CONF-960448--7

Resolving Kinematic Redundancy with Constraints Using the FSP (Full Space Parmeterization) Approach*

François G. Pin and Faithlyn A. Tulloch
Oak Ridge National Laboratory
Robotics and Process Systems Division
P.O. Box 2008, Bldg. 7601, MS-6305
Oak Ridge, TN 37831-6305, U.S.A.

Submitted to: 1996 IEEE International Conference on Robotics and Automation, April 22–28, 1996, Minneapolis, Minnesota

*This research was supported in part by the U.S. Air Force Combat Command (ACC), the U.S. Air Force Munition Material Handling Equipment (MMHE) Focal Point, the U.S. Air Force Reliability and Maintainability Technology Insertion Program (PRAM-RAMTIP), the U.S. Department of Defense , Office of the Secretary of Defense (OSD), and the U.S. Advanced Project Research Agency (ARPA), under Interagency Agreement 2146-H055-A1 between the U.S. Air Force Material Command (AFMC) San Antonio Air Logistics Center, Robotics and Automation Center of Excellence (SA/ALC-RACE) and the U.S. Department of Energy, and in part by the DOE-sponsored Science and Engineering Research Program administerd by the Oak Ridge Institute for Science and Engineering.

Resolving Kinematic Redundancy with Constraints Using the FSP (Full Space Parmeterization) Approach

Corresponding Author: Dr. François G. Pin
Oak Ridge National Laboratory
P.O. Box 2008
Building 7601, MS-6305
Oak Ridge, TN 37831-6305
Telephone: (423)574-6130
Fax: (423)574-4624
E-Mail: pin@ornl.gov

Abstract

A solution method is presented for the motion planning and control of kinematically redundant serial-link manipulators in the presence of motion constraints such as joint limits or obstacles. Given a trajectory for the end-effector, the approach utilizes the recently proposed Full Space Parameterization (FSP) method to generate a parameterized expression for the entire space of solutions of the unconstrained system. At each time step, a constrained optimization technique is then used to analytically find the specific joint motion solution that satisfies the desired task objective and all the constraints active during the time step. The method is applicable to systems operating in *a priori* known environments or in unknown environments with sensor-based obstacle detection. The derivation of the analytical solution is first presented for a general type of kinematic constraint and is then applied to the problem of motion planning for redundant manipulators with joint limits and obstacle avoidance. Sample results using planar and 3-D manipulators with various degrees of redundancy are presented to illustrate the efficiency and wide applicability of constrained motion planning using the FSP approach.

Summary

The new approach which is presented differs from previous work in that it allows treatment of constraints and task criteria that may vary in real time. The method is thus particularly applicable to rapidly changing environments where obstacles and other constraints are *a priori* unknown and must be detected using sensor-based techniques. In addition, the analytical solutions that the approach provides for given types of criteria and constraints are the matter of only a few additional statements in the code so that a wide variety of dynamically changing conditions can be handled using a single code with no "extra" computational burden.

Resolving Kinematic Redundancy with Constraints Using the FSP (Full Space Parmeterization) Approach

François G. Pin and Faithlyn A. Tulloch
Oak Ridge National Laboratory
Robotics and Process Systems Division
P.O. Box 2008, Bldg. 7601, MS-6305
Oak Ridge, TN 37831-6305, U.S.A.

Abstract

A solution method is presented for the motion planning and control of kinematically redundant serial-link manipulators in the presence of motion constraints such as joint limits or obstacles. Given a trajectory for the end-effector, the approach utilizes the recently proposed Full Space Parameterization (FSP) method to generate a parameterized expression for the entire space of solutions of the unconstrained system. At each time step, a constrained optimization technique is then used to analitically find the specific joint motion solution that satisfies the desired task objective and all the constraints active during the time step. The method is applicable to systems operating in *a priori* known environments or in unknown environments with sensor-based obstacle detection. The derivation of the analytical solution is first presented for a general type of kinematic constraint and is then applied to the problem of motion planning for redundant manipulators with joint limits and obstacle avoidance. Sample results using planar and 3-D manipulators with various degrees of redundancy are presented to illustrate the efficiency and wide applicability of constrained motion planning using the FSP approach.

