Proceeding of the IEEE
International Conference on Robotics and Biomimetics
Dali, China, December 2019

Constraint Path Planning for an Autonomous Wall Spray Coating Robot

Ganggang Yang
Department of Mechanical Engineering and Automation
Shanghai Jiao Tong University
Shanghai, China
yanggang@sjtu.edu.cn

Chen Hu
Department of Mechanical Engineering
Carnegie Mellon University
Pittsburgh, USA
chu3@andrew.cmu.edu

Hao Meng
College of Architecture and Urban Planning
Tongji University
Shanghai, China
1610098@tongji.edu.cn

Shih Yuan Wang
Graduate Institute of Architecture
National Chiao Tung University
Taiwan, China
yuan@arch.nctu.edu.cn

Abstract— The wall spray coating task requires robot to handle a spray-head to cover the whole interior wall surface. In this paper a manipulator is used to complete such task where a Cartesian constraint path is firstly generated by breadth first search (BFS) based on the boustrophedon decomposition, and then the Cartesian path will be converted to joint sequences. In addition, corners formed by two walls as the non-differentiable area are analyzed and fitted by Bezier curve. Finally, a sphere cap constraint planner is proposed to deal with wall suspended with multiple aligned pipelines. A 6DOF KUKA robot arm was employed to do the spray coating task in a building, and the experiments will demonstrate the effectiveness and efficiency of the proposed approach.

Index Terms— path planning, spray coating, sphere cap constraint

I. Introduction

Robots have many sophisticated applications in a lot of fields with the assistance of artificial intelligence, such as in assembly, cleaning and sorting. However, architectural robot is not so prevailing [1][2] in building construction compared with industrial application. The completely autonomous robotic system for construction usage which has practical feasibility and technological adaptability needs more intelligent methods [3].

Wall painting is a general process in construction, which requires robot to spray coating on the whole interior wall surface. There are many researchers designing multiple contraption to make painting robotic system more dexterous [1][4][5]. While this paper mainly focuses on path planning methods for the high DOF manipulator arm to do spray coating work, as in Fig. 1, where a 6DOF KUKA robot arm is spraying pigment on the wall. The distance between the spray head and the wall surface is denoted as d, and the cross section for the spraying coating from the nozzle is assumed as a circle with the diameter D under such distance d. This process is actually the complete coverage task of a known region while avoiding obstacles with a path of width D, which is common to numerous applications, such as demining, cleaning indoor workspace and so on. There are many previously proposed methods [6] [16] to manage the coverage task, including decomposition methods and optimization methods. However, these methods are almost based on mobile cars and the situation may be different when adopted in manipulator arms on the walls. Firstly, the planned path of coverage region is just the followed trajectory of the mobile vehicles. Nevertheless, the path only represents the Cartesian path of the tools on manipulator arms and needs to be converted to joint sequences. And the path could be broken as we just need to guarantee the continuity in joint space. Secondly, the

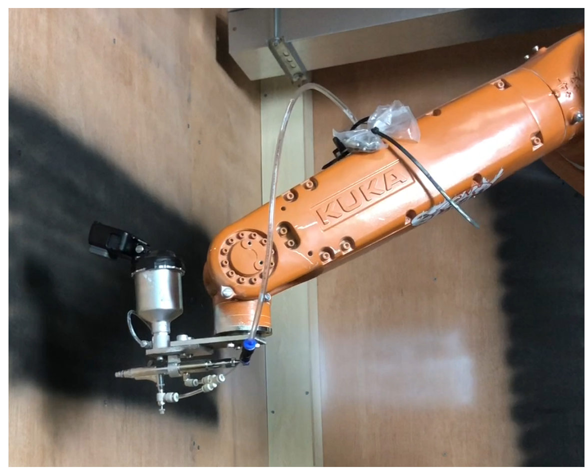


Fig. 1. A KUKA robot arm is spraying pigment on the wall.

978-1-7281-6321-5/19//$31.00 ©2019 IEEE

prior studies concentrate on one square area, not the interior wall of a building which generally consists of several square areas. And the adjacent walls will form the non-differentiable area, in other words, a corner, which requires particular treatment. Thirdly, in the face of walls suspended with multiple pipelines which is common in the building, the robot will definitely collide with these pipelines while simply following the end-effector path. On this occasion, a sphere cap constraint path planner is presented in this paper to deal with it, which will be analyzed in detail in Section III.

