24th International Symposium on Automation & Robotics in Construction (ISARC 2007)
Construction Automation Group, I.I.T. Madras

A 3-DIMENSIONAL FORCE FIELD METHOD FOR ROBOT COLLISION AVOIDANCE IN COMPLEX ENVIRONMENTS

P. Chotiprayanakul, D. K. Liu, D. Wang and G. Dissanayake
ARC Centre of Excellence for Autonomous Systems
Faculty of Engineering, University of Technology, Sydney
Broadway, NSW 2007, Australia
{pchotipr, dkliu, Da-Long.Wang, gdissa}@eng.uts.edu.au

ABSTRACT

This paper proposes a three-dimensional force field (3D-F^2) method for efficient motion planning and collision avoidance of a 6DOF manipulator in complex and dynamic environments while keeping the planned end-effector's path and speed unchanged. The 3D-F^2 is defined as ellipsoid shapes covering selected links of a manipulator. When the manipulator moves and its ellipsoid force field approaches to an obstacle in a tolerant range, a repulsive force will be generated and considered in the robot kinematic and dynamic analyses. In infrastructure maintenance, spray-painting and sand-blasting operations require that the operating spot "moves" smoothly and continuously along planned path on a work surface at a constant speed, and allow changes in length and orientation of the spray/blasting stream. Thus, the stream is supposed to be another link and the end of stream performs as a spherical joint fixed on the target surface. Various simulations in a construction area show that the 3D-F^2 can retain the operating path and effectively avoid potential collisions.

KEYWORDS

Force Field, 3D-F^2, Collision Avoidance

1. INTRODUCTION

This paper addresses the motion planning and collision avoidance problem for a robot arm in complex environments. Path/motion planning includes off-line and real-time path planning that generates paths automatically for a robot arm to move from one position to another. When planned paths are executed by a robot arm, the robot control system needs to generate a collision-free trajectory and drive every joint of the robot safely and smoothly. Collision avoidance for a whole robot manipulator has been considered as a very important issue, especially in complex and dynamic environments. Distance measuring is a procedure, before collision avoidance, to determine the possibility of collisions. As computing the distance from a manipulator to obstacles using accurate topology is very time consuming, many researchers have been trying to find efficient approaches to speed up the calculation, for example, covering the links of a robot manipulator with polytope, polyhedron [5], [6], sphere [7], [8], [10], [11], ellipsoid, etc., which can significantly reduce computing time. Based on the measured distance between the robot arm and obstacles in the environment, collision avoidance algorithms are then used to steer the movement of a robot arm away from obstacles while fulfilling its task. Many techniques for this objective, for example the Configuration Space (C-Space) method [2], [3], potential field approach [4], [9] and force field method [11], [12], have been developed.

Path/motion planning and collision avoidance have been extensively investigated. Biegelbauer et al. generalised the painting path planning process [1], which is capable of generating collision-free trajectory for the end-effector. Lozano-Perez presented approaches to distance measuring

between polyhedrons which are used for path planning in discrete C-space [2], [3]. Khatib applied the potential field concept for real-time obstacle avoidance for manipulators and mobile robots [4]. Juang applied polyhedrons to represent a manipulator and potential field method to dynamically control manipulators [5], [6]. Greenspan and Burtnyk modelled a robot arm and obstacles with sets of spheres, measured the distances with a weighted voxel map and demonstrated successful collision avoidance and on-line path planning [7], [8]. Lin and Chuang used potential fields in 3-D workspace to generate a collision free path by locally adjusting the robot configuration for minimum potential [9]. Brock and Khatib presented elastic strip framework generating an obstacle bended free space tunnel for a robot to move from point to point [10]. Xie et al. used spheres covering a robot and converted the repulsive forces to joint torques to find the collision free trajectory [11]. D.Wang et al. presented a variable speed force field (VSF^2) method in 2D for multi-mobile robot collaboration and collision avoidance [12]. A relationship between speed and repulsive force is defined to reduce the oscillation in robot speed and orientation.

