Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments

Anthony A. Maciejewski and Charles A. Klein

The International Journal of Robotics Research 1985; 4; 109
DOI: 10.1177/027836498500400308

Anthony A. Maciejewski
Charles A. Klein
Department of Electrical Engineering
The Ohio State University
2015 Neil Avenue
Columbus, Ohio 43210

Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments

Abstract

The vast majority of work to date concerned with obstacle avoidance for manipulators has dealt with task descriptions in the form of pick-and-place movements. The added flexibility in motion control for manipulators possessing redundant degrees of freedom permits the consideration of obstacle avoidance in the context of a specified end-effector trajectory as the task description. Such a task definition is a more accurate model for such tasks as spray painting or arc welding. The approach presented here is to determine the required joint angle rates for the manipulator under the constraints of multiple goals, the primary goal described by the specified end-effector trajectory and secondary goals describing the obstacle avoidance criteria. The decomposition of the solution into a particular and a homogeneous component effectively illustrates the priority of the multiple goals that is exact end-effector control with redundant degrees of freedom maximizing the distance to obstacles. An efficient numerical implementation of the technique permits sufficiently fast cycle times to deal with dynamic environments.

1. Introduction

Kinematically redundant manipulators can be defined as manipulators that possess more than the required six degrees of freedom to arbitrarily position and orient their end-effectors in space. These extra, so-called redundant degrees of freedom result in greater dexterity and flexibility in the specification of motion for the manipulator. This paper presents a formulation that allows the use of these redundant degrees of freedom so that a manipulator can avoid obstacles in the workplace while completing a specified task. Obstacles are defined as any portion of an object with which contact is undesirable.

Clearly, obstacle avoidance is essential for satisfactory task completion. Current manipulator applications typically involve removal of potential obstacles from the manipulator's workspace and the use of fixed motion commands. While preventing manipulator collisions, this method is unduly restrictive because of its inability to deal with unpredictable or dynamic environments. The satisfaction of the obstacle avoidance criteria for such environments would greatly increase the autonomy of the manipulator and result in a wider range of applications that could benefit from the advantages of automation.

This motivation has resulted in a number of various approaches to the resolution of the obstacle avoidance problem (Pieper 1968; Loeff and Soni 1975; Udupa 1977; Khatib and LeMaitre 1980; Lozano-Pérez 1981). A basic premise applied to the development of virtually all of these approaches has been that the task can be defined in terms of a known initial and final configuration for the end-effector. Any set of joint motions that attains the goal configuration without collision is considered satisfactory completion. While this method of task definition is well suited for many applications, particularly of the pick-and-place variety, it

This work is the result of research supported by the National Science Foundation under Grant ECS-8121519 and a National Science Foundation graduate fellowship.

The International Journal of Robotics Research,
Vol. 4, No. 3, Fall 1985,

does not characterize those tasks that require a specified end-effector trajectory throughout the task, such as spray painting or arc welding. Implementation of these obstacle avoidance schemes for such tasks, therefore, is at best awkward. Since most tasks can be described in terms of desired trajectories for a workpiece, the basic task definition used here will be that of a specified end-effector velocity (Paul 1979).

2. Theory

The end-effector velocity, described by the six-dimensional vector $\dot{\mathbf{x}}$, is related to the joint velocities, denoted by the n-dimensional vector $\dot{\theta}$, where n is the number of degrees of freedom, by the equation

$$\dot{\mathbf{x}} = J\dot{\theta}, \tag{1}$$

where J is the Jacobian matrix (Paul 1981). In the case of redundant manipulators, the inverse of the Jacobian is not defined because J is rectangular. Useful solutions to Eq. (1), however, can be obtained through the use of generalized inverses (Boullion and Odell 1971), one of which can be defined as the best approximate solution.

Definition. The vector $\dot{\theta}_0$ is a best approximate solution of Eq. (1) if for all vectors $\dot{\theta}$ either

$$\| J\dot{\theta} - \dot{\mathbf{x}} \| > \| J\dot{\theta}_0 - \dot{\mathbf{x}} \|$$

or

$$\| J\dot{\theta} - \dot{\mathbf{x}} \| = \| J\dot{\theta}_0 - \dot{\mathbf{x}} \| \quad \text{and} \quad \| \dot{\theta} \| > \| \dot{\theta}_0 \|,$$

where $\| \ \|$ denotes the Euclidean norm.