1. Introduction

This paper deals with the control of kinematically redundant serial-link manipulators with motion constraints such as joint limits and/or obstacles. The kinematic equations of the system can be written as:

$$\overline{X} = \overline{F}(\overline{q}) \qquad (1)$$

where $\overline{X}$ represents the $n \times 1$ vector of position and orientation of a given point of interest on the manipulator (generally the end-effector) expressed in the n-dimensional Cartesian task space, $\overline{q}$ is the $m \times 1$ vector of joint coordinates for the m degrees of freedom (d.o.f.) manipulator, and $\overline{F}$ represents the forward kinematic vector function for the system. For motion control, the velocity equations obtained by taking the time-derivatives of Eq. (1) are used:

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

where an upper dot denotes a time-derivative, and $J(\overline{q})$ is the $n \times m$ Jacobian matrix of the system with component $J_{ij} = \partial F_i / \partial q_j$. Since Eq. (2) is typically highly non-linear, control of the system is generally performed using a linearized version of Eq. (2) providing first-order approximations for the displacement vectors $\Delta\overline{X}$ and $\Delta\overline{q}$ over the discretized time domain with time steps Δt:

$$\Delta\overline{X}/\Delta t \approx J_{\Delta t}\, \Delta\overline{q}/\Delta t \qquad (3)$$

When the manipulator is a redundant system, the Jacobian is a rectangular $n \times m$ matrix with $n < m$, and the system of equations represented by Eq. (3) is underspecified. Given an end-effector trajectory and the corresponding incremental displacement vectors $\Delta\overline{X}$, the matrix $J_{\Delta t}$ cannot be directly inverted, and the system has an infinity of solutions for $\Delta\overline{q}$.

In addition to Eq. (3), we consider that, at each time step, the system is subject to a set of r constraints with r varying between 0 and a finite number. The constraints, which themselves can vary at each time step, are assumed to be of the form

$$C(\overline{q}, \Delta\overline{q}) = 0 \quad , \qquad (4)$$

which many manipulator-related constraints (e.g., work space obstacles, joint limits, velocity limits) can be reduced to, as discussed in the following sections.

Several authors have previously addressed this problem (e.g., see [1], [2], [3], [4], for applications to obstacle and joint limit avoidance). All of them have used one of the two main techniques for resolution of underspecified systems of equation: constrained generalized inverse-based approaches or augmented task space methods with "extended Jacobians" [5]. In [6], we pointed out some of the shortcomings encountered when using either of these two general resolution approaches for application to real-time systems where constraints and/or task requirements may change widely and rapidly (e.g., at loop–rate and/or on a sensor–based basis) during a single trajectory. Among these shortcomings are the implicit task priority requirement of inverse–based techniques, i.e., the fact that a solution which is the sum of a particular solution obtained from a primary criterion and a homogenous solution obtained from a secondary criterion, typically does not satisfy both together (e.g., a solution obtained from a least–norm particular solution to which an obstacle–avoiding, self-motion solution has been added, is not the least-norm solution of the obstacle-avoiding solution); and the "artificial" algorithmic singularities that may be encountered with extended Jacobian and augmented task space approaches.

In recent papers [6], [7], [8], [9], we introduced a new method for the resolution of underspecified systems of algebraic equations. This method has been named the Full Space Parameterization (FSP) method because it provides, in simple form, a parameterized expression for the entire space of solutions of the basic system. With the entire solution space parameterized, specific solutions corresponding to wide ranging criteria and/or sets of constraints can be analytically found using simple constrained optimization techniques. In [6], an application of the FSP method to the problem of inverse kinematic joint velocity calculations of redundant manipulators was presented. Analytical solutions were derived for a general least norm criterion and the method's results were compared with results from the "standard" pseudo-inverse in unconstrained cases. The results appeared very promising for application of the method to other general cases, in particular the constrained problem.

In this paper, we present an FSP-based solution method to the constrained inverse kinematic problem of Eq. (3) and (4) with the constraints and the number of constraints varying at loop–rate. The FSP framework is briefly reviewed in the next section and the general analytical solutions for the constrained case are derived. Example applications to the treatment of work space obstacles and joint limit constraints are described and, in the following section, sample results using 2-D and 3-D manipulators are presented to illustrate the approach. The last section includes our concluding remarks.

