Trajectory Planning of Transformer Cleaning System Based on IRB4600 Robotic Arm

Yong-qiang LI¹, Wei-cai LIU²,* and Zhi-fei ZHANG²
¹Guangdong Raising Synthesis Energy Services Co., Ltd, Foshan, China
²School of Automation, Foshan University, Foshan, China
*Corresponding author

Keywords: Transformers, IRB4600 robotic arm, Trajectory planning.

Abstract. Clean with power for oil-immersed transformer by robotic arm is investigated. We establish kinematic model of IRB4600 Robotic Arm, coordinate transfer is analyzed. Space near inline and outline of transformer is treated as obstacle, surface to be sprayed is departed into several areas, and RRT algorithm is modified in form of point to multi-points to search optimal path for cleansing. Simulation and test by virtual robot platform show that scheme proposed in this paper is valid.

Introduction

10kV oil-immersed transformer is a kind of static electrical equipment commonly used in low-voltage power stations and small or medium-sized enterprises. It is generally used in enterprises and low-voltage substations. The operating environment is complicated, which causes the surface of the transformer to be easily contaminated with oil and dust, having an influence on heat dissipation and insulation performance of transformer, so surface decontamination is an important part of the daily maintenance of such oil-immersed transformers. At present, the cleaning of the surface of the transformer can only be manually performed by pulse water cleaning or insulating cleaning agent. This cleaning method requires the transformer to be powered off to ensure the safety of the workers and has a great impact on the normal production of the transformer users. At the same time, since the descaling effect of manual cleaning depends on labor, the actual cleaning quality is not proved. Therefore, the development of a robotic system for descaling the surface of the transformer has multiple meanings such as saving labor and enterprise cleaning costs, avoiding safety accidents and improving descaling effect [1-2].

The problem of charged cleaning robot can be transformed into robotic arm motion planning without collision in high-dimensional space (avoiding obstacles such as high-voltage intake line), which is essentially a problem of optimal track planning under constraint conditions. In recent years, the sampling-based search algorithms which was named rapidly-exploring random tree (RRT) [3-5] have been studied extensively and in depth. The RRT algorithm, proposed by LaValle and Kuffner, was originally used to solve path planning problems with kinematic constraints. Because the RRT algorithm is randomly sampled in the state space and its fast search speed, especially in the high-dimensional planning space, the advantage of search speed is more particularly obvious, at the same time, the RRT algorithm also has the advantage of probability completeness [6]. However, the RRT algorithm itself also has some defects, because of the global random uniform sampling, and checks whether it will collide with obstacles every time it expands. That will consume more computing resources, making the convergence speed slower. Because of the random sampling, resulting in more bends in the resulting path, which make a possibility that the robotic arm cannot be executed directly, and for this reason Karaman and Frazoll propose an RRT* algorithm that can yield an asymptomatic optimal path [7]. This article will select the robotic arm according to the cleaning requirements and use the RRT* algorithm to solve track planning of cleaning robotic arm.
**Motion Model of Cleaning Robotic Arm**

The structure of the transformer cleaning system includes the robotic arm, cleaning object, information collector and controller, etc. Our investigation focus on the robotic arm track planning problem. The schematic diagram 1 omitted the part of information acquisition and controller.

The robotic arm is the core of the system and undertakes the task, so the structure and characteristics of robotic arm determine the performance of the system. The IRB4600 robotic arm is one of ABB's robotic arms, and this model adopts optimized design that has excellent adaptability to target applications. The slim body makes the layout of production unit more compact and has the high precision, fast operation speed, large operating range and body insulation protection, which is suitable for spray cleaning environment. The three-dimensional structure diagram of the IRB4600 series of robotic arm is shown as in Figure 2 [8].

As shown in Figure 3, robotic arm is usually described with four linkage parameters: link length $a_i$, link twist $\alpha_i$, the link offset $d_i$, and joint angle $\theta_i$. 