In operations such spray-painting, the end-effecter of a manipulator is required to move at constant speed along pre-planned path while considering the collision avoidance of the whole robot arm. Collision avoidance for the whole body of a robot arm has not been extensively studied. We propose a 3D-F^2 method in this research. The definition of ellipsoidal three-dimensional Force Field (3D-F^2) covering the links of a manipulator are introduced in Section 2. Section 3 presents a robot model and the approach using 3D-F^2 to fulfil a task. Simulations carried out in a maintenance environment shows that the 3D-F^2 effectively protects the manipulator from possible collisions with obstacles in a complex environment. The remarkable computing time shows that the 3D-F^2 method can be used in real-time applications.

2. DEFINITION OF THE 3D-F^2

This section introduces the definition of the 3D-F^2. To determine the ellipsoid covering a link of manipulator, two points on a link are selected as the foci of the resulted ellipsoid (P_1 and P_2 in Fig. 1). Transformations from the robot joint coordinate system to the global system are given by transfer function ($^i\mathbf{T}_j$) and rotation-translation metrics ($^i\mathbf{A}_j$). $\mathbf{P}_{i(u,v,w)}$ and $\mathbf{P}_{i(x,y,z)}$ are points on the robot's coordinate system and the global system, respectively.

$$ {}^0\mathbf{T}_{n^{th}} = \prod_{m=0}^{n^{th}-1} {}^m\mathbf{A}_{m+1} \tag{1} $$

$$ \mathbf{P}_{i(x,y,z)} = {}^0\mathbf{T}_{n^{th}}\,\mathbf{P}_{i(u,v,w)} \quad ; \text{where } i = (1,2) \tag{2} $$

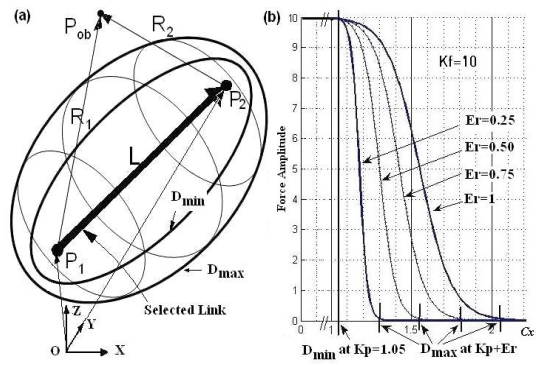

Figure 1 (a) Factors of 3D-F^2 (b) Amplitude of Repulsive Force Between D_{min} and D_{max}

The equations of D_{min} and D_{max} ellipsoids are given in (3) and (4). To ensure that this D_{min} ellipsoid will cover the whole body of the link, the length of the major axis is set to be equal to L*K_p, where L is the distance between the foci and K_p is a constant larger than 1. For any point in the 3D space (P_{ob} in Fig. 1), if the sum of distances, R_1 and R_2, between this point to the foci is equal to the length of the major axis, that is, L*K_p, this point is on the D_{min} ellipsoid. If R_1+R_2 is smaller than L*K_p, this point is inside D_{min}. On the contrary, when R_1+R_2 is larger than L*K_p, the point is outside D_{min} (see Fig. 1). To take the linear speed of the end-effecter into account, we introduce a new factor, E_r, which is the ratio of the instant speed (V_i) and the maximum speed (V_{max}), so E_r is between 0 to1. The length of major axis of D_{max} is set to be L*(K_p+E_r).

$$ \mathbf{L} * K_p = \left|\vec{\mathbf{R}}_1\right| + \left|\vec{\mathbf{R}}_2\right| \tag{3} $$

$$ \mathbf{L} * (K_p + Er) = \left|\vec{\mathbf{R}}_1\right| + \left|\vec{\mathbf{R}}_2\right| \tag{4} $$

$$ E_r = \frac{V_i}{V_{max}} \tag{5} $$

From (3) and (4), if we define *Cx*, is a ratio between $\mathbf{R}_1$+$\mathbf{R}_2$ and **L**, the K_p and K_p+*Er* will

indicate if $\mathbf{P}_{ob}$ is inside or between or outside the D_{min} ellipsoid and the D_{max} ellipsoid.

$$Cx = \frac{|\vec{\mathbf{R}}_1| + |\vec{\mathbf{R}}_2|}{|\mathbf{L}|} \tag{6}$$

Then, the amplitude of repulsive force is given by

$$|\vec{\mathbf{F}}_{rep}| = Kf - \left(\frac{Kf}{1 + e^{\left(-\frac{10\,(Cx - Kp - 0.5\,Er)}{Er} \right)}} \right) \tag{7}$$