2. FSP-Based Approach

In a previous paper, [6], we showed that the entire space of solutions, S, of the unconstrained Eq. (3) could be parameterized as

$$S = \left\{ \Delta\bar{q} \in \Re^m, \Delta\bar{q}(t_1,\dots,t_{m-n+1}) = \sum_{i=1}^{m-n+1} t_i \bar{g}_i; \ \sum_{k=1}^{m-n+1} t_k = 1 \right\} \tag{5}$$

where each of the $m-n+1$ linearly independent vectors $\bar{g}_i$ includes $m-n$ zero components and can be easily calculated from inversion of square $(n \times n)$ submatrices of J. It was also shown that the null space $\mathcal{N}$ of the mapping J can be parameterized using the same $\bar{g}_i$ vectors as:

$$\mathcal{N} = \left\{ \Delta\bar{q} \in \Re^m, \Delta\bar{q}(t_1,\dots,t_{m-n+1}) = \sum_{i=1}^{m-n+1} t_i \bar{g}_i; \ \sum_{k=1}^{m-n+1} t_k = 0 \right\} \tag{6}$$

At each time step therefore, a calculation of the vectors $\bar{g}_i$ for Eq. (3) provides a parameterization of the entire spaces of solutions of Eq. (3), be it for an end-effector motion or a motion in the null space. With the entire spaces of solutions of Eq. (3) now parameterized, the calculation of the specific solution satisfying the particular task requirement and all the constraints of the time step is then the matter of only a few code statements embodying the analytical expression of the corresponding parameters $t_k, k = 1, m-n+1$. A wide variety of these parameter solutions, each corresponding to particular types of requirements and constraints, can be included in the code and selected as appropriate at each time step.

Analytical solutions for the parameters can be obtained from a Lagrangian-type constrained optimization. For a general criterion $Q(\Delta\bar{q}(t_i)), i = 1, m-n+1,$ to be optimized in the space defined by Eq. (5) with a set of r general constraints $C^j(\bar{q}, \Delta\bar{q}(t_i)) = 0, j = 1, r;$ the Lagrangian is:

$$\mathcal{L}(t_i, \mu, \nu_j) = Q(t_i) + \mu\left(\sum_{i=1}^{m-n+1} t_i - 1\right) + \sum_{j=1}^{r} \nu_j C^j(t_i) \tag{7}$$

and the optimality conditions are:

$$\frac{\partial \mathcal{L}}{\partial t_i} = 0, i = 1, m-n+1; \frac{\partial \mathcal{L}}{\partial \mu} = 0; \frac{\partial \mathcal{L}}{\partial \nu_j} = 0, j = 1, r \tag{8}$$

With these $m-n+r$ conditions, analytical solutions can be found for the Lagrange multipliers μ and ν_j, $j = 1, r$, and for the vector $\bar{t}$ with components $(t_1, \dots t_{m-n+1})$. The resulting joint displacement solution $\Delta\bar{q} = \sum_{i=1}^{m-n+1} t_i \bar{g}_i$ will optimize Q while satisfying all the constraints. As an example of such an analytical derivation, consider a general criterion

$$Q = \left\| \Delta\bar{Z}(\bar{q}, \Delta\bar{q}) - \Delta\bar{Z}_r \right\|^2 \tag{9}$$

where $\Delta\bar{Z}_r$ represents a given reference operational vector characterizing the state to be acheived by the system, and $\Delta\bar{Z}$ is an operational vector function of the joint positions and displacements. Let $B(\bar{q})$ be a matrix such that

$$\Delta\bar{Z} = B(\bar{q})\Delta\bar{q} \tag{10}$$

and define the vector $\overline{H}$ and matrix G as:

$$\overline{H}, H_k = \Delta\overline{Z}_r^T B\overline{g}_k\,; k = 1, m-n+1 \quad (11)$$

$$G, G_{ij} = \overline{g}_i^T B^T B\overline{g}_k; i = 1, m-n+1; j = 1, m-n+1 \quad (12)$$

where the upper T sign denotes a transpose. Note that if B is set as the identity matrix and $\overline{H} = \overline{O}$, then the criteria reduces to the least norm of $\Delta\overline{q}$, as was utilized in the comparisons of Ref. [6].