This paper is organized as follows. In Section II, we provide a short state of the art for the path planner of coverage region. Section III will give the description of the work-flow for the manipulator arm to do the wall painting job, and propose the BFS search, Bezier curve and sphere cap constraint planner to improve its work performance. In Section IV, a 6DOF KUKA robot arm was employed to do spray coating task in a building to show the effectiveness and efficiency of the proposed approach. Finally, a concise conclusion is presented in Section V.

II. Related Work

In the early researches, several path planning approaches have been proposed. The template based approach [7][8] is first popularly applied in the automatic vehicle guidance for floor cleaning or inspection. It uses several predefined turning curve to guide the vehicle, but the path is sometimes weird and may get stuck in an closed loop. The trapezoidal decomposition [9] introduces cellular decomposition to give a complete coverage path solution. However, a majority of extra motions is produced in trapezoidal decomposition to guarantee completeness. The boustrophedon decomposition [10] is a generalization of the trapezoidal decomposition to reduce the back and forth turns which combines the narrow cells into one cell. And then it seeks an exhaustive path through the topological graph which represents the adjacency relationship of the separated cells. The Morse decomposition [11], which is another enhancement of the cellular decomposition, uses the critical points of Morse functions to indicate the locations of cell boundaries. This decomposition allows to change the pattern by which a robot covers its free space according to the different definition of the Morse function. An alternative algorithm for Morse decomposition is the landmark-based topological coverage algorithm [12], which uses the topological map as the coverage region instead of the grid-based map. It could detect obstacles more efficient and the generated coverage path is shorter. To combine the sensor scan message, [13] proposed a novel sensor-based coverage path planning algorithm. It divides the workspace into cells at each scan sample and uses a two-direction oriented rectilinear decomposition (ORD) to create a closed map and memorizing the covered space to define the next target cell from the map. The spatial cell diffusion (SCD) [15] diffuses the occupied cells outwards by moving spirally. To save power, a battery-constrained sweep algorithm (BC Sweep) [14] based on boustrophedon decomposition simply approximates a lower cost path through the coverage graph.

Another common approaches for coverage region path planning belongs to the graph optimization techniques. In greedy algorithms, the planner always traverses the minimum cost point to gain the most benefit, such as Dijkstra [17], D* [18], A* [19], but may get stuck in local minima. In evolutionary algorithms which includes particle swarm optimization (PSO) [20], ant colony optimization (ACO) [21] and genetic algorithm (GA) [22], the environment is represented as a grid map where a random path is generated, and then the path will be optimized by stochastically changing the nodes on the path according to a parameterized cost function.

Pictobot [1] is a wall painting robotic system, which could cooperate with human to complete the interior painting task. It uses a sensor to percept the surroundings and detect the wall and corners, and could raise high to get a large workspace and is more time-consuming and material-saving than human. However, the planner in Pictobot simply adopts the boustrophedon steps and the windows which are not necessary to paint have not been descriped. The corner planning method is only feasible at one single corner, not general when facing polyline corners. When there are pipelines aligned beside the walls, the planner may fail to generate a valid path.

III. Constrained Path Planning

The path planning of coverage region in this paper consist of three parts: the construction of the map, a feasible cartesian path searching and end-effector path converting to joint path. The Cartesian path will also be discretized by Δx. The spray-head moves in a uniform linear speed s, and the duration Δt added as the time parameterization is calculated by: $\Delta t = \frac{\Delta x}{s}$

A. Path Planning of Coverage Region

In practice, the boustrophedon back and forth motion for the spray-head is more effective and natural than the spiral or other orientation-changed methods. The turning and orientation changing number will increase the path length and path overlapping and the overlapping could destroy the coating uniformity. In the wall painting, there are generally some openings for windows or doors, which should be viewed as obstacles in the coverage region and need not to paint, as in Fig. 2. What's more, these obstacles are usually regular geometry which most are

