Path Planning for Robot Manipulators in Polyhedral Objects Environment

Ching-Shiow Tseng* and Tsun-Sun Lue
Department of Mechanical Engineering
National Central University
Chungli, Taiwan

Received January 18, 1994; revised January 25, 1995; accepted April 12, 1995

A recent development in robotics is the increase of intelligence in robots. One of the research fields is to enable robots to autonomously avoid collisions with surrounding objects. This article presents an efficient method for planning collision-free paths for an articulated robot that is surrounded by polyhedral objects. The algorithm plans a hypothetical Archimedes's spiral path from the initial position to the goal position. When a collision among the arms and obstacles is detected, the hypothetical path will be modified to avoid the collision. The algorithm applies geometric methods to determine the upper and lower bounds of the reachable area of the wrist and then determines a collision-free path point on that reachable area. Because the equations, which represent the upper and lower bounds, are simple, the algorithm can rapidly determine a collision-free path. Moreover, with minor modifications, this path planning algorithm can also be applied to other robots such as spherical, cylindrical, and Cartesian types of robots. © *1995 John Wiley & Sons, Inc.*

ロボット学における最近の進歩は、ロボット知能の増大である。この研究分野の課題の1つは、ロボットと周辺物体の衝突を自動的に回避できるようにすることである。この発表では、多面体的物体に囲まれた連結ロボット用の無衝突経路計画を効果的に実行する方法を説明する。アルゴリズムは、始点から終点までの仮のアルキメデス・スパイラル経路を計画する。アームと障害物の衝突が検出されると、仮の経路が修正されて衝突を回避する。幾何学的方法にアルゴリズムを適用して、リストが到達できる範囲の上限と下限を特定し、またリスト到達範囲上の無衝突経路ポイントを決定する。上限と下限を表す方程式は単純なので、アルゴリズムは即座に無衝突経路を決定できる。さらに、若干の変更によって、この経路計画アルゴリズムは、球形、円筒形、カルテシアン型のロボットにも応用できる。

*To whom all correspondence should be addressed.

Journal of Robotic Systems 12(10), 637–646 (1995)
© 1995 by John Wiley & Sons, Inc.
CCC 0741-2223/95/0100637-10

1. INTRODUCTION

Most industrial robots currently available in the market have little capabilitiy of thinking and decision-making. Operators have to teach or program robots so that they can operate without any collision or damage. Inefficient teaching or programming restricts the application fields of robots. Robots should have the capability to autonomously decide their collision-free motion so that they can operate in various environments such as houses, offices, nuclear plants, and even space. This article discusses an algorithm for robot manipulators to autonomously generate collision-free paths.

Since Udupa[1] presented the collision detection and avoidance problems of manipulators in 1977, many researchers have been involved in exploring rapid and efficient generation of collision-free paths. The methods developed include geometric methods,[2-6] control algorithms,[7] artificial intelligent search methods,[8-11] and sensor-based motion planning.[12] Most of the geometric methods found the free space from the robot's configuration space.[1] The search for collision-free paths in configuration space may be global,[2,6] local,[5,13,14] or both.[4] The global method may explore an optimum collision-free path[15] if a completely known environment is available. The algorithm presented in this article applies the principle of plane geometry to detect collisions and to determine collision-free path points. It approaches the 3-D collision-free path planning problem by searching for reachable path points on 2-D cross-sections of the wrist's workspace. It is a local method made global.

In this article, the surrounding polyhedral objects are assumed to have vertical sides and horizontal or inclined top/bottom surfaces. Actually, many objects can be divided and considered as a composition of several polyhedral objects with vertical sides. The objects are hung under the ceiling or stand on the floor as shown in Figure 1. To reduce the computing time for planning collision-free paths, the upper and lower arms of the robot are replaced by line segments with the same length as those arms. Contrarily, the objects are expanded according to the sizes of the cross-sections of the upper and lower arms. Although the approach method for planning collision-free paths is different, the object expansion algorithm is somewhat similar to Lozano-Pérez's work.[16]

The expanded objects are checked to see if they are within the workspace of the robot. If the objects are within the workspace of the robot, they are obstacles. Otherwise, they are not obstacles. In process

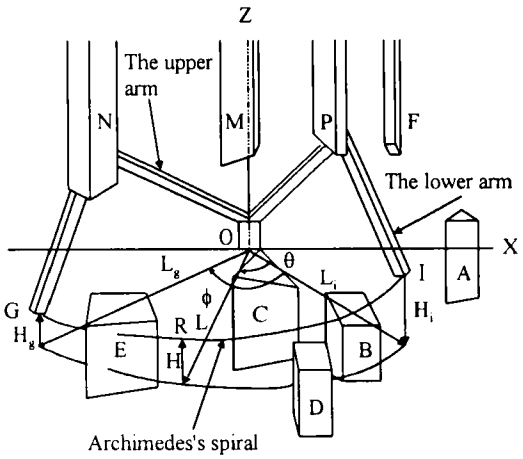


Figure 1. A robot with surrounding objects.

of the path planning, only obstacles are under consideration. Each corner of the top (bottom) surfaces of the standing (or hanging) obstacles and the Z-axis of the robot's fixed frame determine a vertical plane that may cut through the obstacles and will generate a cross-section of the workspace of the wrist. The number of cross-sections of the wrist's workspace generated is thus equal to that of the corners of all the obstacles. On each of the cross-sections, a reachable path point will be determined. This reachable path point is used to determine a reachable end effector point. Finally, the spiral path, which sequentially connects the initial point, the reachable end effector points, and the goal point, will be modified to generate a collision-free path.