The best approximate solution is given by the pseudoinverse (Penrose 1956) and is denoted here by J^+.

It can be shown (Greville 1959) that the general solution to Eq. (1) is given by

$$\dot{\theta} = J^+ \dot{\mathbf{x}} + (I - J^+J)\mathbf{z}, \tag{2}$$

where I is an $n \times n$ identity matrix and $\mathbf{z}$ is an arbitrary vector in $\dot{\theta}$-space. Thus the resultant joint angle rates can be decomposed into a combination of the least-squares solution of minimum norm (Whitney 1972) plus a homogeneous solution created by the action of a projection operator $(I - J^+J)$, which describes the redundancy of the system, mapping an arbitrary $\dot{\theta}$ into the null space of the transformation. By applying various functions of θ to compute the vector $\mathbf{z}$, reconfiguration of the manipulator can be obtained to achieve some desirable secondary criterion under the constraint of the specified end-effector velocity (Liegeois 1977; Ribble 1982; Klein and Huang 1983; Trevelyan, Kovesi, and Ong 1984).

Recently, Yoshikawa (1984) has implemented an obstacle avoidance scheme in which the vector $\mathbf{z}$ is a specified velocity toward a safe—i.e., collision-free—joint space vector that is predetermined with a priori knowledge of the workspace. The homogeneous solution is thus used to reconfigure the manipulator to be as close as possible to the safe vector while tracking the desired trajectory. In the case of dynamic environments, however, no universal safe joint space vector exists, and the current state of the environment must be used in the motion planning to obtain effective obstacle avoidance.

The obstacle avoidance approach presented here is to identify for each period in time the point on the manipulator that is closest to an obstacle, termed the *obstacle avoidance point,* and assign to it a desired velocity component in a direction that is directly away from the obstacle surface. Thus the primary goal of specified end-effector velocity and the secondary goal of obstacle avoidance (see Fig. 1) are described by the equations

$$J_e \dot{\theta} = \dot{\mathbf{x}}_e \quad \text{and} \tag{3}$$

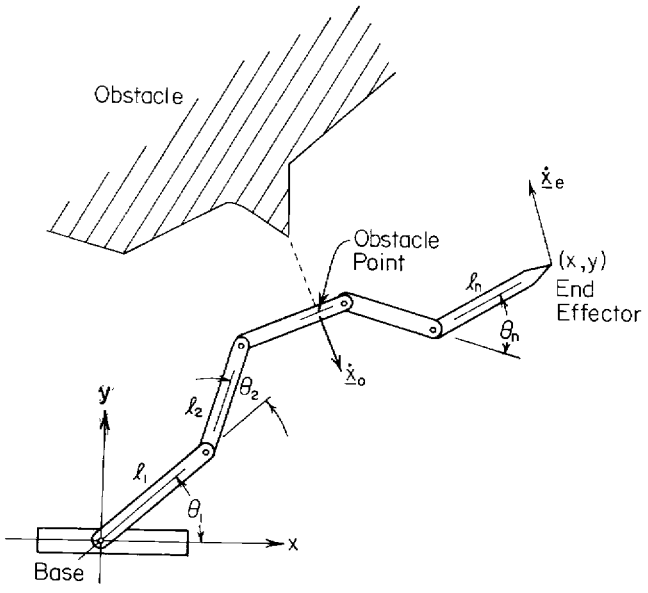
$$J_0 \dot{\theta} = \dot{\mathbf{x}}_0, \tag{4}$$

where

J_e = end-effector Jacobian,
J_0 = obstacle avoidance point Jacobian,
$\dot{\mathbf{x}}_e$ = specified end-effector velocity, and
$\dot{\mathbf{x}}_0$ = specified obstacle avoidance point velocity.

