Robotica (2011) volume 29, pp. 1025–1035. © Cambridge University Press 2011
doi:10.1017/S0263574711000233

Developing contour surfaces of manipulators with specified dexterities

K. Y. Tsai*, P. J. Lin and H. Y. Yu

Department of Mechanical Engineering, National Taiwan University of Science and Technology, 43 Keelung Road, Section 4, Taipei 10672, Taiwan

(Received in Final Form: March 10, 2011. First published online: April 11, 2011)

SUMMARY

Dexterity and workspace are two of the most important design criteria in developing manipulators. This paper presents algorithms for developing contour surfaces with specified dexterities and evaluating the area of the surfaces or the volume of the enclosed regions. The obtained results can be utilized to evaluate the dexterity and the rate of change of dexterity. Any closed curve or surface can be used to determine the singularity-free workspace of a manipulator with better dexterity. The proposed algorithms can be employed to study the dexterity and singularity-free workspace of 3-DOF manipulators and 4-DOF redundant serial manipulators. The contour surfaces in some subspaces of 6-DOF manipulators can also be investigated.

KEYWORDS: Manipulator; Isotropy; Dexterity; Workspace; Singularity; Optimal design.

1. Introduction

Except for manipulators used in some special applications, dexterity is a very desirable property for almost any type of manipulator. Dexterity can be evaluated by several measures, in which the condition number is the most commonly used mathematical tool to evaluate local dexterity of a manipulator or search for the design with optimum dexterity.[1–4] If the condition number (or the reciprocal of the condition number) of a related matrix reaches the optimum value of unity, then a manipulator is at an isotropic configuration where the manipulator can be controlled equally well in all directions, and sensitivity in velocity and force analysis is at a minimum. The dexterity of a manipulator, in general, is evaluated using the global conditioning index (GCI) defined by $\int_W \frac{1}{\kappa} dW / \int_W dW$, where κ is the condition number at a particular point in workspace, and the denominator is the volume of the workspace.[5–8] The GCI that gives the mean value of $1/\kappa$ for all the points in workspace can be used to evaluate the average performance of a manipulator. This measure, however, cannot evaluate the rate of change of dexterity, so local dexterity might change significantly even for manipulators with good GCI. Furthermore, the GCI cannot determine the branch of the inverse kinematic solutions of a manipulator having better dexterity. For 2-DOF or 3-DOF manipulators, some researchers have used the GCI atlas to develop manipulators with better dexterity.[9,10] The global dexterity, the rate of change of dexterity, and the branch of solutions having better dexterity can be roughly estimated from the distribution of the contour curves in 2D subspaces.

The workspace is another important criterion in designing manipulators. It is desirable to develop manipulators with a larger singularity-free workspace. The index $1/\kappa$ is not zero for all the points inside the singularity-free workspace, but a relatively large subset of the workspace still has poor dexterity with $1/\kappa$ very close to zero. The GCI atlas can be employed to roughly determine the region with $1/\kappa$ larger than a specified value, so regions with better dexterity can be chosen for higher precision tasks. Therefore, the atlas of GCI is helpful in searching for the design with better dexterity or singularity-free workspace. However, some properties in 3D space (such as the gradient, the area of a surface, or the volume of a region) are difficult to evaluate using the atlas.

In this paper, algorithms based on the analytical solutions of the cubic are employed to develop contour surfaces with specified dexterities. Some related properties of dexterity (such as the dexterity for a particular family of contour surfaces or a particular branch of kinematic solutions, the rate of change of dexterity, the area of a boundary surface or the volume enclosed in the surface with desired dexterity, the shape of any contour surface, or the relative positions of two neighboring surfaces) can be estimated from the contour surfaces and some related data obtained in the developing process. Any surface in joint space with specified dexterity can be utilized to develop the boundary of a region in workspace with same dexterity. The algorithms can be employed to develop manipulators with optimum dexterity or singularity-free workspace. In theory, the methods can be extended to higher degree-of-freedom manipulators because the dexterity of 6-DOF or redundant manipulator can also be evaluated by the solutions of a cubic.[11]

2. Dexterity of a Manipulator

This section provides formulas for the maximum and minimum solutions of the 3rd degree characteristic equations of a positive semidefinite matrix. The solutions are then employed to develop contour surfaces with specified dexterities. The characteristic equation of matrix JJ^t can be arranged as

$$f(x) = x^3 + qx^2 + rx + s, \tag{1}$$

* Corresponding author. E-mail: kytsai@mail.ntust.edu.tw

https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press

where J is the $3 \times n$ Jacobian matrix with $n \geq 3$. Let $a_1 = q/3$, $a_2 = r/3$, $a_3 = s$, $E = a_2 - a_1^2$, $G = a_3 - 3a_1a_2 + 2a_1^3$, and $p = \frac{1}{2}(-G + i\sqrt{|G^2 + 4E^3|})$, then the maximum solution and the minimum solution of the equation can be obtained by ref. [11] as

$$x_{\max} = 2r^{1/3}\cos\left(\frac{\theta}{3}\right) - a_1, \tag{2}$$

$$x_{\min} = 2r^{1/3}\sin\left(\frac{7}{6}\pi + \frac{\theta}{3}\right) - a_1, \tag{3}$$

where r and θ denote the amplitude and the argument of p, respectively. In this work, the dexterity is evaluated by the reciprocal of the condition number, or

$$\lambda = \sqrt{x_{\min}/x_{\max}}. \tag{4}$$

A point is on a curve or surface with a specified λ if the following equation is satisfied:

$$\sqrt{x_{\min}} - \lambda\sqrt{x_{\max}} = 0. \tag{5}$$

The equation, the bisection method, and a grid-scanning technique are employed to develop contour surfaces in joint space, workspace, or orientation space with $\lambda = k/n$ for $k = 1, 2, 3, \ldots, n-1$, and the mod (modulus after division) function is utilized to count the number of desired points that can be developed between the two given points. The number of points between points A and B with λ_A and λ_B, respectively, can be determined by the following function:

$$h(\lambda_A, \lambda_B) \equiv |g(\lambda_A) - g(\lambda_B)| \tag{6}$$

with

$$g(\lambda) \equiv n\lambda - \mathrm{mod}(n\lambda, 1). \tag{7}$$

Note that $\mathrm{mod}(r, 1)$ yields the part after the decimal point of a real number r. Suppose that $\lambda_B > \lambda_A$, $n = 20$, and $h = 2$, then two points P_1 and P_2 with $\lambda_{P_1} = (g(\lambda_B - 1))/20$ and $\lambda_{P_2} = g(\lambda_B)/20$ can be obtained. For example, if $\lambda_B = 0.21$ and $\lambda_A = 0.12$, then $\lambda_{P_1} = 0.15$ and $\lambda_{P_2} = 0.2$ can be developed by the bisection method. First, points A and B are utilized to develop point P_1 with $\lambda_{P_1} > \lambda_A$. Then P_2 is developed from P_1 and B.

The dexterity of serial manipulators is independent of the first joint displacement q_1, so taking q_2–q_3 cross-section is a better choice for developing contour curves or surfaces. Figure 1 shows the cross section of a contour surface in joint space with $\lambda = \lambda_S$. The straight line specified by $q_2 = c$ intersects the curve at points Q_2, Q_3, Q_4, and Q_5. Points $Q_1 = (c, q_{3\min})$ and $Q_6 = (c, q_{3\max})$. The line segments in the region with $\lambda > \lambda_S$ are Q_1–Q_2, Q_3–Q_4, and Q_5–Q_6. Let $Q_1 = (c, q_{3\min})$ and $Q_{m_1} = (c, q_{3\max})$, then the desired segments can be determined by the following rules:

1. Segment Q_1–Q_2 is included if λ at Q_1 is larger than λ_S.
2. Segment Q_{m_1-1}–Q_{m_1} is included if λ at Q_{m_1} is larger than λ_S.