Assume the r constraints $C^i(\overline{q}, \Delta\overline{q}(\overline{t})) = 0$ are expressed as

$$\overline{\beta}^{j^T}\overline{t} - 1 = 0; j = 1, r \quad (13)$$

a form to which many kinematic constraints (e.g., joint limits, obstacle avoidance, etc.) can be reduced as discussed in the next section. Then the optimality conditions [Eq. (8)] become:

$$\begin{cases} G\overline{t} + \overline{H} + \mu\overline{e} + \sum_{i=1}^{r} v_i\overline{\beta}^i = \overline{o} \\ \overline{e}^T\overline{t} = 1 \\ \overline{\beta}^{j^T}\overline{t} = 1; j = 1, r \end{cases} \quad (14)$$

where $\overline{e}$ and $\overline{o}$ are the $m-n+1$ dimensional vectors $\overline{e}^T = (1,1,1,\ldots 1)$ and $\overline{o}^T = (0,0,\ldots 0)$, respectively. Setting $\overline{v}^T = (v_1, \ldots v_r)$ and $a = \overline{e}^T G^{-1}\overline{e}$; and defining the vector $\overline{b}$, $\overline{c}$, and $\overline{d}$, and the matrix A by: $b_i = \overline{e}^T G^{-1}\overline{\beta}^i$, $c_i = \overline{\beta}^{i^T} G^{-1}\overline{e}$, $d_i = 1 + \overline{\beta}^{i^T} G^{-1}\overline{H}$, $A_{ij} = c_i b_j - a\overline{\beta}^{i^T} G^{-1}\overline{\beta}^j, i = 1, r, j = 1, r$; the solution of Eq. (14) for the Lagrange multipliers and parameter set can be written as:

$$\overline{v} = A^{-1}(a\overline{d} - \overline{c}(1 + \overline{e}^T G^{-1}\overline{H})) \quad (15)$$

$$\mu = -(1 + \overline{v}^T\overline{b} + \overline{e}^T G^{-1}\overline{H})/a \quad (16)$$

$$\overline{t} = -G^{-1}\left(\mu\overline{e} + \sum_{i=1}^{r} v_i\overline{\beta}^i + \overline{H}\right) \quad (17)$$

In a very similar manner, if a constrained solution in the null space of Eq. (6) is desired, the solution for $\overline{t}$ is given by Eq. (17) with the Lagrange parameters given by

$$\overline{v} = A^{-1}(a\overline{d} - \overline{c}(\overline{e}^T G^{-1}\overline{H})) \quad (18)$$

$$\mu = -(\overline{v}^T\overline{b} + \overline{e}^T G^{-1}\overline{H})/a \quad (19)$$

3. Applications to Obstacle and Joint Limits Avoidance

This section presents applications of the framework to two of the most common constraints encountered with manipulators: obstacles and joint limits. In practical applications, these constraints are active on the system only at certain times during a trajectory, in particular, when the configuration of the system approaches one of the limits imposed by the constraints. Thus the constraints and their number will change with time, with the corresponding $\overline{\beta}^j$ vectors in Eq. (13) calculated as non-zero when the system reaches "danger zones" in the vicinity of the absolute limits.

3.1. Obstacle Constraint

Figure 1 depicts the two situations in which the manipulator may come dangerously close to obstacles: (1) with a link or (2) with an elbow. In the figure, the distance d indicates the "danger distance" below which the constraint becomes active. This distance is of course a parameter that is set by the user as a function of the particular manipulator characteristics (e.g., maximum joint velocities and accelerations) or sensing scheme (e.g., sensor location and spacing, sensor sampling rate). The scheme which we use to implement the obstacle avoidance in both the link and elbow cases, consists in moving the point X_j, calculated as the closest point to an obstacle, away from the obstacle using:

$$\Delta\overline{X}_j \bullet \overline{n} = L \quad , \quad (20)$$

where $\Delta\overline{X}_j$ is the position displacement of the point $\overline{X}_j$, $\overline{n}$ represents the normal to the obstacle surface and L is the desired "push away" distance. Other schemes tailored to the system's specific pattern and capabilities could obviously be selected to embody this "push away" concept and that would produce similar effects. Calling J^X the $3\times m$ Jacobian matrix for the position displacement of a point X_j characterizing a given obstacle constraint, Eq. (20) can be written as

$$J^X\Delta\overline{q}(\overline{t}) \bullet \overline{n} = L \quad (21)$$

which, put in the form of Eq. (13), give the expression for the vector $\overline{\beta}$ representing the constraint as

$$\overline{\beta}, \beta_k = \sum_{i=1}^{3}\sum_{j=1}^{m}(J_{ij}^X g_{k_j} n_i)/L \quad (22)$$

where g_{k_j} and n_i represent the components of the vectors $\overline{g}_k$ and $\overline{n}$, respectively.

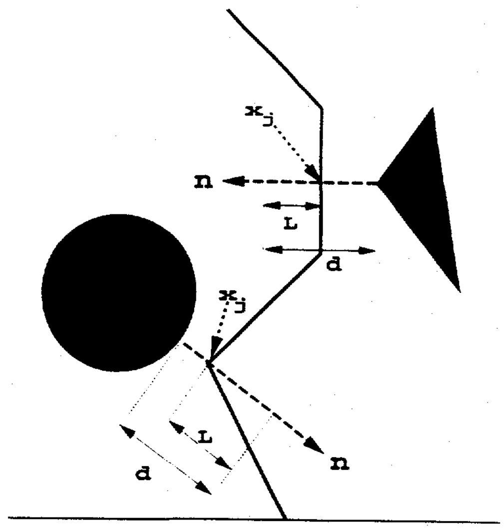

Fig. 1. Schematic of the obstacle avoidance scheme.

3.2. Joint Limit Constraint

In a similar fashion, if any joint, i, of the manipulator is approaching one of its limits $\theta_{i\,\min}$ or $\theta_{i\,\max}$, and requires an angle displacement, d, to return outside of its "danger zone" (angles within a θ_{dang} range of the limit), the constraint can be expressed as:

$$\Delta q_i = \sum_{k=1}^{m-n+1} t_k \overline{g}_{k_i} = d \qquad (23)$$

and the $\overline{\beta}$ vector corresponding to the constraint in the form of Eq. (13) is:

$$\overline{\beta}, \beta_k = \overline{g}_{k_i} / d \qquad (24)$$

4. Sample Results

Implementation and testing of the FSP with obstacle avoidance and joint limit constraints were performed on several manipulator systems. Sample results using a 2-D planar manipulator are first presented here to ease the visualization of the effect of the changing constraints through a display of the detailed step-by-step motion of the manipulator. Results using 3-D manipulators are also shown to illustrate a more realistic implementation.

Figure 2 shows four cases of a 4 d.o.f. planar manipulator controlled in position only (thus, with two degrees of redundancy) while following a semi-circular end-effector trajectory from point A to point B, as indicated in Fig. 2a. A least-norm optimization criterion (i.e., $\overline{H} = \overline{o}$ and B is the identity matrix in Eqs. (10) to (19)) is used in all four trajectories for ease of comparison. In Fig. 2a, no constraints are imposed on the system. In Figs. 2b and 2c obstacles are placed in the way of the manipulator, while in Fig. 2d obstacles and a joint limit constraint on joint 1 have been imposed.

The initial motions of the manipulator are identical in all four plots confirming that the constraints are not active during these early time steps. Only when the manipulator is in the near vicinity of an obstacle or a joint limit do the constraints have any effect on the motion. Figures 2b and 2c depict the obstacle avoidance behavior resulting from the expression of the obstacle constraint using the formalism of Eqs. (20) and (21). Comparison of the final portions of the motions on Figs. 2c and 2d shows the joint limit constraint on joint 1 becoming active and the resulting effect on the manipulator motion.