---

**Figure 1.** Transformer automatic cleaning robotic arm system.

**Figure 2.** 3D structure of the IRB4600 series of robotic arms.
According to the parameter method of standard D-H, the corresponding parameters for IRB4600 are shown as in Table 1.

Table 1. DH parameters table for IRB4600 robotic arm.

| Joint | Link length [mm] | Link twist [rad] | Link offset [mm] | Jointed angle [rad] |
|-------|------------------|------------------|------------------|--------------------|
| 1     | 0                | 0                | 0                | $\pi/2$            |
| 2     | 175              | $\pi/2$          | 459              | 0                  |
| 3     | 1075             | 0                | 0                | $\pi/2$            |
| 4     | 175              | $\pi/2$          | 0                | 0                  |
| 5     | 0                | $\pi/2$          | 1230             | 0                  |
| 6     | 0                | $\pi/2$          | 0                | 0                  |

Modeling the IRB4600 robotic arm by using MATLAB 2018B, Robotics Toolbox 10.3 and DH parameter table, and the model of the robotic arm is shown as in Figure 4.
Workspace and Analysis of Motion

Calculation of Workspace

The working space of a robot is a collection of points where a robotic arm can arrive during operation. There are generally two methods for solving the whole workspace: numerical method and Monte Carlo method. For the numerical method, the first is to set the joint angle limit, and then to accumulate each joint with a certain step value and calculate the positive solution to obtain the end point position. It would not be shut down until each joint reaches the maximum limit angle, and finally all the acquired end points are processed to draw the boundary curve of the robot. Based on these boundary curves, one can draw the boundary surface that represents the robot’s working space.

In this paper, Monte Carlo method is used to solve the work space, which has fast calculation, and the amount of data is provided largely enough, more accurate results can be obtained. The solution steps include 1) generate random variables for each joint; 2) calculate a positive solution.

Using the MATLAB 2018B, Robotics Toolbox 10.3 and the model established in the previous section to program, then setting the associated joint limit angle and the number of 100,000 sampling points. The effect diagram of program implementation shows in Figure 5, the blue domain is the whole work space, where coordinate unit is mm.

![Figure 5. Working space map of IRB4600 robotic arm.](image)

Kinematic Analysis

Link transformation is the basis of kinematic analysis of robots, and its establishment mainly involves coordinate shift and rotations around X, Y, and Z axes of the coordinate system. The rotation transformation can be described as follows:

\[
R(X, \alpha) = \begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \alpha & -\sin \alpha \\
0 & \sin \alpha & \cos \alpha 
\end{bmatrix}, \quad (1)
\]

\[
R(Y, \phi) = \begin{bmatrix}
\cos \phi & 0 & \sin \phi \\
0 & 1 & 0 \\
-\sin \phi & 0 & \cos \phi 
\end{bmatrix}. \quad (2)
\]
\[
R(Z, \theta) = \begin{bmatrix}
\cos \theta & -\sin \theta & 0 \\
\sin \theta & \cos \theta & 0 \\
0 & 0 & 1
\end{bmatrix}.
\] (3)

The coordinate shift along the X, Y, and Z axes is parallel. Let \( p_x \), \( p_y \) and \( p_z \) be the distances from the original coordinate, the shift matrix is

\[
P = \begin{bmatrix}
p_x \\
p_y \\
p_z
\end{bmatrix}
\] (4)

Usually all the link transformations can be obtained by coordinate rotation and shift. The composite transformation formula is (see Figure 6):

\[
^A P \approx ^B A R^B P + ^A P_{B0}.
\] (5)

The values of DH parameters of IRB4600 are as follows:

\[
\theta_1 = \pi / 2, \theta_2 = 0, \theta_3 = \pi / 2, \theta_4 = 0, \theta_5 = 0, \theta_6 = 0, a_2 = 175, a_3 = 1075, a_4 = 175.
\]

Figure 6. Compound coordinate transformation.