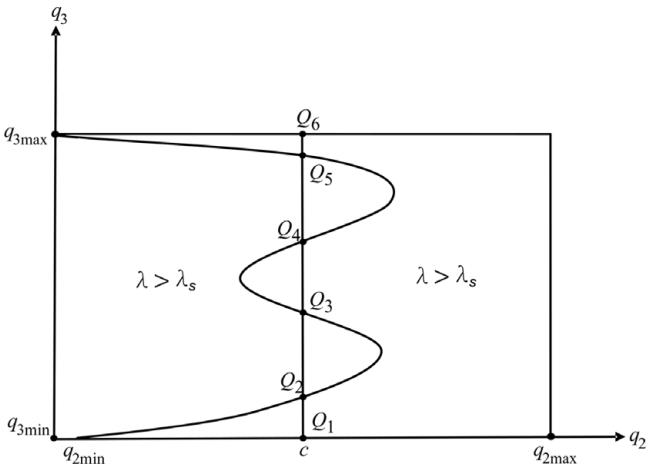


Fig. 1. Line segments in joint space with $\lambda \geq \lambda_S$.

3. For the rest of the segments, a segment is included if λ at the middle point of the segment is larger than λ_S.

If q_3 represents the displacement of a revolute joint and $q_{3\max} = q_{3\min} + 2\pi$, then both Q_1–Q_2 and Q_{m_1-1}–Q_{m_1} are included when Q_1 is larger than λ_S. In the scanning process, number m_1 and related data of points Q_1, $Q_2, \ldots, Q_{m_1}$ are recorded. If $m_1 = 2$ or $m_1 = 3$ (where the straight line just touches the curve at one point), then the whole segment from $q_{3\min}$ to $q_{3\max}$ is included if λ at Q_1 is larger than λ_S. For general cases ($m_1 \neq 2$ or $m_1 \neq 3$), the legitimate line segments are determined using the three proposed rules. The area dA_i of the region from q_2 to $q_2 + \Delta q_2$ can be approximated by the multiplication of Δq_2 with the summation of lengths of the line segments, the area of the region at a cross section specified by q_1 can be determined by the summation of all dA_i, and the summation of the multiplication of Δq_1 with the area of a cross section yields the volume of the region with $\lambda \geq \lambda_S$ in 3D space.

The steps in the scanning process can be summarized as follows:

1. Input maximum q_i, minimum q_i and Δq_i (for $i = 1, 2, 3$), λ_S, the number n_p of the desired points to be determined in one step, and the number n to obtain surfaces with specified dexterities.
2. From the starting point P_o, let $A = P_o$ and $B = A + \Delta q_3$.
3. Evaluate λ_A, λ_B, and function $h(\lambda_A, \lambda_B)$ in Eq. (6).
4. Repeat letting $\Delta q_3 = 2\Delta q_3$ if $h = 0$, or $\Delta q_3 = \frac{1}{2}\Delta q_3$ if $h > n_p$, and re-evaluating λ_B and h until condition $0 < h \leq n_p$ is satisfied.
5. Employ the bisection method to develop the desired points.
6. Save related data for points with $\lambda = \lambda_S$ and use them to compute dA_i.
7. Let $A = B$, $B = A + \Delta q_3$, and repeat steps 3–6 (and modify q_2 or q_1) until the scanning process is completed.
8. Output the volume of the region with $\lambda \geq \lambda_S$.

The procedure for developing surfaces with $\lambda = k/n$ for $k = 1, 2, 3, \ldots, n-1$ is shown in Fig. 2. It is easier to study the contour surfaces if surfaces with different λ are plotted on different coordinate frames. The searching process can be

https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press

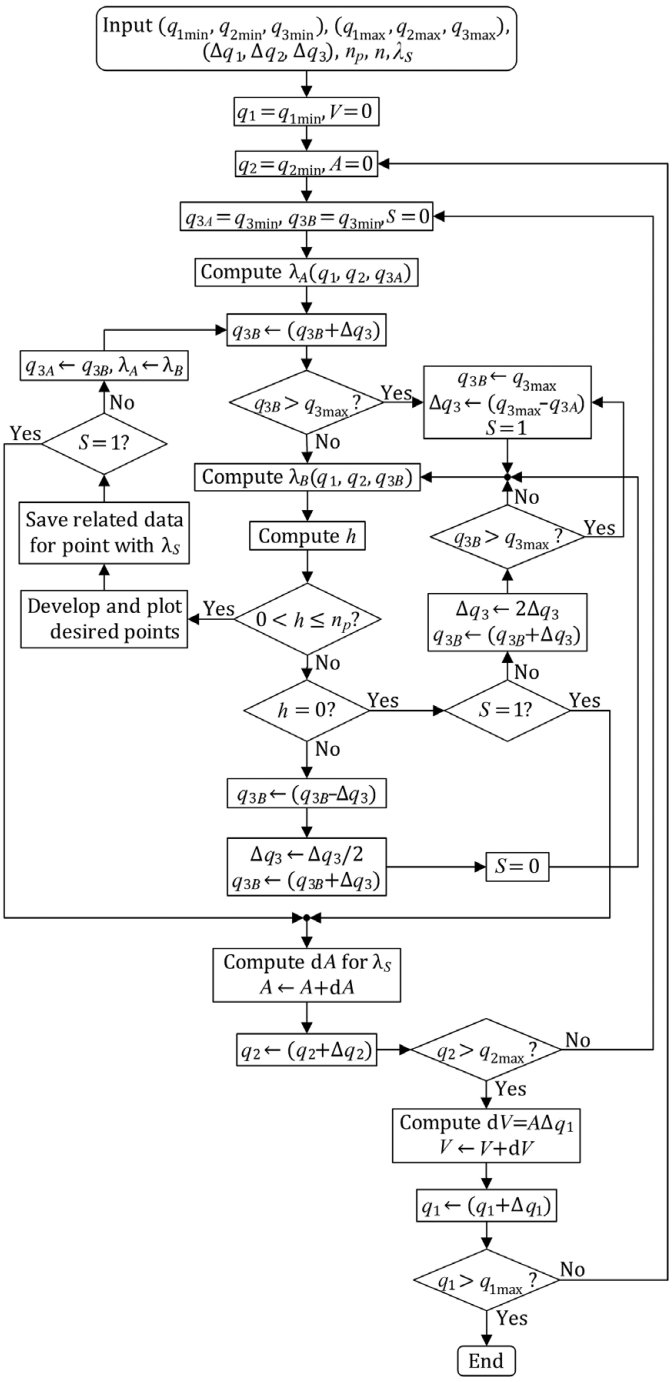


Fig. 2. Procedure for developing contour surfaces with specified dexterities.

improved if a larger step is chosen, but the results obtained might not be closed curves or surfaces. When two points with $\lambda_A \cong \lambda_B$ are used in the bisection method with a large step, function h yields zero but points between A and B might have λ that increases first and then decreases (or decreases first and then increases). Therefore, some desired points might not be obtained for a larger initial Δq_3 or a larger ratio of n_p/n because the algorithm keeps modifying Δq_3 to make it proportional to n_p/n. For parallel manipulators, it might be easier to develop contour surfaces in the task space. In this case, the points with illegitimate inverse kinematic solutions (for manipulators with limited joint ranges) should be excluded from the contour surfaces.

3. Dexterity of a Family of Contour Surfaces

This section presents methods to evaluate the dexterity of a particular family of contour surfaces or a particular branch of kinematic solutions. The algorithm in the previous section can be used to evaluate the dexterity if the ranges of the contour surfaces or the kinematic solutions are given. For serial or parallel manipulators whose inverse kinematic solutions can be obtained by solving quadratic equations, the range of each branch of inverse kinematic solutions can be determined using the factors from the Jacobian determinant.[12] The range for other types of manipulators, however, is very difficult to define.