A real-scale articulated robot is applied to describe the algorithm. It is an ABB IRB-2000 industrial robot. Because the angular displacement of joint 3 is limited between 30 degrees and 150 degrees, this robot has only one configuration corresponding to a given position and orientation of the end effector.

2. EXPANSION OF THE OBJECTS

Generally, the shapes of the cross-sections of the robot's arms are approximately circular or rectangular. If the arms are substituted by cylinders or rectangular solids, their cross-sections will be circles or rectangles. This substitution will simplify the collision detection problem between the arms and obstacles. Figure 2 shows an example in which the lower arm of the ABB robot is substituted by a rectangular solid.

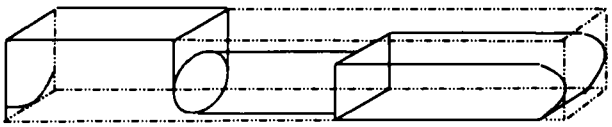


Figure 2. The substitution of the lower arm by a rectangular solid.

To represent the arms of the robot by line segments, all surrounding objects are expanded by suitable scales. The expansion scales should be large enough to compensate for the sizes of the cross-sections of the arms. Because the cross section of the upper arm is larger than that of the lower arm, the expansion rule is defined as below.

> When any side of the objects can be reached by the upper arm and/or lower arm, it will be expanded outward. If the side can be reached by the upper arm (or only by the lower arm), the expansion distance is equal to the longest distance measured from the axial point of the cross section of the upper arm (or lower arm) to its periphery.

The above expansion rule is similar to Newman's work.[17,18] Figure 3 shows an application example of the above expansion rule. The circle "*o*" with the radius *N* is the cross-section of the lower arm, and the rectangle "*p*" with the longest diagonal distance *M* is the cross-section of the upper arm. The polygon *a*-*b*-*c*-*d* is the top view of an arbitrary object A. Suppose that the sides *a*-*b* and *b*-*c* can be reached by the upper arm while the other two sides can be reached only by the lower arm. After having applied the above expansion rule to the object A, the top view of the expanded object A will be the polygon, *a*1-*a*2-*b*1-*b*2-*c*1-*c*2-*d*1-*d*2. The new corners such as *b*1 and *b*2 are determined by following the steps below.

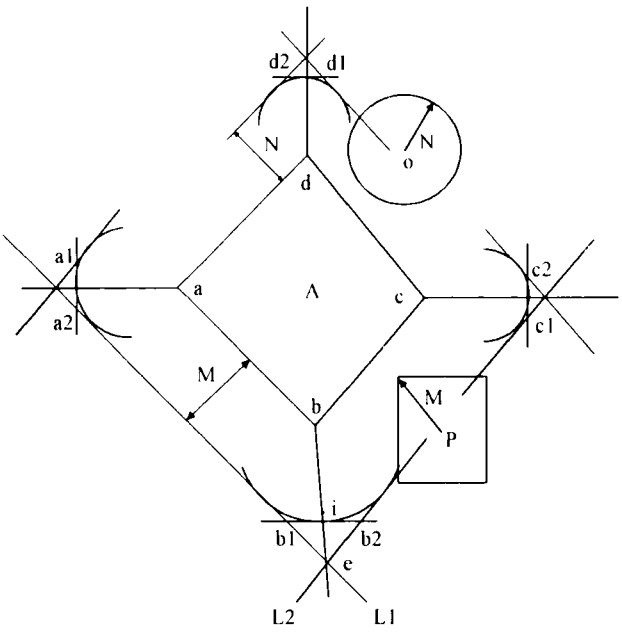


Figure 3. The object A before and after expansion.

1. Sides *a*-*b* and *b*-*c* are shifted out into lines *L*1 and *L*2, respectively. Point "*e*" is their intersection.
2. Point "*i*" is on the line segment *b*-*e*, and the length of line segment *b*-*i* equals *M*.
3. The line that passes through the point "*i*" and is perpendicular to the line b-e intersects with the lines *L*1 and *L*2 at points *b*1 and *b*2, respectively.
4. The coordinates of the points *b*1 and *b*2 can be calculated by applying plane geometry to the first three steps.

If two adjacent sides can be reached by different arms, both sides will be shifted out by the distance of max (*N*, *M*). Their new corners can be determined by following the same procedure as the above four steps. Figure 3 shows an example where sides *b*-*c* and *c*-*d* can be reached by the upper and lower arms, respectively. The new corners *c*1 and *c*2 are determined by expanding both sides by a distance equal to *M*.

If two adjacent sides form a concave shape, they will have only one new corner. Figure 4 shows an example where sides *a*-*b* and *b*-*c* form a concave shape. Their new corner "b'" is the intersection of the expanded sides.

The expanded object A of Figure 3 can be used to check if the circle "*o*" or the rectangle "*p*" touches the original object A. The circle will not touch the original object A if the center "*o*" of the circle is outside the expanded object A. Similarly, the rectangle will not touch the original object A if the axial point "*p*" of the rectangle is outside the expanded object A. Therefore, the collision problem between the object A and the circle "*o*" (or rectangle "*p*") is equivalent to that between the expanded object A and the point "*o*" (or point "*p*").