One possible way to find a common solution to both Eqs. (3) and (4) would be to adjoin the two

Fig. 1. Planar manipulator illustrating the primary goal of end-effector velocity specification and the secondary goal of maximizing the distance to the obstacle.



matrices and the right-hand sides into a single matrix equation:

$$\begin{bmatrix} J_e \\ \hline J_0 \end{bmatrix} \dot{\theta} = \begin{bmatrix} \dot{\mathbf{x}}_e \\ \hline \dot{\mathbf{x}}_0 \end{bmatrix}. \tag{5}$$

If the new matrix has the rank n, then mathematically the system has been fully specified and is no longer redundant. If the matrix in Eq. (5) is not square or of full rank, then obviously its pseudoinverse could be applied to give a best approximate solution. However, for this application it is not desirable to treat end-effector and obstacle velocities in the same way. Instead a method is described that first satisfies the primary command of end-effector velocity and then uses the system's redundancy to best match the secondary command of obstacle velocity.

The set of solutions to exactly satisfy the primary goal can be given in the form of Eq. (2). Substituting this solution into Eq. (4) yields

$$J_0 J_e^+ \dot{\mathbf{x}}_e + J_0(I - J_e^+ J_e)\mathbf{z} = \dot{\mathbf{x}}_0, \tag{6}$$

which can now be solved for the desired homogeneous solution. Since the degree of available redundancy may not be sufficient to exactly achieve the secondary goal, a solution that increases the minimum obstacle distance is provided by the pseudoinverse, given by

$$\mathbf{z} = [J_0(I - J_e^+ J_e)]^+(\dot{\mathbf{x}}_0 - J_0 J_e^+ \dot{\mathbf{x}}_e). \tag{7}$$

This equation is identical to the one obtained by Hanafusa, Yoshikawa, and Nakamura (1981) for two goals of different priority. This result is substituted back into Eq. (2) to provide the desired solution to both goals under the constraints imposed by the number of available degrees of freedom, yielding

$$\begin{aligned} \dot{\theta} = {} & J_e^+ \dot{\mathbf{x}}_e \\ & + (I - J_e^+ J_e)[J_0(I - J_e^+ J_e)]^+(\dot{\mathbf{x}}_0 - J_0 J_e^+ \dot{\mathbf{x}}_e). \end{aligned} \tag{8}$$

This solution can be simplified to

$$\dot{\theta} = J_e^+ \dot{\mathbf{x}}_e + [J_0(I - J_e^+ J_e)]^+(\dot{\mathbf{x}}_0 - J_0 J_e^+ \dot{\mathbf{x}}_e) \tag{9}$$

since the projection operator is both hermitian and idempotent. This result is proved in the Appendix.

Each of the terms in the preceding formulation has an easily visualized physical interpretation. As stated previously, the pseudoinverse solution ($J_e^+ \dot{\mathbf{x}}_e$) of the desired end-effector trajectory composes the particular portion of the solution that in the redundant case guarantees the exact desired end-effector velocity with the minimum joint velocity norm. The added homogeneous solution sacrifices the minimum norm solution to satisfy a different goal, that of obstacle avoidance. The matrix composed of the obstacle Jacobian times the projection operator, $J_0(I - J_e^+ J_e)$, represents the degrees of freedom available to move the obstacle point while creating no motion at the end-effector. This matrix is used to transform the desired obstacle point motion from Cartesian obstacle velocity space into the best available solution in joint velocity space again through the use of the pseudoinverse. The vector describing the desired obstacle point motion is composed of the commanded motion, $\dot{\mathbf{x}}_0$, obtained from environmental information, modified by subtracting the motion caused at the obstacle point due to satisfaction of the end-effector velocity constraint ($J_0 J_e^+ \dot{\mathbf{x}}_e$).

3. Implementation

The preceding obstacle avoidance scheme has been implemented in both two-dimensional and fully gen-

eral three-dimensional computer graphics simulations on a PDP 11/70 computer provided with a Hewlett-Packard 1350 vector graphics display. The primary goal, a specified end-effector trajectory, is entered either through a file or interactively with a joystick. The secondary goal information, obstacle avoidance point and velocity, is determined from a mathematical description of the world. In a physical implementation, the secondary goal information would be obtained from sensory range-finding devices (Espiau and Andre 1984). Details of the implementation can be obtained from Maciejewski (1984).