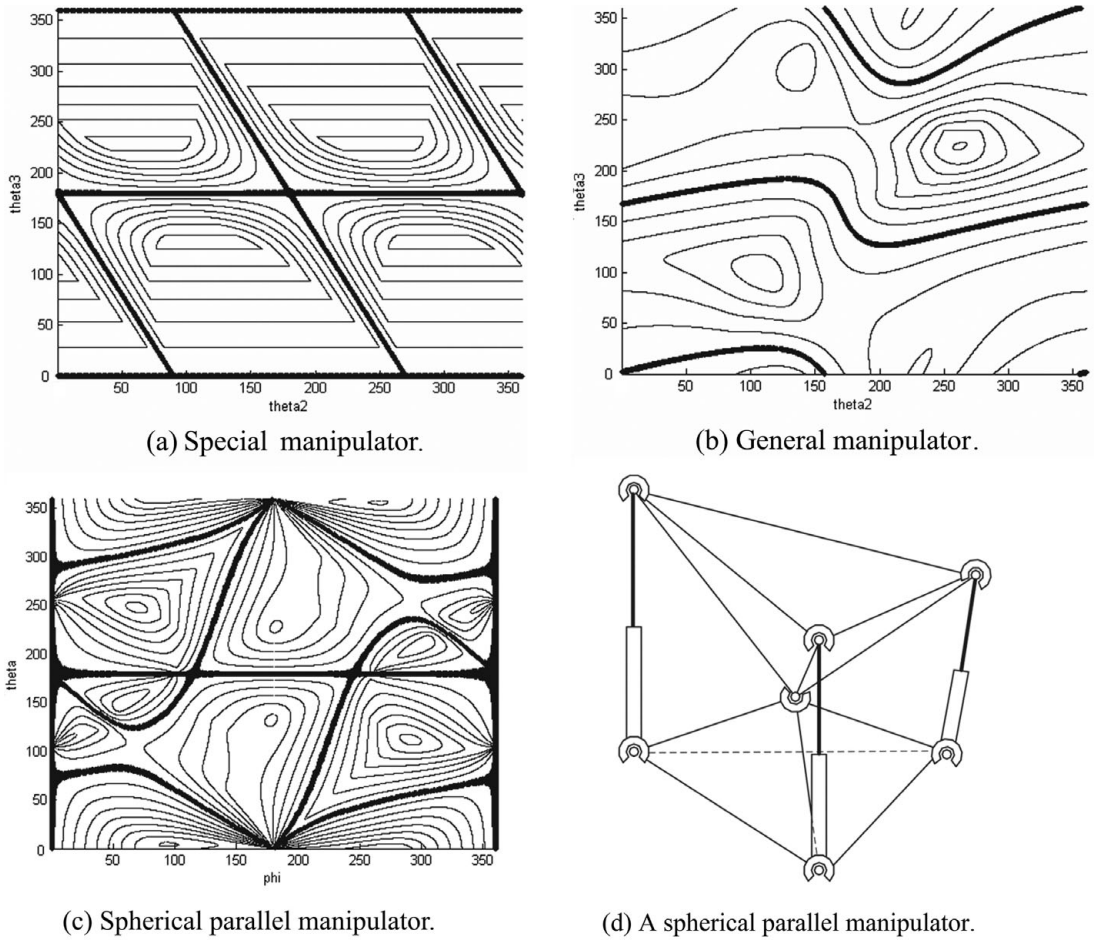
A study, by the first method, on the contour curves or surfaces of different types of 3-DOF manipulators shows that a particular family of contour curves or surfaces is surrounded by some closed curves or surfaces with lower dexterity. Figure 3 shows some contour curves of three different types of manipulators with $n = 10$, and the thick and black curves represent singular curves with $\lambda = 0$. Figure 3(a) shows that the contour curves with $\lambda \neq 0$ of the four branches of inverse kinematic solutions of a special 3R manipulator are simple closed curves surrounded by singular curves. For a general 3R manipulator, Fig. 3(b) shows the contour curves of the four branches of solutions (identified by the four local optimum of λ), which are surrounded by four closed curves with $\lambda = 0.3$. In Fig. 3(c), a cross section of contour surfaces (developed without considering the joint limits) in the orientation workspace of the spherical parallel manipulator (with three S–P–S chains) in Fig. 3(d) shows that there are several families of contour curves and each family has closed curves with $\lambda = 0.1$ as the boundary. The figure also shows that there is one local optimum of λ for each branch of solutions or each family of contour surfaces.

A closed curve with specified dexterity of $\lambda = \lambda_S$ on a cross section (with q_2 and q_3 as variables) can be developed using the bisection method. Figure 4(a) shows the steps to develop a closed curve with $\lambda = \lambda_S$. The point P_o can be any point belonging to a particular branch of kinematic solution or a point that can be used to locate the region where contour surfaces are to be developed. Point P_o can be obtained by finding two points A_o and B_o with $\lambda_{A_o} < \lambda_S < \lambda_{B_o}$. To avoid getting two points from different families of contour curves, the gradient (approximated by $[\,\Delta\lambda/\Delta q_2 \;\; \Delta\lambda/\Delta q_3\,]^t$) at the first point A_o in Fig. 4(a) can be utilized to find the second point. Search for point B_o in the direction of the gradient if $\lambda_{A_0} < \lambda_S$, otherwise search in the opposite direction of the gradient. Next, the gradient at point P_o can be employed to find the second point P_1 on the curve. Let unit vector $\boldsymbol{n}_o = [n_{ox}, n_{oy}]^t$ represent the direction of the gradient. Then the associated unit tangent vector in the counter-clockwise direction can be defined as $\boldsymbol{t}_1 = [n_{oy}, -n_{ox}]^t$. With $\boldsymbol{n}_1 = \boldsymbol{n}_o$, point Q_1 on the tangent and points A_1 and B_1 on the normal can be determined by

$$\begin{aligned} Q_1 &= P_o + \Delta_1 t_1, \\ A_1 &= Q_1 - \Delta_2 n_1, \\ B_1 &= Q_1 + \Delta_2 n_1, \end{aligned} \tag{8}$$

where Δ_1 and Δ_2 denote the increments in the corresponding directions. If λ at point Q_1 is very close to λ_S (or

https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press



(a) Special manipulator. (b) General manipulator.

(c) Spherical parallel manipulator. (d) A spherical parallel manipulator.

Fig. 3. Contour curves of different types of manipulators.

$|\lambda_{Q_1} - \lambda_S| < \varepsilon$), then Q_1 is the second point P_1 on the curve. Otherwise use points Q_1 and B_1 first (if the curve is developed in the counter-clockwise direction) to find the second point P_1 on the curve, or use points Q_1 and A_1 to obtain the second point P_1 if λ_S is between λ_{Q_1} and λ_A. The other points on the curve can be developed using a similar approach except that the tangents can be approximated by $\boldsymbol{t_i} = \frac{P_{1-1} - P_{1-2}}{\|P_{1-1} - P_{1-2}\|} = [t_{ix}, t_{iy}]^t$, and $\boldsymbol{n}_i$ can be directly obtained from $\boldsymbol{t}_i$ for $i \geq 2$. Point P_i, however, may not be on the normal through Q_i when the curve reaches points (especially for the points on a curve with larger λ) with relatively large curvature. Figure 4(b) shows that the normal does not intersect the curve, so P_i cannot be obtained by the bisection method. In this case, the straight line defined by vector $\boldsymbol{t_i}$ and point $H_i = Q_i + \Delta[\begin{smallmatrix} -t_{iy} \\ t_{ix} \end{smallmatrix}]$, instead of $\boldsymbol{n}_i$ and point Q_i, is employed to develop P_i. A closed curve can be obtained if the path returns to the starting point. The coordinates and the number of points obtained are recorded for computing the length of the boundary curve and the area of the enclosed region. The area of the region enclosed by the sequence of points (q_{2i}, q_{3i}) for $i = 1, 2, \ldots, m$ on the boundary can be determined by

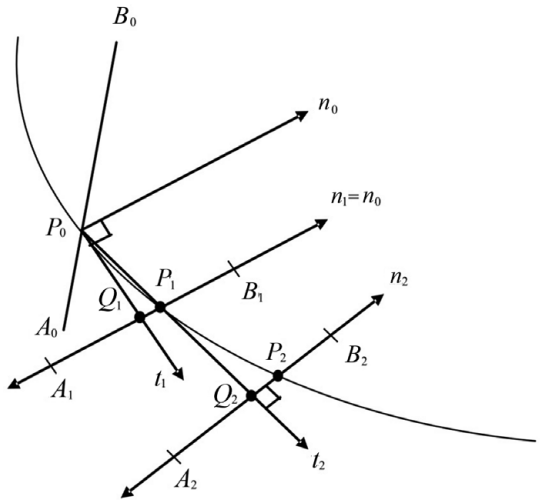
$$A = \frac{1}{2}\begin{vmatrix} q_{21} & q_{22} & q_{23} & & q_{2(m-1)} & q_{2m} & q_{21} \\ \times & \times & & \cdots\cdots & & \times & \times \\ q_{31} & q_{32} & q_{33} & & q_{3(m-1)} & q_{3m} & q_{31} \end{vmatrix}$$

$$= \frac{1}{2}\|q_{21}q_{32} + q_{22}q_{33} + \cdots + q_{2(m-1)}q_{3m} + q_{2m}q_{31} - q_{22}q_{31} - q_{23}q_{32} \cdots - q_{2n}q_{3(m-1)} - q_{21}q_{3m}\| \quad (9)$$

