A 3D Path Following Control Scheme for Robot Manipulators

Yalun Wen and Prabhakar Pagilla [*]

[*] *Mechanical Engineering Department, Texas A&M University, USA,
(email: ywen27@tamu.edu, ppagilla@tamu.edu)*

Abstract: We describe a novel path following control scheme for robot manipulators where constant tool velocity of travel on a surface is desirable. The path following scheme is applicable to general situations where the surface geometry is typically given in terms of measured data from a sensor. Considering the measured data points as control points, we utilize a cubic spline interpolation to generate a closed-form geometric description for the 3D path. Since joint velocity control is quite common in many industrial robots and most surface finishing tasks require travel with constant velocity along the path, we consider a 3D kinematic model for the end-effector with control inputs as rate of change of orientation and translational velocity that is locally tangent to the surface along the path. By utilizing a path variable and the tangent vector along the path, we formulate a converging path as the path that is traversed from a given robot end-effector position to the desired path and subsequent travel on the desired path on the surface. To evaluate the performance of the scheme, we have conducted a number of real-time experiments on a six degree-of-freedom industrial robot for several paths which can be employed for sanding of structures or deburring of large industrial cast parts and gears.

Keywords: Path following, Robotic surface finishing, Arc-length based path parameterization.

1. INTRODUCTION

Many problems in robotics involve the end-effector to follow an ordered set of control points. The control points can be generated from trajectory optimization schemes (see King et al. (2013)) or from 3D scanners or proximity sensors as discrete data obtained from scanning of freeform surfaces as suggested in Wen et al. (2019). In this paper, we consider robotic tasks (such as sanding and grinding) where the robot end-effector is expected to travel on the surface with a constant tangential velocity to facilitate uniform material removal and uniform finished surfaces. The problem has been extensively studied as a constrained trajectory planning problem. When the trajectory of the set of control points is generated with time-space parameterization (see Ezair et al. (2014)), then timed-trajectory tracking is utilized. In the case of trajectory tracking, Xiang et al. (2011), the robot is forced to catch up with the evolving reference on the time-parameterized trajectory. As pointed out in Encarnaçao and Pascoal (2001), aggressive dynamic behavior and saturation of actuator can occur during the trajectory tracking operation. In addition, a time-parameterized signal is often difficult to execute and maintain from a temporal viewpoint as disturbance and/or process/workpiece uncertainty would result in deviation from this trajectory. As an alternative, path following is seen as more suitable and advantageous for such tasks for several reasons. First, the path generation task from the list of control points is purely a geometric task, and a time-parameterized trajectory generation algorithm is not needed. Second, from Lapierre et al. (2006), a path following strategy usually results in smoother convergence and higher accuracy than trajectory tracking; for example, Aguiar et al. (2008) points out that the fundamental limitations imposed on trajectory tracking by unstable zero dynamics do not apply to the path following problem; the control problem of following a geometric path is less restrictive than trajectory tracking since a path following strategy allows the freedom to design a timing law.

As discussed in Aguiar et al. (2005), given a geometric description of the desired path, the path following problem is concerned with the design of a control law without a time parameterization associated with the path. For the geometric task, when the path curve relies on a parameterization, one path following scheme is provided in Skjetne et al. (2004); a different formulation can be found in Nielsen et al. (2010) where transversal control addresses the geometric task and the tangential control addresses the dynamic task. Many path following problems are studied using the Frenet-Serret Frames, such as the one in Lapierre et al. (2006). Depending on how the reference point is defined, different control strategies can be selected, see Lapierre and Jouvencel (2008). In Micaelli and Samson (1993), the target point is designed to be the closest point on the path, relative to the current position of the object; Lapierre et al. (2006) treats the target point as a virtual moving point. A path following scheme based on model predictive control can be found in Kanjanawanishkul et al. (2010). However, one aspect that has been ignored in existing path following schemes is that the designed path does not guarantee that the end-effector would reach the desired path with appropriate tool orientation along the path; it is often assumed in path following that the convergence direction is normal to the desired path. Although this allows for rapid convergence to the path, in the case of many robotic tasks, the end-effector position overshoot or uncertainty in the part location can cause transition problems as discussed in Pagilla and Yu (2001b). This issue is of particular interest to the robotic surface finishing community (see Pagilla and Yu (2001a)) since without a well-designed path for the end-effector to follow, any contact with the surface, especially if the path has a velocity component normal to the surface, will create impact at contact which may lead to workpiece damage.