In implementing Eq. (9) several types of decisions must be made, one of which involves the rank of the matrices for which the pseudoinverse is computed. One of the aesthetic attractions of pseudoinverse formulations is that they are valid independently of the rank of the matrix. If a matrix has full rank, then the pseudoinverse yields finite values as weights for the input singular vectors. If a matrix becomes singular, then the pseudoinverse yields a zero coefficient for the direction of the input singular vector in the null space. A simple example demonstrates that these two cases cannot blend smoothly into each other. The matrix $\mathbf{A}$ is its own pseudoinverse:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} = \mathbf{A}^{+}. \tag{10}$$

However, the matrix $\mathbf{B}$ has a pseudoinverse that diverges from $\mathbf{A}^{+}$:

$$\mathbf{B} = \begin{bmatrix} 1 & 0 \\ 0 & \epsilon \end{bmatrix}, \qquad \mathbf{B}^{+} = \begin{bmatrix} 1 & 0 \\ 0 & 1/\epsilon \end{bmatrix} \tag{11}$$

as the parameter ϵ makes $\mathbf{B}$ approach $\mathbf{A}$. Therefore, the main problem in a pseudoinverse formulation is not at a singularity when the pseudoinverse formulation assigns a zero component in the missing degree of freedom but near a singularity where a very large component may be applied. The solution involves evaluating the effective rank of $\mathbf{B}$ and treating it as singular whenever it would yield unacceptably larger answers (Noble 1975). Since this is used in a real-time controller for a physical system, limits must be based on physical speed limitations rather than loss of significance as would apply in a numerical analysis situation. However, independent of the value of the threshold of rank, there will be a discontinuity when the change of rank is noted.

If assumptions regarding the rank of a matrix can be made, then the computation of the pseudoinverse can be simplified. Unless the specified task is unachievable, the end-effector Jacobian is of full rank, and its pseudoinverse need not be explicitly calculated. The approach used here is to solve, by using Gaussian elimination, for β, a six-element vector, and γ, a $6 \times n$ matrix, in the equation

$$[J_e J_e^*][\beta|\gamma] = [\dot{\mathbf{x}}_e|J_e], \tag{12}$$

where * denotes the complex conjugate transpose. It is then easily shown (Klein and Huang 1983) that the desired quantities $J_e^{+}\dot{\mathbf{x}}_e$ and $J_e^{+}J_e$ are obtained from

$$J_e^{+}\dot{\mathbf{x}}_e = J_e^{*}\beta, \tag{13}$$

$$J_e^{+}J_e = J_e^{*}\gamma. \tag{14}$$

The elements of the Jacobians are efficiently computed by using the screw axis formulation (Waldron 1982).

On the other hand, the $3 \times n$ matrix $[J_0(I - J_e^{+}J_e)]$ cannot always be assumed to be of full rank, and therefore the Gaussian elimination approach used above is not applicable. In this case the pseudoinverse is explicitly calculated using the recursive method presented by Greville (1960). Within this algorithm one implicitly makes decisions regarding the rank in terms of evaluating the linear independence of matrix columns. Therefore, a sufficiently large threshold must be applied to limit high velocities near singularities. Although theoretically this technique works as well on a matrix as on its transpose, numerically it is very important to transpose this matrix so that Greville's algorithm is applied to an $n \times 3$-order matrix ($n > 3$). Since this algorithm operates on columns, the correct decision for rank is much more easily applied to 3, rather than n, columns.

Proper decision making regarding rank is especially important since the system tends to move such that either the obstacle point cannot move any farther without changing the end-effector or the system finds a new point to become the obstacle point. Besides

Fig. 2. The form of the homogeneous term gain, α_η, and the obstacle avoidance gain, α_0, as a function of the obstacle distance.

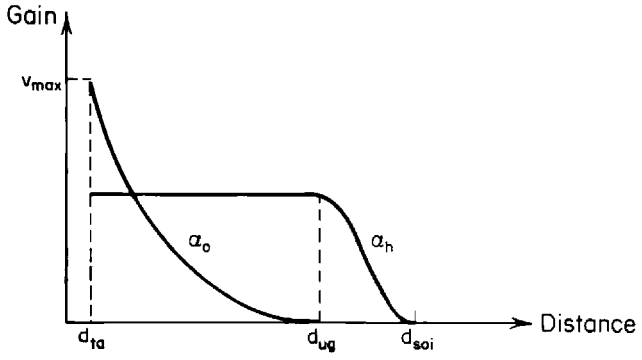
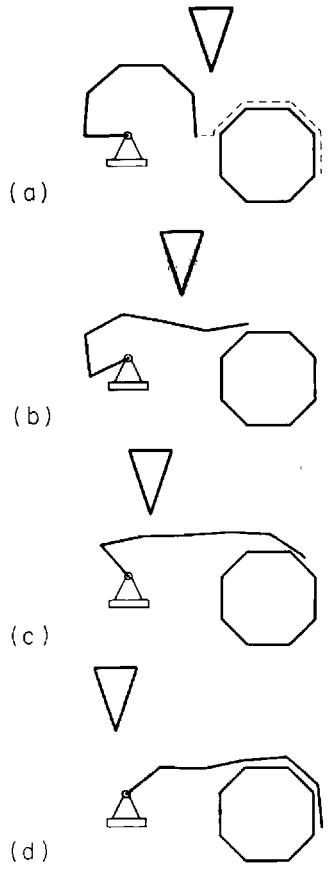