**Reverse Kinematics Calculation**

The inverse kinematics of joint robot is to solve the angle of each joint of the robot based on the end position and pose. Unlike positive kinematics, in reverse kinematics a posture may have several groups of joint angles, so constraints need to be added to get unique solution.

Figure 7 is a flowchart of inverse calculation of the robotic arm. It can be concluded that one firstly checks whether the coordinates of the position and pose of the actuator at end of robotic arm can reach to are in the work space or not, and if it does, then calculates inverse solutions.

The six joint angles are divided into two groups, the first, second and third joint are gathered into a group, which are calculated previously with geometric methods since these three angles are independent of others. The first angle mainly controls the rotation of the whole robotic arm, which can...
be projected into the XOY coordinate system for calculation. The second and the third angle are independent of the rotation, and can be projected into the XOZ or OZY coordinate system for calculation. It is forth to note that geometric method is greatly affected by the configuration of the robotic arm, it probably loss generality. All results calculated above are substituted into the DH matrix of the robotic arm in preparation for consequent calculations.

The fourth, fifth and sixth angles are solved by solving following equations:

$$T_5 = T_6 A_6^{-1}.$$  

(6)

$$A_1^{-1} A_2^{-1} A_3^{-1} T_6 = A_5 A_6.$$  

(7)

where $T_6 = A_1 A_2 \cdots A_6$, and A1 to A6 are

$$A_1 = \begin{bmatrix} \cos \theta_1 & 0 & \sin \theta_1 & 0 \\ \sin \theta_1 & 0 & -\cos \theta_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 & 0 & \cos \theta_2 a_2 \\ \sin \theta_2 & \cos \theta_2 & 0 & \sin \theta_2 a_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} \cos \theta_3 & -\sin \theta_3 & 0 & \cos \theta_3 a_3 \\ \sin \theta_3 & \cos \theta_3 & 0 & \sin \theta_3 a_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} \cos \theta_4 & 0 & -\sin \theta_4 & \cos \theta_4 a_4 \\ \sin \theta_4 & \cos \theta_4 & 0 & \sin \theta_4 a_4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} \cos \theta_5 & 0 & \sin \theta_5 & 0 \\ \sin \theta_5 & 0 & -\cos \theta_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_6 = \begin{bmatrix} \cos \theta_6 & -\sin \theta_6 & 0 & 0 \\ \sin \theta_6 & \cos \theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Since the solutions obtained may not be unique, it is necessary to screen the obtained solution according to the constraint conditions, excluding the solution that is not inside the constraint condition and select the optimal solution as the final result according to the principle of shortest path.

**Motion Planning Algorithm**

The motion path plan of transformer cleaning robot is one of the key technologies in this field. The path should be 0.5 meters away from the high-voltage inline and does not collide with the outline. Path plan faces two challenges: First, the robotic arm is usually equipped with joints of up to six degrees of freedom, its configured space is high-dimensional, so the algorithm based on the structure space dispersion is difficult to apply. Second, it is the configuration space that requires high-precision collision check of potential trajectories due to the presence of obstacles.

This paper will use the technique developed in [9] named RRT* algorithm to solve the path planning problem, due to this method can rapidly solve the generation of high-dimensional configuration space.

**Problem Description of the Path Plan**

Let $X \in \mathbb{R}^n$ be configuration space, denote $X_{obst}, X_{goal} \subset X$ to be obstacle and target space respectively, $X_{free} = X \setminus X_{obst}$ is free space. There is a path $\sigma(\tau)|\tau \in [0,1] \rightarrow X$, $\tau = 0.1$ corresponding to the start and end point of the path. If $\forall \tau \in [0,1]$ $\sigma(\tau) \in X_{free}$, the $\sigma$ is said to be collision free. Sets of all feasible and collision-free paths $\sigma \in \Sigma_{free}$, start point $x_{init} \in X_{free}$. Now the problem for path
plan can be stated as follows: seek a feasible path \( \sigma : \rightarrow X_{free} \) such that the performance index \( c(\sigma) \) (or cost function) is minimized, i.e. \( \sigma^* = \min_\sigma c(\sigma) \).