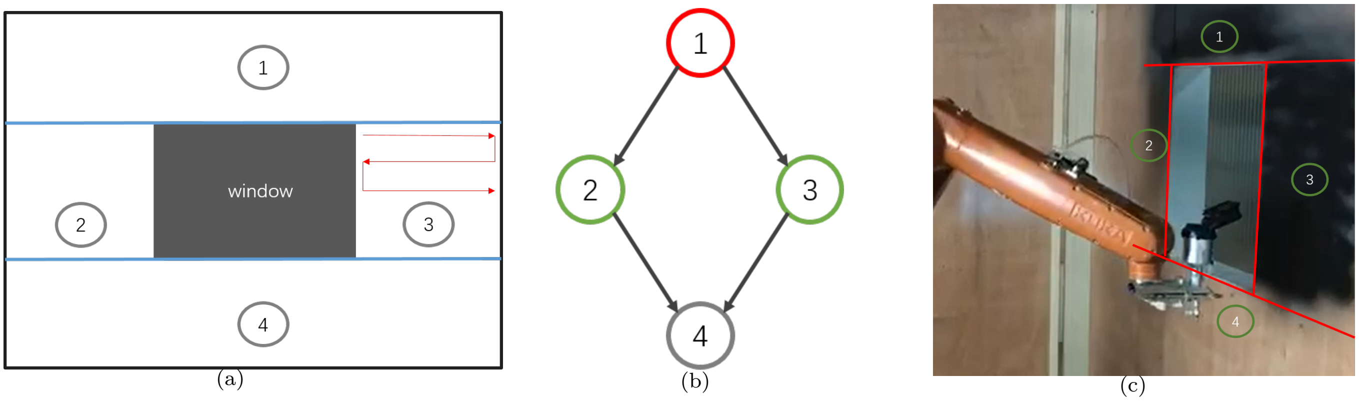


Fig. 2. Painting on the wall with window as the unreacheable area. (a): the boustrophedon decomposition of the coverage region, and the window will be viewed as th obstacle which won't be painted. (b): the adjacent topology graph for the cells. Under the BFS rules, node 1 will be firstly visited, and the connected unvisited neighbors, node 2 and node 3, will be pushed to stack and wait for visiting. Node 4 will be visited lastly. (c): the KUKA robot arm is painting on the wall in the order: 1-3-2-4.

rectangles embedded in walls and the number of such obstacles are few. From the above, the boustrophedon decomposition is used to construct the cell map. This decomposition could generate fewer cells than trapezoidal decomposition and is more suitable for such environment with vertical and horizontal lines than the Morse decomposition as the obstacles are convex rectangles most of the time. Besides, the decomposed cells often have a large scale, and the path planning in each single cell is simple, as the boustrophedon motion indicated by the red line in Fig. 2 (a), so the boustrophedon decomposition will cost less time than the sequences optimization methods.

However, the search method in the adjacent topology graph differs with the plane mobile vehicles which try to find an exhaustive path through every single node in the graph. With assist of the dexterous manipulator arm, the tool head could move from one node to another node even if these two nodes are not connected to each other. For example, in Fig. 2 (c), the manipulator could directly move from section 3 to section 2 under a point-to-point motion (P2P) even no path goes through node 2 and node 3. This means that the path can be broken and the problem reduces to the full traversal of the decomposition graph. Based on this, the BFS method is proposed to traverse all nodes in the topology graph in substitute of the exhaustive search. When traversing the graph, one node is first visiting, the unvisited neighbors will be stored in a candidate stack. After the node is visited, nodes in the candidate stack will be visited in order. The BFS guarantees that the adjacent section will be visited with higher priority, which seems intuitive on the painting work. In Fig. 2 (b), the path generated by BFS is 1-2-3-4 (or 1-3-2-4) which is in line with human choice while the order 1-2-4-3 which is an exhaustive search path is not consistent with human nature.

In each single cell, the boustrophedon path will be discretized during a small time Δt with every fraction Δx, and the whole path reduces to a series of points with position and orientation in Cartesian coordinate system. The IKFast kinematics solver is used to convert this Cartesian path to joint sequences and the joint values in each sequence will be commanded to manipulator controller to drive the spray head. As for the transition between cells, a general P2P motion planner in joint space is adopted form the end of one path to the begin of the other path with OMPL as the planning backend. There is a switch to control the valve to open or close by the programming under different motion mode. As there is no end effector constraint, the P2P motion planning is fast, which gives a more efficient transition than the traditional mobile vehicles moving on the plane.

B. Corner Fitting

The biggest difference between the wall painting and general plane coverage is the existence of the corner. Corner forms from the intersection of walls, and is a saltation of the wall surface. On account of the bodies size of the spray head and the robot itself, the surroundings around the corner is hard to be painted by the decomposition method, which require a subsidiary method to eliminate such non-differentiable part.

