

Fault-Tolerant Path Planning of the Space Manipulator with Single Joint Failure Based on the Dexterity Space

Jia-Yi Tan¹, Gang Chen², and Yu-Qi Wang²

¹Beihang University, Beijing 100191, China

²Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract. To enable the space manipulator to complete the original task efficiently after any single joint fails, a fault-tolerant path planning method for the manipulator with single joint failure is proposed based on dexterity space in this paper. On the base of solving the degraded workspace, the dexterity space of the manipulator with single joint failure is established by constructing the dexterity index, and then the traditional A* algorithm is improved to complete fault-tolerant path planning in the dexterity space. The correctness and validity of fault-tolerant path planning based on improved A* algorithm are verified by simulating experiments with 7R manipulator.

1 Introduction

Due to the harshness of the space environment and the complexity of its own structure, joint locked failure occurs easily during the orbit operation for the space manipulator[1]. The degree of freedom of the space manipulator with the joint failure is reduced, which reduces the reachability of its terminal attitude, thereby causing the space operation task to fail to complete normally. Therefore, it is necessary to carry out fault-tolerant path planning with taking into account the reachability of the terminal attitude to make the space manipulator continue the task after joint locked failure occurs.

Aiming at fault-tolerant path planning for faulty manipulator, Khatib [2] et al. proposed an artificial potential field method, which make the proposed path smoother, but with local optimal problems still existing. Du[3] et al. combined the genetic algorithm with simulated annealing algorithm to propose a new algorithm. At present, many scholars utilize A* algorithm widely to search the shortest path in the workspace by selecting the appropriate valuation function[4]. Therefore, the A* algorithm is going to be utilized to carry out fault-tolerant path planning research of the manipulator in this paper.

Much path planning often requires perceive and analyze the environment and according to the performance of manipulator itself. Roberts[5] et al. carried out joint movement planning under the premise that there are high manipulability at the time of any joint failure based on the degraded manipulability. Zhao[6] et al. completed the algorithm design of fault-tolerant path planning with the assessment of fault tolerance performance based on the central degree. When the space

manipulator performs certain tasks, we expect to obtain a set of all possible gestures of the spatial position to ensure completing the task with high quality. Therefore, the dexterity index that represents the reachability of the terminal attitude[7] is going to be constructed in this paper. The traditional A* algorithm is improved by increasing the flexibility index in the cost function, and then fault-tolerant path planning is completed utilizing the improved A* algorithm.

In summary, aiming at the space manipulator with any joint locked failure, the dexterity index that represents the reachability of the terminal attitude is constructed by considering three posture parameters of the end of the space manipulator, and then the dexterity space is established on the base of solving the degraded workspace of the manipulator. The traditional A* algorithm is improved by increasing the dexterity index in the cost function, and then fault-tolerant path planning of the space manipulator with single joint failure is completed utilizing the improved A* algorithm. The dexterity index constructed in this paper realizes the characterization of the terminal attitude of the space manipulator with single joint failure. The trajectory searched based on the improved A* algorithm has meet the higher requirements of the terminal attitude for the manipulator and ensured that the manipulator can still complete the task.

2 The establishment of the dexterity space for the space manipulator

In this section, the dexterity space is constructed by considering three attitude parameters of the end of manipulator, and then the degraded workspace of the manipulator is solved based on the Monte Carlo method.

At the same time, the dexterity of each space point is calculated and the dexterity space of the space manipulator with single joint failure is established later.

2.1 Kinematic modeling of the manipulator with joint failure

The kinematic model of n DOF manipulator is established first before the establishment of the dexterity space, as shown in Figure 1.

Under normal condition, the symbols $L_i (i=1,2,\dots,n)$ and $J_i (i=1,2,\dots,n)$ represent the link i and joint i of the manipulator respectively and $\Sigma_i (i=0,2,\dots,n)$ represents the body coordinate system of L_i . As to joint angle sequence $\theta = \{\theta_1, \theta_2, \dots, \theta_{n-1}, \theta_n\}$, the positive kinematics equation of the manipulator is:

$${}^0T = {}^0T(\theta_1) {}^1T(\theta_2) \dots {}^{n-2}T(\theta_{n-1}) {}^{n-1}T(\theta_n) \quad (1)$$

When the joint J_i locked failure occurs, the link L_{i-1} and L_i are going to be connected to be a new link \tilde{L}_i . If the joints, links and coordinate systems of the manipulator are located before the locked joint, their symbols are unchanged; on the contrary, they only need to add the “~” above their symbols, and the body coordinate systems fixed to the connecting link \tilde{L}_i is Σ_{i-1} and $\tilde{\Sigma}_{i+1}$, as is shown in Figure 2.