Kf is maximum repulsive force. Fig 1b shows how the repulsive force varies with E_r and the position between D_{min} and D_{max}. This definition ensures the smooth change of the force. The direction of the repulsive force is defined to be the unit vector that points from $\mathbf{P}_{ob}$ to $\mathbf{P}_1$ or $-R_1$.

$$\vec{\mathbf{F}}_{rep} = \frac{\vec{\mathbf{P}}_1 - \vec{\mathbf{P}}_{ob}}{|\vec{\mathbf{P}}_1 - \vec{\mathbf{P}}_{ob}|} * |\vec{\mathbf{F}}_{rep}| \tag{8}$$

3. KINEMATICS AND DYNAMICS

In the section, we analysis the manipulator's kinematics and dynamics in operations such as spray-painting and then introduce how the 3D-F^2 method can be applied for planning and collision avoidance.

3.1 Model of Robot

In a spraying operation, the end effector of the manipulator needs to follow a pre-planned path and move continuously. To maintain the target point, the manipulator model has to be adapted by assuming a virtual link from the robot end-effecter to the target point and a virtual joint at the spraying spot which contains three rotating axes ($R_{z,\psi}R_{y,\varsigma}R_{x,\rho}$). Thus, the model of the robot arm can be represented in Figure 2. This modified model has two fixed pivots at both ends of a serial chain-link. The stream pivot is an instant point that moves along a pre-planned path.

3.2 The Effects of Repulsive Force

When an obstacle or any part of an obstacle is inside of the D_{max} ellipsoid of the virtual stream link, a repulsive force will be generated on a point that has the smallest *Cx* and has maximum repulsive force. Moment ($\vec{\mathbf{M}}_{1,\psi,\varsigma,\rho}$), which is a cross product of the distance vector of stream direction ($\vec{\mathbf{l}}_{\text{stream link}}$) and the force ($\vec{\mathbf{F}}_1$), is defined by (11) and will change the stream direction. The repulsive force at the lower arm ($\vec{\mathbf{F}}_2$) is defined from the nearest point of an obstacle and is toward to the end of lower arm at $\mathbf{P}_{1,2}$.

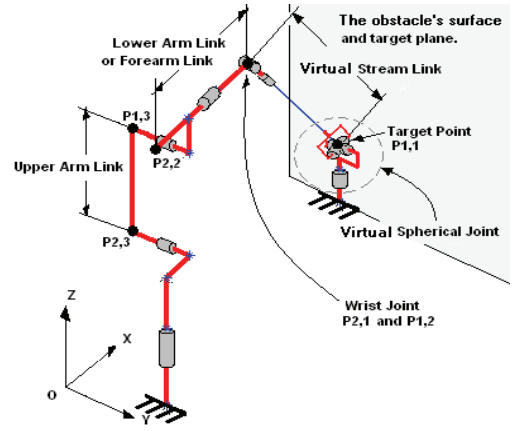

Figure 2 The Stream Link, Virtual Spherical Joint and Points of the Robot Arm

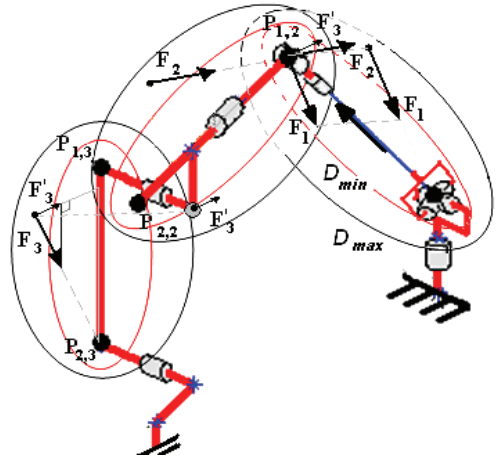

Figure 3 Repulsive Forces on the Links of a Robot Arm

$$\vec{\mathbf{M}}_{1,\psi,\varsigma,\rho} = \begin{bmatrix} M_{1,\psi} \\ M_{1,\varsigma} \\ M_{1,\rho} \end{bmatrix} = \vec{\mathbf{l}}_{\text{stream link}} \times \vec{\mathbf{F}}_1 \tag{9}$$