The Pictobot [1] uses a simple way to deal with this situation where the path goes along the intersection line of the corner and orientates towards the middle of the intersection angle between walls. This idea makes the corner planning very easy and could handle a major of the corner problems. However, it is not the perfect one. First, the diameter D should be larger enough to cover the corner and the troublesome surroundings. Secondly, in front of the continuous turns, as in Fig. 3 (a), the

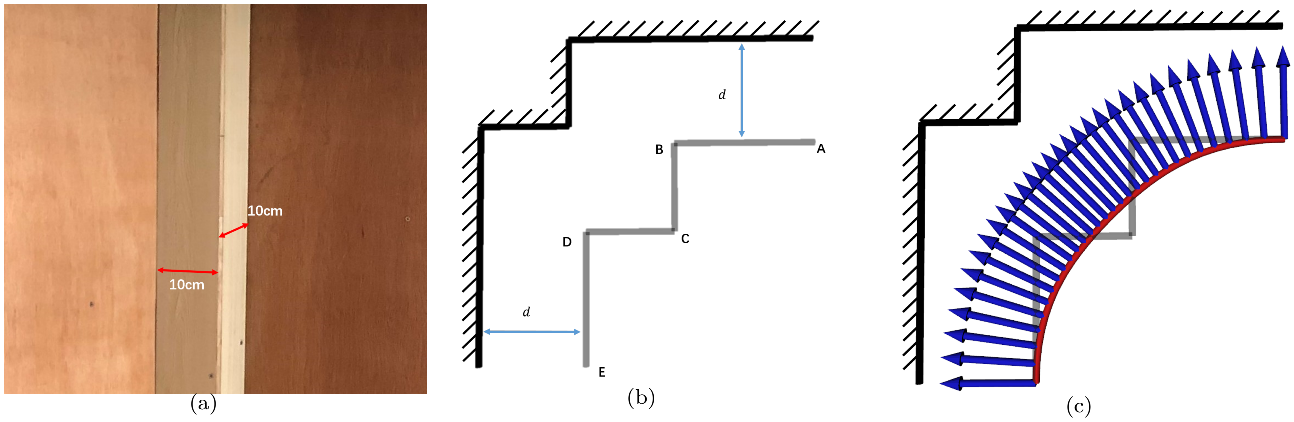


Fig. 3. Corner path planning by curve fitting. (a): a two-continuous turns where the two strip walls has a small width of 10cm. (b): the cross section view of the corner. The corner polyline moves towards inside by d, the distance between the spray nozzle and the wall surface. After this offset, the endpoints will be viewed as the control points, which are labled as A, B, C, D and E. (c): The Beizer curve fitting. Red lin is the fitted curve by the control points and the blue arrows represent the normal vectors.

middle line may be ambiguous and the coat thickness will lose its uniformity.

This paper proposes the corner fitting method to generally change all kinds of corners to a smooth curve surface. As the corner can be viewed as a 3D model extruded form a 2D section, the surface fitting reduces to line fitting. Considering the cross section of a two-continuous turns in Fig. 3 (b), the corner polyline will move towards inside by an offset d, and the new polyline is totally defined by the end control points. Suppose the new polyline as a function of time: $f(t), t \in [0,1]$. Then we need find another function $g(t)$ to represent $f(t)$. As the orientation of spray head should be perpendicular to the wall surface, the difference of the normal vectors should be as small as possible. The tangent vector is always perpendicular to the normal vector, so the normal vector errors equal to the tangent vector errors. To guarantee the smoothness between the neighbor walls, the tangent vectors should be the same at the begin and the end of these two functions. Then the fitting is actually the following optimization:

$$\begin{aligned} &arg\ \min \frac{\int_0^1 ||\dot{g}(t) - \dot{f}(t)||dt}{\int_{g(t)} ds} \\ &s.t.\ \dot{g}(0) = \dot{f}(0) \\ &\qquad \dot{g}(1) = \dot{f}(1) \end{aligned} \tag{1}$$

The Beizer curve is very common in computer graphics because of its excellent performance. This curve is always tangent to the start point and end point, and each of intermediate points will pull this curve to bend towards itself. These particular properties enable the Beizer very suitable for the corner fitting. Taking the end points of the $f(t)$ as the control points, the curve $g(t)$ could be generated by Beizer curve fitting, as showed by the red line in Fig. 3 (c). The red line represents the spray head path when painting the corner and the blue arrows as the normal vectors of this line indicate the nozzle orientation.