Based on the given scanned freeform surface data, we consider the class of admissible paths to be spline-interpolated curves. Splines are extensively used to model complex freeform surfaces such as complex dies, aerospace and automotive parts since splines could be used to obtain smooth paths, and therefore spline-interpolated paths can result in better surface quality, reduced vibration and can provide longer tool life, see Siu (2011) and Cohen et al. (2001). Therefore, the proposed path following strategy with spline interpolated paths is suitable for a number of robotic path following applications. A 2D path following strategy for robot end-effectors was presented in Wen and Pagilla (2020). In this work, we discuss the limitations of 2D path following and develop a 3D path following strategy.

In this paper, given a list of control points, we consider an interpolated-spline as the general path. An arc-length based spline where arc-length is the path parameter ensures that the path curve is of unit speed. For our purpose, we require a path curve to be of some constant speed. Employing a scaled arc-length as the path parameter, we generate a desired path on the surface that ensures the curve of the given constant speed, which ensures the resulting path curve to be of fixed speed. The proposed path following strategy constructs a novel converging path, the path from a given end-effector position onto the desired path. We utilize a numerical projection method to relate the current end-effector position to the one on the desired path via the path variable, which ensures that the end-effector would reach the desired path with the correct tool surface orientation along the desired path. To enable control of end-effector translational velocity to be a constant and orientiation that ensure the end-effector is normal to the path, we model the end-effector motion using a 3D kinematic model. We corroborate the effectiveness of the approach via a number of experiments with a six degree-of-freedom (DOF) industrial robot for following a path on an aircraft blade surface and a gear profile.

In the following, section II provides the problem description. The construction of the desired 3D path and the converging path are described in Section III. Section IV describes the kinematic model used to describe the end-effector motion and derives a control law to achieve motion along the converging path with the desired orientation of the end-effector. Experimental results on an six-degree-of-freedom industrial robot are presented and discussed in V. Concluding remarks are provided in Section VI.

2. PROBLEM DESCRIPTION

Fig. 1 shows the profile of an aircraft blade with a sample path that can be followed for sanding of the blade surface to cover the surface; the dotted points on the path are the control or measured points and one sample finishing path is shown by the green curve. Since the entire freeform surface is treated uniformly and the extent of material being removed at each location should be uniform, the end-effector (surface finishing tool) should travel on the surface with a constant translation velocity, and the direction of the translation velocity should be aligned with the tangent of the path constructed from the list of points; thus, the end-effector orientation should change along with the surface geometry. Since most of the engineered products have freeform surfaces, such as turbine blades, car bodies and boat hulls, there is a significant need for the development of a 3D path following scheme that can be employed to finish these surfaces with robots.

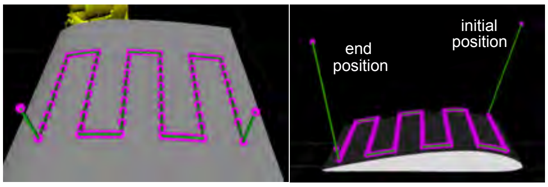


Fig. 1. Path on a workpiece surface.

Given a list of control points on the freeform surface, the main task is to follow a parameterized desired path. The problem to be solved is how to command the robot end-effector to converge and stay onto the desired path with a constant translation velocity. We extend the idea of path following such that there always exist a converging path for the robot end-effector to follow at any moment of time: when the end-effector lies on the desired path, the tangent of the converging path is aligned with the local tangent on the desired path; when the end-effector is away from the desired path, the converging path points towards the desired path. Since we would like the end-effector to have constant velocity of travel, we impose a kinematic model for the end-effector with constant velocity of travel, and design a control law that acts on the end-effector orientation to drive the direction of the velocity vector to the desired path by following the tangent of the vector field formulated from the converging path. It should be noted that the proposed method is developed for commercial articulated manipulators where the joint velocity can be commanded; this is the case for most industrial robots which are either position or velocity controlled and joint torque is not accessible to the end-user.

3. 3D PATH CONSTRUCTION

In this section, we will provide a novel method for the construction of the desired path and the design of the converging path that the end-effector must follow from its current position to reach the desired path.

