Ellipsoidal Modeling for Articulated Robot Manipulators for Interactive Motion Planning

Ming-Yi Ju[*,], Jing-Sin Liu[*] and Kao-Shing Hwang[**]**

[*]**Institute of Information Science 20**
Academia Sinica
Nankang, Taipei 115, Taiwan, R.O.C.

[**]**Department of Electrical Engineering**
National Chung Cheng University
Chiayi 160, Taiwan, R.O.C.

Abstract

The report presents an enclosing and enclosed ellipsoids based solid modeling technique for serial links of articulated robot manipulators. This modeling allows the collision detection for interactive robot motion planning in a graphical simulation environment or in a telerobotic simulator be performed efficiently and precisely. A systematic way of determining the kinematic transformation matrices between links and its associated ellipsoids model is presented. An example, a PUMA560 manipulator, is shown in details to demonstrate the proposed modeling technique.

Keywords: solid modeling, enclosing ellipsoid, enclosed ellipsoid, PUMA560

1. INTRODUCTION

Methods for detecting intersections of three-dimensional objects play an important role for many robotic applications [1]. Usually, a collision detection problem in real applications is equivalent to that of collision detection between two convex polyhedra. Methods based on polyhedral model for computing the Euclidean distance between two polyhedra are proposed [2]. However, such kind of methods has a computational expense which is nearly linear with the total number of vertices. Therefore, a lot of research efforts devoted to accurately model convex polyhedra with a simpler shape for efficient distance computation and collision detection. How to model the shapes of robots' links is an important problem to an efficient collision detection algorithm. For a modeling method, its criteria can be summarized by

- The model must represent the physical system as precisely as possible.
- The model must be simple enough to ensure that the algorithms can be solved fast enough to secure the real-time operation of the manipulator.

In order to increase the accuracy of the model, it is practical to model each link individually by creating a volume representative of each link, rather than to model a robot as a single body. Generally, there are three methods which are usually used to achieve this: polyhedral modeling, cylindrical modeling and spherical modeling. Some comparisons are made in the following [19].

Although accuracy can be obtained in polyhedral model representation for each link, it is computationally intensive to construct. The description of complex shapes represented by a polyhedral model often requires the whole data structure of the faces and edges. Moreover, high quality rendering of complex or curved shapes requires hundreds of planar faces. Besides, it becomes more difficult when rotations are introduced. As a result, the detection of interactions with other objectives becomes more complex and impairs the real-time operation of the arm. In cylindrical modeling, each link can be effectively 'inserted' into a tube. The problem is that, when we try to view a cylindrical model with respect to a Cartesian coordinate system, the mathematical expression becomes complicated that restricts the real-time operation. In sphere modeling, an immediate advantage is simplicity in mathematical formulation. Since a sphere is rotationally invariant, only the calculation of the original position is needed. From the point of view of collision detection, it just needs to compare the distance between an obstacle and the sphere model with the radius of the sphere. However, spherical representation for robot links often leads to a large volume of waste.

It can be seen that the polyhedral method is accurate but computationally intensive; the representation of cylinders is elegant but complicated; the sphere modeling is quite inaccurate. Therefore, a model able to represent the link of a robot simply and estimate the distance efficiently is important for collision-detection in interactive motion planning. Based on the principles mentioned previously, the minimum-volume enclosing ellipsoid (LJ ellipsoid) [8] for shape modeling is used as the basis in modeling a robot link.

In bounding volume schemes, it is very common to assume that an object is represented by an enveloping ellipsoid or sphere [3, 4]. The reason for adopting such a simple primitive is to reduce the complexity in representation and collision detection. However, for robot manipulator, such kind of representation does not have enough accuracy since the link of a robot arm is usually rectangular. A representation called dynamic spheres [5], which are parametric volumes composed of an infinite number of spheres, the positions and radii of which vary linearly over the extent of the object. Hierarchical spherical representations were also proposed for more accurate representations [6, 7]. The collision radii are chosen so that the collision detection time is low and the available free workspace for each object is as large as possible. Unfortunately, a large number of spheres are needed to represent slender objects with reasonable accuracy, and hence the computation cost for collision detection increases. Besides, when the geometric primitives are near or in collision, such method must check all the pairs of spheres for intersection detection, not efficient in the real-time applications.

Superquadric models [16, 17] can approximate complex shape precisely through inside-outside function by tuning its parameters. One of the drawbacks for superquadric model is time consuming. Ellipsoids are also used as primitives for collision detection due to their simple geometric features [8, 18, 20]. However, detecting intersection between two ellipsoids is not an easy task and such bounding volume models also suffer from representation accuracy for collision detection.

Bounding volume schemes regard the distance between two enclosing ellipsoids as the minimum distance between the two polyhedra. Obviously, it will induce a large estimate error in most cases. The objective of this work is to develop an ellipsoidal modeling technique for efficient and accurate distance computation to overcome the problems encountered in bounding volume schemes. The proposed approach first approximates the link shape by a convex polyhedron and then computes its enclosing and enclosed ellipsoids. The distance between two convex polyhedra can be estimated accurately based on those ellipsoids [9], [10]. Geometric relations among the polyhedra and the modeling ellipsoids are depicted in Fig. 1. Briefly speaking, our approach uses the enclosing ellipsoid as the initial value for enclosed ellipsoid computation. By combining the geometric information of polyhedra and enclosed ellipsoids, a tight distance estimate between polyhedra can be computed in an efficient way. The details about the distance computation based on enclosed ellipsoids can be found in the work [9], [10].

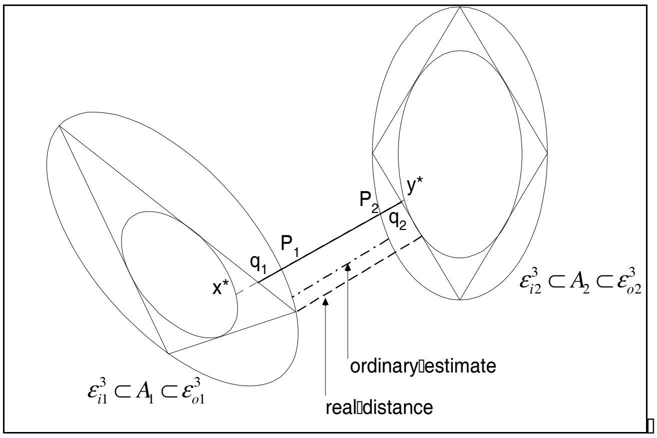

y*
P2
q2
P1
q1
x*
$\varepsilon_{i2}^{3} \subset A_2 \subset \varepsilon_{o2}^{3}$
ordinary estimate
real distance
$\varepsilon_{i1}^{3} \subset A_1 \subset \varepsilon_{o1}^{3}$

Fig. 1. The distance estimates based on enclosed ellipsoids

The organization of this paper is as follows. Section 2 describes the approach for the construction of enclosing and enclosed ellipsoids of convex polyhedra. Forward kinematics for articulated kinematic chain is introduced in Section 3. An implementation example of how to model a PUMA560 robot manipulator with a set of ellipsoids is presented in Section 4. Finally, Section 5 concludes the paper.

2. ELLIPSOID COMPUTATION FOR LINK MODELING

For most practical applications in robotics and computer graphics, complex objects are generally represented as a union of simpler primitives [11]. However, the primitives should reflect a good balance between the efficiency of primitive-primitive intersection detections and the number of primitives required to adequately represent the objects. For the proposed technique, the ellipsoids are selected as a basic element to represent objects. An ellipsoid is capable of representing a convex polyhedron, such as robot's links, in the direction of its axis. Besides, the main advantage of ellipsoid model is that it is very simple in mathematical representation; therefore it can reduce the complexity of ensuing computations. An ellipsoid is represented as $\varepsilon^{n}(y, \mathbf{Y})$ in this report, where *n* is the dimension, *y* is the center, and **Y** is the characteristic matrix.

2.1 Enclosing Ellipsoid Computation

Löwner-John (L-J) ellipsoid, the minimum-volume enclosing ellipsoid of a body, is an intuitively appealing means to lump the detailed geometry into a single quadratic surface. The computation of the L-J ellipsoid is a convex optimization problem [8] whose solution can be derived by applying the ellipsoid algorithm [12].

2.1 Enclosed Ellipsoid Computation

In order to generate an enclosed ellipsoid that fits to the polyhedron, a 3-phase approach is proposed. Our approach roach is to derive the enclosed ellipsoid of a convex polyhedron by means of shrinking, stretching, and then scaling an L-J ellipsoid to fit the polyhedron as tight as possible.