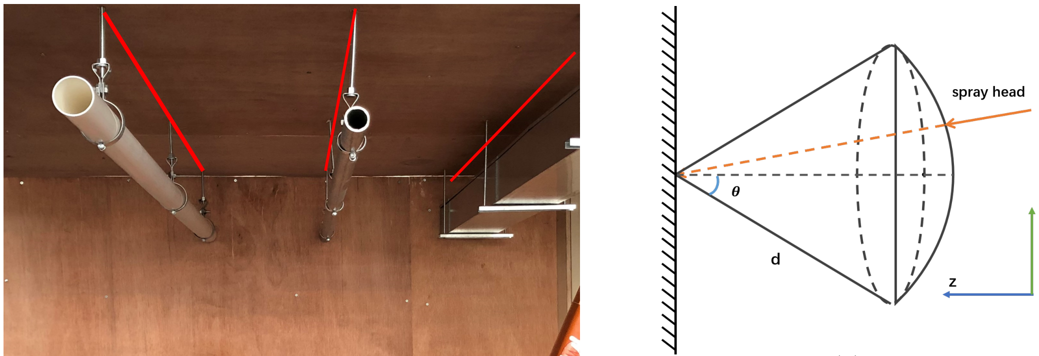
The time duration between two interpolated points is decided by the path length. On the plane, the spray head generally moves in a straight line with no orientation changing, and the path length depends on the linear distance. However, the orientation difference can not be ignored in the corner painting, else the coating may be not uniform. The duration is expressed as:

$$\Delta t = \frac{\Delta x + \eta \Delta\theta}{s} \tag{2}$$

In (2), $\Delta\theta$ is the angle between the two normal vectors, and η is a constant factor to determine the weight of the angular fraction.

C. Sphere Cap Constraint

The building is often equipped with water or circuit pipelines, as in Fig. 4 (a), which remain as obstacles in path planning. The intermediate region between pipelines and the wall is tough to reach because when the nozzle tries to be perpendicular to the wall, the manipulator arm will collide with these pipelines. On the other hand, the pipelines are different from the windows or doors embded in walls as obstacles. Windows impede Cartesian path to pass through, while pipelines prevent the joint path. To solved such complex problem, the constraint could be relaxed so that the nozzle orientation vector could rotate about the normal vector by a tolerant angle θ and the nozzle head moves on the spherical cap with a radius of d, as illustrated by Fig. 4 (b). This relxation takes the point on the wall as the center point of the sphere cap, which keeps the initial Carteisan path of the



(a) (b)

Fig. 4. Wall painting path planning with pipelines suspended. (a): roof wall with pipelines hangging below it. The red paths denote the remaing inaccessible region. (b): the sphere cap constraint. The spray head moves on the sphere cap while keeping to orient to the sphere center.

Algorithm 1 SpherecapConstraint(P_{in}, P_{out})

$d, \alpha, \beta, \gamma, M, N, P, T_{A0}$;
$num \leftarrow P_{in}.Size()$;
G.Init(num);
P_{out}.Init(num);
for $i = 0$ to dim,$m = 0$ to M,$n = 0$ to N,$p = 0$ to P do
  $T_A = T_{A0} \cdot Sample(d, \alpha_m, \beta_n, \gamma_p)$;
  q=InverseKine(T_A);
  $G[i]$.push_back(q);
end for
GraphSearch(G,P_{out});
return

Algorithm 2 ExhaustiveGraphSearch(G, P_{out})

InitCost($G[0]$);
$last\_size \leftarrow G[0]$.size;
$num \leftarrow G$.size;
for $i = 1$ to num do
  for $j = 0$ to G[i].size do
    dis.Init($last\_size$);
    for $k = 0$ to $last\_size$ do
      $dis[k] \leftarrow$ Cost(G[i][j],G[i-1][k])+G[i-1][k].cost;
    end for
    $min_cost, min_id \leftarrow$MinCost($dis$);
    G[i][j].cost$\leftarrow$ min_cost;
    G[i][j].parent_id $\leftarrow$ min_id;
  end for
  $last\_size \leftarrow G$[i].size;
end for
$min_id \leftarrow$MinElement(G.end);
while $num \geq 1$ do
  P_{out}.push_back(min_id);
  $min\_id \leftarrow G[num-1][min\_id]$.parent_id;
end while
Reverse(P_{out});
return

ejected coating while changing the position of the spray head to avoid colliding with the pipelines. We named this as the sphere cap constraint planning. Under the sphere cap constraint, one Cartesian path point corresponds to not a single pose but an area for the end effector pose. Thus we can sample a feasible pose in this area to the corresponding Cartesian point.