As shown in Figure 3, $\vec{\mathbf{F}}_2$ is going to change the stream link direction by pushing or pulling the wrist of the robot arm. In a different way, the repulsive force of the upper arm ($\vec{\mathbf{F}}_3$) is divided into two perpendicular vectors, one points to the upper arm link at $P_{2,3}$ the shoulder joint and another ($\vec{\mathbf{F}}_3'$) to its perpendicular plane. The force would have

an effect to the rotation of shoulder indirectly because it could make the robot missing the target. While the force $\vec{F}_3'$ is pushing the elbow of the robot, there is a reaction against the force. With this force, the forearm transfers the $\vec{F}_3'$ in parallel along itself to push the stream as well as the repulsive force from lower arm.

$$\vec{\mathbf{M}}_{2,\psi,\varsigma,\rho} = \vec{\mathbf{l}}_{\text{stream link}} \times \vec{\mathbf{F}}_2 \tag{10}$$

$$\vec{\mathbf{M}}_{3,\psi,\varsigma,\rho} = \vec{\mathbf{l}}_{\text{streamlink}} \times \vec{\mathbf{F}}_3' \tag{11}$$

$$\vec{\mathbf{M}}_{rep\ \psi,\varsigma,\rho} = \sum_{i=1}^{3} \vec{\mathbf{M}}_{i,\psi,\varsigma,\rho} \tag{12}$$

If there are no obstacles, repulsive forces become zero, the end-effecter of the robot end-effecter has to return to the planned orientation as the result of a virtual torque $\vec{\mathbf{M}}_{sp,\psi,\varsigma,\rho}$ defined by (13). $\mathbf{T}_{\psi,\varsigma,\rho}$ and $\mathbf{A}_{\psi,\varsigma,\rho}$ are the planned orientation and the actual orientation of the end-effecter, respectively, and *Ksp* is a virtual spring constant.

$$\vec{\mathbf{M}}_{sp,\psi,\varsigma,\rho} = Ksp*(\mathbf{A}_{\psi,\varsigma,\rho} - \mathbf{T}_{\psi,\varsigma,\rho}) \tag{13}$$

Thus the torques are applied to the robot arm will be given by

$$\vec{\mathbf{M}}_{\psi,\varsigma,\rho} = \begin{cases} \vec{\mathbf{M}}_{rep_{\psi,\varsigma,\rho}} & ;\text{one of } \mathbf{F}_{i,\text{rep}} > 0 \\ \vec{\mathbf{M}}_{sp_{\psi,\varsigma,\rho}} & ;\text{all of } \mathbf{F}_{i,\text{rep}} = 0 \end{cases} \tag{14}$$

3.3 Inverse Kinematics and Dynamics

The dynamics of the stream link can be described with spatial motion equations (16-18). Parameters in the equations include target position ($\mathbf{S}_t$), start position ($\mathbf{S}_s$), current position ($\mathbf{S}_c$), previous position ($\mathbf{S}_p$), speed (v_p), differential time (dt), direction of the path ($\mathbf{D}$), current orientation ($\mathbf{O}_c$), previous orientation ($\mathbf{O}_p$) and the mass property of the virtual stream link (*Km*).

$$\mathbf{D} = \frac{\mathbf{S}_t - \mathbf{S}_s}{\|\mathbf{S}_t - \mathbf{S}_s\|} \tag{15}$$

$$\mathbf{V} = v_p*\mathbf{D} \tag{16}$$

$$\mathbf{S}_c = \mathbf{S}_p + \mathbf{V}*dt \tag{17}$$

$$\mathbf{O}_c = \mathbf{O}_p + Km*0.5*\vec{\mathbf{M}}_{\psi,\varsigma,\rho}*dt^2 \tag{18}$$

The poses of the manipulator joints, [$\mathbf{\Theta}$], can be obtained from the position and orientation in the Cartesian space [x y z ψ ζ ρ] by applying inverse kinematics (IK):

$$[\mathbf{\Theta}] = \boldsymbol{IK}\ (\ [\mathbf{S}_c\ \ \mathbf{O}_c], [\mathbf{\Theta}_{\text{previous}}]\ ) \tag{19}$$