Phase 1 – Isotropically shrinking all principal axes

An initial enclosed ellipsoid is given by shrinking the L-J ellipsoid along its principal axes isotropically to be contained in the polyhedron in phase 1. Let $\varepsilon^n(y,\mathbf{Y})$ be the minimum volume *n*-ellipsoid containing a convex polyhedron in *n*-dimensional space. Then, the initial enclosed ellipsoid is given as $\varepsilon^n(y,(n+1)^2\mathbf{Y})$, formed by shrinking $\varepsilon^n(y,\mathbf{Y})$ from its center by a factor of (*n*+*1*), to guarantee that the polyhedron contains the initial ellipsoid [11]. Therefore, the ellipsoid $\varepsilon^3(y,16\mathbf{Y})$ is selected to be the initial guess for enclosed ellipsoid computation in 3-dimensional case. The regulation of the shrinking factor is based on the bisection methods. The phase terminates with a user-defined error while the ellipsoid cannot extend further without overlapping with the facets of a polyhedron.

Phase 2 - Stretching

The phase 1 terminated while the enclosed ellipsoid is very close to one of the polyhedron's facets; however, it still has some free space to enlarge the enclosed ellipsoid. Stretching operation [13] is applied to expand the enclosed ellipsoid along a given direction in phase 2. Let *p* be the point to adapt to and $\varepsilon^3(c,\mathbf{M})$ be the enclosed ellipsoid generated in phase 1. The idea is to move the ellipsoid's center towards to the point, i.e. *p*, and then stretch the ellipsoid along the movement direction such that the old border point in the opposite direction remains a border point. Therefore the new center is represented as

$$c' = c + \beta(p-c),$$

where β determines how far to move the ellipsoid's center. With the normalized distance vector

$$\mathbf{a} = \mathbf{M}^{1/2}(p-c)/\left\|\mathbf{M}^{1/2}(p-c)\right\|, \qquad (1)$$

the new transformation matrix is given as

$$\mathbf{M}'^{1/2} = (\mathbf{I}+(\alpha-1)\mathrm{aa}^T)\mathbf{M}^{1/2}, \qquad (2)$$

where

$$\alpha = 1/(1 + \|\beta \mathbf{M}^{1/2}(p - c)\|).$$

It is worth to notice that enlarging an ellipsoid means that its transformation matrix makes the vectors shorter, therefore α is always smaller than 1. In the stretching operation, p is given as

$$p = L \cdot (p_m - c)/\|p_m - c\|,$$

where L is the distance from the farthest vertex of the polyhedron to c, the center of enclosed ellipsoid, and p_m is the mass center of vetices of the farthest face to c. In our implementation, β is initialized as 1 and inside the range from 0 to 1. The selection of β is based on the bisection method. The algorithm terminates while the variation of β is smaller than 0.005.

As mentioned the old border point in the opposite of the stretching direction is still a border point of the new ellipsoid, it implies that perhaps there is free space for the ellipsoid to expand in the opposite side. Therefore, the stretching operation is applied once again for possibly enlarging the ellipsoid. In order to hold the interface point between the facet of a polyhedron and the ellipsoid, the new p, which needs to be adapted to, is given as

$$p = L \cdot (c - p_i)/\|c - p_i\|,$$

where p_i is the interface point.

Phase 3 – One by one enlarging each radius

Let $\varepsilon^3(c', \mathbf{M}')$ be the enclosed ellipsoid generated by means of stretching. Since the matrix $\mathbf{M}'$ is symmetric and positive- definite, it can be diagonalized through a rotational matrix **V**. The relation is expressed as

$$\mathbf{D} = \mathbf{V}^{-1}\mathbf{M}'\mathbf{V}.$$

In fact, matrix **V** is the matrix of eigenvectors of matrix $\mathbf{M}'$ and matrix **D** is the canonical form of $\mathbf{M}'$ — a diagonal matrix with $\mathbf{M}'$'s eigenvalues on the main diagonal. Since the inverse square roots of matrix $\mathbf{M}'$'s eigenvalues are equivalent to the length of principal axes of the enclosing ellipsoid, the change of ellipsoid's each radius can be performed individually by means of multiplying matrix **D** with a scaling matrix **S**, which is also diagonal. Therefore, each new radius of the enclosed ellipsoid can be written as

$$\mathbf{D}' = \mathbf{S}\mathbf{D};$$

and the enlarged enclosed ellipsoid can be represented as

$$\mathbf{M}'' = \mathbf{V}\mathbf{D}'\mathbf{V}^{-1}.$$

By the use of scaling operations, the length of each principal axis of the enclosed ellipsoid is extended individually until the enlargement induces the ellipsoid to intersect with facets of the polyhedron.

Since $\varepsilon^3(c,\mathbf{M}'')$ is generated by stretching along a specified vector and, then, enlarging some axes of $\varepsilon^3(c,\mathbf{M})$ generated in phase 1, the following relationship $\varepsilon^3(c,\mathbf{M}) \subseteq \varepsilon^3(c,\mathbf{M}'')$ always holds.

3. FORWARD KINEMATICS OF ARTICULATED ROBOT MANIPULATORS

An articulated robot manipulator consists of a series of segments connected by single degree of freedom translational or rotational joints. For each segment there is a Cartesian coordinate system and the segment is described in its own coordinate system. Denavit-Hartenberg representation [14] is the most popular method to describe the forward kinematics of manipulators. D-H representation results in a 4x4 homogeneous transformation matrix representing each link's coordinate system at the joint with respect to the previous link's coordinate system. Therefore, through sequential transformation from the base and end effector coordinate frames, the end effector expressed in the hand coordinate frame can be transformed and expressed in the base coordinate frame.

In each D-H coordinate system of robot joints, the transformation is represented by two rotations and two translations:

$Tran(0,0,d)Rot(z,\theta)Tran(a,0,0)Rot(x,\alpha)$,

where θ is the angle between the adjacent links; d is the distance between the adjacent links; α is the twist angle of link; and a is the length of link.

Every coordinate frame is determined and established on the basis of three rules:

1. The z_{i-1} axis lies along the axis of motion of the ith joint.
2. The x_i axis is normal to the z_{i-1}, and pointing away it.
3. The y_i axis completes the right-handed coordinate system as required.

As for the D-H representation, the four parameters are defined as follows:

θ_i is the joint angle from the x_{i-1} axis to the x_i axis about the z_{i-1} axis.

d_i is the distance from the origin of the $(i-1)$th frame to the intersection of the z_{i-1} axis with the x_i axis along the z_{i-1} axis.

α_i is the offset angle from the z_{i-1} axis to the z_i axis.

a_i is the offset distance from the intersection of the z_{i-1} axis with the x_i axis to the origin of the ith frame along the x_i axis.

The four operations can be expressed by a composite transformation matrix ${}^{i-1}A_i$ for adjacent coordinate frame, *i* and *i*-1. Thus

$$
\begin{aligned}
{}^{i-1}A_i &= Tran(0,0,d_i)Rot(z,\theta_i)Tran(a_i,0,0)Rot(x,\alpha_i) \\
&= \begin{bmatrix} \cos\theta_i & -\cos\alpha_i\sin\theta_i & \sin\alpha_i\sin\theta_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\alpha_i\cos\theta_i & -\sin\alpha_i\cos\theta_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}.
\end{aligned}
$$

Therefore, the position and orientation of joint *i* coordinate frame with respect to the base coordinate frame can be expressed as:

$$
{}^{0}T_i = {}^{0}A_1 \cdot {}^{1}A_2 \cdots\cdots {}^{i-1}A_i\,.
$$

4. MODELING EXAMPLE – PUMA560 ROBOT MANIPULATOR

A PUMA 560 robot is depicted in Fig. 2. According to the primary parameters of PUMA560, a graphic model of PUMA560, as shown in Fig. 3, is constructed and used to demonstrate how to model a manipulator with a set of ellipsoids for use in collision detection in a dynamic environment.