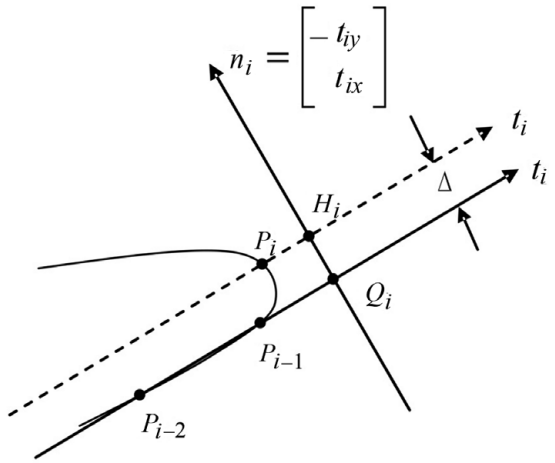
Let $N[1]$, $N[2], \ldots, N[m]$ denote the points on the first closed curve with $\lambda = \frac{k}{n}$. Then the coordinates of the center point, denoted as P_c, of the enclosed region can be approximated by $[\frac{\sum_i^m q_{2i}}{m}, \frac{\sum_i^m q_{3i}}{m}]$. If function h (in Eq. (6)) yields j for points $N[1]$ and P_c, then j points can be obtained and used as the starting points to develop closed curves with $\lambda = \frac{k+1}{n}, \frac{k+2}{n}, \ldots, \frac{k+j}{n}$. Next, find point P_c inside the closed curve with $\lambda = \frac{k+j}{n}$ and repeat the process until $h = 0$ or the number of points obtained (on a closed curve) is less than three. Compute the length of the boundary curve and the area of the enclosed region if the number of points obtained is $m \geq 3$.

With some modifications, the approach can be employed to develop contour curves of manipulators with limited joint ranges. On the closed curve with $\lambda = \lambda_S$ shown in Fig. 5, the curve from A to B and the curve from C to D can be replaced by line segments AB and CD, respectively, and the curve from E to G can be represented by line segments EF and FG. For example, if the curve reaches point E with $q_{2\max} = q_2$ (in the counter-clockwise direction), then related data (denoted as $N[k]$) is recorded. When the closed curve reaches point G with $q_{3\max} = q_3$, the coordinates of point F with $q_{2\max} = q_2$

https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press



(a)



(b)

Fig. 4. Tangent and normal vectors for developing a closed curve.

and $q_{3\,\max} = q_3$ can be determined. The coordinates of points F and G are saved in $N[k+1]$ and $N[k+2]$, respectively, or the segments EF and FG are divided into smaller segments and the related data is saved for developing the boundary surface of a region in task space with $\lambda \geq \lambda_S$.

Since points with λ close to zero might not be associated with any branch of kinematic solutions or any family of contour curves or surfaces, points with λ larger or equal to 0.1 and 0.3, respectively, are suggested as the starting point P_o for developing the outmost contour curve or surface of parallel and general 3-DOF manipulators. The λ for the outmost contour curve can be much closer to zero for special 3R manipulators. Contour curves with larger λ usually have shorter length and larger curvature, so the proposed method searches for points in the tangent direction (shown in Fig. 4(b)) first and uses a smaller step when developing contour curves with $\lambda \geq 0.5$. Some data of one cross section can be utilized to facilitate the developing process for the curves at the next cross section. For example, the coordinates of point P_o (or point $N[1]$) of one cross section can be employed to find the corresponding point on the next cross section, and the tangent vector at point P_o can be directly used

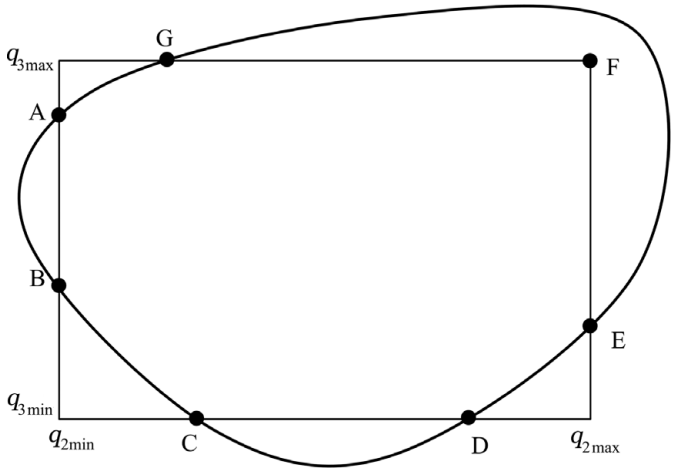


Fig. 5. Modifications of a closed curve considering joint limits.

as the tangent vector at point P_o of the next cross section (or the tangent vector at the first point of the next curve on the same cross section).

Let A_i denote the area of the 2D region on the ith cross section and s_i represents the length of its closed boundary curve. Then the volume of the 3D region V_j with $\lambda \geq j/n$ and the area of the boundary surface Ω_j with $\lambda = j/n$ (or $\lambda \geq j/n$ for manipulators with limited joint ranges) can be evaluated by

$$V_j = \sum_{i=1}^{nc-1} \frac{A_i + A_{i+1}}{2} \Delta q_1, \tag{10}$$

$$\Omega_j = \sum_{i=1}^{nc-1} \frac{(s_i + s_{i+1})}{2} \Delta q_1, \tag{11}$$

where nc and Δq_1 denote the number of cross sections taken and the distance between two neighboring sections, respectively. The steps to develop contour surfaces with $\lambda \geq \frac{k^*}{n}$ are summarized below:

1. Input numbers n and k^*. Let $k = k^*$, $S = 0$, and JL = "N". For manipulators with joint limits, input joint ranges and set JL = "Y."
2. Develop the first point $P_o = (q_{1o}, q_{2o}, q_{3o})$ with $\lambda = \frac{k}{n}$ (note that joint displacement q_{1o} must satisfy $q_{1\,\min} < q_{1o} < q_{1\,\max}$ if JL = "Y"). Let $q_1^* = q_{1o}$.
3. Use point P_o to develop the closed curve with $\lambda = \frac{k}{n}$. Modify the obtained coordinates if JL = "Y." If the number of points obtained $m < 3$, then go to Step 11.
4. Record the related data. Compute the length of the closed curve and the area of the enclosed region.
5. Find the center point P_c of the current closed curve and let $j = h(\lambda_{p_c}, \lambda_{N[1]})$. If $j = 0$, then go to Step 9.
6. Let count= $k + j$.
7. Repeat

- Let $k = k + 1$.
- Develop the closed curve with $\lambda = \frac{k}{n}$.
- Modify the obtained coordinates if JL = "Y."