Fig. 3. Simulation results illustrating a redundant manipulator following a specified trajectory (shown in A) while avoiding a moving triangular obstacle.



choosing the proper threshold, an additional method can be applied to minimize rank change effects. This is to consider obstacle avoidance only when sufficiently close to an obstacle, which would be compatible with limited-range proximity sensors. In order not to introduce a discontinuity, the obstacle avoidance term should be tapered as a function of distance.

To effectively control the degree of influence that an obstacle has on the resultant manipulator motion, the solution for the joint angle rates was modified to

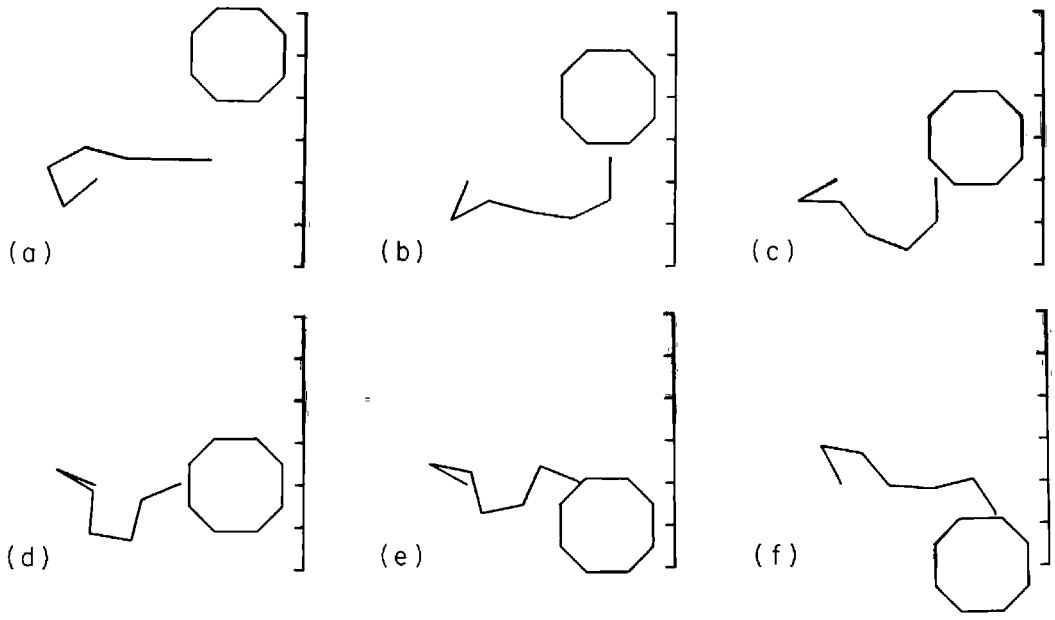
$$\dot{\theta} = J_e^+ \dot{\mathbf{x}}_e + \alpha_\eta [J_0 (I - J_e^+ J_e)]^+ (\alpha_0 \dot{\mathbf{x}}_0 - J_0 J_e^+ \dot{\mathbf{x}}_e), \quad (15)$$

where d is the distance between the obstacle surface and the obstacle avoidance point, $\dot{\mathbf{x}}_0$ is now considered a unit vector indicating direction, $\alpha_0(d)$ is the magnitude of the secondary goal velocity, and $\alpha_\eta(d)$ is a gain term for the amount of the homogeneous solution to be included in the total solution. The functions α_0 and α_η are described by polynomials of the form shown in Fig. 2.