3.1 Construction of the desired path

Suppose that $(n+1)$ control points from the initial to the end position are given by $(x_k, y_k, z_k), k = 0 : n$. First, we will consider the cubic-spline interpolation between these control points using a parameterization. The spline interpolation allows for retrieval of the geometric properties, such as local tangent at a point originally not included in the list of control points and the overall path arc-length. Let ϕ be the path parameter associated with this parameterization. The interpolated path is a differentiable function $\sigma(\phi) : K \to \mathbb{R}^3$ from an open interval K into $\mathbb{R}^3$ given by:

$$\sigma(\phi) = (\tilde{x}(\phi), \tilde{y}(\phi), \tilde{z}(\phi)). \tag{1}$$

The path parameter ϕ is monotonically increasing and one can specify $n+1$ discrete break points $(\phi_0, \phi_1, ..., \phi_n)$. For a ϕ in the interval $(\phi_j \leq \phi < \phi_{j+1})$, we have:

$$\begin{aligned} &\tilde{x}_j(\phi) = \sum_{i=0}^{3} a_{ji}(\phi - \phi_j)^i, \quad \tilde{y}_j(\phi) = \sum_{i=0}^{3} b_{ji}(\phi - \phi_j)^i, \\ &\tilde{z}_j(\phi) = \sum_{i=0}^{3} c_{ji}(\phi - \phi_j)^i. \end{aligned} \tag{2}$$

where $a_{ji}, b_{ji}, c_{ji}, i = 0 : 3, j = 0 : n$ are the cubic spline parameters. We will refer to $\sigma(\phi)$ parameterized in this manner as the original curve. Consider that when the original curve is reparameterized by arc-length s, we have $P : J \to \mathbb{R}^3$ as the arc-length parameterized curve, where J is an open interval, and $||dP(s)/ds||_2 = 1$, i.e., the curve is of unit speed. Since we would like the end-effector to travel along the parameterized curve with constant speed of magnitude $v > 0$, our goal is to obtain another reparameterization with a new path parameter $\bar{s}$ such that the following condition is met: $||d\bar{P}(\bar{s})/d\bar{s}||_2 = v$, where $\bar{P} : I \to \mathbb{R}^3$ is the reparameterized arc-length-based curve. Since we are modeling a curve on a freeform surface with changing curvature, we assume that the interpolated curve to be regular. Thus, there exists a differentiable function $h : J \to I$ such that $P = \bar{P}(h)$ and $\bar{s} = h(s)$; since we would like the reparameterization to be orientation preserving, we have $d(h(s))/ds > 0$. Since the arc-length interpolated curve $P(s)$ is of unit-speed, we have the following relation:

$$\begin{aligned} 1 &= \left\| \frac{dP(s)}{ds} \right\|_2 = \left\| \frac{dh(s)}{ds} \frac{d\bar{P}(h(s))}{dh(s)} \right\|_2 \\ &= \left| \frac{dh(s)}{ds} \right| \left\| \frac{d\bar{P}(h(s))}{dh(s)} \right\|_2 = \frac{dh(s)}{ds} v. \end{aligned} \tag{3}$$

Since $v \neq 0$, the reparameterization function h is bijective and continuous. We would like the reparameterization to have zero initial condition, thus the constant speed parameterization is related to the arc-length by:

$$\bar{s} = h(s) = \frac{1}{v} s. \tag{4}$$

In the following, $\bar{s}$ will be referred to as the scaled arc-length. Note that the given control points are measured points on a curved surface with changing curvature, thus, it is unlikely that the original path variable ϕ is naturally an arc-length parameterization. Denote the line integrand defined for the polynomials at index j with respect to the original path parameter ϕ by:

$$F(\phi, j) = \sqrt{(\tilde{x}_j(\phi)')^2 + (\tilde{y}_j(\phi)')^2 + (\tilde{z}_j(\phi)')^2}. \tag{5}$$

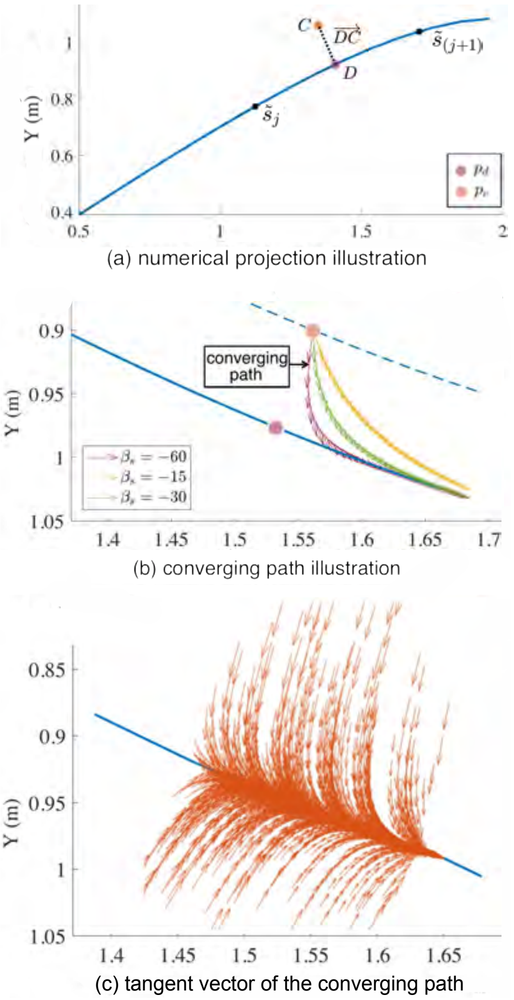
where the prime on $\tilde{x}_j(\phi)'$ denotes derivative of $\tilde{x}_j$ with respect to ϕ. The arc-length can be calculated from:

$$\begin{aligned} s(\phi) =& S(j-1) + \int_{\phi_j}^{\phi} F(\phi, j) d\phi, \\ S(j-1) =& \sum_{l=0}^{j-1} \int_{\phi_l}^{\phi_{l+1}} F(\phi, l) d\phi. \end{aligned} \tag{6}$$

where $S(j-1)$ is the arc-length for the path starting from ϕ_0 to ϕ_j. For the list of arc-length parameterized coordinates $(s_0, ..., s_m)$ that divide the original curve into m segments. Denote $\bar{s}_i = s_i/v, i = 0 : m$ and $(\bar{x}_i, \bar{y}_i, \bar{z}_i)$ as the Cartesian coordinates associated with $\bar{s}_i$, we reinterpolate the spline curve by interpolating $[(\bar{s}_0, \bar{x}_0), ..., (\bar{s}_m, \bar{x}_m)]$, $[(\bar{s}_0, \bar{y}_0), ..., (\bar{s}_m, \bar{y}_m)]$ and $[(\bar{s}_0, \bar{z}_0), ..., (\bar{s}_m, \bar{z}_m]$ with cubic spline, and then we have the desired path $\bar{P}(\bar{s}) = (\bar{x}(\bar{s}), \bar{y}(\bar{s}), \bar{z}(\bar{s}))$. For a $\bar{s}$ in the interval $(\bar{s}_j \leq \bar{s} < \bar{s}_{j+1})$, the desired path is given by:

$$\bar{x}_j(\bar{s}) = \sum_{i=0}^{3} \bar{a}_{ji}\bar{s}_e^i, \; \bar{y}_j(\bar{s}) = \sum_{i=0}^{3} \bar{b}_{ji}\bar{s}_e^i, \; \bar{z}_j(\bar{s}) = \sum_{i=0}^{3} \bar{c}_{ji}\bar{s}_e^i. \tag{7}$$

where $\bar{s}_e = \bar{s} - \bar{s}_j$, $\bar{a}_{ji}$, $\bar{b}_{ji}$ and $\bar{c}_{ji}$ are the cubic spline parameters for the re-interpolated cubic spline segment with the scaled arc-length parameter.



(a) numerical projection illustration

(b) converging path illustration

(c) tangent vector of the converging path

Fig. 2. Numerical projection and converging path.

3.2 Design of the converging path

The converging path is the path that the end-effector must follow from its current position to reach the desired path and follow it. When the end-effector is on the desired path, the converging path shares the same tangent with that of the desired path. When the end-effector is away, the converging path connects the current end-effector position to the desired path. The design of the converging path is particularly useful for robotic surface following/finishing operations since the end-effector may deviate from the desired path due to the uncertainty of the location of the work piece or due to other disturbances. Given the current end-effector position to be $p_c = (x_c, y_c, z_c)$, we need to find a corresponding point on the desired path, denoted by $p_d = (x_d, y_d, z_d)$, and the numerical value of its scaled arc-length $\bar{s}_c$; since there is no analytical expression for the scaled arc-length based curve, we will employ a numerical projection approach to achieve this.

Numerical Projection: The process of relating the given end-effector coordinate p_c to its scaled arc-length $\bar{s}_c$ is illustrated in Fig. 2. In plot (a), point C denotes the current end-effector position and one can use the nearest neighborhood algorithm (well-known for finding the shortest Euclidean distance between a given point and the second point from a list of points) to find the point D (coordinate p_d) on the desired path which is of the shortest distance from p_c. With this p_d, the process of finding its associated scaled arc-length coordinate $\bar{s}_c$ can be formulated as another NN problem.