https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press

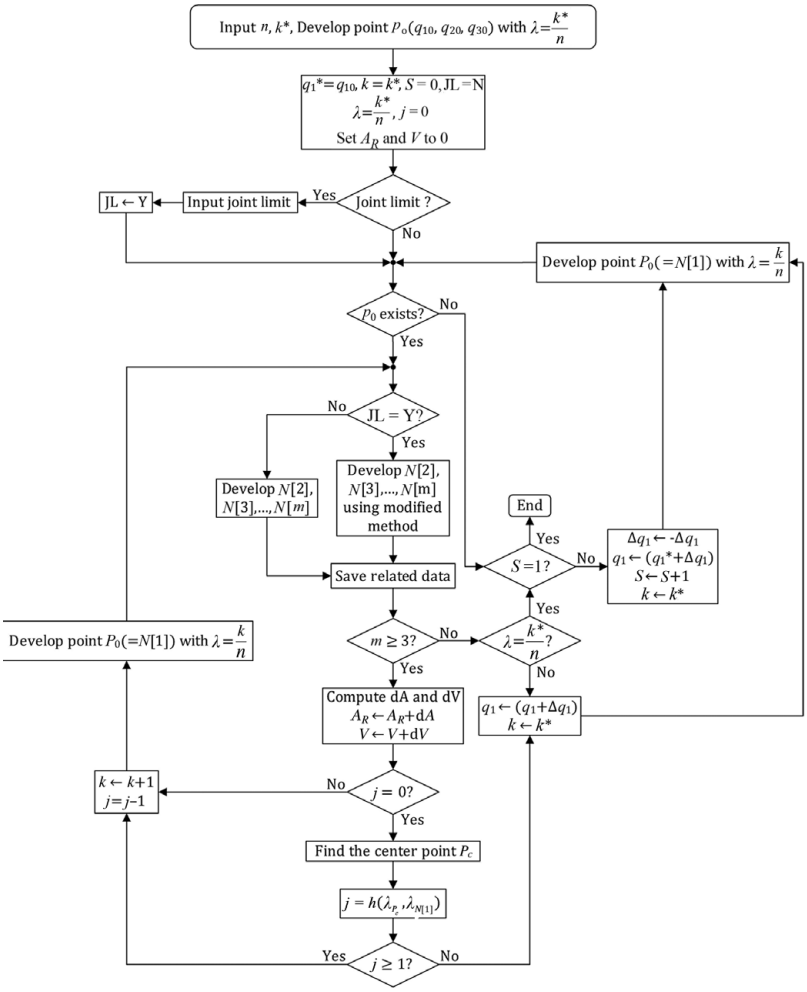


Fig. 6. Procedure for evaluating dexterity for a family of contour surfaces.

- If less than three points are obtained for any closed curve, then go to Step 9.
- Compute the length of the boundary curve and the area of the enclosed region, respectively, until count $= k$.

8. Go to Step 5.
9. Let $q_1 = q_1 + \Delta q_1$ and $k = k^*$. Develop the first point P_o with $\lambda = \frac{k}{n}$ on the cross section at q_1. Go to Step 11 if the point cannot be obtained.
10. Go to Step 3.
11. Exit if $S = 1$. $\Delta q_1 = -\Delta q_1, q_1 = q_1^* + \Delta q_1, S = S + 1$, $k = k^*$. Develop the first point P_o with $\lambda = \frac{k}{n}$ on the cross section at q_1. Exit if the point cannot be obtained.
12. Go to Step 3.

The procedure of this algorithm is shown in Fig. 6. For manipulators with limited joint ranges, center point P_c of a closed curve might not be close to a local optimum of λ, so the method might miss some closed curves near the local optimum. Better results can be obtained by searching (in the direction of the gradient at P_c) for a point with larger λ to replace P_c if $j = h(\lambda_{p_c}, \lambda_{N[1]}) = 0$.

For parallel manipulators, a particular family of contour surfaces is easier to develop using spherical coordinates with the origin at the local optimum of λ. Since many isotropic designs can be easily developed by existing methods, the method can be employed to determine which design has better dexterity. The three variables (in joint space, task space, or orientation space) in spherical coordinates can be defined by

$$\begin{aligned} q_1 &= q_1^* + \rho \sin\phi \cos\theta, \\ q_2 &= q_2^* + \rho \sin\phi \cos\theta, \\ q_3 &= q_3^* + \rho \cos\theta, \end{aligned} \tag{12}$$

where $0 < \phi < \pi$, $0 \leq \theta < 2\pi$, and $q^* \equiv (q_1^*, q_2^*, q_3^*)$ denote the point with $\lambda_{\max}$. When parameters ϕ and θ are given, points on the straight line (through the origin with ρ as the variable) with desired dexterities can be easily developed. If a point on the straight line with $\lambda = \frac{1}{n}$ is obtained, then points on the line with $\lambda = \frac{2}{n}, \frac{3}{n}, \ldots, \frac{m_2}{n}$ can be developed by the bisection method with the origin and the point with $\lambda = \frac{1}{n}$ as two starting points, where m_2 is the maximum integer satisfying $\frac{m_2}{n} < \lambda_{\max}$. Therefore, a family of contour surfaces with different dexterities can be developed using a 2D grid-scanning method through parameters ϕ and θ.

The area of the ith grid on the surface with a specified λ can be approximated by $A_{gi} = \rho_i^2 \Delta\phi_i \Delta\theta_i$, where ρ_i is the distance from the origin to the center of the grid. The area of the surface can be estimated by $\sum_{i=1}^{ng} A_{gi}$ with ng representing the number of grids. If each grid uses the same $\Delta\phi$ and $\Delta\theta$, then the area is proportional to $\sum_{i=1}^{ng} \rho_i^2$. Let

https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press

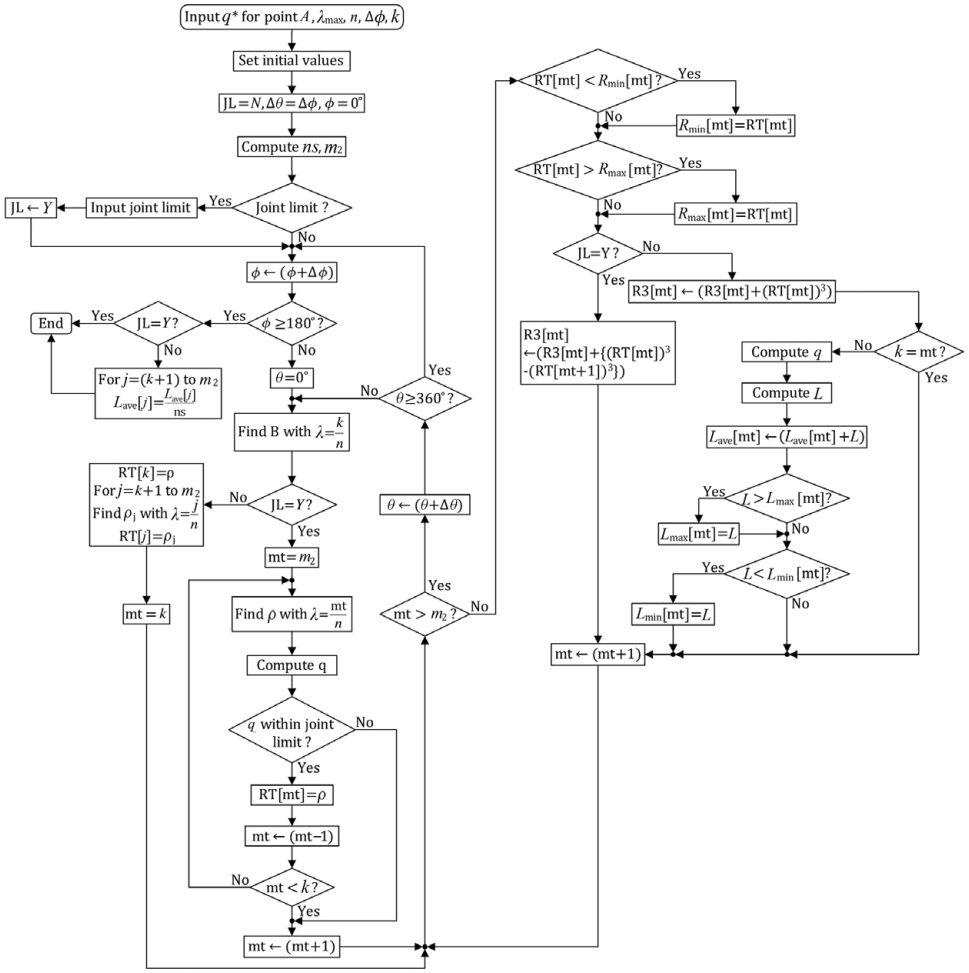


Fig. 7. Procedure for evaluating dexterity using spherical coordinates.