From Fig. 2 it can be seen that there are three distances that characterize changes in the value of the gain functions. These distances are defined as the task abort distance, d_{ta}, the unity gain distance, d_{ug}, and the sphere of influence distance d_{soi}. These distances define three zones that encompass each obstacle. If the manipulator is further from the obstacle than the d_{soi}, then the obstacle has no influence on the manipulator and the homogeneous solution can be used to satisfy some other desirable criterion. Between the d_{soi} and the d_{ug} there is a smooth transition where avoidance of the obstacle is included in the motion specification of the manipulator. Inside of the d_{ug}, the influence of the obstacle in repelling the manipulator is inversely related to the distance. If the manipulator reaches the d_{ta}, then a collision is considered imminent and the specified task is suspended. This distance is determined by such factors as the maximum velocity and physical dimension of the manipulator, thus allowing the manipulator to slow down and stop before physical contact occurs.

Since use of a single worst-case obstacle point for motion planning may result in oscillations for some configurations or environments, the use of multiple secondary goals has been considered as a way of minimizing switching between homogeneous solutions. In this implementation the two worst-case obstacles were considered by modifying the solution for the joint

Fig. 4. Simulation results of the obstacle avoidance scheme for redundant manipulators applied to environments with workpieces on a moving conveyor.



angles to

$$\dot{\theta} = J_e^{+}\dot{\mathbf{x}}_e + \alpha_1(d_2/d_1)\mathbf{h}_1 + \alpha_2(d_2/d_1)\mathbf{h}_2, \qquad (16)$$

where $\mathbf{h}_i$ is the ith homogeneous solution, α_i is its corresponding gain, and d_i is the distance to the obstacle where the subscript 1 denotes the worst-case obstacle. The greater the disparity between d_1 and d_2, the closer α_1 comes to unity, with α_2 approaching zero. With d_1 approximately equal to d_2, $\alpha_1 = \alpha_2 = 0.5$ with the overall homogeneous solution split between the two goals. When we blend the homogeneous solutions in this manner, the manipulator has the capability to make smooth transitions to allow for the varying priorities of the secondary goals.

In implementation, certain tradeoffs in performance must be recognized. Desirable, but contradictory, qualities are smoothness of motion, responsiveness, and automatic operation without planning. For example, it is possible to formulate tasks that force the obstacle point arbitrarily close to the obstacle even after the configuration has been fully optimized. In this situation arbitrary end-effector velocities can no longer be permitted, and the primary goal must be modified. By including more obstacle points, motion is smoother but the most critical obstacle distance may not be as large. If specific knowledge about the manipulator configuration is used, the algorithm can run faster but at the expense of generality.

4. Results

Most of the evaluation of the obstacle avoidance formulation was performed using the two-dimensional simulation due to the greater degree of redundancy available with respect to the task and the corresponding reduction in computation time that allows for real-time joystick specification of the end-effector trajectory. The cycle time for a seven-degree-of-freedom, two-dimensional manipulator, including secondary goal calculation and graphic display time, was 47 ms. Figure 3 depicts a typical simulation run illustrating the effectiveness of the formulation. The manipulator is commanded along a specified end-effector trajectory that is required to complete the task. The maximum velocity is 20% of the total manipulator length per second. The moving triangular obstacle is avoided without deviation from the assigned end-effector trajectory. The secondary goal terms generated from environmental information serve to reorient the ma-

Fig. 5. Simulation results for a redundant manipulator working in a cavity with inboard links avoiding the throat.