Figures 3 and 4 show selected frames during the motion of a seven-degree-of-freedom 3-D manipulator controlled in position only (thus with four degrees of redundancy). Figure 3 shows the motion of the manipulator under a least-norm criterion with no obstacle avoidance constraint. An obstacle, depicted by the big sphere in Figs. 3 and 4, has been placed in the path of the manipulator. Intersection of the obstacle and several of the manipulator links and elbow are clearly illustrated in Fig. 3 where the obstacle constraint is kept inactive. Figure 4 shows the obstacle avoidance behavior of the manipulator when the obstacle and joint limit avoidance schemes are turned on. (The motion data log for this case shows that several joint limits, principally at the sperical wrist of the manipulator, are reached and compensated for during this motion.) Here too, the initial portions of the two motions of Figs. 3 and 4 are found identical, confirming that the constraints are active and affect the motion only in the near vicinity of the obstacle.

5. Conclusion

A new method has been presented for resolving the inverse kinematic motion of redundant manipulators with constraints and number of constraints that can vary in real time (sensor sampling rate). The method utilizes the FSP approach to find analytical joint motion solutions that satisfy the task criteria and all constraints active at each time step. A change of criterion and/or number of constraints does not require a major change of algorithms and is only a matter of switching from one analytical solution to another in the code, each solution consisting of only a few explicit statements. Complex motions with widely varying constraints and task criteria can therefore be considered with a single code. Example applications of the constraints formulation have been described for the two most common manipulator constraints, obstacle and joint limit. Sample results using various manipulator test beds have been presented and discussed to illustrate the general algorithm and the real-time sensor-based control applicability of the method.

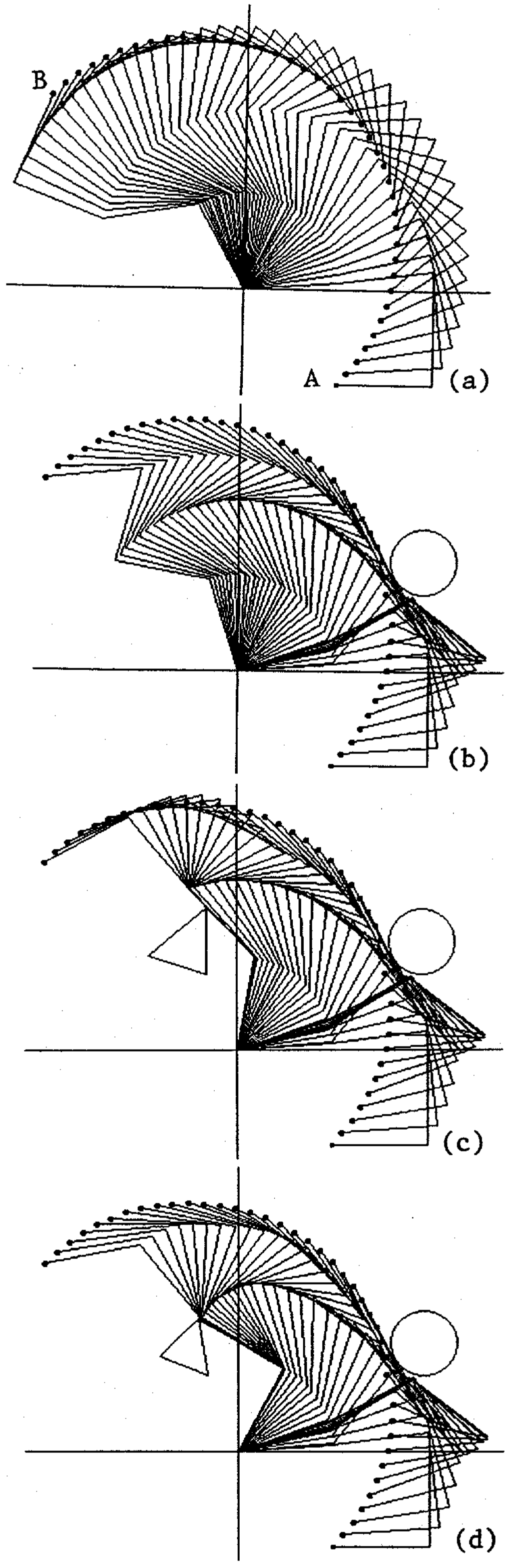

Fig. 2. Sample trajectories of a 4 d.o.f. planar manipulator, a) without constraints, b) and c) with obstacle constraints, d) with obstacle and joint limit constraints.