Converging Path: With the numerical projection method, we can find the corresponding point p_d for a given end-effector position, where p_d is associated with one of the spline segments whose index is denoted as j:

$$x_d = \sum_{i=0}^{3} \bar{a}_{ji}\bar{s}_e^i,\ y_d = \sum_{i=0}^{3} \bar{b}_{ji}\bar{s}_e^i,\ z_d = \sum_{i=0}^{3} \bar{c}_{ji}\bar{s}_e^i. \quad (8)$$

We define the vector $\overrightarrow{DC}$ as:

$$\overrightarrow{DC} = (x_{dc}, y_{dc}, z_{dc}) = (x_c - x_d, y_c - y_d, z_c - z_d). \quad (9)$$

Based on this, the 3D converging path parameterized by $\bar{s}$ is $Q(\bar{s}) = (x(\bar{s}), y(\bar{s}), z(\bar{s}))$ (See Fig. 2 (b)). Denote $\bar{s}_{de} = \bar{s} - \bar{s}_c$, the converging path from C to D is:

$$x(\bar{s}) = x_d + x_{dc}\exp(\beta_s \bar{s}_{de}), y(\bar{s}) = y_d + y_{dc}\exp(\beta_s \bar{s}_{de}),$$
$$z(\bar{s}) = z_d + z_{dc}\exp(\beta_s \bar{s}_{de}). \quad (10)$$

where $\beta_s < 0$ is a design constant whose magnitude governs the convergence behavior of the end-effector as shown in Fig. 2 (b): faster convergence to the desired path is achieved for β_s with a larger magntitude. We are interested in finding the tangent of the converging path since the orientation of the end-effector translation velocity has to be aligned with that of the converging path tangent if the end-effector is to travel along the path. The components of the tangent vector $\boldsymbol{t} = (t_x, t_y, t_z) = (\frac{dx(\bar{s})}{d\bar{s}}, \frac{dy(\bar{s})}{d\bar{s}}, \frac{dz(\bar{s})}{d\bar{s}})$ are given by:

$$t_x = \sum_{i=1}^{3} i\bar{a}_{ji}\bar{s}_e^{i-1} + \beta_s x_{dc}\exp(\beta_s \bar{s}_{de}),$$
$$t_y = \sum_{i=1}^{3} i\bar{b}_{ji}\bar{s}_e^{i-1} + \beta_s y_{dc}\exp(\beta_s \bar{s}_{de}),$$
$$t_z = \sum_{i=1}^{3} i\bar{c}_{ji}\bar{s}_e^{i-1} + \beta_s z_{dc}\exp(\beta_s \bar{s}_{de}). \quad (11)$$

When $p_c \in \bar{P}(\bar{s})$, we have $p_c = p_d$ and $\overrightarrow{DC} = 0$, and the converging path given in equation (10) is the same as the desired path from equation (8), i.e., the tangent of the converging path $\boldsymbol{t}$ shares the same tangent with that of the desired. A 2D illustration of the converging path is given in Fig. 2 (b): the dashed line represents the translated spline via vector $\overrightarrow{DC}$ and the blue line represents the desired path. For drawing the illustration in Fig. 2 (c), we randomly generated a list of points around the desired path and calculated the corresponding converging path. The tangent vectors around the converging paths, as shown by the arrows, form an attractive field around the desired path, where the tip of each arrow can be treated as a given current end-effector position. When $p_c \notin \bar{P}(\bar{s})$, the converging path connects the current end-effector position to the desired path in a way that the end-effector would reach the desired path with tool orientation along the path; when $p_c \in \bar{P}(\bar{s})$, the converging path is the same as the desired path. To constrain the end-effector position onto the desired path, the velocity vector must travel in the direction of the tangent of the converging path. Therefore, the objective of the control action must be to align the velocity vector of the end-effector orientation with that of the tangent of the converging path.

4. MANIPULATOR END-EFFECTOR VELOCITY CONTROL STRATEGY

Since most of the industrial robots are controlled in the joint position or velocity mode, we propose a control law for the robot end-effector velocity to follow the converging path. The desired end-effector velocity is converted to joint velocity as system inputs. We have assumed that there is an inner torque loop that provides the necessary compensation for torque control that takes into consideration of the robot dynamics; the torque loop is generally not accessible to the end-user in commercial robots.

