robotics Res. 6 (1987) 3-16.

[4] F.-T. Cheng, T.-H. Chen and Y.-Y. Sun,: Efficient Algorithm for Resolving Manipulator Redundancy – The Compact QP Method. IEEE Int. Conf. on Robotics and Automation (1992) 508-513.

[5] F.-T. Cheng, W.-M. Wang and F.-C. Kung: Priority Considerations for Multiple Goals of Redundant Manipulators. IEEE Int. Conf. on Robotics and Automation (1996) 264-269.

[6] F.-T. Cheng, M.-S. Shih, F.-C. Kung and Y.-Y. Sun: The Improved Parallel Scheme for Multiple-Goal Priority Considerations of Redundant Manipulators. IEEE Int. Conf. on Robotics and Automation (1997) 2409-2414.

[7] F. -T. Cheng, T.-H. Chen, Y.-S. Wang and Y.-Y. Sun: Obstacle Avoidance for Redundant Manipulators Using the Compact QP Method. IEEE Int. Conf. on Robotics and Automation (1993) 262-269.

[8] C. Pholsiri and D. Tesar: Task-Based Decision making and control of Robotic Manipulators. Ph.D. Dissertation, University of Texas at Austin (2004).

[9] R. Hooper and D. Tesar: Multicriteria Inverse Kinematics for General Serial Robots. Ph.D. Dissertation, University of Texas at Austin (1994).

[10] M. Cetin and D. Tesar: Performance Identification and Multi-Criteria Redundancy Resolution for Robotic Systems. Ph.D. Dissertation, University of Texas at Austin (1999).

[11] M. Pryor and D. Tesar: Task-Based Resource Allocation for Improving the Reusability of Redundant Manipulators. Ph.D. Dissertation, University of Texas at Austin (2002).

[12] D. E. Whitney: Resolved motion rate control of manipulators and human prostheses. IEEE Transactions of Man-Machine system, vol. MMS-10 (1969) 47-53.

[13] J. Baillieul: Kinematic programming alternative for redundant manipulators. IEEE Int. Conf. on Robotics and Automation (1985) 722-728.

[14] A. Liegeois: Automatic supervisory control of the configuration and behavior of multibody mechanisms. IEEE Trans. On Syst. Man and Cyber, vol. SMC-7.12 (1977) 868-871.

[15] J. Baillieul: Avoiding Obstacles and Resolving Kinematic Redundancy. IEEE Int. Conf. on Robotics and Automation (1986) 1698–1704.

[16] S. I. Choi and B. K. Kim: Obstacle Avoidance Control for Redundant Manipulators Using Collidability Measure. Robotica, vol. 18 (2000) 143-151.

[17] Y. Zhang and J. Wang: Obstacle Avoidance for Kinematically Redundant Manipulators Using a Dual Neural Network. IEEE Trans. On Syst. Man and Cyber, vol.34 (2004) 752-759.

[18] Z. Mao and T. C. Hsia: Obstacle avoidance inverse kinematics solution of redundant robots by neural networks. Robotica, vol. 15 (1997) 3-10.

[19] S.S. Chern and W.H. Chern: lecture notes of differential geometry.2nd edition. University of Beijing. Beijing (2001).

[20] N. S. Bedrossian: Classification Of Singular Configurations For Redundant Manipulators. IEEE Int. Conf. Robotics Automat (1990) 818-823.

[21] O. Khatib: A unified approach for motion and force control of robot manipulators: the operational space formulation. IEEE J. of Robotics and Automat. , Vol. RA-3 (1987) 43-53.