Acknowledgment

This research was supported in part by the U.S. Air Force Combat Command (ACC), the U.S. Air Force Munition Material Handling Equipment (MMHE) Focal Point, the U.S. Air Force Reliability and Maintainability Technology Insertion Program (PRAM-RAMTIP), the U.S. Department of Defense , Office of the Secretary of Defense (OSD), and the U.S. Advanced Project Research Agency (ARPA), under Interagency Agreement 2146-H055-A1 between the U.S. Air Force Material Command (AFMC) San Antonio Air Logistics Center, Robotics and Automation Center of Excellence (SA/ALC-RACE) and the U.S. Department of Energy, and in part by the DOE-sponsored Science and Engineering Research Program administered by the Oak Ridge Institute for Science and Engineering.

References

[1] A. Liégeois, "Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms," *IEEE Transactions on Systems Man, and Cybernetics* **7**, 868–871, 1977.

[2] A. A. Maciejewski and C. A. Klein, "Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments," *International Journal of Robotics Research* **4**(3), 109–117, 1985.

[3] R. Colbaugh, H. Seraji, and K. Glass, "Obstacle Avoidance for Redundant Robots Using Configuration Control," *Journal of Robotic Systems*, **6**(6), 721–744, 1989.

[4] P. Chiacchio et al., "Closed-Loop Inverse Kinematics Schemes for Constrained Redundant Manipulators with Task Space Augmentation and Task Priority Strategy," *The International Journal of Robotics Research* **10**, 410–425, 1991.

[5] J. Baillieul, "Kinematic Programming Alternatives for Redundant Manipulators," *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*, IEEE Computer Society Press, Silver Spring, 722–728, 1985.

[6] F. G. Pin et al., "A New Solution Method for the Inverse Kinematics Joint Velocity Calculations of Redundant Manipulators," *IEEE Conference on Robotics and Automation*, San Diego, 96–102, 1994.

[7] K. A. Morgansen and F. G. Pin, "Enhanced Code for the Full Space Parameterization Approach to Solving Underspecified Systems of Algebraic Equations, V.1.0," Oak Ridge National Laboratory Technical Report No. ORNL/TM-12816, March 1995.

[8] G. A. Fries, C. J. Hacker, and F. G. Pin, "FSP (Full Space Parameterization), Version 2.0," Oak Ridge National Laboratory Technical Report No. ORNL/TM-13021, 1995.

[9] C. J. Hacker and F. G. Pin, "Inverse Kinematics on Redundant Systems, IKOR Driver (V.1.0 and V.2.0): FSP with Orientation Control, Platform Mobility, and Portability," Oak Ridge National Laboratory Technical Report No. ORNL/TM-13096, 1995