**The principle of RRT & RRT* Algorithm**

RRT is a motion planning algorithm with probability sampling, developed by LaValle in Iowa State University in 1998. The method can efficiently search for high-dimensional space, mapping obstacles from the original three-dimensional work space to configuration space and judging whether the robot collides with obstacles through collision detection. The operation of RRT can be summarized as:

1) Sampling: the sample(.) function collects independent and evenly distributed samples and stores them in \( X_{free} \);
2) collision_free(\( \sigma \)): collision detection for a given path \( \sigma(t) \),
3) Guide: given \( x, x' \in X \), by steering(\( x, x' \)) to form a path \( \sigma \) according to the following rule:

\[
\sigma(0) = x, \sigma(1) = x'; \forall \tau \in (0,1), \sigma(\tau) = (1-\tau)x + \tau x'.
\]
4) Find the nearest vertex: given the nearest vertex set \( V \subseteq X \) and \( x \in X \), the nearest(\( V, x \)) finds the \( x' \in V \) to minimize \( \|x' - x\|_2 \), where \( \|\cdot\|_2 \) represents the vector's Euclidean norm.

Finally, we will introduce a function \( A.T(a) \), \( D = \{\text{add}, \text{remove}\} : \text{add or remove element } \{a\} \).

---

| Algorithm 1: RRT* |
|-------------------|
| 1: set \( x_{start} \)  |
| 2: for i=1 to n do  |
| 3: \( x_{rand} = \text{sample}(X) \)  |
| 4: \( x_{nearest} = \text{nearest}(D, x_{rand}) \)  |
| 5: \( x_{new} = \text{steering}(x_{nearest}, x_{rand}) \)  |
| 6: if (\( x_{nearest}, x_{new} \)) feasible then  |
| 7: \( V = D.\text{add_vertex}(x_{new}) \)  |
| 8: \( E = D.\text{add_edge}(x_{nearest}, x_{new}) \)  |
| 9: \( x_{near} = \text{nearest}(D, x_{new}, x) \)  |
| 10: for all (\( x_{near}, x_{near} \)) do  |
| 11: \( \text{rewrite}_RRT^*(x_{near}, x_{new}) \)  |
| 12: for all (\( x_{near}, x_{near} \)) do  |
| 13: \( \text{rewrite}_RRT^*(x_{new}, x_{near}) \)  |

---

| Algorithm 2: rewrite_RRT* |
|---------------------------|
| 1: if (\( x_{potential\_parent}, x_{child} \)) is feasible then  |
| 2: \( c = \text{cost}(x_{potential\_parent}, x_{child}) \)  |
| 3: if (cost D(\( x_{potential\_parent} \)) + c < cost D(\( x_{child} \))) then  |
| 4: \( D.\text{parent}(x_{child}) = x_{potential\_parent} \)  |

RRT* is obtained by adding rewrite_RRT*, where the pseudo code of the rewrite_RRT* function is shown in Algorithm 2.
Simulation

The RRT* algorithm solves the point-to-point path plan problem in complex environments. In the specific cleaning operation process, not only the complicated environment but also the path traversal to cover all surface of transformer should be considered. The actual size of the transformer, cleaning range of head injection, and the safe operation standard are basis. Take the upper surface as an example, the area is divided into eight parts, and the center of each area is regarded as the start point of to another point, as shown in Figure 8, the yellow is bound line.

