Available online at www.sciencedirect.com
ScienceDirect
Mechanism and Machine Theory 43 (2008) 1225–1235
Mechanism and Machine Theory
www.elsevier.com/locate/mechmt

A methodology for implementing the curvature theory approach to path tracking with planar robots

Satyajit Ambike *, James P. Schmiedeler

Department of Mechanical Engineering, The Ohio State University, 201 W. 19th Avenue, Columbus, OH 43210, USA

Received 28 November 2006; received in revised form 29 October 2007; accepted 30 October 2007
Available online 20 December 2007

Abstract

Planar motion can be synthesized with two-degree-of-freedom curvature theory by using the second-order Taylor series to coordinate planar path-tracking systems. Second-order control-variable coordination generates tracking error away from the reference point, and tracking becomes erratic when the driving variable experiences a dwell in its trajectory. This paper extends the technique through an algorithm that addresses both tracking errors and control variable dwells. The algorithm corrects the coefficients in the coordinating Taylor series when an error parameter is exceeded and switches the control variable when the current driving variable approaches a dwell. The algorithm is applied to a revolute–revolute mechanism to track a polynomial path. The approach utilizes local path information and executes tracking using an incremental approach. This is an advantage over the traditional method which requires knowledge of the entire path to compute exact joint variable motions before the motion of the system begins.

Keywords: Path planning; Curvature theory; Planar path-tracking systems

1. Introduction

The trajectory tracking problem for robotic manipulators is well-known. Most traditional approaches to solving this problem use dynamics to obtain time-based joint trajectories corresponding to the desired output trajectory [1,2]. However, time-independent joint kinematics can be obtained from the local geometry of the tracked path alone. Curvature theory is used to obtain time-independent joint motion by analyzing the differential geometric properties of the desired path in the output space. Bottema and Roth [3] developed planar two-degree-of-freedom (DOF) curvature theory, while Lorenc et al. [4] generalized it to arbitrary degrees of freedom and applied it to the motion synthesis problem for two-DOF systems. This approach provides the trajectories of the control variables (joint angles) that allow the system to track a locally predefined path. The method maps the geometry of the desired path in the output space to the geometry of a corresponding

* Corresponding author. Tel.: +1 614 247 8411; fax: +1 614 292 3163.
E-mail address: ambike.1@osu.edu (S. Ambike).

0094-114X/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mechmachtheory.2007.10.010

path in the control space (joint space). The time-invariant nature of the mapping is the advantage of the curvature theory approach to path tracking.

For single-DOF planar systems, Roth [5] shows how to determine all time-dependent motions that generate paths with identical geometric properties. Similarly for two-DOF systems, a particular output path can be traversed at any speed. Lorenc et al. [4] track desired paths by selecting one control variable as the driving variable and coordinating the motion of the two variables using a second-order Taylor series. By changing the velocity and the acceleration of the driving variable, the same path can be traversed at a different speed by maintaining the relation between the two variables.

Local path geometry at a reference point is used to specify the instantaneous coordination of the motion variables. As a result, error is produced in the tracking away from the reference point. Also, when the driving variable experiences a dwell in its trajectory, the system cannot be coordinated using that control variable. The contribution of this paper is an algorithm that addresses both tracking errors and dwells in the control variable. The central ideas of the algorithm are to correct the coefficients in the coordinating Taylor series, called the speed ratios, when a predefined error parameter is exceeded and to switch to the second control variable to coordinate the system when the current driving variable experiences a dwell in its trajectory. The algorithm assumes knowledge of the path to be tracked in the neighborhood of the current end-effector position and tracking system geometry and provides trajectories of the control variables that allow the system to execute tracking.

The methods of this paper are applicable to any planar tracking system. With different physical tracking systems, such as a revolute–revolute (RR) or prismatic–revolute mechanism, the instantaneous invariants of motion for the mechanism have to be obtained. The expressions for the speed ratios are then obtained as functions of these invariants. As an example, this paper includes a method developed by Lorenc et al. [4] to find the instantaneous invariants for the RR mechanism, whereas Stanisic et al. [6] provides this information for a planar prismatic–revolute positioner, and Lorenc et al. [4] provides the invariants for a non-holonomic cart system.

The problem of tracking a predefined path is traditionally handled by formulating two 'exact' motion laws for the joint variables. To compare the traditional approach with the new one presented here, the following points are of interest. First, the traditional approach requires the knowledge of the entire path for computing the joint trajectories before the motion commences. Secondly, the success of the traditional approach requires accurate calibration of the tracking system. The new approach uses only local path information to execute tracking. Throughout this paper it is assumed that the methods described are executed using a vision feedback system. This makes the new approach more adaptable to external disturbances. It reduces the reliance of tracking performance on accurate calibration of the tracking system as the process of geometry mapping and path correction occurs in the output space. A visual image of the desired path in the vicinity of the end-effector can be analyzed to obtain the current tracking error and also the local geometric properties of the desired path. This paper provides an alternate method to path tracking. The choice of method will be decided by the requirements of a particular application.