One can consider 2D motion on the surface; however, this may not suffice because the orientation of the end-effector frame may change in accordance with the direction of the tangent on the converging path, which is constructed based on the end-effector travel on the work piece. For most general freeform surfaces, it is not possible to cover the entire surface with a path generated on a fixed 2D plane. To resolve this issue, we consider the 3D kinematic model for end-effector motion as described by

$$\dot{x_b} = v\sin\psi\cos\theta, \quad \dot{y_b} = v\sin\psi\sin\theta, \quad \dot{z_b} = v\cos\psi,$$
$$\dot{\theta} = u_1, \quad \dot{\psi} = u_2. \quad (12)$$

where x_b, y_b, z_b are the components of the end-effector velocity expressed in the base frame and u_1 and u_2 are the control inputs. The variables associated with this 3D kinematic model for the end-effector are defined through the base and end-effector frames as shown in Fig. 3.

Now we consider the control of the end-effector velocity vector to satisfy the above kinematic model and the motion of the end-effector to follow the converging path. Without loss of generality, we can assume the direction of the velocity vector $\boldsymbol{v}$ to be aligned with the Y-axis of the end-effector frame. The problem now reduces to the control of the orientation of the end-effector Y-axis. Denote the component of the velocity vector $\boldsymbol{v}$ in the base frame as $\boldsymbol{v} = (v_{xb}, v_{yb}, v_{zb})$. The orientation θ and ψ can be calculated according to:

$$\theta = \text{atan2}(v_{yb}, v_{xb}),\ \psi = \text{atan2}(\sqrt{v_{xb}^2 + v_{yb}^2}, v_{zb}). \quad (13)$$

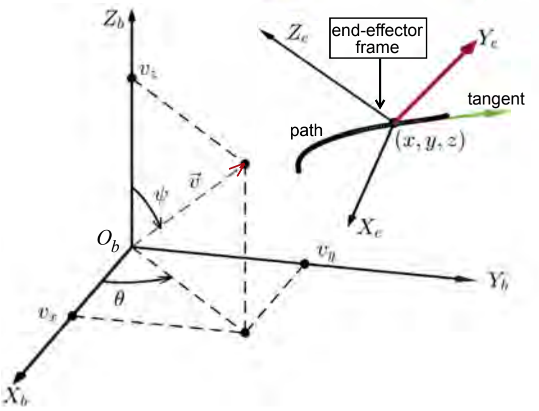


Fig. 3. Illustration of the 3D kinematics model.

For a given end-effector position at p_c, there corresponds a desired tangent vector from the construction of the converging path. The desired orientation of the end-effector can be calculated from the local tangent $\boldsymbol{t}$ on the converging path. As described in equation (11), the orientation of the path tangent can be derived from:

$$\theta_d = \text{atan2}(t_y, t_x), \ \psi_d = \text{atan2}(d_{xy}, t_z). \tag{14}$$

where $d_{xy} = \sqrt{t_x^2 + t_y^2}$. Thus, the control objective for the end-effector angular velocity is to stabilize the orientation error between the end-effector velocity and the orientation of the path tangent. As shown in Fig. 3, the objective of the control laws for u_1 and u_2 should be to align the velocity orientation, shown by the red arrow, with the that of the path tangent, shown by the green arrow. Define orientation errors as $e_1(t) = \theta - \theta_d(t)$ and $e_2(t) = \psi(t) - \psi_d(t)$. When the end-effector is traveling along the path with constant translational velocity, we have $\frac{d\bar{s}}{dt} = v$. The error governing equations after simplifications are are given by:

$$\dot{e}_1 = u_1 - \frac{v}{d_{xy}} \left(\frac{\partial t_y}{\partial s} t_x - \frac{\partial t_x}{\partial s} t_y \right) =: u_1 - e_1(s_e), \tag{15}$$

$$\begin{aligned} \dot{e}_2 &= u_2 - \frac{v}{d_{xy} d_{xyz}^2} \left((\frac{\partial t_x}{\partial s} t_x + \frac{\partial t_y}{\partial s} t_y) t_z - \frac{\partial t_z}{\partial s} d_{xy}^2 \right) \\ &=: u_2 - e_2(s_e). \end{aligned} \tag{16}$$

where $d_{xyz} = \sqrt{t_x^2 + t_y^2 + t_z^2}$. Therefore, one can choose the orientation control laws of the form $u_1 = e_1(s_e) - k_{p1} e_1$ and $u_2 = e_2(s_e) - k_{p2} e_2$ to follow the converging path. Since the converging path converges to the desired path and includes the desired path, the above control law guarantees the end-effector would converge and stay on the desired path.