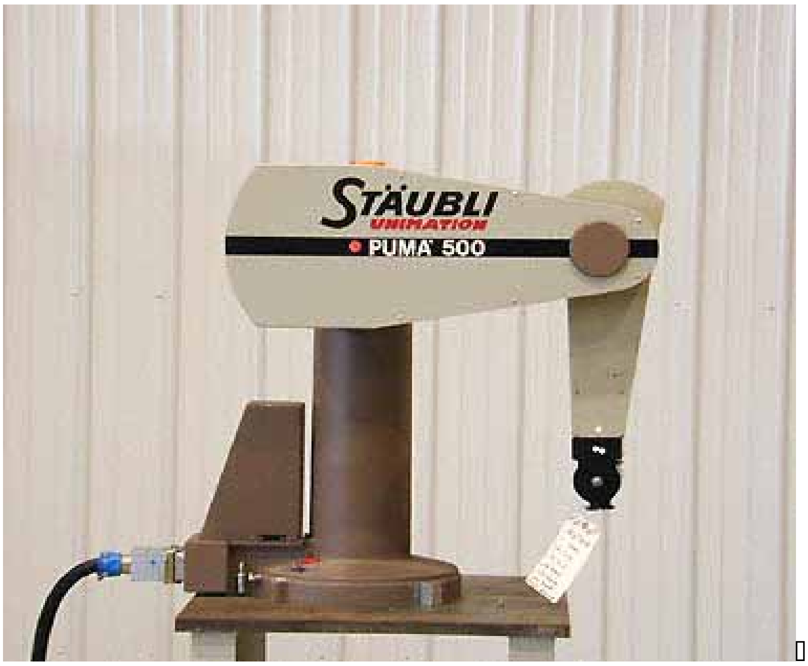


Fig. 2. PUMA560 [21].

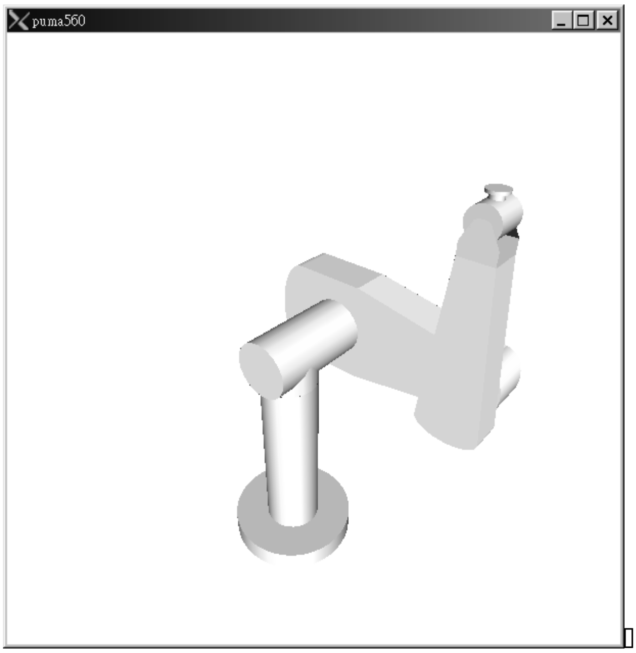


Fig. 3. Simulated PUMA560 manipulator.

Since the L-J ellipsoid computation is based on polyhedral model, therefore, all the curved surfaces of links are first approximated with a union of several polyhedral facets. In our implementation, the cylindrical part of links of PUMA560 is first approximated by an octahedron with even number of vertices. From Fig. 4, the maximal approximation error induced by the polyhedral modeling is derived as

$$D_{err} \le R(1-\sin((n-2)*180/2n)),$$

where R is the radius of the circle and n=8 in our case.

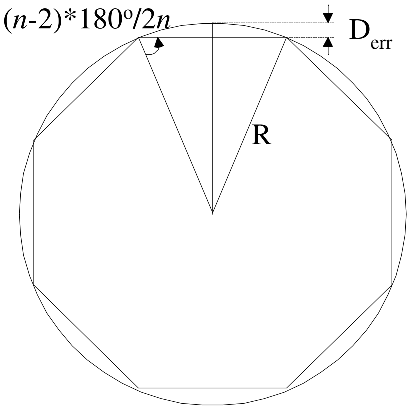


Fig. 4. Circular cross section of a cylinder and its approximated n-polygon model.

Although the volume of approximated model is smaller than the real cylinder, it is noted that the vertices of polyhedron are symmetrical if n is even. This property guarantees that the distance between two symmetrical vertices is equal to the cross section's diameter of the cylinder. It can be deduced from this property that the enclosing ellipsoid of polyhedron is guaranteed to contain the original cylinder as well.

Besides, the link 4 and link 5 of PUMA560 are lumped together heuristically in our implementation. The reason for the simplification is that to rotate the joint 5 will not affect the lumped geometric shape of link 4 and link5. Thus, for convenience and the saving of computation time, the lumped object is approximated with a bounding box.

The polyhedral model of PUMA560 for ellipsoid computation is shown in Fig. 5. Based on the polyhedral model, the enclosing and enclosed ellipsoid models for PUMA560 are depicted in Fig. 6 and Fig. 7, respectively.

It is noticing that if the elements of the column vector that represent the center of an ellipsoid is not equal to zero, it means that there is displacement between the ellipsoid's center and the modeled link's origin of local coordinate. Furthermore, an additional translation operation is needed to correctly locate the ellipsoid's center. Besides, rotation operations are also requested to locate the central coordinate of an ellipsoid with correct orientation with respect to the manipulator's configuration. Since all the vertices of each link are constructed with respect to the link's local coordinate, therefore, it can be done easily by directly utilizing the forward kinematics of PUMA560. It can be represented as a homogeneous matrix as:

$$ {}^{base}T_i \cdot T_{displacement} = \begin{bmatrix} R_i & P_i \\ 0 & 1 \end{bmatrix}, $$

where T_i is given in Appendix A.

Therefore, in summary, the correct central coordinate, depending on the PUMA560 configuration, can be rewritten as:

$$ c_i = P_i , $$

Correspondingly, the right orientations for enclosing and enclosed ellipsoids are given as:

$M_i = R_i^{-1} M_i R_i$ and $N_i = R_i^{-1} N_i R_i$, respectively.

All the detailed computations for constructing the enclosing and enclosed ellipsoids of each link of PUMA560 are described in Appendix B.

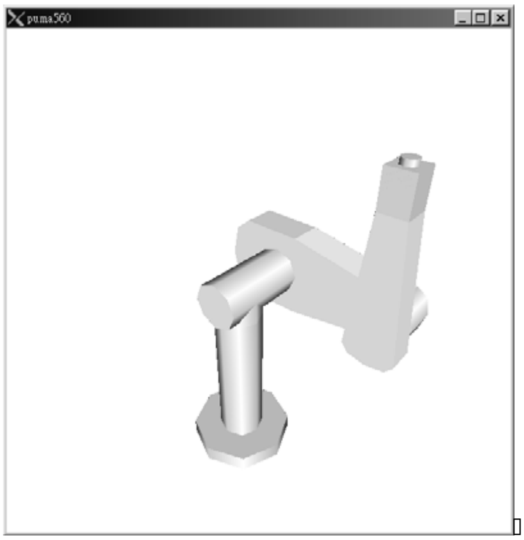


Fig. 5. Polyhedral model of PUMA560.

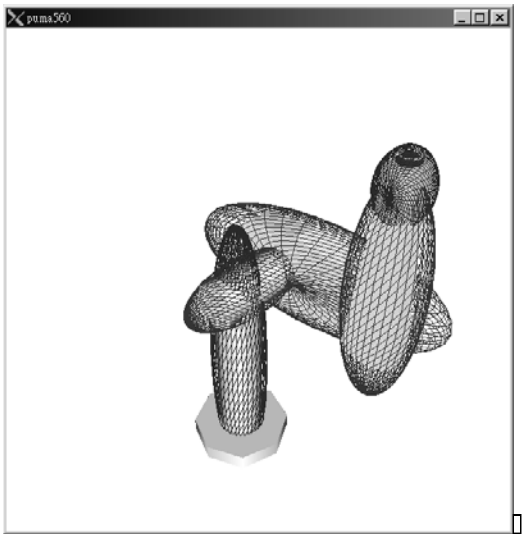


Fig. 6. PUMA560 modeled with L-J ellipsoids.