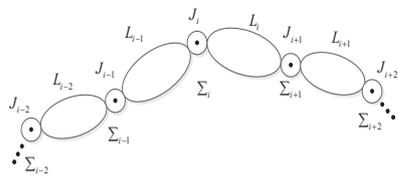


Figure 1. The symbols of links, joints and coordinate systems under normal condition.

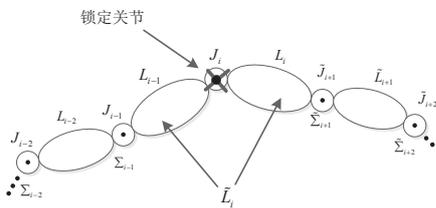


Figure 2. The symbols of links, joints and coordinate systems under fault condition.

2.2 The construction of the dexterity index for the faulty manipulator

The dexterity index which represents the reachability of the terminal attitude is going to be constructed by three attitude parameters of the manipulator.

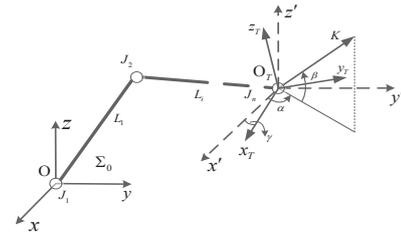


Figure 3. The diagram of solving dexterity.

Figure 3 is a configuration of the manipulator in the base coordinate system Σ_0 . Combined with the coordinates of the end of the manipulator in the certain space point $P = [P_x, P_y, P_z]$, The terminal attitude of the manipulator is:

$$T = \begin{bmatrix} k_x k_x \text{vers} \gamma + c \gamma & k_y k_x \text{vers} \gamma - k_z s \gamma & k_z k_x \text{vers} \gamma + k_y s \gamma & p_x \\ k_x k_y \text{vers} \gamma + k_z s \gamma & k_y k_y \text{vers} \gamma + c \gamma & k_z k_y \text{vers} \gamma - k_x s \gamma & p_y \\ k_x k_z \text{vers} \gamma - k_y s \gamma & k_y k_z \text{vers} \gamma + k_x s \gamma & k_z k_z \text{vers} \gamma + c \gamma & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The dexterity can be defined as follows:

$$F(\alpha, \beta, \gamma) = \begin{cases} 0 & \text{unreachable attitude} \\ 1 & \text{reachable attitude} \end{cases} \quad (3)$$

$$D = \frac{\sum_{m=1}^a \sum_{n=1}^b \sum_{k=1}^c F(\alpha, \beta, \gamma)}{\sum_{m=1}^a \sum_{n=1}^b \sum_{k=1}^c 1}$$

where $F(\alpha, \beta, \gamma)$ indicates whether the attitude of the certain point is reachable and D represents the dexterity at the point.

2.3 The establishment of the dexterity space for the faulty manipulator

The dexterity space is going to be established by introducing the dexterity index in every point in the degraded workspace, to represent the reachability of the terminal attitude. Before the dexterity space, the degraded workspace corresponding to joint locked failure needs constructing first.

2.3.1 The establishment of the degraded workspace

The process of the establishment of the degraded workspace, which is constructed by the Mont Carlo method [8], is as follows:

STEP1. Locking the fault joint at θ_{i0} ; Generating N random numbers between 0 and 1, and obtaining the random angle values of the healthy joints: $\theta_i = \theta_{i\min} + (\theta_{i\max} - \theta_{i\min})N_i$. $\theta_{i\max}$ and $\theta_{i\min}$ are the upper bound and the lower bound of the i^{th} joint angle respectively;

STEP2. Substituting the random healthy joint angles and the locked fault joint angle from STEP1 into Eq.(1), and then obtaining corresponding terminal positions of manipulator;

STEP3. The degraded workspace consists of all terminal positions generated in STEP2.

The construction of dexterity space is based on the established degraded workspace.

2.3.2 The establishment of the dexterity workspace

According to Eq.(2) and Eq.(3), the dexterity index of all points in the degraded workspace can be calculated, and then the dexterity space is obtained.