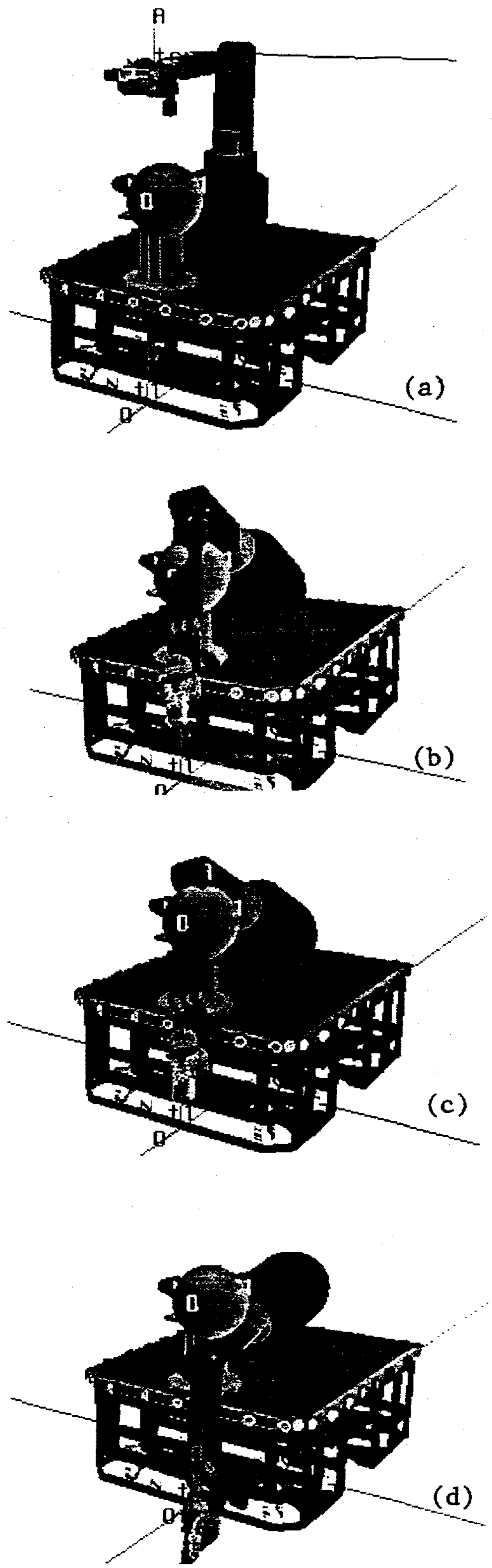

Fig. 3. Sample trajectory of a 7 d.o.f. manipulator with no obstacle constraint.

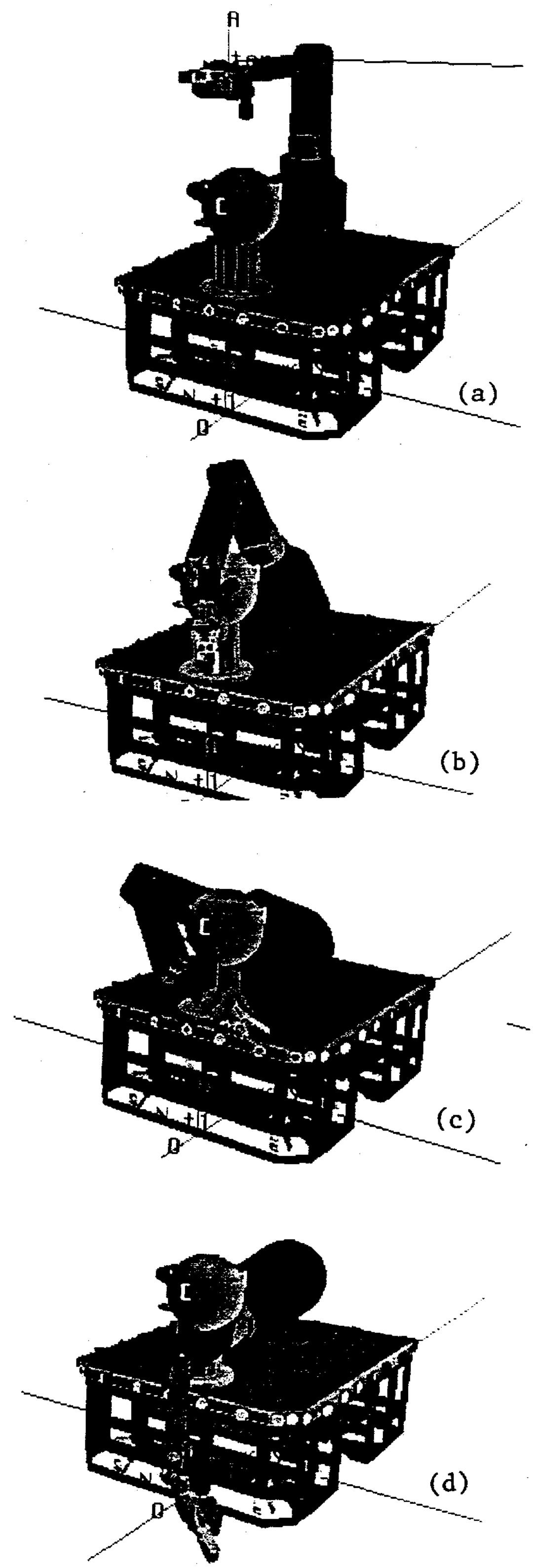

Fig. 4. Same as Fig. 3 with obstacle and joint limit constraints.