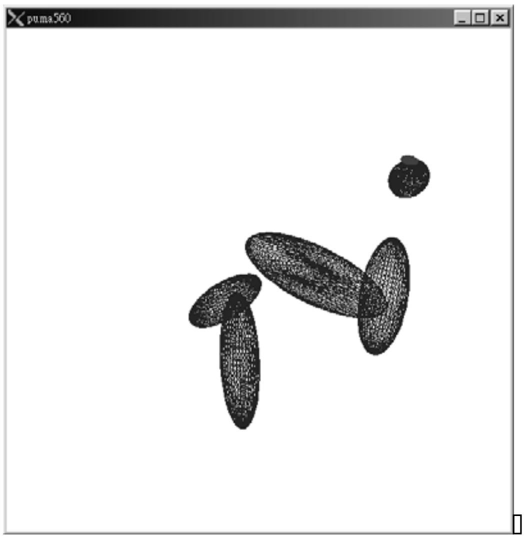
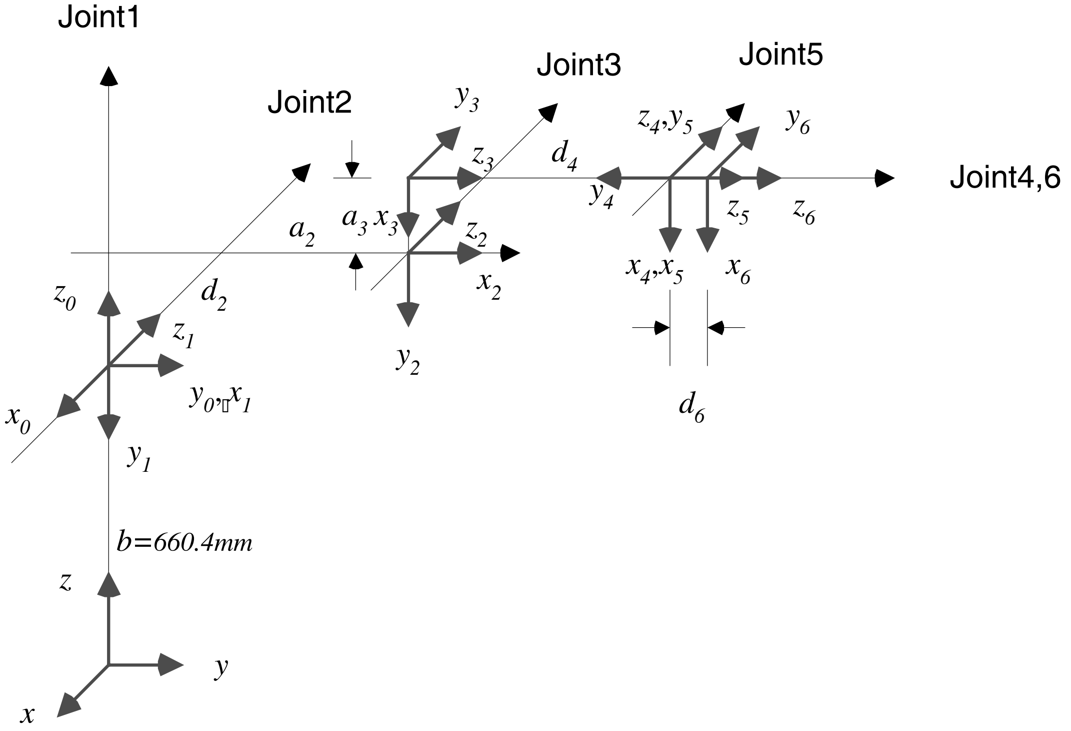


Fig. 7. PUMA560 modeled with enclosed ellipsoids.

5. CONCLUSION

A systematic ellipsoid modeling technique for the links of articulated robot manipulators, PUMA560 in particular, is proposed for use in interactive motion planning in a graphical simulation environment or in a telerobotic simulator. In comparison with the traditional bounding volume approaches that only the enveloping ellipsoids or spheres for convex polyhedra modeling are utilized for conservative distance estimate, the proposed technique makes use of additional interior and exterior representations to provide more accurate estimate about the minimal distance and to reduce the number of inaccurate detections of collisions for robotic applications. Moreover, due to the use of ellipsoid models to simplify the representation complexity of convex polyhedra, the computational complexity for minimum distance computation and collision detection is also significantly reduced, as compared with the polyhedral models.

APPENDIX A. Forward Kinematics of PUMA560



PUMA robot arm link coordinate parameters				
Joint i	α_i	a_i	d_i	Joint range
1	-90	0	0	-160 to 160
2	0	431.8*mm*	149.09*mm*	-225 to 45
3	90	-20.32*mm*	0	-45 to 225
4	-90	0	433.07*mm*	-110 to 170
5	90	0	0	-100 to 100
6	0	0	56.25*mm*	-266 to 266

Fig. A1. The link coordinate systems for PUMA560 robot [14, 15].

As shown in Fig. A1, the homogeneous transformation matrix of D-H representation for each link of PUMA560 are given as follows:

$$^{Base}A_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & b \\ 0 & 0 & 0 & 1 \end{bmatrix},\ ^0A_1 = \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & b \\ 0 & 0 & 0 & 1 \end{bmatrix},\ ^1A_2 = \begin{bmatrix} C_2 & -S_2 & 0 & a_2C_2 \\ S_2 & C_2 & 0 & a_2S_2 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$^2A_3 = \begin{bmatrix} C_3 & 0 & S_3 & a_3C_3 \\ S_3 & 0 & -C_3 & a_3S_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ ^3A_4 = \begin{bmatrix} C_4 & 0 & -S_4 & 0 \\ S_4 & 0 & C_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ ^4A_5 = \begin{bmatrix} C_5 & 0 & S_5 & 0 \\ S_5 & 0 & -C_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$^5A_6 = \begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Given the above transformation matrix, the local joint coordinate system with respect to the base for each link can be compute as:

$$T_0 = {}^{base}A_0 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & b \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$T_1 = {}^{Base}A_0\,{}^0A_1 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & b \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$T_2 = {}^{Base}A_0\,{}^0A_1\,{}^1A_2 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C_1C_2 & -C_1S_2 & -S_1 & a_2C_1C_2 - d_2S_1 \\ S_1C_2 & -S_1S_2 & C_1 & a_2S_1C_2 + d_2C_1 \\ -S_2 & -C_2 & 0 & -a_2S_2 + b \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_3 = {}^{Base}A_0\,{}^0A_1\,{}^1A_2\,{}^2A_3 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C_1C_{23} & -S_1 & C_1S_{23} & C_1(a_2C_2 + a_3C_{23}) - d_2S_1 \\ S_1C_{23} & C_1 & S_1S_{23} & S_1(a_2C_2 + a_3C_{23}) + d_2C_1 \\ -S_2 & -C_2 & 0 & -a_2S_2 - a_3S_{23} + b \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_4 = {}^{Base}A_0\,{}^0A_1\,{}^1A_2\,{}^2A_3\,{}^3A_4 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} C_1C_{23}C_4 - S_1S_4 & -C_1S_{23} & -S_1C_4 - C_1C_{23}S_4 & C_1(a_2C_2 + a_3C_{23} + d_4S_{23}) - d_2S_1 \\ S_1C_{23}C_4 + C_1S_4 & -S_1S_{23} & C_1C_4 - S_1C_{23}S_4 & S_1(a_2C_2 + a_3C_{23} + d_4S_{23}) + d_2C_1 \\ -C_4S_{23} & -C_{23} & S_{23}S_4 & -a_2S_2 - a_3S_{23} + b + d_4C_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_5 = {}^{Base}A_0\,{}^{0}A_1\,{}^{1}A_2\,{}^{2}A_3\,{}^{3}A_4\,{}^{4}A_5 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned}
n_x &= C_1(C_{23}C_4C_5 - S_{23}S_5) - S_1S_4C_5 \\
n_y &= S_1(C_{23}C_4C_5 - S_{23}S_5) + C_1S_4C_5 \\
n_z &= -S_{23}C_4C_5 - C_{23}S_5 \\
o_x &= -S_1C_4 - C_1C_{23}S_4 \\
o_y &= C_1C_4 - S_1C_{23}S_4 \\
o_z &= S_{23}S_4 \\
a_x &= C_1(C_{23}C_4S_5 + S_{23}C_5) - S_1S_4S_5 \\
a_y &= S_1(C_{23}C_4S_5 + S_{23}C_5) + C_1S_4S_5 \\
a_z &= -S_{23}C_4S_5 + C_{23}C_5 \\
p_x &= C_1(C_2a_2 + C_{23}a_3 + S_{23}d_4) - S_1d_2 \\
p_y &= S_1(C_2a_2 + C_{23}a_3 + S_{23}d_4) + C_1d_2 \\
p_z &= -S_2a_2 - S_{23}a_3 + C_{23}d_4 + b
\end{aligned}$$