The above expansion rule will be applied to all objects. After all objects have been expanded, their new top/bottom surfaces (top surfaces are for standing objects while bottom surfaces are for hanging

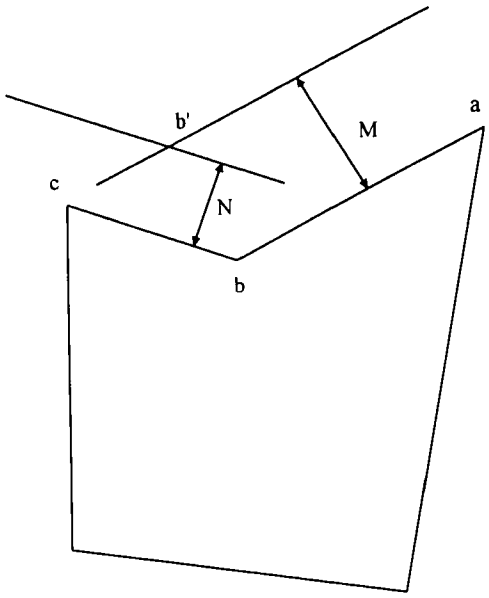


Figure 4. The expansion of concave sides.

objects) are shifted upward/downward a distance. This distance is equal to $M/\cos\phi$ if the top/bottom surface can be reached by the upper arm, where ϕ is the inclined angle of the top/bottom surface. Otherwise, the distance is equal to $N/\cos\phi$. This shift is to avoid the arms touching the top/bottom surface.

Note: The sides of an object can be reached by the upper arm if their orthogonal projections (line segments) on the floor intersect with the circle whose center is the origin of the fixed frame and whose radius equals the length of the upper arm.

The above expansion process will overestimate the unreachable volume around the obstacles. Thus, some possible maneuvers are removed from the potential solution set. However, the overestimated volume is small, and the influence in path planning is insignificant.

3. IDENTIFICATION OF THE OBSTACLES

Generally, robots are planned to move along linear paths. It is simple to define a linear path and to determine its path points in Cartesian space. However, a linear path for the articulated, cylindrical, and spherical types of robots may pass through the void: the unreachable space whose center is at the center of joint 2. The robot cannot move along a path that passes through the void. To avert this problem, a smooth Archimedes's spiral, instead of a linear path, is selected as the reference path for determining a collision-free path. For general types of robots, an Archimedes's spiral path will always stay within the robot's workspace if its two end points are inside the robot's workspace. Figure 1 ilustrates an Archimedes's spiral path that is determined by the initial position I and the goal position G. The equation of this Archimedes's spiral path is defined as

$$L = L_i + (L_g - L_i)\theta/\phi$$
$$H = H_i + (H_g - H_i)\theta/\phi \qquad (1)$$

where

L_i, L_g, and L are the horizontal distances from the Z-axis to the initial point I, goal point G, and an arbitrary point R on the spiral respectively.
H_i, H_g, and H are the vertical heights of the points I, G, and R, respectively.
θ is the angle between the planes R-O-Z and I-O-Z.
ϕ is the angle between the planes G-O-Z and I-O-Z.

Because path planning is related to the identification of obstacles, it is necessary to describe how a collision-free path is planned prior to identifying obstacles. As described above, the Archimedes's spiral path determined by the given initial and goal positions is the reference path. The wrist point of the robot is commanded to move along the spiral path if it is collision-free. However, the spiral path will be modified if a collision between the arms and objects is detected at any path point on the spiral path. The procedure to modify the spiral path is to find a new collision-free path point straight upward or downward from the current path point. If such a path point cannot be found, a search for a new collision-free path point from the current path point toward the Z-axis of the fixed frame will be proceeded. If no collision-free path point can be found, it is considered that the robot cannot move to the goal position without colliding with the objects.

Suppose that a new collision-free path point is found. Then, the coordinates of this path point and the goal position will be used to generate a new Archimedes's spiral path between them. The wrist point of the robot will follow this new spiral path and move toward the goal position.

The above procedure will be applied whenever a collision is detected. Finally, a collision-free path from the initial position to the goal position will be generated. This path planning method restricts the arms to move within the triangular prim space determined by the planes IOZ and GOZ, and the vertical

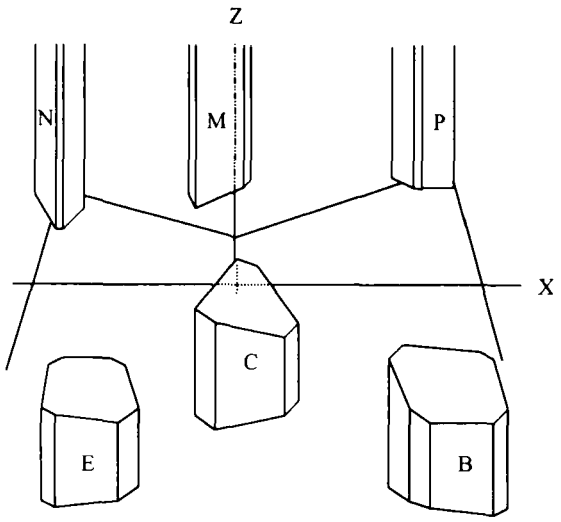


Figure 5. The environment simplified from the case of figure 1.