$\Delta\phi = \Delta\theta$. Then the volume of the region enclosed by the surface can be estimated by $\sum_{i=1}^{ng} \Lambda_i = \sum_{i=1}^{ng} \frac{1}{3}\rho_i^3(\Delta\phi)^2$, where Λ_i is the volume of a pyramid with the ith grid as the base and the origin as the vertex. Therefore, the volume of a region with dexterity better than the specified λ is proportional to $\sum_{i=1}^{ng} \rho_i^3$. Either $\sum_{i=1}^{ng} \rho_i^2$ or $\sum_{i=1}^{ng} \rho_i^3$ can be used to evaluate dexterity. Better dexterity and a larger singularity-free workspace can be expected for a larger $\sum_{i=1}^{ng} \rho_i^2$ or $\sum_{i=1}^{ng} \rho_i^3$. If a manipulator has joint limits, then ρ_i^2 and ρ_i^3 are excluded from the summations if any joint displacement exceeds its limits.

The distance from a point P_i with $\lambda = \frac{j+1}{n}$ to an associated point Q_i with $\lambda = \frac{j}{n}$ can be used to detect the relative position of the two boundary surfaces. Point Q_i is on the straight line defined by point P_i and the direction specified by the gradient (in 3D space) of λ at P_i. The point, which is at the opposite direction of the gradient, can be easily obtained using the bisection method. The average length L_{ave} of line segments of P_iQ_i can be employed to evaluate the average rate of change of dexterity, and the ratio of the minimum length L_{min} to the maximum length L_{max} can be utilized to check the relative position of two neighboring surfaces. It is desirable to have larger L_{ave} and smaller L_{min}/L_{max}.

With the local optimum as the origin and the outmost contour surface specified by $\lambda_{min} = k/n$, the steps for developing contour surfaces are given below:

(1) Input the origin (with λ_{max} and use it as the starting A), joint limits (if they exist), $\Delta\phi$, k, and n.
(2) Let $\Delta\theta = \Delta\phi$ and $m_2 = g(\lambda_{max})/n$ (where function g is defined in Eq. (7)). Search for ρ at the starting orientation with $\lambda = k/n$ and use it as the second point B.
(3) Develop points (with $\lambda = \frac{k+1}{n}, \frac{k+2}{n}, \ldots,$ and $\frac{m_2}{n}$) on the line segment between points A and B using the bisection method.
(4) Compute the distance from point A to each point obtained. If there is no joint limit, then save the related data and go to Step 6.
(5) Compute the joint displacements using Eq. (12) for the contour surfaces in joint space or solve the inverse kinematics for the contour surfaces in task space with each point obtained. Save the related data for points whose joint displacements are all within the related joint limits.
(6) Find the next (ϕ, θ) using a grid-scanning approach. Exit if all possible directions have been searched.
(7) Search for ρ at the new direction with $\lambda = \frac{k}{n}$ and use it as the second point B. Go to Step 3.

Since the points with $\phi = 0°$, $\phi = 180°$, or $\theta = 360°$ are excluded, the number of sample points reached for each λ is ns $= (\frac{180°}{\Delta\phi} - 1)(\frac{360°}{\Delta\theta})$. Figure 7 shows the procedure for the approach. In general, the steps on the left-hand

https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press

side input or set the initial data, compute ρ with different λ, and temporarily save the obtained results using array RT, which are then employed to update the recorded data using the steps on the right-hand side. $R3[j]$ records the summation of ρ_j^3 with ρ_j denoting the length associated with $\lambda = j/n$. The volume of the region with $\lambda \geq j/n$ is proportional to $R3[j]$. $R_{\min}[j]$ and $R_{\max}[j]$ record $\rho_{j\,\min}$ (the minimum ρ_j) and $\rho_{j\,\max}$ (the maximum ρ_j), respectively, for $j = k, k+1, \ldots, m_2$. This algorithm also investigates the relative positions of two neighboring surfaces. The maximum distance, the minimum distance, and the average distance between two neighboring surfaces with $\lambda = j/n$ and $\lambda = (j-1)/n$ are saved in $L_{\max}[j]$, $L_{\min}[j]$, and $L_{ave}[j]$, respectively for $j = k+1, k+2, \ldots, m_2$. The initial values for all RT$[j]$, $R_{\max}[j]$, $R3[j]$, $L_{\max}[j]$, and $L_{ave}[j]$ are set to zero, and a very large initial number is assigned to all $R_{\min}[j]$ and $L_{\min}[j]$. For manipulators with joint limits, $R3[j]$ records $(\rho_j^3 - \rho_{j+1}^3)$ for $j = k, k+1, \ldots, m_2$ with $\rho_{m_2+1} = 0$. The volume of the region with $\lambda \geq j/n$ is proportional to $\sum_{j=i}^{m_2} R3[i]$. The relative positions of two neighboring surfaces cannot be determined by a small number of sample points, so the related data are not evaluated for manipulators with joint limits. The process for determining the relative positions can be added to the previous two algorithms.

4. Numerical Examples

Example 1. Figures 8 and 9 show some contour surfaces, developed by the first algorithm with $n = 10$, of two 4-DOF redundant isotropic manipulators. The first manipulator with $a_1 = 4.3802$, $d_1 = 1.8892$, $\alpha_1 = 190.5131^\circ$, $a_2 = 4.7562$, $d_2 = 2.7281$, $\alpha_2 = 81.7690^\circ$, $a_3 = 2.154$, $d_3 = 4.5862$, $\alpha_3 = 190.4890^\circ$, and $a_4 = 2.8017$, $d_4 = 2.6529$, $\alpha_4 = 130.7402^\circ$ can reach only one isotropic configuration. The second special design manipulator (shown in Fig. 10), which has $a_1 = 2, a_2 = a_3 = 3.7, a_4 = 2.6, d_1 = d_2 = d_3 = d_4 = 0$, $\alpha_1 = 90^\circ$, and $\alpha_2 = \alpha_3 = \alpha_4 = 0$, can reach up to 16 isotropic configurations.[13] Figure 9(a) shows that there are four similar regions with $\lambda > 0.9$. Each region contains four isotropic configurations. The volumes of the regions with $\lambda \geq 0.7$ for the two designs are 0.6157 and 6.0882, respectively. The results show that the second design has much better dexterity. If $a_1 = 1$ for the second design, then the manipulator can reach only eight isotropic configurations. In this case, the volume for the region with $\lambda \geq 0.7$ is 4.4768. It is clear that isotropic manipulators with more isotropic configurations have better global dexterity.

Example 2. The second algorithm is employed to develop the singularity-free workspace with $\lambda \geq 0.1$ of a special 3R manipulator with link parameters $a_1 = 0, d_1 = 0, \alpha_1 = 90^\circ, a_2 = 5$, $d_2 = 0$, $\alpha_2 = 0$, $a_3 = 5$, $d_3 = 0$, $60^\circ \leq \theta_2 \leq 300^\circ$, and $30^\circ \leq \theta_3 \leq 330^\circ$. The manipulator is similar to the design given in Fig. 10 except that it only has the first three axes with $a_1 = 0$. Figure 11 shows the contour curves in joint space and workspace of four branches of the inverse kinematic solutions with $n = 10$. The areas of the regions in joint space with $\lambda \geq 0.1$ for the four branches of solutions are 56.9616, 123.2846, 56.9616, and 123.2846, respectively. This indicates that the second or fourth solution has better dexterity and larger singularity-free workspace (with $\lambda \geq 0.1$).