$$T_6 = {}^{Base}A_0\,{}^{0}A_1\,{}^{1}A_2\,{}^{2}A_3\,{}^{3}A_4\,{}^{4}A_5\,{}^{5}A_6 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned}
n_x &= C_1(C_{23}(C_4C_5C_6 - S_4S_6) - S_{23}S_5C_6) - S_1(S_4C_5C_6 + C_4S_6) \\
n_y &= S_1(C_{23}(C_4C_5C_6 - S_4S_6) - S_{23}S_5C_6) + C_1(S_4C_5C_6 + C_4S_6) \\
n_z &= -S_{23}(C_4C_5C_6 - S_4S_6) - C_{23}S_5C_6 \\
o_x &= -C_1(C_{23}(C_4C_5S_6 + S_4C_6) - S_{23}S_5S_6) + S_1(S_4C_5S_6 - C_4C_6) \\
o_y &= -S_1(C_{23}(C_4C_5S_6 + S_4C_6) - S_{23}S_5S_6) - C_1(S_4C_5S_6 - C_4C_6) \\
o_z &= S_{23}(C_4C_5S_6 + S_4C_6) + C_{23}S_5S_6 \\
a_x &= C_1(C_{23}C_4S_5 + S_{23}C_5) - S_1S_4S_5 \\
a_y &= S_1(C_{23}C_4S_5 + S_{23}C_5) + C_1S_4S_5 \\
a_z &= -S_{23}C_4S_5 + C_{23}C_5 \\
p_x &= C_1(C_{23}(C_4S_5d_6 + a_3) + S_{23}(C_5d_6 + d_4) + C_2a_2) - S_1(S_4S_5d_6 + d_2) \\
p_y &= S_1(C_{23}(C_4S_5d_6 + a_3) + S_{23}(C_5d_6 + d_4) + C_2a_2) + C_1(S_4S_5d_6 + d_2) \\
p_z &= -S_{23}(C_4S_5d_6 + a_3) + C_{23}(C_5d_6 + d_4) - S_2a_2 + b
\end{aligned}$$

APPENDIX B. Ellipsoid Modeling for Links of PUMA560

This appendix describes the detailed calculations in link-by-link modeling of PUMA560 with the enclosing and enclosed ellipsoids. In the following, v_i, f_i denote the vertex and face of polyhedron, respectively. The unit of length is *mm*. All computations are performed by Matlab.

Base

Base is a cylinder. It is first approximated by a polyhedron. The set of vertices, which are with respect to the local frame, and the facets of the octagon are given as:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \\ v_{10} \\ v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \\ v_{15} \\ v_{16} \end{bmatrix} = \begin{bmatrix} 80.0000 & 0.0000 & 0.0000 \\ 56.5685 & 56.5685 & 0.0000 \\ 0.00000 & 80.0000 & 0.0000 \\ -56.5685 & 56.5685 & 0.0000 \\ -8.0000 & 0.0000 & 0.0000 \\ -55.5685 & -56.5685 & 0.0000 \\ 0.0000 & -80.0000 & 0.0000 \\ 56.5685 & -56.5685 & 0.0000 \\ 80.0000 & 0.0000 & -660.4000 \\ 56.5685 & 56.5685 & -660.4000 \\ 0.0000 & 80.0000 & -660.4000 \\ -56.5685 & 56.5685 & -660.4000 \\ -80.0000 & 0.0000 & -660.4000 \\ -56.5685 & -56.5685 & -660.4000 \\ 0.0000 & -80.0000 & -660.4000 \\ 56.5685 & -56.5685 & -660.4000 \end{bmatrix} \quad \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9 \\ f_{10} \end{bmatrix} = \begin{bmatrix} v_1 & v_8 & v_7 & v_6 & v_5 & v_4 & v_3 & v_2 \\ v_1 & v_2 & v_{10} & v_9 & & & & \\ v_1 & v_9 & v_{16} & v_8 & & & & \\ v_2 & v_3 & v_{11} & v_{10} & & & & \\ v_3 & v_4 & v_{12} & v_{11} & & & & \\ v_4 & v_5 & v_{13} & v_{12} & & & & \\ v_5 & v_6 & v_{14} & v_{13} & & & & \\ v_6 & v_7 & v_{15} & v_{14} & & & & \\ v_7 & v_8 & v_{16} & v_{15} & & & & \\ v_9 & v_{10} & v_{11} & v_{12} & v_{13} & v_{14} & v_{15} & v_{16} \end{bmatrix}$$

According to the above data, the center coordinate and the characteristic matrix of the L-J ellipsoid for base are computed as:

$$c_{o\_base} = \begin{bmatrix} -0.0014 \\ 0.0183 \\ -330.2026 \end{bmatrix} \quad M_{base} = \begin{bmatrix} 0.0001034 & 0.0000000 & 0.0000000 \\ 0.0000000 & 0.0001034 & 0.0000000 \\ 0.0000000 & 0.0000000 & 0.0000031 \end{bmatrix}.$$

By using the L-J ellipsoid as the initial value, the center coordinate and the characteristic matrix of the enclosed ellipsoid of base are computed as:

$$c_{i\_base} = \begin{bmatrix} 0.0124 \\ 0.542 \\ -329.9760 \end{bmatrix} \quad N_{base} = \begin{bmatrix} 0.0001831 & -0.0000211 & 0.0000000 \\ -0.0000211 & 0.0003025 & 0.0000000 \\ 0.0000000 & 0.0000000 & 0.0000092 \end{bmatrix}.$$

The results are shown in Fig. B1.

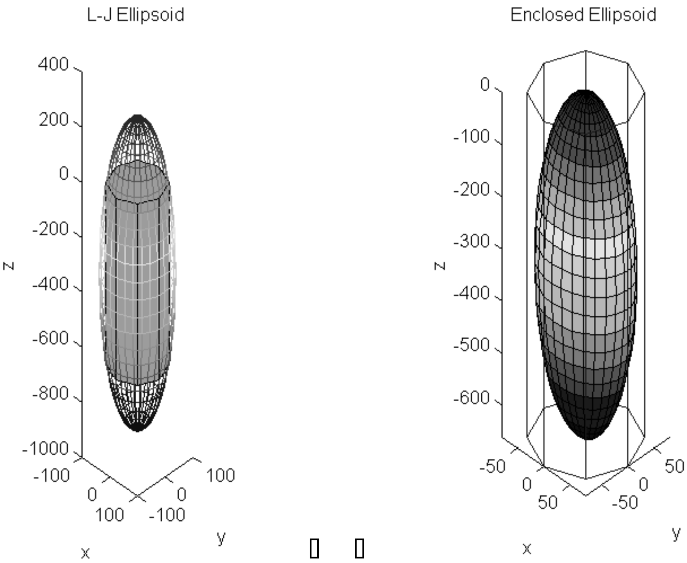


Fig. B1. The enclosing and enclosed ellipsoids of base.

Link 1

Link1 is also the cylindrical part of PUMA560, therefore the same means for L-J ellipsoid computation of PUMA560's base is applied. The set of vertices, which are with respect to the local frame, and the facets constructed the octagon are given as:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \\ v_{10} \\ v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \\ v_{15} \\ v_{16} \end{bmatrix} = \begin{bmatrix} 80.00000 & 0.0000 & -100.0000 \\ 56.5685 & 56.5685 & -100.0000 \\ 0.0000 & 80.0000 & -100.0000 \\ -56.5685 & 56.5685 & -100.0000 \\ -80.0000 & 0.0000 & -100.0000 \\ -56.5685 & -56.5685 & -100.0000 \\ 0.0000 & -80.0000 & -100.0000 \\ 56.5685 & -56.5685 & -100.0000 \\ 80.0000 & 0.0000 & 192.0900 \\ 56.5685 & 56.5685 & 192.0900 \\ 0.0000 & 80.0000 & 192.0900 \\ -56.5685 & 56.5685 & 192.0900 \\ -80.0000 & 0.0000 & 192.0900 \\ -56.5685 & -56.5685 & 192.0900 \\ 0.0000 & -80.0000 & 192.0900 \\ 56.5685 & -56.5685 & 192.0900 \end{bmatrix} \quad \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9 \\ f_{10} \end{bmatrix} = \begin{bmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 \\ v_1 & v_9 & v_{10} & v_2 & & & & \\ v_1 & v_8 & v_{16} & v_9 & & & & \\ v_2 & v_{10} & v_{11} & v_3 & & & & \\ v_3 & v_{11} & v_{12} & v_4 & & & & \\ v_4 & v_{12} & v_{13} & v_5 & & & & \\ v_5 & v_{13} & v_{14} & v_6 & & & & \\ v_6 & v_{14} & v_{15} & v_7 & & & & \\ v_7 & v_{15} & v_{16} & v_8 & & & & \\ v_9 & v_{16} & v_{15} & v_{14} & v_{13} & v_{12} & v_{11} & v_{10} \end{bmatrix}$$

According to the above data, the center coordinate and the characteristic matrix of the L-J ellipsoid, the enclosing ellipsoid, for PUMA560's link 1 are computed as:

$$c_{o\_link1} = \begin{bmatrix} 0.0015 \\ 0.0057 \\ 46.0371 \end{bmatrix}, \quad M_{link1} = \begin{bmatrix} 0.0001039 & -0.0000001 & 0.0000000 \\ -0.0000001 & 0.0001039 & 0.0000000 \\ 0.0000000 & 0.0000000 & 0.0000157 \end{bmatrix}.$$

By using the L-J ellipsoid as the initial value, the center coordinate and the characteristic matrix of the enclosed ellipsoid are computed as:

$$c_{i\_link1} = \begin{bmatrix} 0.0153 \\ -0.0667 \\ 45.9585 \end{bmatrix}, \quad N_{link1} = \begin{bmatrix} 0.0003107 & 0.0000000 & 0.0000000 \\ 0.0000000 & 0.0003101 & -0.0000001 \\ 0.0000000 & -0.0000001 & 0.0000469 \end{bmatrix}.$$

The geometrical relation of link1, enclosing ellipsoid, and enclosed ellipsoid are shown in Fig. B2.