STEP1. Selecting a certain point in the degraded workspace, and recording its coordinate value;

STEP2. Assigning the values of α, β, γ in $[0, 2\pi]$ at the regular intervals of $\Delta\theta$, and the total number of attitudes is:

$$N_{\text{post}} = \left(\frac{2\pi}{\Delta\theta}\right)^3$$

STEP3. Combining the position of the selected point with every assigned attitude, getting the corresponding configurations based on Eq.(2), and then carrying out the inverse kinematic[9];

STEP4. If there exists solution of inverse kinematic, the corresponding attitude is reachable, otherwise it is unreachable. Then the dexterity in the current point D is the ratio of the number of reachable attitudes and the total number of attitudes;

STEP5. Traversing all points in the degraded workspace, and repeating the process in STEP1~STEP4, the dexterity space is obtained.

Thus, the dexterity space that is the significant reference to the fault-tolerant path planning is established.

3 Fault-tolerant path planning of the space manipulator

Considering that the A* algorithm has the characteristics of strong autonomy and avoid the unreachable region flexibly, fault-tolerant path planning is going to be carried out utilizing A* algorithm, which is going to be improved based on the dexterity. The dexterity space is going to be rasterized after being established first and then the traditional A* algorithm is improved, completing fault-tolerant path planning in the dexterity space to obtain a path that represents the reachability of the terminal attitude for the manipulator.

3.1 The rasterization of the dexterity space

Set a cuboid with its length, width and height in parallel with x, y, z of the inertial frame respectively and then set the side lengths a, b, c of the rectangular so that the cuboid can envelop the dexterity space fully. The number of small cube is:

$$N_{\text{cube}} = \lceil a/\Delta l \rceil \lceil b/\Delta l \rceil \lceil c/\Delta l \rceil \quad (4)$$

Thus, the dexterity space can be seen as composed by a number of small cube units, and the coordinate and dexterity of each cube unit can be represented by the center point of the small cube unit.

3.2 Fault-tolerant path planning based on improved A* algorithm

As a heuristic search algorithm, the A* algorithm is widely utilized in solving the optimal path and its cost function is:

$$f(n) = g(n) + h(n) \quad (5)$$

To obtain the trajectory meeting the terminal attitude of the manipulator, the cost function is improved according to the dexterity index, getting the following expression:

$$f(n) = g(n) + h(n) + Q(D) \quad (6)$$

where $Q(D)$ represents the function evaluating the reachability of the terminal attitude for the manipulator at a certain point and is just concerned with the dexterity D . Set the threshold $k \in [0, 1]$ and $Q(D)$ is:

$$Q(D) = \begin{cases} \infty, & D < k \\ \delta/D, & D \geq k \end{cases} \quad (7)$$

where δ represents the search coefficient of the dexterity. δ and k can be determined according to the actual needs of the task, changing the weight of the dexterity in the path search.

After adding the dexterity index, $Q(D)$ becomes the priority determination item in the path search.

According to the improved A* algorithm and the actual operational requirements, the process searching for a reachable path is as follows:

STEP1. Regarding the starting point as the current node, calculating to the costs $f(n)$ of all the small cube value adjacent to the current node, and taking the center point of the small cube whose $f(n)$ is the smallest as the next node of the current node;

STEP2. Regarding the next node obtained at STEP1 as the current node, and repeating STEP1;

STEP3. Repeating the above step, until the next node is the target node. And then by connecting all the obtained node, a reachable path is formed from the starting point to the target point.

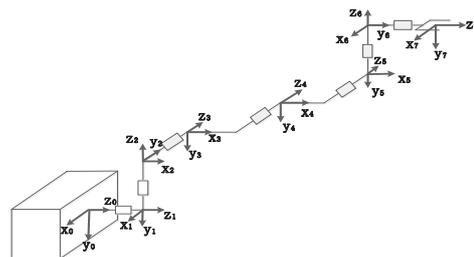


Figure 4. The initial configuration of 7R redundant manipulator.

The planning path based on the improved A^* algorithm may not be the shortest path. So we select the appropriate search coefficient of the dexterity δ and threshold k according to the actual task requirements, to meet both the expected attitude and the shortest distance.

4 Simulation

4.1 The establishment of kinematic model for the space manipulator

As shown in Figure 4 7R redundant manipulator is chosen as the object of study, whose DH parameters are shown in Table 1.

Table 1. The DH parameter table of the initial configuration for 7R manipulator.