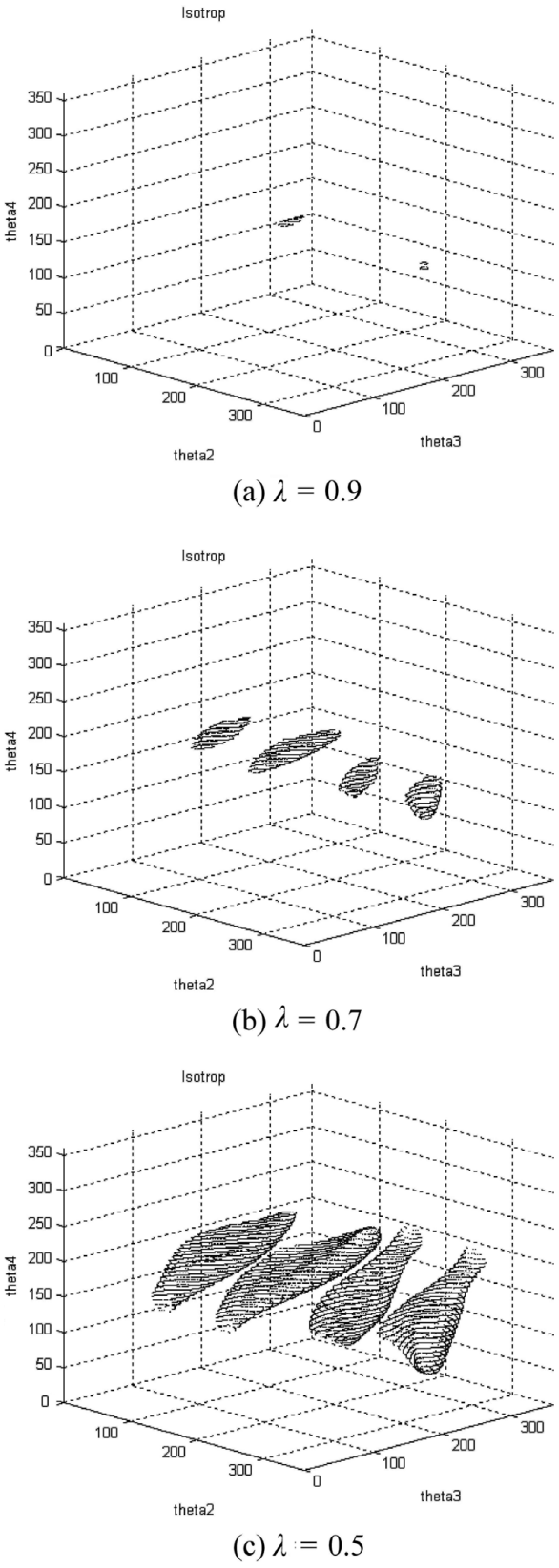
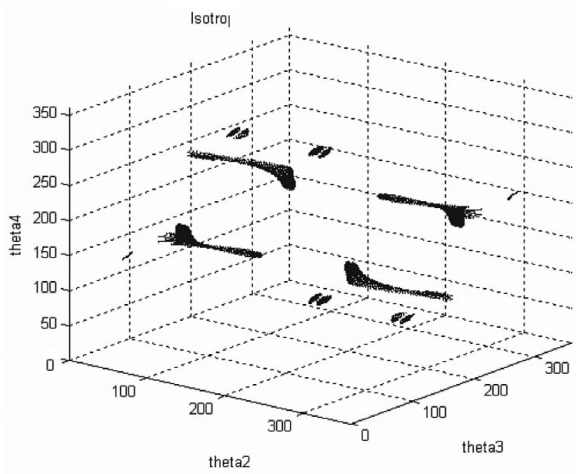


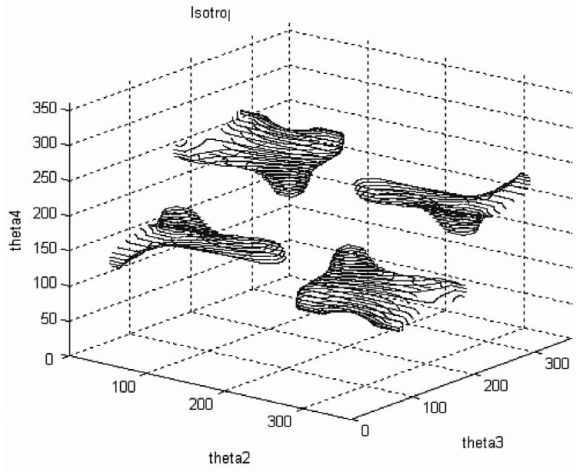
Fig. 8. Contour surfaces of a general 4-DOF isotropic redundant manipulator.

Example 3. Figure 12 and Table I show the parameters that define a group of 6-DOF Stewart–Gough parallel manipulators with the same local optimum dexterity at the initial assembly configuration.[14] The third algorithm is employed to find the design with the best dexterity.

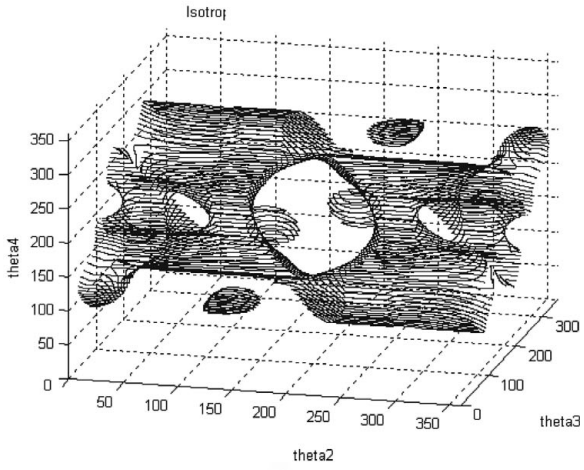
https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press



(a) $\lambda = 0.9$



(b) $\lambda = 0.7$



(c) $\lambda = 0.5$

Fig. 9. Contour surfaces of a special 4-DOF isotropic redundant manipulator.

Dexterity in the orientation space is investigated with limited joint ranges $\rho_{oi} - 6 \leq \rho_i \leq \rho_{oi} + 6$ for $i = 1, 2, \ldots, 6$, where ρ_i and ρ_{oi} denote, respectively, the ith limb length and the ith limb length at the initial assembly configuration. The orientation space is specified by Z–Y–Z Euler angles, and the 3×6 submatrix consisting of first three rows of the

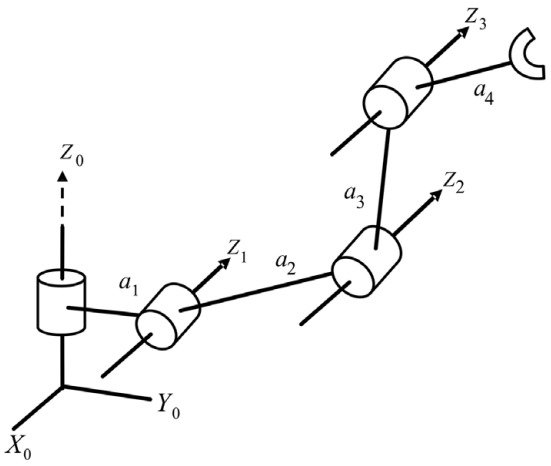
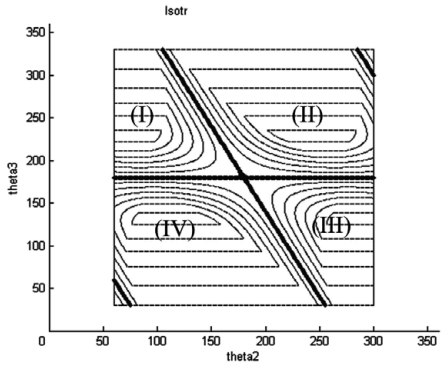
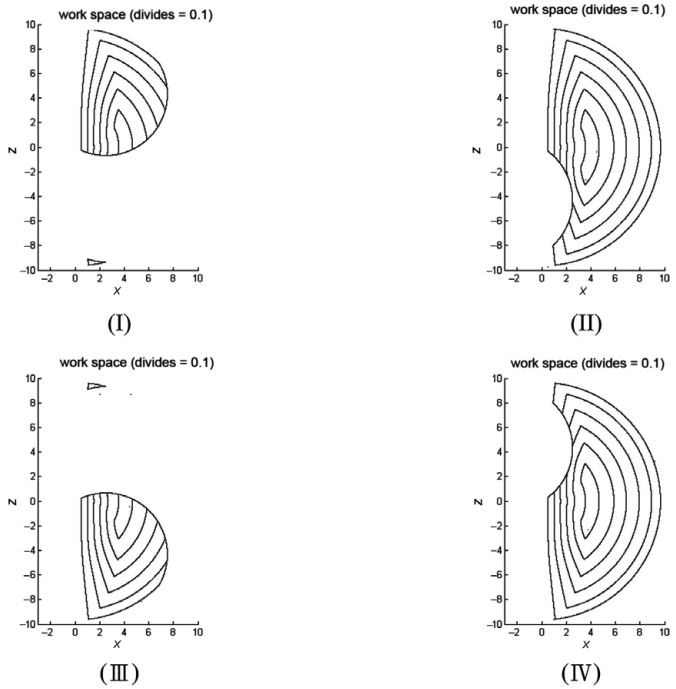


Fig. 10. A special 4R redundant manipulator.