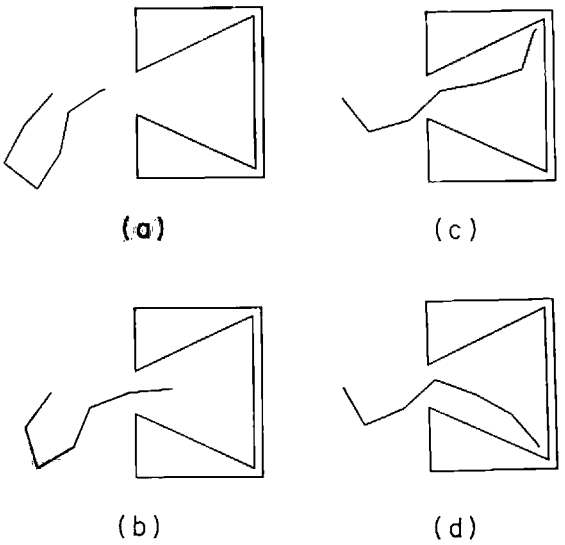
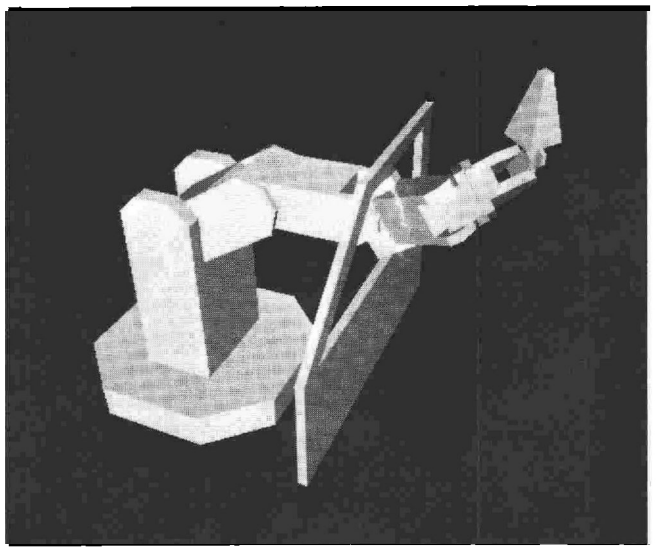


Fig. 6. A single frame from a simulation of a 3-D redundant manipulator operating through the window of an automobile door.



nipulator so as to conform to the obstacle surface, thereby retaining maximum use of the redundant degrees of freedom in avoiding the obstacle. Note that the workpiece itself is seen as an obstacle by the inboard links. From Fig. 4 it can be seen that the formulation works equally well for a moving workpiece as for an assembly line employing conveyors. Results obtained from a somewhat more imposing obstacle (Fig. 5) illustrate the effectiveness of the multiple secondary goal formulation. The close quarters present with the end-effector investigating the interior of the obstacle result in multiple worst-case goals. The appropriate blending of homogeneous solutions, however, keeps the manipulator close to the center of the opening, maintaining an optimum orientation without oscillations.

The three-dimensional simulation was evaluated for a nine-degree-of-freedom manipulator design proposed by Waldron (1980). A single frame from a simulation in which the manipulator was commanded to spray-paint a car door illustrates the action of the homogeneous solution in preventing contact of inboard links with the car door while the end-effector follows the required trajectory (Fig. 6). The computation cycle time for the simulation, excluding graphics and secondary goal calculation, was 102 ms.

While planned for dynamically varying and unpredictable environments, this technique also has applications in repetitive situations. By using this formulation in the simulation of a task using a CAD/CAM data base for obstacle descriptions, optimal joint configurations can be saved and reused in the homogeneous term (Yoshikawa 1984). In this way the advantages of the technique presented here can be realized at an even faster cycle time. This would be particularly advantageous where moving conveyors are employed and a single joint state would not be sufficient.

5. Conclusion

Past uses for the homogeneous solution of redundant manipulators have described the secondary goals in terms of the joint space variables. The formulation presented here illustrates an extension to the uses of the homogeneous solution to include criteria that are better described in Cartesian world space coordinates. Obstacle avoidance is clearly one such criterion.

The real-time demands of obstacle avoidance in practical applications, particularly in dynamic environments, demands a computationally efficient algorithm. This formulation, with a cycle time of 102 ms on a general-purpose computer, clearly demonstrates the potential for a viable system in a physical realization.