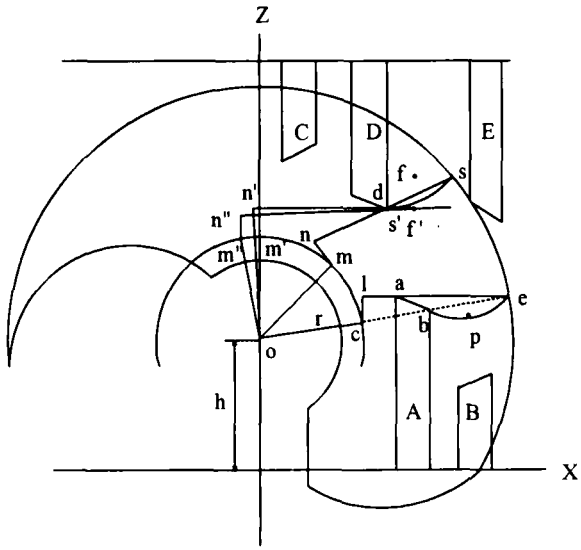


Figure 6. The cross sections of the wrist's workspace and obstacle.

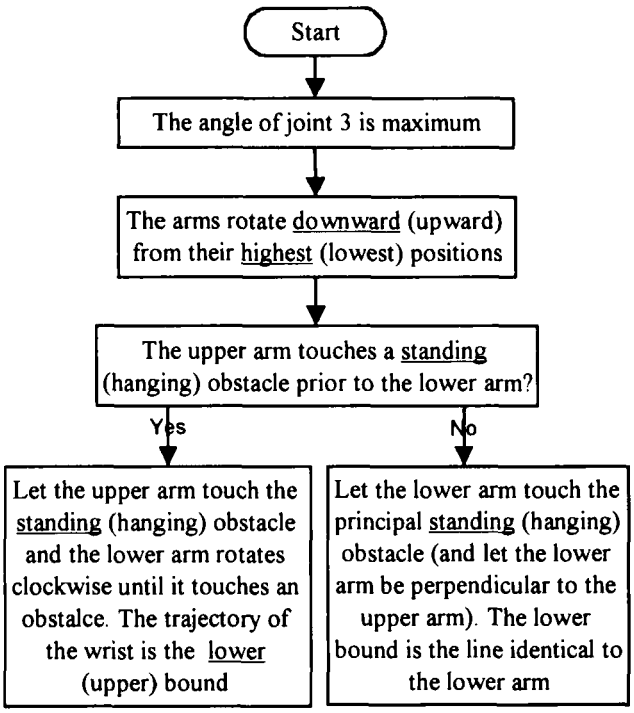
and curved plane passing through the Archimedes's spiral *I-R-G*. Therefore, the expanded objects outside this triangular prim space are not obstacles. Because there is no difficulty in deriving the equations of the three boundary planes, it is easy to identify all obstacles.

In Figure 1, the robot is surrounded by nine objects. Only six of them are obstacles. The objects F, A, and D are not obstacles and can be removed. Figure 5 shows the simplified robot and expanded obstacles from the case of Figure 1.

4. DETERMINATION OF THE REACHABLE PATH POINTS

Each corner of the top/bottom surfaces of the obstacles and the Z-axis determine a vertical plane. This plane will cut through the wrist's workspace and one or more obstacles, and will generate their cross sections. Figure 6 illustrates an example: Polygons A, B, C, D, and E are cross-sections of the obstacles; and the moon-shaped close curve is the cross-section of the workspace of the wrist.

Because of the existence of obstacles (or the cross-sections of obstacles), the wrist of the robot can only reach part of the cross-section of the wrist's workspace without the occurrence of a collision. This reachable area is the wrist's reachable area by name. The reachable area of the wrist has upper and lower bounds, which are determined by the locations, sizes, and shapes of the obstacles. Figure 7 shows



Note: The words in the parentheses are used to replace the underlined words in front of them. If no underlined words, they are additional words .

Figure 7. The determination of the upper and lower bounds.

a flow chart for determining the upper and lower bounds of the reachable area of the wrist.

Here, a reference path point is defined as the intersection of the Archimedes's spiral path and the reachable area of the wrist. It is a path point on the spiral path. As described previously, a search for reachable path points will proceed if the reference path point is unreachable or cannot be reached without the occurrence of a collision. The search direction is straight upward/downward and/or toward the fixed Z-axis from the reference path point. Therefore, it is not necessary to consider obstacles that are farther than the reference path point, measured from the fixed Z-axis. In other words, only those obstacles that are located at the area between the fixed Z-axis and the line $x = x_f$ are used to determine the upper and lower bounds of the reachable area of the wrist, where x_f is the x-coordinate of the reference path point "f." Figure 6 shows that the objects B and E are farther away than the reference path point "f." Therefore, they are not used for determining the upper and lower bounds.

The determination of the upper and lower bounds of the reachable area of the wrist are based on the principal obstacle, which is the first obstacle touched by the lower arm when it rotates counterclockwise (or clockwise) about the center of joint 2 from a position lower (or higher) than all the hanging (or standing) obstacles. Note: joint 3 must keep its maximum angular displacement when the lower arm rotates clockwise or counterclockwise. Figure 8 shows a case simplified from that of Figure 6 and is used to explain how the equation of the upper bound d-s is determined. The concave curve d-s is the trajectory of the wrist when the upper arm rotates counterclockwise about the center of joint 2 (the point "m") and the lower arm keeps touching the corner "d" of the principal obstacle D. The equation of curve d-s can be derived by solving the following four simultaneous equations. Note: point "e" is an arbitrary point on the curve d-s.