(a) Joint space.



(b) Task space.

Fig. 11. Contour curves of four branches of the inverse kinematic solutions.

https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press

Table I. Parameters of symmetric generators for $H = 10$.

P	R	r	ϕ_1	ϕ_2
3	23.1759	11.9086	89.8116°	30.1884°
4	23.3348	11.7109	31.7976°	88.2024°
5	23.4837	11.5243	86.5247°	33.4753°
6	11.7109	23.3348	31.7977°	88.2023°
7	11.9085	23.1760	30.1892°	89.8108°
8	12.1172	23.0070	28.6476°	91.3524°
9	12.3375	22.8273	92.8282°	27.1718°
10	12.5692	22.6367	25.7663°	94.2337°
11	12.8101	22.4368	24.4195°	95.5805°
12	13.0634	22.2249	96.8592°	23.1408°
13	13.3284	22.0010	21.9280°	98.0720°
14	13.6054	21.7648	20.7801°	99.2199°
15	13.8946	21.5155	19.6968°	100.3032°

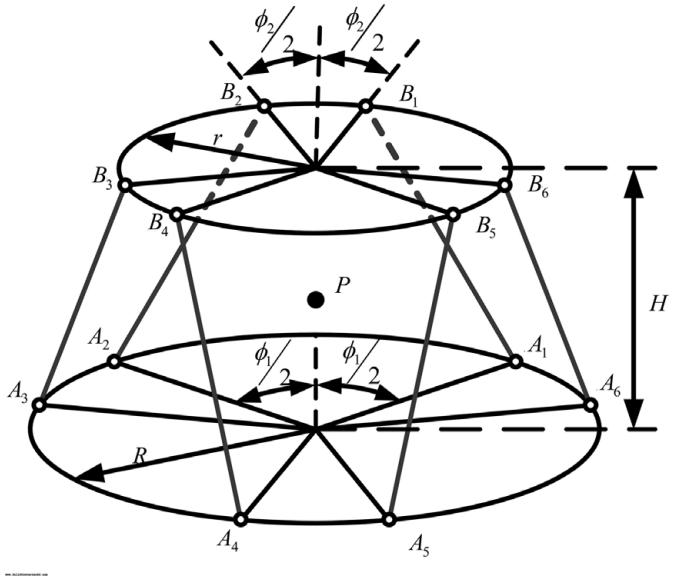


Fig. 12. Parameters for developing 6-DOF isotropic parallel manipulators.

Jacobian is employed to evaluate dexterity. With the origin of the coordinate frame at the tool center point P, $n = 10$ and $\Delta\phi = \Delta\theta = 6^\circ$, the results are listed in Table II. Some regions with lower dexterities cannot be reached because of the joint limits. The dexterity can be compared by V_k, which is proportional to the volume of the region with $\lambda \geq k/10$ for $k = 9, 8, \ldots, 5$. This shows that the design with $P = 5$ (the tool center point is near the geometric center of the manipulator) has the best dexterity and the largest singularity-free orientation workspace.

5. Conclusion

This paper evaluates kinematic properties related to the dexterity of manipulators, presenting three methods to develop all contour surfaces or a particular family of contour surfaces in the joint space or the workspace with specified dexterities. The dexterity can be evaluated by the volume of a closed region, and the rate of change of dexterity can be determined by the related positions of the contour surfaces. The methods are applicable to special, general, or parallel manipulators with or without joint limits. The methods were used to investigate the dexterity and workspace of a 3R serial manipulator and a group of 6-DOF isotropic parallel manipulators. The first method can also be employed to evaluate the dexterity of redundant manipulators. A study of 4R redundant manipulators showed that isotropic manipulators with more isotropic configurations have better global dexterity.

Table II. Related data for orientation workspace and dexterity.

$\sum_{i=1}^{ng} \rho_i^3$					
≥ 0.9	≥ 0.8	≥ 0.7	≥ 0.6	≥ 0.5	≥ 0.4
0.4992	Nil	Nil	Nil	Nil	Nil
1.9904	2.1254	Nil	Nil	Nil	Nil
1.3664	2.6812	2.8696	Nil	Nil	Nil
2.1102	2.4610	Nil	Nil	Nil	Nil
0.4774	0.7926	Nil	Nil	Nil	Nil
0.3397	0.5410	0.7896	1.2186	1.5960	Nil
0.2034	1.0818	1.4640	2.0265	2.1786	Nil
0.1560	0.8794	1.4062	1.8486	2.4345	Nil
0.1204	0.5708	1.2971	1.6910	2.3740	Nil
0.0817	0.4482	1.0311	1.2668	1.3294	Nil
0.0958	0.5007	1.2851	1.6433	1.7002	Nil
0.0888	0.4415	1.1796	1.6588	1.8296	Nil
0.0856	0.4140	1.0770	1.4467	1.5475	Nil

Acknowledgments

The authors would like to acknowledge the financial support from the National Science Council of the Republic of China under grant NSC97-2212-E011-051.

References

1. C. A. Klein and B. E. Blaho, "Dexterity measures for the design and control of kinematically redundant manipulators," *Int. J. Robot. Res.* **6**(2), 72–83 (1987).
2. J. Angeles and A. A. Rojas, "Manipulator inverse kinematics via condition number minimization and continuation," *Int. J. Robot. Autom.* **2**(2), 61–69 (1987).
3. C. Gosselin and J. Angeles, "The optimum kinematic design of a spherical three-degree-of-freedom parallel manipulator," *ASME J. Mech. Trans. Autom. Des.* **111**(2), 202–207 (1989).
4. C. M. Gosselin, "The optimum design of robotic manipulators using dexterity indexes," *Robot. Auton. Syst.* **9**, 213–226 (1992).
5. C. M. Gosselin and J. Angeles, "A global performance index for the kinematic optimization of robotic manipulators," *ASME J. Mech Des.* **113**(3), 220–226 (1991).
6. X.-J. Liu, J. Wang and G. Pritschow, "Performance atlases and optimum design of planar 5R symmetrical parallel mechanisms," *Mech. Mach. Theory* **41**(2), 119–144 (2006).
7. J. P. Merlet, "Jacobian, manipulability, condition number, and accuracy of parallel robots," *ASME J. Mech Des.* **128**(1), 199–208 (2006).
8. K. Miller, "Optimal design and modeling of spatial parallel manipulators," *Int. J. Robot. Res.* **23**(2), 127–140 (2004).
9. X.-J. Liu, Z.-L. Jin and F. Gao, "Optimum design of 3-DOF spherical parallel manipulators with respect to the conditioning

https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press

and stiffness indices," *Mech. Mach. Theory* **35**(9), 1257–1267 (2000).
10. X.-J. Liu, J. Wang and H.-J. Zheng, "Optimum design of the 5R symmetrical parallel manipulator with a surrounded and good-condition workspace," *Robot. Autom. Syst.* **54**(3), 221–233 (2006).
11. K. Y. Tsai and S. R. Zhou, "The optimum design of 6-DOF isotropic parallel manipulators," *J. Robot. Syst.* **22**(6), 333–340 (2005).
12. K. Y. Tsai, I. P. Hsu and D. Kohli, "Admissible motions of special manipulators," *IEEE Trans. Rob. Autom.* **10**(3), 386–391 (1994).
13. K. Y. Tsai, P. Y. Lin and T. K. Lee, "4R spatial and 5R parallel manipulators that can reach maximum number of isotropic positions," *Mech. Mach. Theory* **43**(1), 68–79 (2008).
14. K. Y. Tsai, T. K. Lee and Y. S. Jang, "A new class of isotropic generators for developing 6-DOF isotropic manipulators," *Robotica* **26**(5), 619–625 (2008).

https://doi.org/10.1017/S0263574711000233 Published online by Cambridge University Press