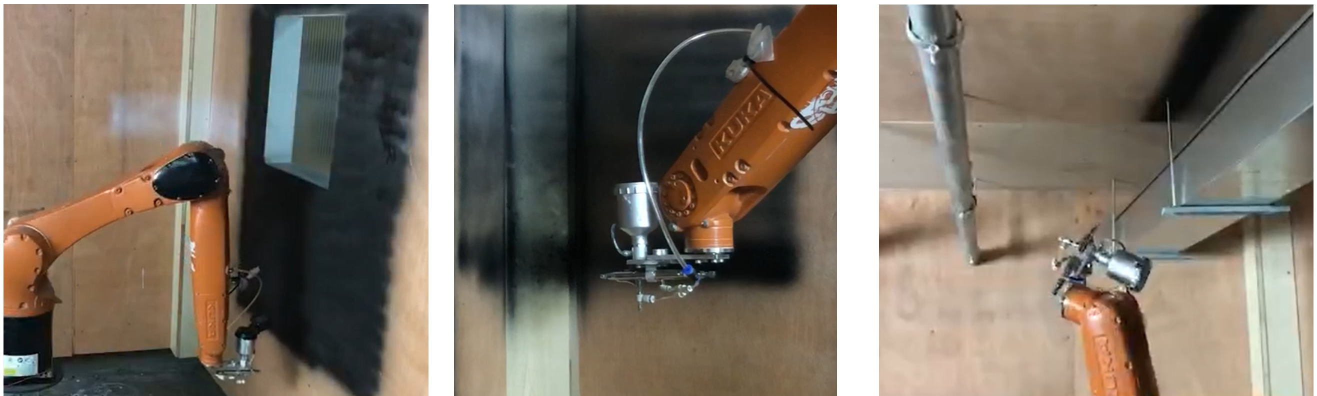
There are two variables to represent spray head on the sphere cap: $\alpha \in [0, \theta], \beta \in [0, 2\pi]$, and a third number $\gamma \in [0, 2\pi]$ to indicate the angle the spray head rotates around its orientation vector. We denote $RotX(a) \in SO3$ as a rotation matrix which means the frame rotates about x-axis by a, and the same principle on $RotY, RotZ$. For convenience, we suppose the sphere center coordinate system T_B has the same orientation as the spray head coordinate system T_A. By default, the z-axis for T_A is towards to the sphere center. Then the transform from spray head frame to sphere center frame is:

$$T_B^A = \begin{bmatrix} 0 & t_B^A \end{bmatrix}, t_B^A = \begin{bmatrix} 0 & 0 & d \end{bmatrix}^T \tag{3}$$

As the position of the sphere center remains unchanged, we could smaple the orientation of T_B and then convert back to T_A by the linear transformation. In the begining, z-axis of T_{A0} and T_{B0} is perpendicular to the wall surface as indicated by Fig. 4 (b) with $\alpha = \beta = \gamma = 0$. When these angles change, the rotation matrix R_{sam} is given as follows:

$$R_{sam} = RotZ(\beta) \cdot RotX(\alpha) \cdot RotZ(\gamma) \tag{4}$$

From the rotation matrix, the spray head frame could be



(a) (b) (c)

Fig. 5. Experiment for wall painting with 6DOF manipulator. (a): painting on the wall with window. (b): painting on the two continuous turns corner. (3): painting on the roof with pipelines.