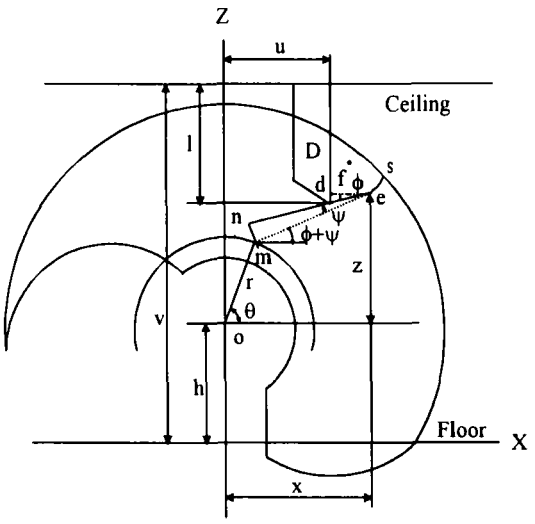


Figure 8. The lower arm and the upper bound.

$$\tan\phi = (l + h + z - v)/(x - u) \tag{2}$$

$$\tan\psi = i/j \tag{3}$$

$$x = r\cos\theta + k\cos(\phi + \psi) \tag{4}$$

$$z = r\sin\theta + k\sin(\phi + \psi) \tag{5}$$

where $i = mn$, $j = ne$, $r = om$, $k = me$

The $\cos(\phi + \psi)$ and $\sin(\phi + \psi)$ can be obtained by solving Eqs. (2) and (3). Then they are brought into Eqs. (4) and (5), and the following equation will be derived.

$$\begin{aligned} f(x,z) &= z^6 + 2pz^5 + (q^2 + p^2 + 2s^2 - 4k^2\cos^2\psi)z^4 \\ &\quad + (4s^2p - 8uk^2\cos\psi)z^3 \\ &\quad + (2s^2q^2 + 2s^2p^2 + s^4 - 4k^2u^2 - 8wk^2\cos\psi)z^2 \\ &\quad + (2ps^4 - 8uwk^2)z + (s^4q^2 + s^4p^2 - 4k^2w^2) \\ &= 0 \end{aligned} \tag{6}$$

where

$$s = x^2 + k^2 - r^2, \quad p = +h - v, q = x - u$$

$$u = p\cos\psi + q\sin\psi - x\sin\psi$$

$$w = xq\cos\psi - xp\sin\psi$$

Eq. (6) inicates that the curve d-s is a sixth-order polynomial curve. The area above the curve d-s and between the lines $x = x_d$ and $x = x_s$ is an unreachable area, where x_d and x_s are the x-coordinates of points "d" and "s" respectively. Therefore, curve d-s is part of the upper bound of the wrist's reachable area. When the wrist moves along this trajectory from point "s" to point "d," the tangent of the lower arm will be smallest if its short part, m-n, is collinear with the upper arm, o-m (see m'-n' and o-m' of Fig. 6). (Note: This short part of the lower arm is link 3 of the ABB 2000 robot.) This can be realized from the fact that "n'''" is the highest position that the twist point "n" of the lower arm can reach (point "n'''" is higher than points "n" and "n''"). Because the line

n'-s' (the lower arm) has the smallest tangent, any point on the curve d-s will be above the line n'-s'. In other words, all points on the line n'-s' and between the lines $x = x_d$ and $x = x_s$ are reachable points. If the line n'-s' is chosen to replace the curve d-s as the upper bound, it will be easier to find a reachable path point on it than on the curve d-s, which requires time-consuming calculations for finding the roots of the sixth-order polynomial equation given in Eq. (6). This replacement is worthy even that the reachable area under the curve d-s and above the line n'-s' is lost. The equation of line n'-s' is

$$t(x - x_d) = z - z_d \tag{7}$$

where

$$t = \tan(\alpha + \beta - \pi/2)$$
$$\alpha = \tan^{-1}[(z_d - h)/x_d]$$
$$\beta = \cos^{-1}[(on')/od]$$

For most industrial robots, the center of joint 3 is located at the axis of joint 4. The length of link 3 is zero. Therefore, the smallest tangent of the lower arm happens when the lower arm is perpendicular to the upper arm.

Similarly, the lower bound of the wrist's reachable area in Figure 6 can be determined by following the flow chart of Figure 7. First, the lower arm touches the corner "a" of the principal obstacle A. Point "e" is the wrist point. Then, the upper arm rotates counterclockwise and about the joint "o" while the lower arm always touches the top of the obstacle A. The trajectory of the wrist point will be the curve of b-e. This curve is composed of two sixth-order polynomial curves, e-p and p-b. The connection point "p" of these two curves is the position of the wrist point when the lower arm lies flatly on top of the obstacle A. Because the equation of the curved trajectory a-e is complicated, the line segment a-e is used to replace this trajectory as the lower bound so that the computing time for determining reachable path points can be reduced.

The area between the upper bound and lower bound is the reachable area of the wrist. If the reference path point is within this area, it is a collision-free path point. If the reference path point is above the upper bound (or below the lower bound), the point on the upper (or lower) bound and with the same x-coordinate as the reference path point will be selected as the reachable path point. For example, the reference path point "f" of Figure 6 is above the upper bound. The point "f''" on the upper bound is selected as the reachable path point. If point "f''" is also below the lower bound, this will initiate a search for finding a reachable path point along the upper bound and toward the Z-axis.

The selection of point "f''" to replace point "f" is because "f''" is the reachable point that is closest to the reference path point. Therefore, the pre-planned spiral path will not change too much. However, it is not absolutely necessary to choose the point "f'." Any point within the reachable area of the wrist is a reachable point and can be selected to replace the unreachable reference path point.