Link	a_{i-1}/m	$\alpha_{i-1}/^\circ$	d_i/m	$\alpha_{i-1}/^\circ$
1	0	0	0.6	0
2	0	90	0.5	90
3	0	-90	0.5	0
4	5	0	0.5	0
5	5	0	0.5	0
6	0	90	0.5	-90
7	0	-90	0.6	0

4.2 The establishment of the dexterity space for the space manipulator

Assume that the joint 4 has failed and the locked angle is 75° , the degraded workspace is obtained via MATLAB simulation, as is shown in Figure 5. The number of traversal is selected as 1000 In this paper.

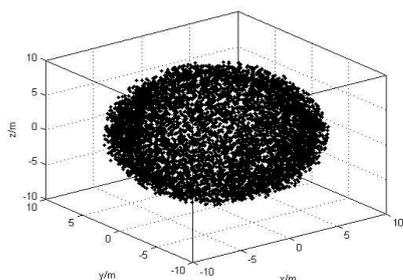


Figure 5. The degraded workspace when $\theta_4 = 75^\circ$.

On the base of solving the coordinates of each point for the degraded workspace, considering that the step size would affect the calculation, take $\Delta\theta = 72^\circ$, and then the dexterity space can be obtained according to the solve steps of the dexterity space, as is shown in Figure 6. It can be seen from Figure 6, that there is a certain rule of distribution for the degraded workspace and it is a gradual process. To reveal the internal dexterity, select some of the interface map, for instance, the planes $x=0$ and $y=0$ where the dexterity distributes with the shape of “X”; in the plane $z=0$, the

dexterity is higher and more uniform, as is shown from Figure 7 to Figure 9.

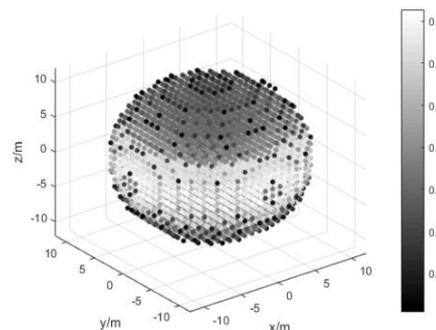


Figure 6. The degraded workspace of the manipulator with single joint failure.

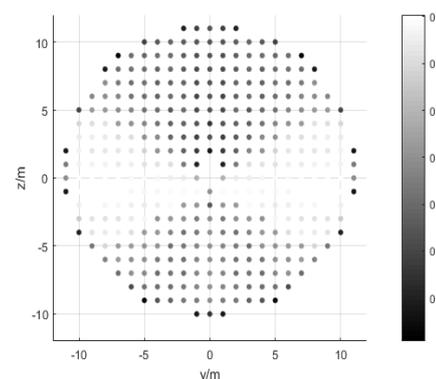


Figure 7. Section view of the degraded workspace when $x=0$.

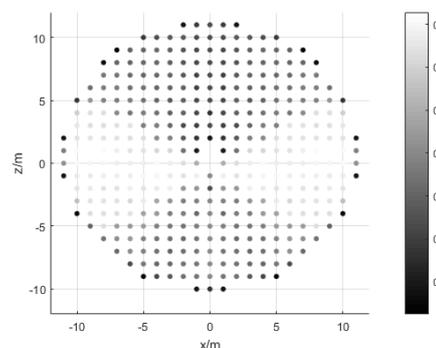


Figure 8. Section view of the degraded workspace when $y=0$.

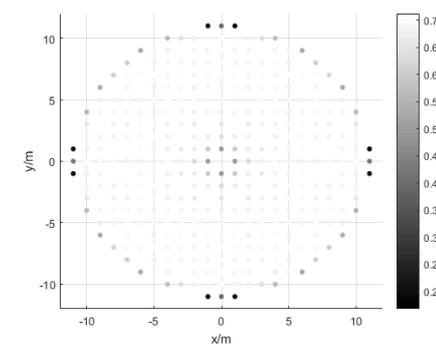


Figure 9. Section view of the degraded workspace when $z=0$.