5. EXPERIMENTS

We have conducted experiments with the proposed approach using a six DOF industrial robot as shown in Fig. 4. The controller implementation and communication is handled via the Robot Operating System. The robot controller is connected to a local PC via the interface provided by the ABB EGM library. In each time step, the local PC reads joint states, calculates the end-effector twist and send desired joint velocity commands, derived from the robot differential kinematics, as a EGM message to the robot controller.

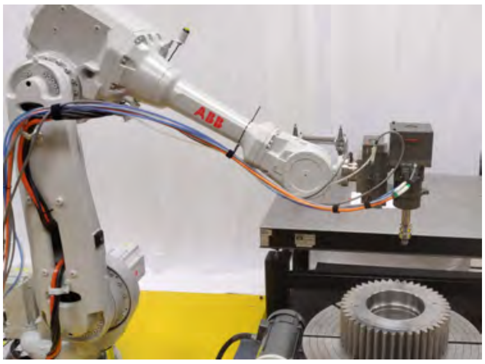


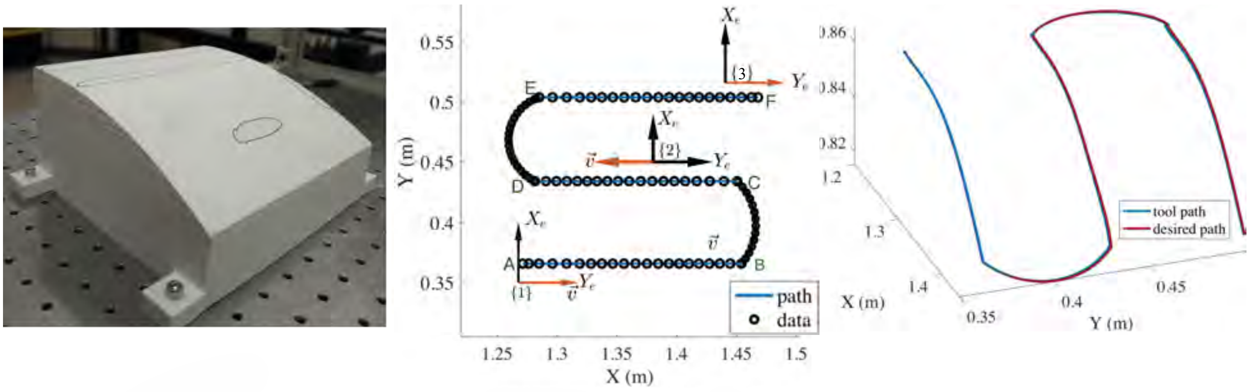
Fig. 4. Robot Setup with the finishing tool.

Results for 3D Path Following We present the 3D path following result for the sanding path on a workpiece as shown in Fig. 5 (a) with the translation velocity at 0.008 m/s. The surface geometry of the workpiece approximates that of the aircraft blade cross section. Fig. 5 (b) shows the top view of the "path", which is interpolated from the list of the control points as shown by "data" that can be generated from a proximity sensor or a 3D scanner using the method specified in Wen et al. (2019). The end-effector is supposed to travel along the desired path from the initial position A to the end position F. The direction of velocity vector can be any direction from the linear combination of end-effector X_e and Y_e axes. However, in practice, this direction has to comply with the path feasibility and manipulator joint limits. Starting from the position $\{1\}$ for the path AB, we assume the velocity is pointing in direction of the Y_e-axis: the velocity direction, shown at $\{1\}$ coincides with the end-effector Y_e. For the path segment from C to D, the velocity vector points towards the negative direction of Y_e-axis at $\{2\}$; this is because the end-effector cannot rotate 180° with respect to the base Z-axis otherwise the finishing tool would hit the manipulator. Similar logic applies to the path segment EF. Note that the velocity direction does not have to coincide with the end-effector axes and can be any vector expressed in the end-effector frame. Fig. 1 (c) demonstrates the the experimental path can be followed precisely with the proposed approach.

Result for Gear Involute Paths We consider the proposed method for a path that is employed for gear chamfering. As shown in Fig. 6, the task of chamfering requires the end-effector (surface finishing tool) to travel along the involute gear teeth. We discretize the involute tooth profile as a list of indexed control points whose interpolation with cubic spline is the desired path as shown by the red curve on the right. We present the result for one gear tooth since the entire gear teeth profile can be derived by continuously rotating one of the gear teeth profile with respect to the center of the gear. For the setup, we set the translation velocity to be at 2 mm/s. The end-effector follows the spline generated path very closely with the proposed strategy as shown in Fig. 6.