[$\mathbf{\Theta}_{\text{previous}}$] is the joint poses in the previous step, which is used in pose selection to reduce jerk in joint movement. The angles of the first three joints, shoulder to elbow, come from the coordinate of the wrist joint ($\mathbf{P}_{2,1\ \text{current}}$) from (20). Input the first three joints' angles, Θ_1, Θ_2, Θ_3 to ${}^0\mathbf{T}_6$ to work out ${}^3\mathbf{T}_6$ in term of [$\mathbf{n'\ s'\ a'}$] in (23) and compare it with ${}^3\mathbf{T}_6$ in (23) from forward function to get $\Theta_4,\Theta_5,\Theta_6$.

$$\begin{bmatrix} \mathbf{P}_{2,1\ current} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{z,Oc(\psi)} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{y,Oc(\varsigma)} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{x,Oc(\rho)} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -l_{\text{streamlink}} \\ 1 \end{bmatrix} + \begin{bmatrix} Sc \\ 1 \end{bmatrix} \tag{20}$$

$${}^0\mathbf{T}_3\,{}^3\mathbf{T}_6 = {}^0\mathbf{T}_6 = \mathbf{R}_{z,Oc(\psi)}\mathbf{R}_{y,Oc(\varsigma)}\mathbf{R}_{x,Oc(\rho)} = \begin{bmatrix} n_x & s_x & a_x \\ n_y & s_y & a_y \\ n_z & s_z & a_z \end{bmatrix} \tag{21}$$

$${}^3\mathbf{T}_6 = \frac{{}^0\mathbf{T}_3^{\,T}}{\left|{}^0\mathbf{T}_3\right|} \begin{bmatrix} n_x & s_x & a_x \\ n_y & s_y & a_y \\ n_z & s_z & a_z \end{bmatrix} = \begin{bmatrix} n'_x & s'_x & a'_x \\ n'_y & s'_y & a'_y \\ n'_z & s'_z & a'_z \end{bmatrix} \tag{22}$$

$${}^3\mathbf{T}_6 = {}^3\mathbf{A}_4\,{}^4\mathbf{A}_5\,{}^5\mathbf{A}_6 \tag{23}$$

4. SIMULATION AND RESULTS

Simulations are carried out in an infrastructure maintenance environment. A manipulator mounted on a lift platform is supposed to spray-paint certain area of a maintenance site. The manipulator model in these simulations is Denso 6556 [13]. The simulation program runs on MATLAB7.1 in a Pentium 4 computer with a 2.8GHz CPU and 1GB RAM. The values of parameters in the simulations are set to be v_p = 30 mm/sec, *dt* = 0.1 sec, *Km*=2, K_f=[50 20 20], K_p=[1.01 1.1 1.1], *Er*=[0.025 0.15 0.15] and *Ksp*= 2.

4.1 Case 1

The robot is lifted under a ceiling and spray-paints the surface of the ceiling. Figure 4 shows simulation snapshots of the robot arm with 3D force field, D_{min}, D_{max}, and the end effector's paths. Points 1 to 4 are a pre-planned path denoted by the solid blue line in Fig. 4a. The spray-painting direction is set to be normal to the ceiling surface

in this task. The robot performs a motion that avoids crashing into a flank on the wall and a hanging ventilator beside it. Figure 5 shows the changes of joint angles from point 1 to 4 in Fig 4a. The end-effecter of the manipulator traveled 4600 mm in total in 153 seconds. The computation time required in this simulation is only 63 seconds.

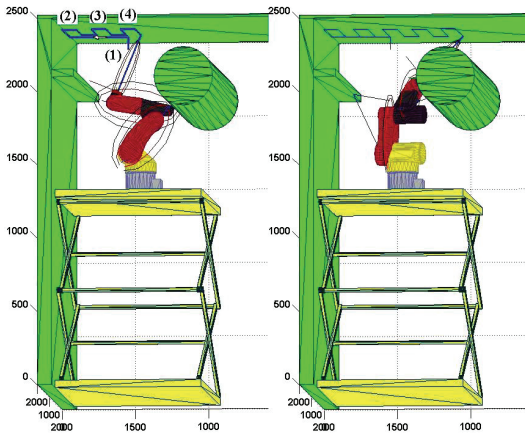

Figure 4 Case1: Ceiling-Spraying Task

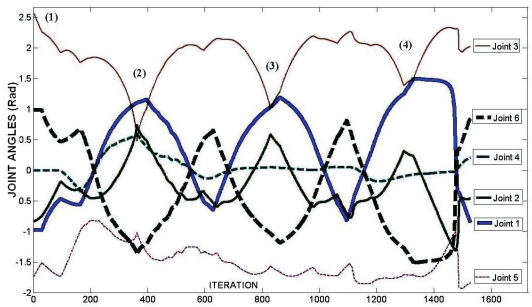

Figure 5 Angles of Joints Along the Path (Case 1)