4.3 The fault-tolerant path planning of the space manipulator

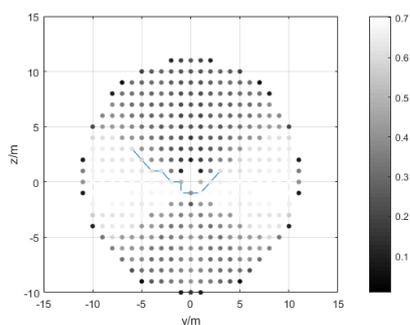
Assume that the joint 6 is faulty and the locked angle is 149° , the coordinates of the starting point and the ending point for the task are as follows:

$$\begin{aligned} x_{int} &= 0, y_{int} = -6, z_{int} = 3 \\ x_{des} &= 0, y_{des} = 3, z_{des} = 1 \end{aligned}$$

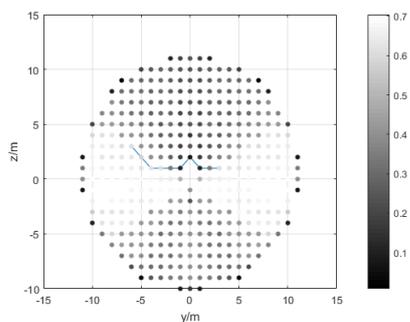
The simulation results of the path planning are shown in Figure 10. Comparing the difference between the two paths, it can be found that the path using improved A* algorithm keeps higher dexterity. The Figure 11 shows the trend of dexterity where the path node passed by. It can be seen that the dexterity of the improved path increased a lot and the dexterity improved overall:

$$\varepsilon = (0.624 - 0.3992) / 0.3992 = 84.37\%$$

As to the distance comparison between the two paths, the total distances of the paths are 10.66m and 11.32m respectively. It explains that although the distance in the path searched by improved A* algorithm has increased, while the dexterity improved, it is not obvious.



(a) Path search based on improved A* algorithm



(b) Path search based on unmodified A* algorithm

Figure 10. The comparison of the experiment for fault-tolerant path planning.

The dexterity obtained by utilizing the improved A* algorithm is overall 84.37% higher than the previous, while the distance is 7% longer than the previous. In the actual missions, it tend to pay an additional 7% of the distance to get the 80% increase in dexterity. Therefore, it is feasible to utilize the improved A* algorithm with dexterity to carry out path planning.

The simulation results show that the improved A* algorithm based on the dexterity can solve the problem of path planning for the manipulator with single joint failure. In summary, the correctness and validity of fault-tolerant path planning method for the manipulator based on the dexterity space can be verified by the simulation results.

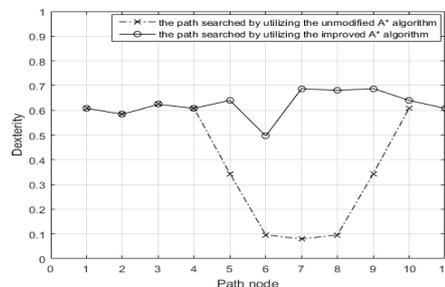


Figure 11. The comparison of the dexterity for the two paths.

5 Conclusion

A fault-tolerant path planning method for the manipulator with single joint failure is proposed based on dexterity space in this paper. The dexterity index is constructed by considering three posture parameters, and then the dexterity space is established on the base of solving the degraded workspace. In the meantime, the traditional A* algorithm is improved based on the dexterity, and then fault-tolerant path planning comes true utilizing the improved A* algorithm. The simulation results show that the planning path based on the improved A* algorithm in the dexterity space can meet the requirements of the higher attitude of the end of the manipulator so that the manipulator can continue to complete the original task after the joint failure occurs.

References

1. Q X. Jia, T. Li, G. Chen, et al. *MFI. IEEE*, 1(2014).
2. O. Khatib. *IEEE. ICRA*, 500(1985).
3. Z Z. Du, G D.Liu. *Computer Simulation*. **26**(12), 118(2009).
4. M Z. Zhao, Z D. Meng. *Jour of HuaZhong Unive of Sci and Tech*. 205(2008).
5. R G. Roberts, A A. Maciejewski. *IEEE Trans on Robotics & Auto*. **12**(4), 543(1996).
6. J. Zhao, K L. Zhang, Y B. Yao. *Mech Sci and Tech for Aero Engi*. **25**(3), 261(2006).
7. S C. Liu, J Q. Xu, X F. Ma. *Robot*. **11**(6), 12(1989).
8. J. Rastegar, B. Fardanesh. *Mech & Mach Theory*. **25**(2), 233(2014).
9. W. Xu, Y. She, Y. Xu. *Acta Astro*, **105**(1), 201(2014)