Appendix: Proof That $B[CB]^+ = [CB]^+$ for the Hermitian and Idempotent Matrix B

For an $m \times n$ matrix **A**, the pseudoinverse, $\mathbf{A}^+$, is defined by the following four properties

$$AA^+A = A, \qquad A^+AA^+ = A^+,$$
$$(A^+A)^* = A^+A, \qquad (AA^+)^* = AA^+,$$

where the operation * denotes the complex conjugate transpose. Given an $m \times n$ matrix **C** and an $n \times n$ idempotent and hermitian matrix **B**, to show that $B[CB]^+ = [CB]^+$, one must show that $B[CB]^+$ satisfies the above conditions that define the pseudoinverse. Let $A = CB$ and $G = B[CB]^+$. Evaluation of the expression AGA results in

$$\begin{aligned} AGA &= CBB[CB]^+ CB \\ &= CB[CB]^+ CB \quad (B \text{ idempotent}) \\ &= CB \quad (\text{by definition}) \\ &= A. \end{aligned}$$

Evaluation of the expression GAG results in

$$\begin{aligned} GAG &= B[CB]^+ CBB[CB]^+ \\ &= B([CB]^+ CB[CB]^+) \quad (B \text{ idempotent}) \\ &= B[CB]^+ \quad (\text{by definition}) \\ &= G. \end{aligned}$$

Evaluation of the expression $(GA)^*$ results in

$$\begin{aligned} (GA)^* &= (B[CB]^+ CB)^* \\ &= ([CB]^+ CB)^* B^* \\ &= [CB]^+ CBB^* \quad (\text{by definition}) \\ &= [CB]^+ CBB \quad (B \text{ hermitian}) \\ &= [CB]^+ CB \quad (B \text{ idempotent}) \\ &= ([CB]^+ CB)^* \quad (\text{by definition}) \\ &= B^*C^* [CB]^{+*} \\ &= BC^* [CB]^{+*} \quad (B \text{ hermitian}) \\ &= BBC^* [CB]^{+*} \quad (B \text{ idempotent}) \\ &= BB^*C^* [CB]^{+*} \quad (B \text{ hermitian}) \\ &= B([CB]^+ CB)^* \\ &= B[CB]^+CB \quad (\text{by definition}) \\ &= GA. \end{aligned}$$

Evaluation of the expression $(AG)^*$ results in

$$\begin{aligned} (AG)^* &= (CB\, B[CB]^+)^* \\ &= (CB\, [CB]^+)^* \quad (B \text{ idempotent}) \\ &= CB\, [CB]^+ \quad (\text{by definition}) \\ &= CBB[CB]^+ \quad (B \text{ idempotent}) \\ &= AG. \end{aligned}$$

The above four results show that $G = A^+$. Since the pseudoinverse of a matrix is unique (Penrose 1956), $B[CB]^+$ must equal $[CB]^+$.

References

Boullion, T. L., and Odell, P. L. 1971. *Generalized inverse matrices.* New York: Wiley-Interscience.

Espiau, B., and Andre, A. 1984 (June 26–29). Sensory-based control for robots and teleoperators. Preprint. *5th Romansy Symp.*, Warsaw: Polish Scientific Publishers.

Greville, T. N. E. 1959. The pseudoinverse of a rectangular or singular matrix and its applications to the solutions of systems of linear equations. *SIAM Review* 1(1):38–43.

Greville, T. N. E. 1960. Some applications of the pseudoinverse of a matrix. *SIAM Review* 2(1):15–22.

Hanafusa, H., Yoshikawa, T., and Nakamura, Y. 1981 (Aug. 24–28, Kyoto). Analysis and control of articulated robot arms with redundancy. Preprints IFAC 8th Triennial World Congress, vol. 14, pp. 78–83.

Khatib, O., and LeMaitre, J. F. 1980 (Sept. 12–15, 1978, Udine, Italy). Dynamic control of manipulators operating in a complex environment. *3rd Symp. Theory Practice Robots Manipulators.* New York: Elsevier, pp. 267–282.

Klein, C. A., and Huang, C. H. 1983. Review of pseudoinverse control for use with kinematically redundant manipulators. *IEEE Trans. Sys. Man, Cyber.* SMC-13(2):245–250.

Liegeois, A. 1977. Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Trans. Sys., Man, Cyber.* SMC-7(12):868–871.

Loeff, L. A., and Soni, A. H. 1975. An algorithm for computer guidance of a manipulator in between obstacles. *J. Eng. for Industry.* Series B, 97(3):836–842.

Lozano-Pérez, T. 1981. Automatic planning of manipulator transfer movements. *IEEE Trans. Sys. Man, Cyber.* SMC-11(10):681–698.

Maciejewski, A. A. 1984. Obstacle avoidance for kinematically redundant manipulators. M.S. thesis, The Ohio State University, Department of Electrical Engineering.