4.2 Case 2

In this simulation, a vertical wall is painted by the robot (Fig. 6) and the resulted joints' angles are shown in Fig. 7. The path is set upon a vertical side of the wall. The simulation is carried out using the same end-effecter speed and the same force field parameters as those in Case 1, except that stream length is set as 400mm the end-effecter of the manipulator traveled 2750 mm in total in 92 seconds and the computation time required is 46 seconds only.

4.3 Case 3

The robot paints a hanging ventilator (Fig. 9) and joints' angles are shown in Fig. 10. The path is set upon the surface of the cylindrical ventilator. The simulation is carried out using the same end-effecter speed and the same force field parameters as those in Case 1, except that stream length is set as 350mm the end-effecter of manipulator traveled 2300 mm in 77 seconds and the computation time required is 32 seconds.

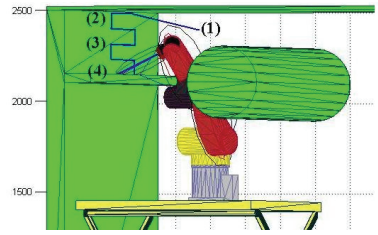

Figure 6 Case2: Wall Spraying Task

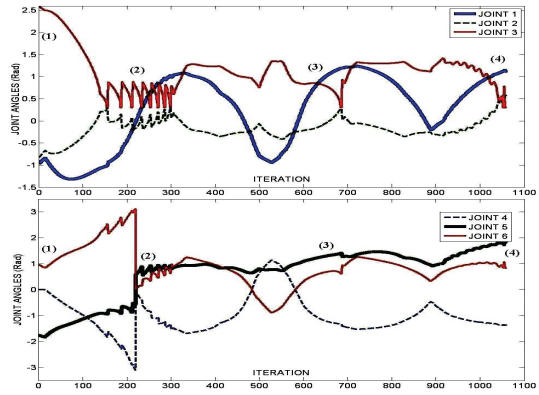

Figure 7 Angles of Joints Along the Path (Case 2)

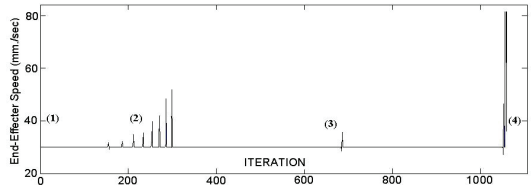

Figure 8 End-Effecter Speed in Case2

4.4 Discussion

The simulation results show that the 3D-F^2 method is able to prevent the robot arm from collisions with obstacles in a complex environment. It also can be seen clearly that the 3D-F^2 method is fast enough to be implemented in the real-time trajectory planning and collision avoidance for manipulators. It can be seen from Fig. 7 and Fig. 10 that the angle changes of some joints are not always smooth although the path is a straight line. Oscillations occur when the robot arm reaches to

some corner areas, such as point 2 in Fig. 6 and point 4 in Fig. 9. In the case of Fig 6, the force field of the stream needs to protect the stream link from obstacles from several directions, i.e. ceiling and wall. The stream link will suffer the repulsive forces from ceiling and wall alternatively, which causes the rapid change of the orientation of stream link and the linear speed of stream link on the working surface (see Fig. 8 and Fig. 11). As a result, the spraying point will fluctuate around the target point. Please note that these fluctuations are instantaneous and only 1 or 2 iterations or 0.1 to 0.2 seconds. They can be easily reduced by applying a damper factor. Besides, there are a few situations that the robot arm reaches its joint limit or workspace limit. These problems will be addressed in our future research work.