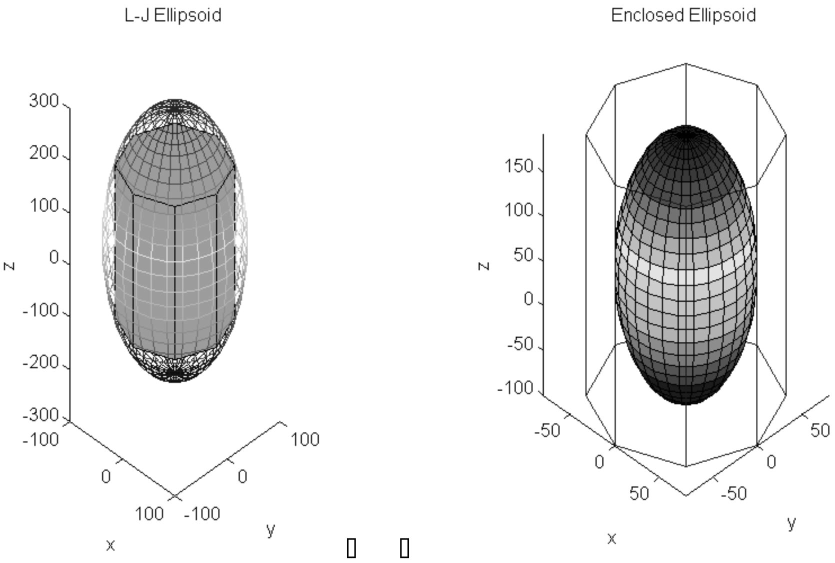


Fig. B2. The enclosing and enclosed ellipsoids of link 1.

Link 2

The geometric shape of PUMA560's link 2 is more complex with comparison with base and link 1. The two extremal ends of link 2 are cylindrical in shape. By applying the previous method used to model a cylinder, each of the two curved ends is first approximated with a union of four planar facets. The set of vertices, which are with respect to the local frame, and the facets constructed the polyhedron are given as:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \\ v_{10} \\ v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \\ v_{15} \\ v_{16} \\ v_{17} \\ v_{18} \\ v_{19} \\ v_{20} \\ v_{21} \\ v_{22} \\ v_{23} \\ v_{24} \end{bmatrix} = \begin{bmatrix} -559.8000 & -144.0000 & 43.0000 \\ -359.8000 & -144.0000 & 43.0000 \\ 12.0000 & -75.0000 & 43.0000 \\ 12.0000 & 75.0000 & 43.0000 \\ -359.8000 & 144.0000 & 43.0000 \\ -559.8000 & 144.0000 & 43.0000 \\ -610.7117 & 101.8234 & 43.0000 \\ -631.8000 & 0.0000 & 43.0000 \\ -610.7117 & -101.8234 & 43.0000 \\ 47.3553 & 53.0330 & 43.0000 \\ 62.0000 & 0.0000 & 43.0000 \\ 47.3553 & -53.0330 & 43.0000 \\ -559.8000 & -144.0000 & 153.0000 \\ -359.8000 & -144.0000 & 153.0000 \\ 12.0000 & -75.0000 & 153.0000 \\ 12.0000 & 75.0000 & 153.0000 \\ -359.8000 & 144.0000 & 153.0000 \\ -559.8000 & 144.0000 & 153.0000 \\ -610.7117 & 101.8234 & 153.0000 \\ -631.0000 & 0.0000 & 153.0000 \\ -610.7117 & -101.8234 & 153.0000 \\ 47.3553 & 53.0330 & 153.0000 \\ 62.0000 & 0.0000 & 153.0000 \\ 47.3553 & -53.0330 & 153.0000 \end{bmatrix}$$

$$\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9 \\ f_{10} \\ f_{11} \\ f_{12} \\ f_{13} \\ f_{14} \end{bmatrix} = \begin{bmatrix} v_1 & v_2 & v_3 & v_{12} & v_{11} & v_{10} & v_4 & v_5 & v_6 & v_7 & v_8 & v_9 \\ v_1 & v_{13} & v_{14} & v_2 & & & & & & & & \\ v_1 & v_9 & v_{21} & v_{13} & & & & & & & & \\ v_2 & v_{14} & v_{15} & v_3 & & & & & & & & \\ v_3 & v_{15} & v_{24} & v_{12} & & & & & & & & \\ v_4 & v_{16} & v_{17} & v_5 & & & & & & & & \\ v_4 & v_{10} & v_{22} & v_{16} & & & & & & & & \\ v_5 & v_{17} & v_{18} & v_6 & & & & & & & & \\ v_6 & v_{18} & v_{19} & v_7 & & & & & & & & \\ v_7 & v_{19} & v_{20} & v_8 & & & & & & & & \\ v_8 & v_{20} & v_{21} & v_9 & & & & & & & & \\ v_{10} & v_{11} & v_{23} & v_{22} & & & & & & & & \\ v_{11} & v_{12} & v_{24} & v_{23} & & & & & & & & \\ v_{13} & v_{21} & v_{20} & v_{19} & v_{18} & v_{17} & v_{16} & v_{22} & v_{23} & v_{24} & v_{15} & v_{14} \end{bmatrix}$$

According to the above data, the center coordinate and the characteristic matrix of the L-J ellipsoid of link 2 are computed as:

$$c_{o\_link2} = \begin{bmatrix} -345.4332 \\ 0.0120 \\ 97.9918 \end{bmatrix} \quad M_{link2} = \begin{bmatrix} 0.0000040 & 0.0000000 & 0.0000000 \\ 0.0000000 & 0.0000239 & 0.0000000 \\ 0.0000000 & 0.0000000 & 0.0001122 \end{bmatrix},$$

By using the L-J ellipsoid as the initial value, the center and the characteristic matrix of the enclosed ellipsoid are computed as:

$$c_{i\_link2} = \begin{bmatrix} -333.0522 \\ 1.3983 \\ 96.5187 \end{bmatrix} \quad N_{link2} = \begin{bmatrix} 0.0000120 & -0.0000006 & 0.0000024 \\ -0.0000006 & 0.0000772 & 0.0000016 \\ 0.0000024 & 0.0000016 & 0.0003497 \end{bmatrix}.$$

The results are shown in Fig. B3.

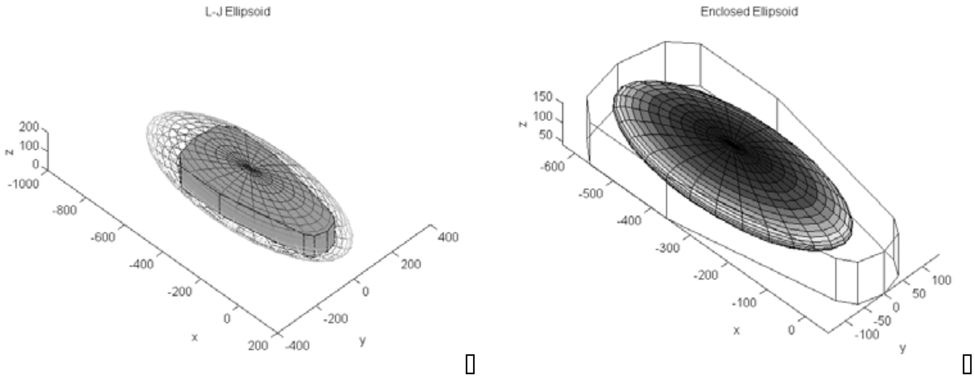


Fig. B3. The enclosing and enclosed ellipsoids of link2.