The above algorithm for determining reachable path points is applied sequentially from the first cross-section of the workspace of the wrist, where the initial point is on it, to the last one, where the goal point is on it. Finally, each cross-section of the workspace of the wrist will have one determined reachable path point.

After all reachable path points are determined, an Archimedes's spiral path segment between two consequent reachable path points will be generated. This spiral path segment is collision-free if no obstacle is above or below it. Moreover, it is also a collision-free path if the spiral path segment is above (or below) the top (or bottom) surfaces of standing (or hanging) obstacles. However, a collision may occur if the path starts from a lower (or higher) position to a position above (or below) the top (or bottom) surface of a standing (or hanging) obstacle or vice versa. Figure 9 shows an example. The wrist may collide with the obstacle when it moves from position "a" lower than the obstacle to position "b" above

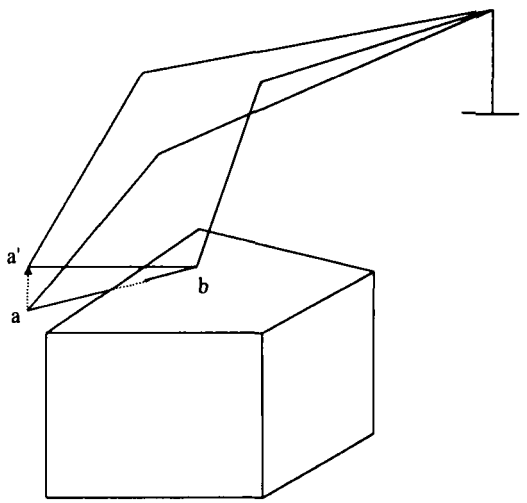


Figure 9. The new path point a' for avoiding a collision.

the obstacle. Such a collision can be avoided by substituting the path point "*a*" with a new path point "*a'*" whose height is equal to that of the path point "*b*."

It is necessary to check if all spiral path segments between two consequent path points are collision-free. If a collision is found on some path segment, this path segment will be modified as described above. After all path segments have been checked for the absence of collisions, the path that sequentially connects the initial position of the wrist, the positions of all reachable path points, and the goal position of the wrist will be a collision-free path for the wrist. If the end effector or a gripper is attached to the wrist, all reachable path points will have to move upward or downward by a distance that is equal to the length of the end effector or gripper to determine a collision-free path for the end effector or gripper. In the meantime, the new path points must be checked again to see if they are still within the reachable area of the wrist. If the path points are not within its reachable area, it will be necessary to determine new reachable path points.

It is interesting to note that the dimension of the end effector or gripper can also be considered in the process of obstacle expansion. The workspace of the end effector is part of a sphere centered at the wrist point and with a radius equal to the length of the end effector. If the radius of the sphere, R, is smaller than the radius of the lower arm, N, or the diagonal distance of the upper arm, M, no modification of the expansion process is necessary. If R is larger than M or N, the sides of the obstacles must be shifted out by a distance equal to R instead of M or N. Also, the top or bottom surfaces of the obstacles should be shifted a distance equal to the largest value among $M/\cos\phi$, $N/\cos\phi$, and R.

Figure 10 illustrates the application of the developed algorithm to an ABB IRB 2000 robot. It is the same case as the sketch in Figure 1. The CPU time spent for generating the collision-free path is 6.6 seconds using a Silicon Graphics 4D-35 workstation, which has a speed of 23 MIPS.

5. APPLICATION TO OTHER TYPES OF ROBOTS

The algorithm developed in this article for generating collision-free paths was based on an articulated robot. Compared to other types of robots, the articulated type robot has much more complicated polynomial equations that represent the upper and lower bounds. If an algorithm for articulated robots to determine collision-free paths can be developed, it will be applicable to other types of robots.

The main concern in applying the developed algorithm to the spherical, cylindrical, and Cartesian types of robots is the determination of the upper and lower bounds of the reachable area of the wrist. As described previously, the robot arms will be shrunk into line segments after the objects have been expanded. Finally, real obstacles will be identified.

The configuration of a spherical type robot is similar to that of an articulated robot. The difference is that joint 3 of the former is a prismatic joint instead of a revolute joint. Because of the prismatic joint 3, the upper and lower bounds of the reachable area of the wrist will be linear, and their equations can be derived easily. Figure 11 shows a typical fan-shaped cross section, *c-d-g-h*, of the reachable area of the wrist of a spherical type robot without considering the existence of obstacles. The line segment *b-r* is the trajectory of the wrist when the lower arm moves from point "*r*" to point "*b*" and always touches the corner "*b*" of obstacle D. This linear trajectory is a part of the upper bound of the reachable area of the wrist. If the reference path point "*f*" is located at the unreachable area above the upper bound, a reachable point "*f'*," which is on the upper bound and has the same x-coordinate as point "*f*," will be selected as the reachable path point. Similarly, the line segment *a-s* is a part of the lower bound of the reachable area of the wrist. If point "*p*" is the reference path point, it will be replaced by the reachable path point "*p'*" which is on the lower bound and has the same x-coordinate as point "*p*."

As for the cylindrical type robot, it has similar configuration as the spherical type robot. The only difference is that joint 2 is prismatic. The prismatic joints 2 and 3 restrict the upper and lower arms to move horizontally or vertically. In other words, the cross-section of the workspace of the wrist will be a rectangle as shown in Figure 12. Therefore, the upper and lower bounds of the reachable area of the wrist will be linear and horizontal. In Figure 12, the upper bound *b-r* and lower bound *a-s* are linear and horizontal. The equations representing the upper and lower bounds are very simple. As with the spherical type robot above, the reachable path point "*f'*" (or "*p'*") used for replacing the unreachable path point "*f*" (or "*p*") can be determined quite easily.