![Figure 8. Departs of upper surface of transformer.](image)

Establishment of the Distribution Room Scene

Design of Model Scene

According to the actual size of the oil-immersed transformer, the modeling based on the plant Solidwork201, 3D map can be obtained as shown in Figure 9. The .stl file(stereolithography), an interface protocol investigated by 3D Systems company in 1988, is then generated, which is needed for subsequent simulation. It is a three-dimensional graphics file used for rapid prototyping services. The STL file consists of the definition of multi-triangle patches, each of which includes the 3D coordinates of each fixed point in the triangle and the normal vector of the triangle patch.
Simulation Platform and Adjustment of Physical Parameters

The simulation platform selects v-rep (virtual robot experimentation platform), it is a powerful 3D integrated development environment of the robot with several common computing modules (reverse kinematics, physics/dynamics, collision detection, minimum distance calculation, etc.), distributed control architecture (control scripts of unlimited number, threads or non-threads), and several extension mechanisms (plugins, client applications, etc.), which provide many functions that can be easily integrated and combined through detailed API and scripting functions. v-rep is the perfect tool for rapid prototyping, remote monitoring, rapid algorithm development, related-education of the robot and automation system simulation of the factory. IRB4600 robotic arm is added to the scene and it is placed at 1.265 meters from the transformer, as shown in Figure 10 (the red box is the cleaning nozzle).

The mode of the six joint motors of robotic arm is set to torque/force mode, as shown in the red frame in Figure 11. The maximum speed of the Joint 1 motor is set to 100 degrees per second, as shown in the blue frame in Figure xx. Joint 1 motor maximum torque set at 1000 N/m, as shown in the yellow box frame in Figure xx. The parameter settings of joint 2 motor and joint 3 motor are the same as those of the Joint 1 motor. The maximum speed of the Joint 4 motor is set to 150 degrees per second and the maximum torque is set to 800 N/m. The parameter settings of joint 5 motor and joint 6 motor are consistent with parameter settings of the joint 4 motor.
The simulation's physics engine is set up as the Newton engine, and the simulation experiment consists of three parts: cleaning the upper surface of the transformer, cleaning the side surface of the transformer; and cleaning the front surface of the transformer.
Conclusions
This paper discusses application of robotic arm in cleaning oil-immersed transformers located in power distribution room to meet with the requirements. According to the characteristics of the cleaning task and the motion planning requirements of the special task, the kinematic model of the IRB4600 robotic arm is established, and the path planning problem with complicated environment is solved by use of the RRT* algorithm. Simulation by virtual robot test platform proves that the cleaning scheme and technology proposed in this paper are valid.

Acknowledgement
This research was financially supported by the National Natural Science Foundation of China (No.60803087)

References
[1] Chen You-ping, Liu Jie-fu. Brief Discussion on Importance to Improve Cleanliness of Transformer Active Part and Its Measures, J. Transformer. 42(2005) 10-12.
[2] Zhou Li-kun, Liu Hong-zhao, Li Yue. Summary for the Key Technologies and Research Status of the Cleaning Robot, J. Mechanical Science and Technology for Aerospace Engineering. 33(2014) 635-642.
[3] Lavalle S M. Rapidly-Exploring Random Trees: A new Tool for Path Planning, R. (1998) 98-11.
[4] Lavalle S M, Kuffner J J. Randomized Kinodynamic Planning, J. International Journal of Robotics & Research. 15(1999) 378-400.
[5] Perez A T, Karaman S, Shkolnik A C, et al. Asymptotically-optimal path planning for manipulation using incremental sampling-based algorithms[C]// 2011 IEEE/RSJ International
[6] L. E. Kavraki, P. Svestka, J. C. Latombe, and M. H. Overmars, Probabilistic roadmaps for path planning in high-dimensional configuration spaces[C], IEEE Trans. on Robotics and Automation, vol. 12, no. 4, pp. 566–580, August 1996.

[7] Kavraki L E, Kolountzakis M N, Latombe J C. Analysis of Probabilistic Roadmaps for Path Planning[C]//IEEE International Conference on Robotics & Automation. IEEE, 1998.

[8] IRB4600 on https://wenku.baidu.com/view/b7ca3bb350e2524de4187e3b.html

[9] Karaman S, Frazzoli E. Incremental Sampling-based Algorithms for Optimal Motion Planning, J. (2010).