6. CONCLUSIONS

We have proposed a novel approach for 3D path following of a robot manipulator. The scheme can be employed when there is only data from scanners/sensors available



(a) workpiece profile

(b) velocity direction illustration

(c) path following results

Fig. 5. Path following result on a workpiece.

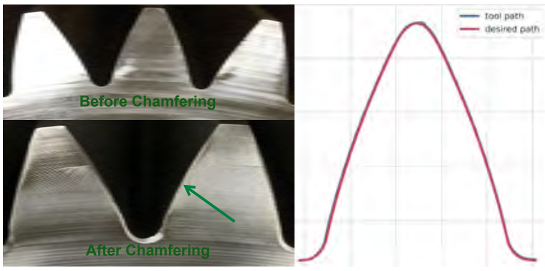


Fig. 6. Involute path following result.

to characterize the surface as discrete points, on which we construct a desired path employing cubic spline interpolation with the path parameter as the scaled arclength. We introduce the design of a converging path based on the desired path for the end-effector to follow with constant velocity. We have modeled the motion of the end-effector with a kinematic model under constant translation velocity for the interpolated 3D path, and demonstrated the proposed approach by conducting experiments on a robot following various tool paths.

REFERENCES

Aguiar, A.P., Hespanha, J.P., and Kokotovic, P.V. (2005). Path-following for nonminimum phase systems removes performance limitations. *IEEE Transactions on Automatic Control*, 50(2), 235.

Aguiar, A.P., Hespanha, J.P., and Kokotović, P.V. (2008). Performance limitations in reference tracking and path following for nonlinear systems. *Automatica*, 44(3), 598–610.

Cohen, E., Riesenfeld, R.F., and Elber, G. (2001). *Geometric modeling with splines: an introduction*. AK Peters/CRC Press.

Encarnaçao, P. and Pascoal, A. (2001). Combined trajectory tracking and path following: an application to the coordinated control of autonomous marine craft. In *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, volume 1, 964–969.

Ezair, B., Tassa, T., and Shiller, Z. (2014). Planning high order trajectories with general initial and final conditions and asymmetric bounds. *The Intl. Journal of Robotics Research*, 33(6), 898–916.

Kanjanawanishkul, K., Hofmeister, M., and Zell, A. (2010). Path following with an optimal forward velocity for a mobile robot. *IFAC Proceedings Volumes*, 43(16), 19–24.

King, J.E., Klingensmith, M., Dellin, C.M., Dogar, M.R., Velagapudi, P., Pollard, N.S., and Srinivasa, S.S. (2013). Pregrasp manipulation as trajectory optimization. In *Robotics: Science and Systems*.

Lapierre, L. and Jouvencel, B. (2008). Robust nonlinear path-following control of an auv. *IEEE Journal of Oceanic Engineering*, 33(2), 89–102.

Lapierre, L., Soetanto, D., and Pascoal, A. (2006). Non-singular path following control of a unicycle in the presence of parametric modelling uncertainties. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 16(10), 485–503.

Micaelli, A. and Samson, C. (1993). Trajectory tracking for unicycle-type and two-steering-wheels mobile robots. *Research report, IRR-2097, INRIA, inria-00074575.*

Nielsen, C., Fulford, C., and Maggiore, M. (2010). Path following using transverse feedback linearization: Application to a maglev positioning system. *Automatica*, 46(3), 585–590.

Pagilla, P.R. and Yu, B. (2001a). Robotic surface finishing processes: modeling, control, and experiments. *Transactions-American Society of Mechanical Engineers Journal of Dynamic Systems Measurement and Control*, 123(1), 93–102.

Pagilla, P.R. and Yu, B. (2001b). A stable transition controller for constrained robots. *IEEE/ASME Transactions on Mechatronics*, 6(1), 65–74.

Siu, A. (2011). *Real time trajectory generation and interpolation*. Ph.D. thesis, University of British Columbia.

Skjetne, R., Fossen, T.I., and Kokotović, P.V. (2004). Robust output maneuvering for a class of nonlinear systems. *Automatica*, 40(3), 373–383.

Wen, Y., Hu, J., and Pagilla, P.R. (2019). A novel robotic system for finishing of freeform surfaces. In *2019 International Conference on Robotics and Automation (ICRA)*, 5571–5577. IEEE.

Wen, Y. and Pagilla, P.R. (2020). A novel path following scheme for robot end-effectors. In *2020 American Control Conference.*

Xiang, X., Lapierre, L., Liu, C., and Jouvencel, B. (2011). Path tracking: Combined path following and trajectory tracking for autonomous underwater vehicles. In *2011 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, 3558–3563.