Finally, the Cartesian type robot has a prismatic joint 1, which is the only difference in its configuration compared to the cylindrical type robot. Because the joints 1, 2, and 3 are all prismatic, the shape of

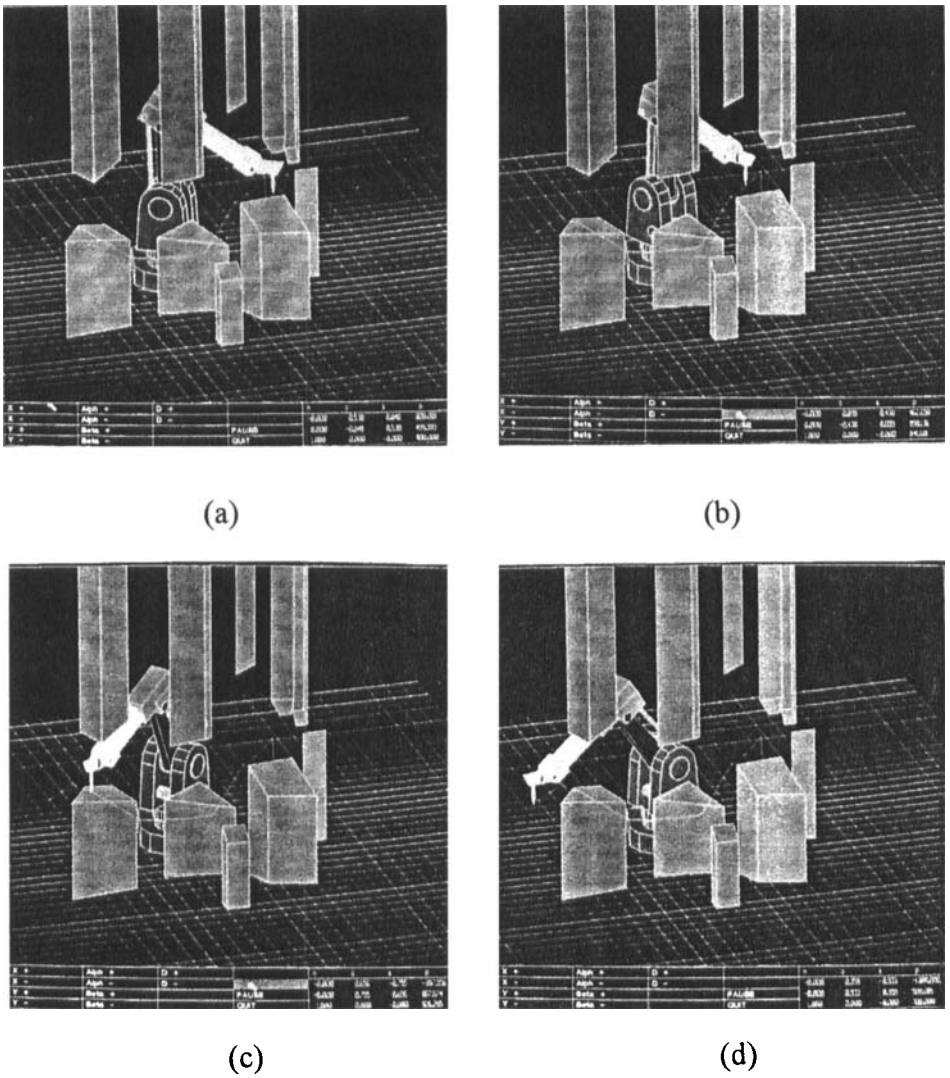


Figure 10. The animation of the ABB IRB 2000 robot.

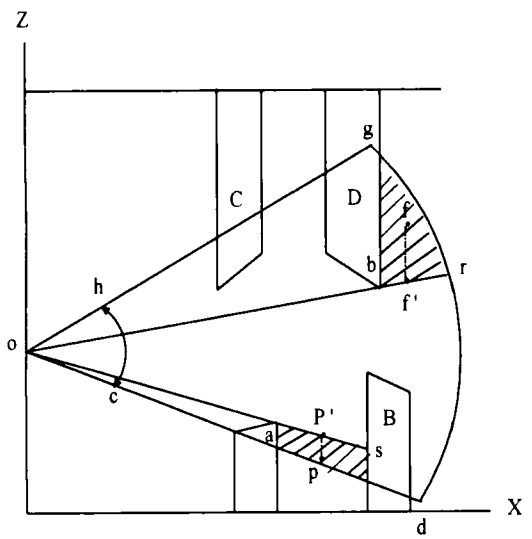


Figure 11. The wrist's reachable area of the spherical type robot.