Link 3

The shape of PUMA560's link3 is similar to link 2 except that it only has one curved surface. As before, the cylindrical surface is approximated with a union of four small facets. The set of vertices, which are with respect to the local frame, and the facets constructed the polyhedron are given as:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \\ v_{10} \\ v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \end{bmatrix} = \begin{bmatrix} 85.0000 & -43.0000 & -80.0000 \\ 43.0000 & -43.0000 & 357.0700 \\ -43.0000 & -43.0000 & 357.0700 \\ -85.0000 & -43.0000 & -80.0000 \\ 60.1041 & -43.0000 & -110.0520 \\ 0.0000 & -43.0000 & -122.5000 \\ -60.1041 & -43.0000 & -110.0520 \\ 85.0000 & 43.0000 & -80.0000 \\ 43.0000 & 43.0000 & 357.0700 \\ -43.0000 & 43.0000 & 357.0700 \\ -85.0000 & 43.0000 & -80.0000 \\ 60.1041 & 43.0000 & -110.0520 \\ 0.0000 & 43.0000 & -122.5000 \\ -60.1041 & 43.0000 & -110.0520 \end{bmatrix} \quad \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9 \end{bmatrix} = \begin{bmatrix} v_1 & v_5 & v_6 & v_7 & v_4 & v_3 & v_2 \\ v_2 & v_3 & v_{10} & v_9 & & & \\ v_3 & v_4 & v_{11} & v_{10} & & & \\ v_4 & v_7 & v_{14} & v_{11} & & & \\ v_7 & v_6 & v_{13} & v_{14} & & & \\ v_6 & v_5 & v_{12} & v_{13} & & & \\ v_5 & v_1 & v_8 & v_{12} & & & \\ v_8 & v_9 & v_2 & v_1 & & & \\ v_8 & v_9 & v_{10} & v_{11} & v_{14} & v_{13} & v_{12} \end{bmatrix}$$

According to the above data, the center coordinate and the characteristic matrix of the L-J ellipsoid, the enclosing ellipsoid, for link 3 is computed as:

$$c_{o\_link\,3} = \begin{bmatrix} 0.0194 \\ -0.0301 \\ 87.1451 \end{bmatrix}, \quad M_{link3} = \begin{bmatrix} 0.0000676 & -0.0000001 & 0.0000000 \\ -0.0000001 & 0.0001798 & 0.0000000 \\ 0.0000000 & 0.0000000 & 0.0000079 \end{bmatrix}.$$

By using the L-J ellipsoid as the initial value, the center coordinate and the characteristic matrix of the enclosed ellipsoid are computed as:

$$c_{i\_link3} = \begin{bmatrix} -0.0173 \\ -0.0660 \\ 87.6122 \end{bmatrix}, \quad N_{link3} = \begin{bmatrix} 0.0002275 & -0.0000002 & -0.0000001 \\ -0.0000002 & 0.0005425 & 0.0000001 \\ -0.0000001 & 0.0000001 & 0.0000264 \end{bmatrix}.$$

The results are shown in Fig. B4.

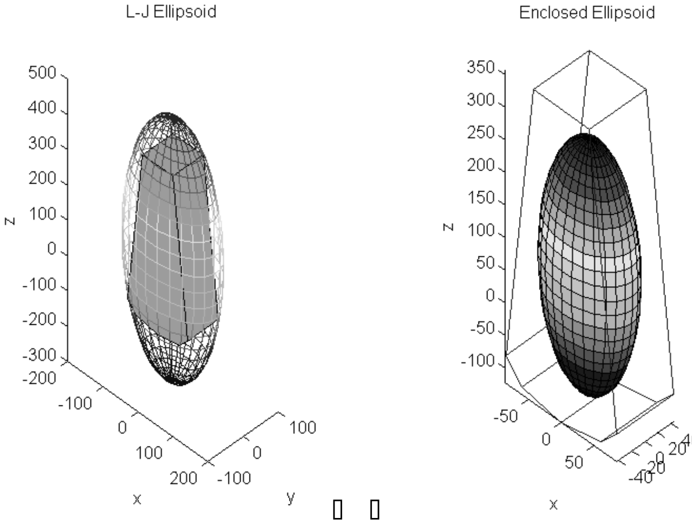


Fig. B4. The enclosing and enclosed ellipsoids of link 3.

Link 4 and Link 5

In our implementation, the link 4 and link 5 of PUMA560 are lumped together. This is due to that the rotation of the joint 5 will not affect the geometric shape. For convenience and the saving of computation time, the lumped links are approximated by a bounding box. The set of vertices, which are with respect to the local frame, and the facets of the bounding box are given as:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \end{bmatrix} = \begin{bmatrix} 43.0000 & 43.0000 & -40.0000 \\ 43.0000 & -43.0000 & -40.0000 \\ -43.0000 & -43.0000 & -40.0000 \\ -43.0000 & 43.0000 & -40.0000 \\ 43.0000 & 43.0000 & 76.0000 \\ 43.0000 & -43.0000 & 76.0000 \\ -43.0000 & -43.0000 & 76.0000 \\ -43.0000 & 43.0000 & 76.0000 \end{bmatrix} \qquad \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{bmatrix} = \begin{bmatrix} v_1 & v_4 & v_3 & v_2 \\ v_1 & v_2 & v_6 & v_5 \\ v_1 & v_5 & v_8 & v_4 \\ v_2 & v_3 & v_7 & v_6 \\ v_3 & v_4 & v_8 & v_7 \\ v_5 & v_6 & v_7 & v_8 \end{bmatrix}$$

According to the above data, the center coordinate and the characteristic matrix of the L-J ellipsoid of lumped link 4 and link 5 are computed as:

$$c_{o\_link45} = \begin{bmatrix} 0.0012 \\ -0.0001 \\ 17.9993 \end{bmatrix}, \; M_{link45} = \begin{bmatrix} 0.0001812 & 0.0000000 & 0.0000000 \\ 0.0000000 & 0.0001793 & 0.0000000 \\ 0.0000000 & 0.0000000 & 0.0000091 \end{bmatrix}.$$

By using the L-J ellipsoid as the initial value, the center coordinate and the characteristic matrix of the enclosed ellipsoid are computed as:

$$c_{i\_link45} = \begin{bmatrix} -0.8405 \\ -0.0001 \\ 17.9993 \end{bmatrix}, \; N_{link45} = \begin{bmatrix} 0.0005655 & 0.0000000 & 0.0000000 \\ 0.0000000 & 0.0005409 & 0.0000000 \\ 0.0000000 & 0.0000000 & 0.0003129 \end{bmatrix}.$$

The results are shown in Fig. B5.

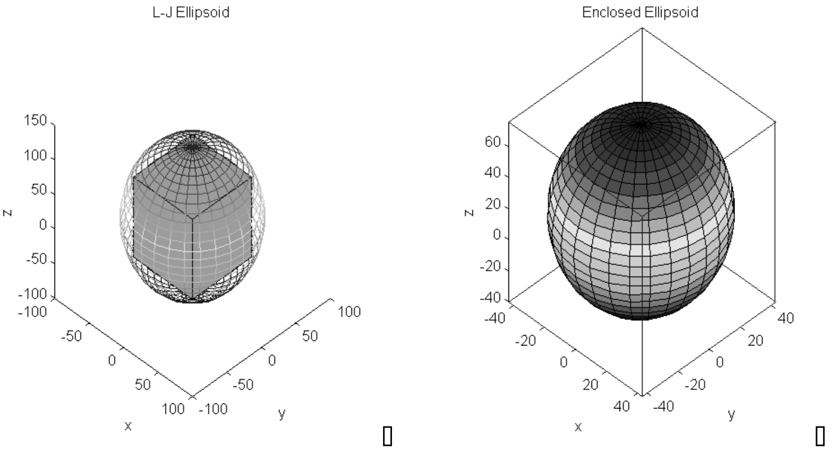


Fig. B5. The enclosing and enclosed ellipsoids of lumped link4 and link5.

Link 6

Link 6 of PUMA560 consists of two cylinders that have different radii. In order to reduce the modeling complexity and time consuming, the cylinder with small radius is enveloped by the large one to re-construct a single cylinder. Then, the same approach used to model the base and the link 1 is applied again.