TABLE I
Sphere Cap Cnstraint Planning Performance

	Random Sample	Exhaustive Graph Search					
θ range	$\pi/6$	$\pi/6$	$\pi/6$	$\pi/6$	$\pi/9$	$\pi/9$	$\pi/9$
angle step	-	$\pi/36$	$\pi/24$	$\pi/12$	$\pi/36$	$\pi/24$	$\pi/12$
time/s	0.107	0.445	0.330	0.184	0.332	0.201	0.162
cost	9.68	3.839	3.923	4.265	4.36	4.877	5.039

derived:

$$\begin{aligned} T_A &= T_{A0} \cdot T_B^A \cdot \begin{bmatrix} & & & 0 \\ & R_{sam} & & 0 \\ & & & 0 \\ & 0 & & 1 \end{bmatrix} \cdot (T_B^A)^{-1} \\ &= T_{A0} \cdot Sample(d, \alpha, \beta, \gamma) \end{aligned} \tag{5}$$

In fact, the random sampling is inefficient. We use a minor steps to discretized the parameter space with $\Delta\alpha, \Delta\beta, \Delta\gamma$ and split the parameter space coordinate axis into m, n and p fractions respectively. For each Cartesian path point, there are $m \cdot n \cdot p$ precomputed candidate frames for spray head. Finally, these candidate frames produce multiple joint values by inverse kinematics. The feasible joint path will be selected among these joint values. This algorithm is illustrated in Algorithm 1.

We use the exhaustive graph searching method to get the joint path with minimum cost, as shown in Algorithm 2. This may be time consuming but could find the global optimal path. Of course, other optimization techniques,such as genetic algorithm and particle swarm optimization could also be used to find the optimal path. In this method the cost function during a path from p to q is defined as:

$$Cost(p, q) = JntDis(p, q) + OriErr(q) \tag{6}$$

The cost contains two parts. One is the distance between these two points in joint space to reduce the joint motion. And another is the error between the wall normal vector and the nozzle orientation vector of q which prevents the nozzle inclining too much .

IV. Experiments

In this section, two experiments are implemented to verify the effectiveness of the proposed methods. The first is a 6DOF KUKA manipulator arm painting in a building under these methods. Another is the sphere cap constraint planning performance comparation in different sampling and searching methods.

A. Wall Painting with 6DOF Manipulator Arm

In this experiment, we use a 6DOF KUKA KR10-R1100-SIXX as the manipulator to hold the spray head doing the painting work, as illustrated in Fig. 1. The distance d away from wall is 0.15m, and the diameter of the ejected coating D is about 0.05m. The Cartesian path step is 0.01 m. The angular offset factor η is 0.5 for the corner planning. In the sphere cap constraint planning, the tolerant angle θ is $\frac{\pi}{6}$ with the angle step segment as $\Delta\alpha = \Delta\beta = \Delta\gamma = \frac{\pi}{36}$. The process of painting is shown in Fig. 5, from which the manipulator could paint the wall dexterously with the assist of the constraint path planner. Even if the area lays under the square pipeline as in Fig. 5 (c), the robot arm could incline its spray head to paint on it.

B. Sphere Cap Cnstraint Planning Performance

To test the sphere cap constraint planning's performance, we compared the random sample and the exhaustive graph search on planning when the pipelines exist as the Fig. 5 (c) shows. From the result shown in Table 1, the random sample method is fast than the exhaustive graph search while the path has a larger cost. In the exhaustive graph search planning, when the angle step decreases, the path obtains a smaller cost with the planning time increasing. A bigger tolerant angle enable the graph search to find a more optimal solution. In a word, the discrete exhaustive search method spends more time to seek a more efficient and feasible path.

V. Conclusions

This paper fully describes the constraint path planning for the wall spray coating robot manipulator and proposes improved approaches for different situations. For the wall with windows or doors, the boustrophedon decomposition will first split the map into multiple cells and then a BFS method will decide the painting order. The Beizer curve fitting is used for smoothing non-differentiable corners to eliminate the irregular inner wall surface. As for the wall equipped with pipelines, a sphere cap constraint planner enlarges the valid sampling range and an exhaustive graph search method will select the feasible path with the lowest cost. The experiment of a KUKA arm painting on a building under the proposed path planner has been implemented and shown its efficient performance. Finally, as the search method guarantees the global optimal while consuming too much time, these is future work to reduce planning time for sphere cap constraint planner.

References

[1] E. Asadi, B. Li and I. Chen,"Pictobot: A Cooperative Painting Robot for Interior Finishing of Industrial Developments" in IEEE Robotics & Automation Magazine, vol. 25, no. 2, pp. 82-94, June 2018.