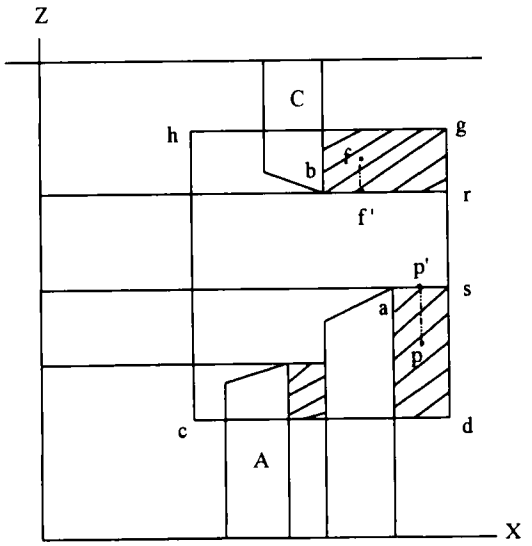


Figure 12. The wrist's reachable area of the cylindrical type robot.

the workspace of the wrist is a rectangular solid. A linear path between the initial and goal positions will always be inside the workspace. Therefore, either an Archimedes's spiral path or a linear path can be the reference path for determining a collision-free path. The upper and lower bounds of the reachable area of the wrist are linear and horizontal. Therefore, the reachable path point can be determined as easily as for the cylindrical type robot.

6. CONCLUSIONS

Computer animation has shown that the algorithm developed is an efficient method for generating collision-free paths. It can be applied not only to the articulated type robot but also the spherical, polar, and Cartesian types of robots. The time required for generating a collision-free path is related to the number of corners on the obstacles. To reduce computing time, the obstacles may be replaced by larger polyhedrons with fewer corners.

The determination of the upper and lower bounds of the reachable area of the wrist makes it possible for robot manipulators to move close to the obstacles. This offers the flexibility of selecting safe paths for special applications or for finding an optimum collision-free path.

The automatic generation of collision-free paths will enable robots to operate autonomously. It is expected that intelligent robots with self-operation capability will be developed soon.

The author acknowledges financial support of the National Science Council, Taiwan, under the contract NSC 80-0422-E008-03.

REFERENCES

1. S. M. Udupa, "Collision detection and avoidance in computer controlled manipulators," *Proc. Int. Joint Conf. Artif. Intel.*, MIT, Cambridge, MA, 1977, pp. 737–748.
2. R. A. Brooks, "Solving the find-path problem by good representation of free space," *IEEE Trans. Syst. Man Cybern.*, **SMC-13**(2), 190–197, 1983.
3. R. A. Brooks, "Planning collision free motions for pick and place operations," *Proc. Int. Rob. Res. Symp.* MIT Press, Cambridge, MA, 1984, pp. 5–38.
4. J. F. Canny and M. C. Lin, "An opportunistic global path planner," *Proc. IEEE Int. Conf. Rob. Autom.*, 1990, pp. 1554–1559.
5. C. S. Tseng, C. Crane, and J. Duffy, "Generation of collision-free paths for robot manipulators," *J. Comput. Mech. Eng.* 7(2), 58–65, 1988.
6. L. Young and J. Duffy, "A theory for the articulation of planar robots: Part I and II," *J. Mech. Transm. Autom. Des. Trans. ASME*, **109**(1), 29–41, 1987.
7. S. Dubowsky, M. Norris, and Z. Shiller, "Time optimal trajectory planning for robotic manipulators with obstacle avoidance: A CAD approach," *Proc. IEEE Int. Conf. Rob. Autom.* Paper No. 276, St. Louis, MS, April 1986.
8. P. Dupont and S. Derby, "Planning collision free paths for redundant robots using a selective search of configuration space," *Proc. ASME Mech. Conf.*, Columbus, OH, 1986.
9. B. Faverjon, "Obstacle avoidance using an octree in the configuration space of a manipulator," *Proc. Int. Conf. Rob.*, Atlanta, GA, March 1984, pp. 504–512.
10. T. Lozano-Pérez, "A simple motion-planning generation algorithm for robot manipulators," *IEEE J. Rob. Autom.*, **RA-3**(3), 224–234, 1987.
11. C. A. Shaffer and G. M. Herb," A real-time robot arm collision-avoidance system," *IEEE Trans. Rob. Autom.* 8(2), 149–160, 1992.
12. V. Lumelsky and E. Cheung, "Toward safe real-time robot teleoperation: Automatic whole-sensitive arm collision avoidance frees the operator for global control," *Proc. IEEE Int. Conf. Rob. Autom.*, Sacramento, CA, April 1991, pp. 797–802.
13. E. Cheung and V. Lumelsky, "Motion planning for a whole-sensitive robot arm manipulator," *Proc. IEEE Int. Conf. Rob. Autom.*, 1990, pp. 344–349.
14. Y. K. Hwang and N. Ahuja, "A potential field approach to path planning," *IEEE Trans. Rob. Autom.*, 8(1), 23–32, 1992.
15. G. Heinzinger, P. Jacobs, J. Canny, and B. Paden, "Time-optimal trajectories for a robot manipulator: A provably good approximation algorithm," *Proc. IEEE Int. Conf. Rob. Autom.*, 1990, pp. 150–156.
16. T. Lozano-Pérez, "An algorithm for planning collision free paths among polyhedral obstacles," *Comm. ACM*, **22**(10), 560–570, 1979.
17. W. S. Newman and M. S. Branicky, "Real-time configuration space transforms for obstacle avoidance," *Int. J. Rob. Res.*, **10**(6), 1991.
18. M. S. Branicky and W. S. Newman, "Rapid computation of configuration space obstacles," *Proc. IEEE Int. Conf. Rob. Autom.*, 1990, pp. 304–310.
19. S. Dubowsky and T. D. Blubaugh, "Time optimal robotic manipulator motions and work places for point to point," *Proc. IEEE Conf. Decis. Control*, Fort Lauderdale, FL, December 1985, pp. 11–13.