The set of vertices, which are with respect to the local frame, and the facets constructed the polyhedron are given as:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \\ v_{10} \\ v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \\ v_{15} \\ v_{16} \end{bmatrix} = \begin{bmatrix} 25.0000 & 0.0000 & -16.2500 \\ 17.6777 & 17.6777 & -16.2500 \\ 0.0000 & 25.0000 & -16.2500 \\ -17.6777 & 17.6777 & -16.2500 \\ -25.0000 & 0.0000 & -16.2500 \\ -17.6777 & -17.6777 & -16.2500 \\ 0.0000 & -25.0000 & -16.2500 \\ 17.6777 & -17.6777 & -16.2500 \\ 25.0000 & 0.0000 & 0.0000 \\ 17.6777 & 17.6777 & 0.0000 \\ 0.0000 & 25.0000 & 0.0000 \\ -17.6777 & 17.6777 & 0.0000 \\ -25.0000 & 0.0000 & 0.0000 \\ -17.6777 & -17.6777 & 0.0000 \\ 0.0000 & -25.0000 & 0.0000 \\ 17.6777 & -17.6777 & 0.0000 \end{bmatrix} \quad \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9 \\ f_{10} \end{bmatrix} = \begin{bmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 \\ v_1 & v_9 & v_{10} & v_2 & & & & \\ v_1 & v_8 & v_{16} & v_9 & & & & \\ v_2 & v_{10} & v_{11} & v_3 & & & & \\ v_3 & v_{11} & v_{12} & v_4 & & & & \\ v_4 & v_{12} & v_{13} & v_5 & & & & \\ v_5 & v_{13} & v_{14} & v_6 & & & & \\ v_6 & v_{14} & v_{15} & v_7 & & & & \\ v_7 & v_{15} & v_{16} & v_8 & & & & \\ v_9 & v_{16} & v_{15} & v_{14} & v_{13} & v_{12} & v_{11} & v_{10} \end{bmatrix}$$

According to the above data, the center coordinate and the characteristic matrix of the L-J ellipsoid of link6 are computed as:

$$c_{o\_link6} = \begin{bmatrix} 0.0063 \\ -0.0004 \\ -8.1252 \end{bmatrix} \quad M_{link6} = \begin{bmatrix} 0.0011 & 0.0000 & 0.0000 \\ 0.0000 & 0.0011 & 0.0000 \\ 0.0000 & 0.0000 & 0.0050 \end{bmatrix},$$

By using the L-J ellipsoid as the initial value, the center coordinate and the characteristic matrix of the enclosed ellipsoid are computed as:

$$c_{i\_link6} = \begin{bmatrix} -0.6056 \\ 0.4153 \\ -8.3033 \end{bmatrix} \quad N_{link6} = \begin{bmatrix} 0.0026 & 0.0007 & -0.0002 \\ 0.0007 & 0.0031 & 0.0001 \\ -0.0002 & 0.0001 & 0.0159 \end{bmatrix}.$$

The results are shown in Fig. B6.

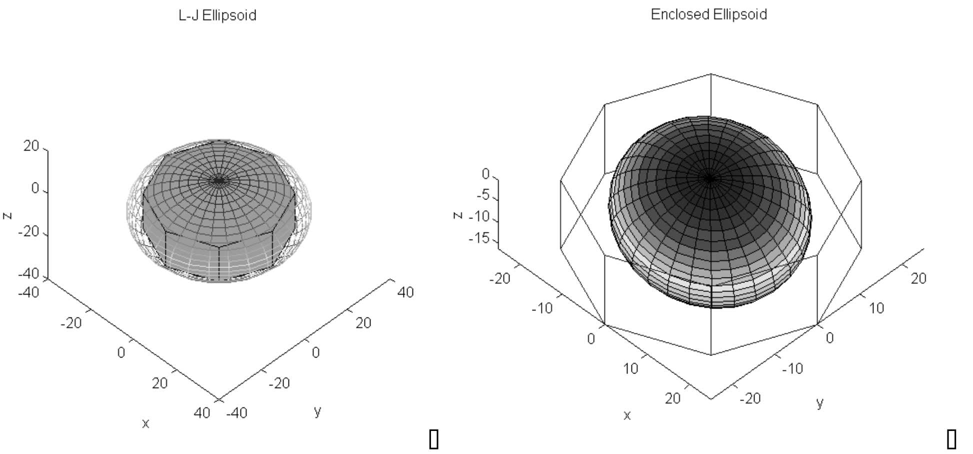


Fig. B6. The enclosing and enclosed ellipsoids of link6.

REFERENCES

[1] F. Dai, "Collision-free motion of an articulated kinematic chain in a dynamic environment," *IEEE Computer Graphics and Applications*, vol.1, pp.70-74, Jan. 1989.

[2] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi, "A fast procedure for computing the distance between complex objects in three dimensional space," *IEEE Trans. Robot. Automat.*, Vol. 4, pp. 193-203, April 1988.

[3] C.J. Wu, "On the representation and collision detection of robots," *Journal of Intelligent and Robotic Systems*, vol. 16, pp 151-168. 1996.

[4] R.A. Basta, R.Mehrotra, and M.R. Varanasi, "Collision detection for planning collision-free motion of two robot arms," *Proceedings of the IEEE Int. Conference on Robotics and Automation*, pp. 638-640, April 1988.

[5] J. Tornero, J. Hamlin, and R.B. Kelley, "Spherical-object representation and fast distance computation for robotic applications," *Proceedings of the IEEE Conference on Robotics and Automation*, pp. 1602-1608, April 1991.

[6] S. Bonner and R. B. Kelley, "A novel representation for planning 3-D collision free path," *IEEE Trans. Syst. Man. and Cybernet.*, Vol. 20, pp. 1337-1352, Nov.-Dec., 1990.

[7] A. P. del Pobil, M. A. Serna, and J. Llovet, "A new representation for collision avoidance and detection," *Proceedings of the IEEE Conference on Robotics and Automation*, pp. 246-251, Nice, France, May 1992.

[8] E. Rimon and S. P. Boyd, "Obstacle collision detection using best ellipsoid fit," *Journal of Intelligent and Robotic System*, Vol. 18, pp. 105–126, 1997.

[9] S. P. Shiang, J. S. Liu, and Y. R. Chien, "Estimate of minimum distance between convex polyhedra based on enclosed ellipsoids," *Proceedings of the IEEE/RSJ Int. Conference on Intelligent Robots and Systems*, 2000.

[10] M. Y. Ju, J. S. Liu, S. P. Shiang, Y. R. Chien, K. S. Hwang, and W. C. Lee, "Fast and accurate collision detection based on enclosed ellipsoid," submitted to 2000 *International Conference on Control, Automation, Robotics and Vision*, 2000.

[11] D. E. Johnson and E. Cohen," A framework for efficient minimum distance computations," *Proceedings of the IEEE Conference on Robotics and Automation*, pp. 3678-3684, Leuven, Belgium, May 1998.

[12] M. Grotschel, L. Lovasz, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, 2nd corrected ed., Springer-Verlag, Berlin, 1993.

[13] T. Brychcy and M. Kinder, " A neural network inspired architecture for robot motion planning," *Proceedings of the International Conference on Engineering Applications of Artificial Neural Networks (EANN'95)*, pp. 103-109, Helsinki, Finland, Aug. 1995.

[14] K. S. Fu, R. C. Gonzalez and C. S. G. Lee, *Robotics: control, sensing, vision, and intelligence*, McGraw-Hill, Inc., Singapore, 1987.

[15] P. I. Corke and B. Armstrong-Hélouvry, “A search for consensus among model parameters reported for the PUMA 560 robot,” *Proceedings of the IEEE Conference on Robotics and Automation*, vol. 2, pp. 1608-1613, May 1994.

[16] F. Solina and R. Bajcsy, “Recovery of parametric models from range images: the case for superquadrics with global deformations,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 2, pp. 131-147, Feb. 1990.

[17] E. I. Agba, T. L. Wong, and M. Z. Huang, “Solid modeling of kinematic chains,” *Proceedings of International Conference on Advanced Robotics*, vol. 2, pp. 959-964, June 1991.

[18] S. I. Choi and B. K. Kim, “Obstacle avoidance control for redundant manipulators using collidability measure,” *Robotica*, vol. 8, pp. 143-151, 2000.

[19] R.G. Beaumont and R. M. Crowder, “Real-time collision avoidance in two-armed robotic systems,” *Computer-Aided Engineering Journal*, pp.233-240, Dec. 1991.

[20] T. Kimoto and Y. Yasuda, “Shape description and representation by ellipsoids,” *Signal Processing: Image Communication*, vol. 9, pp. 275-290, 1997.

[21] http://www.antenen.com/htdocs/frame.html