[2] Huang Y., Carstensen J., Tessmer L., Mueller C. (2019) Robotic Extrusion of Architectural Structures with Nonstandard Topology. In: Willmann J., Block P., Hutter M., Byrne K., Schork T. (eds) Robotic Fabrication in Architecture, Art and Design 2018. ROBARCH 2018.

[3] Kahane B , Rosenfeld Y . Balancing Human-and-Robot Integration in Building Tasks[J]. Computer-Aided Civil and Infrastructure Engineering, 2010, 19(6):393-410.

[4] Berardo N , Alberto G , Alessandro C . Set up of an Automated Multi-Colour System for Interior Wall Painting[J]. International Journal of Advanced Robotic Systems, 2007, 4(4):50-.

[5] Sulaiman, Esraa AbdElrahman Ibraheem Omar, Esraa Muaz MohammedSalih Ahmed, Sara Mahfouz Hassan . Design of an Interior Wall Painting Robot[D]. 2017.

[6] Galceran E , Carreras M . A survey on coverage path planning for robotics[J]. Robotics and Autonomous Systems, 2013, 61(12):1258-1276.

[7] C. Hofner and G. Schmidt, "Path planning and guidance techniques for an autonomous mobile cleaning robot," Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94), Munich, Germany, 1994, pp. 610-617 vol.1.

[8] R. N. De Carvalho, H. A. Vidal, P. Vieira and M. I. Ribeiro, "Complete coverage path planning and guidance for cleaning robots," ISIE '97 Proceeding of the IEEE International Symposium on Industrial Electronics, Guimaraes, Portugal, 1997, pp. 677-682 vol.2.

[9] Jean-Claude Latombe. 1991. Robot Motion Planning. Kluwer Academic Publishers, Norwell, MA, USA.

[10] Choset H . Coverage of Known Spaces: The Boustrophedon Cellular Decomposition[J]. Autonomous Robots, 2000, 9(3):247-253.

[11] Acar, E. U., Choset, H., Rizzi, A. A., Atkar, P. N., & Hull, D. (2002). Morse Decompositions for Coverage Tasks. The International Journal of Robotics Research, 21(4), 331–344.

[12] Wong S , Macdonald B . A topological coverage algorithm for mobile robots[C]// IEEE/RSJ International Conference on Intelligent Robots & Systems. IEEE, 2003.

[13] Dugarjav, BatsaikhanLee, SoonGeulKim, et al. Scan Matching Online Cell Decomposition for Coverage Path Planning in an Unknown Environment[J]. International Journal of Precision Engineering & Manufacturing, 2013, 14(9):1551-1558.

[14] Strimel G P , Veloso M M . Coverage planning with finite resources[C]// IEEE/RSJ International Conference on Intelligent Robots & Systems. IEEE, 2014.

[15] José Prado, Marques L . Energy Efficient Area Coverage for an Autonomous Demining Robot[J]. Advances in Intelligent Systems and Computing, 2014, 253:459-471.

[16] Khan, Amna et al. "On Complete Coverage Path Planning Algorithms for Non-holonomic Mobile Robots: Survey and Challenges." J. Inf. Sci. Eng. 33 (2017): 101-121.

[17] Waanders, Marten. "Coverage Path Planning for Mobile Cleaning Robots." (2011).

[18] Marija Dakulović, Sanja Horvatić, Ivan Petrović. Complete Coverage D* Algorithm for Path Planning of a Floor-Cleaning Mobile Robot[C]// 2011:5950-5955.

[19] Viet H H , Dang V H , Laskar M N U , et al. BA*: an online complete coverage algorithm for cleaning robots[J]. Applied Intelligence, 2013, 39(2):217-235.

[20] Lee T K , Baek S H , Choi Y H , et al. Smooth coverage path planning and control of mobile robots based on high-resolution grid map representation[J]. Robotics & Autonomous Systems, 2011, 59(10):801-812.

[21] Zhou K , Leck Jensen A , S?Rensen C G , et al. Agricultural operations planning in fields with multiple obstacle areas[J]. Computers and Electronics in Agriculture, 2014, 109:12-22.

[22] Yakoubi M A , Laskri M T . The path planning of cleaner robot for coverage region using Genetic Algorithms[J]. Journal of Innovation in Digital Ecosystems, 2016:S2352664516300050.