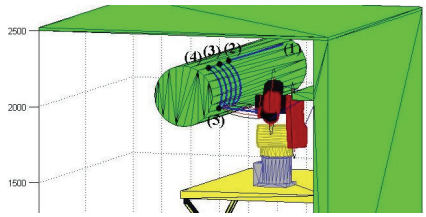

Figure 9 Case3: Ventilator Spraying Task

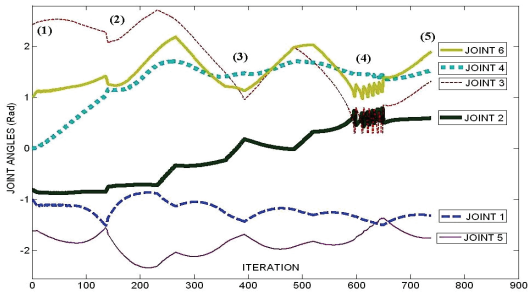

Figure 10 Angles of Joints Along Path (Case3)

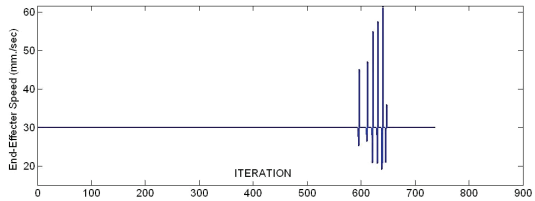

Figure 11 End-Effecter Speed of Case3

5. CONCLUSION AND FUTURE WORK

This paper presented a novel 3D-F^2 method for real-time collision avoidance and motion planning of manipulators in known complex and dynamic environments. With the 3D-F^2, a manipulator is capable of moving its end-effector along a pre-planned path in a constant speed while avoiding obstacles simultaneously. Simulations carried out in a complex environment demonstrate the efficiency and feasibility of the 3D-F^2. In a spray operation, stream length is allowed to change in a certain range, which introduces another degree of freedom. In the next stage of research, the stream link will be represented with a prismatic joint. Further experiments with a real robot in a real working environment will also be conducted.

6. ACKNOWLEDGMENT

This work is supported in part by the ARC Centre of Excellence for Autonomous System, funded by Australian Research Council (ARC) and the New South Wales State Government, Australia.

7. REFERENCES

[1] G. Biegelbauer, A. Pichler, M. Vincze, C. Nielsen, H. J. Andersen and K. Haeusler (2005) Automatic Generation of Robot Painting Motions for Unknown Parts, IEEE Robotics & Automation Magazine, 1070-9932,pp.24-34.

[2] T. Lozano-Perez (1983) Spatial Planning: A Configuration Space Approach, IEEE Transactions on Computers, vol. C-32, no.2, pp.108-119.

[3] T. Lozano-Perez (1987) A Simple Motion-Planning Algorithm for General Robot Manipulators, IEEE Journal of Robotics and Automation, vol. RA-3, no.3, pp. 224-238.

[4] O. Khatib (1985) Real-Time Obstacle Avoidance for Manipulators and Mobile Robots, IEEE Int. Conf. on Robotics and Automation, pp.500-505.

[5] J.-G. Juang (1998) Collision Avoidance using Potential Fields ,Int. J. of Industrial Robot, MCB Uni Press, vol. 25,pp.408-415.

[6] J.-G. Juang (2004) Application of Repulsive Force and Genetic Algorithm to Multi-manipulator Collision Avoidance Proceeding of 5th Asian Conference, vol 2, pp.971-976.

[7] M. Greenspan and N. Burtnyk (1996) Obstacle Count Independent Real-Time Collision Avoidance, IEEE Int. Conf. on Robotics and Automation, Minnesota, pp.1073-1080.

[8] M. Greenspan and N. Burtnyk (1994) Real Time Collision Detection, US Patent, Patent Number 5,347,459.

[9] C.-C. Lin and J. Chuang (2003) Potential-Based Path Planning for Robot Manipulators in 3-D Workspace, Int. Conf. on Robotics Automation, Taiwan, pp.3353-3358.

[10] O. Brock and O. Khatib (1999) Real-Time Obstacle Avoidance and Motion Coordination in a Multi-Robot Workcell, IEEE International Symposium on Assembly and Task Planning, Porto, Portugal, pp. 274-279.

[11] H. P. Xie, R. V. Patel, S. Kalaycioglu, H. Asmer (1998) Real-Time Collision Avoidance for Redundant Manipulator in an Unstructured Environment, IEEE Int. Conf. on Intelligent Robots and Systems, Victoria, B.C, Canada.

[12] D. Wang, D. Liu, G. Dissanayake (2006) A Variable Speed Force Field Method for Multi-robot Collaboration, Int. Conf. on Intelligent Robots and Systems, Beijing, China.

[13] http://www.denso-wave.com/en/robot/