The remainder of the paper is organized as follows. Section 2 provides a review of two-DOF curvature theory. Section 3 describes the path-tracking algorithm in detail. Section 4 presents an example wherein the algorithm is applied to track a predefined path using an RR mechanism. Section 5 contains concluding remarks.

2. Curvature theory of planar two-degree-of-freedom motion

A brief review of second-order control of planar two-DOF motion developed by Lorenc et al. [4] is presented here. Consider a reference frame M moving with respect to a fixed reference frame F, as shown in Fig. 1. A point C has coordinates (x, y) in frame M and coordinates (X, Y) in frame F related by

$$\begin{Bmatrix} X \\ Y \end{Bmatrix} = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \begin{Bmatrix} x \\ y \end{Bmatrix} + \begin{Bmatrix} a \\ b \end{Bmatrix}, \tag{1}$$

where (a, b) are the coordinates of the origin of frame M in frame F and ϕ is the rotation of frame M relative to frame F. For two-DOF motion, the motion variables a, b, and ϕ are assumed to be functions of two

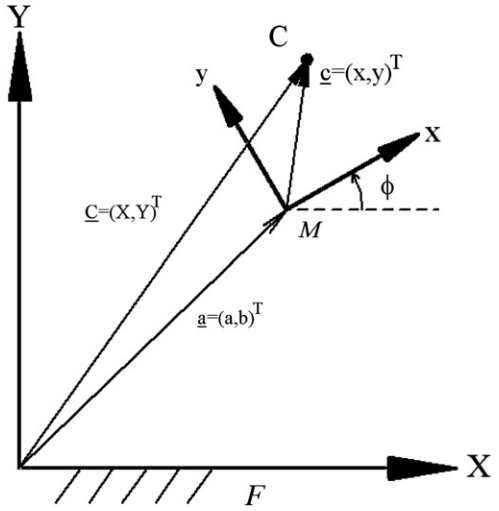

Fig. 1. General planar motion of a point C in moving frame M.

independent control variables or motion parameters, $\lambda(t)$ and $\mu(t)$, such that $a = a(\lambda, \mu), b = b(\lambda, \mu)$, and $\phi = \phi(\lambda, \mu)$. The zero position is defined as the instant when $t = 0, \lambda = 0, \mu = 0$, and the frames M and F are coincident, so $a = 0$, $b = 0$, and $\phi = 0$.

The partial derivative of any variable with respect to λ or μ, evaluated in the zero position, is denoted with trailing subscripts. For example, $\frac{\partial X}{\partial \lambda}\big|_{\lambda=0,\mu=0} \equiv X_\lambda$ and $\frac{\partial^2 X}{\partial \lambda \partial \mu}\Big|_{\lambda=0,\mu=0} \equiv X_{\lambda\mu}$. Time derivatives evaluated in the zero position are denoted using the dot notation. For example, $\frac{dX}{dt}\big|_{\lambda=0,\mu=0} \equiv \dot{X}$.

The motion of point C in the fixed frame can be expressed using the second-order Taylor series

$$\begin{Bmatrix} X \\ Y \end{Bmatrix} = \begin{Bmatrix} x + X_\lambda \lambda + X_\mu \mu + \frac{1}{2}(X_{\lambda\lambda}\lambda^2 + 2X_{\lambda\mu}\lambda\mu + X_{\mu\mu}\mu^2) \\ y + Y_\lambda \lambda + Y_\mu \mu + \frac{1}{2}(Y_{\lambda\lambda}\lambda^2 + 2Y_{\lambda\mu}\lambda\mu + Y_{\mu\mu}\mu^2) \end{Bmatrix}, \tag{2}$$

where

$$\begin{aligned} X_\lambda &= a_\lambda - y\phi_\lambda, \quad Y_\lambda = b_\lambda + x\phi_\lambda, \\ X_\mu &= a_\mu - y\phi_\mu, \quad Y_\mu = b_\mu + x\phi_\mu, \\ X_{\lambda\lambda} &= a_{\lambda\lambda} - x\phi_\lambda^2 - y\phi_{\lambda\lambda}, \quad Y_{\lambda\lambda} = b_{\lambda\lambda} + x\phi_{\lambda\lambda} - y\phi_\lambda^2, \\ X_{\mu\mu} &= a_{\mu\mu} - x\phi_\mu^2 - y\phi_{\mu\mu}, \quad Y_{\mu\mu} = b_{\mu\mu} + x\phi_{\mu\mu} - y\phi_\mu^2, \\ X_{\lambda\mu} &= a_{\lambda\mu} - x\phi_\lambda\phi_\mu - y\phi_{\lambda\mu}, \quad Y_{\lambda\mu} = b_{\lambda\mu} + x\phi_{\lambda\mu} - y\phi_\lambda\phi_\mu. \end{aligned} \tag{3}$$

The values of the constants, i.e. the first- and second-order partial derivatives of a, b, and ϕ with respect to λ and μ, depend on the geometry of the particular tracking system (the link lengths of an RR mechanism for example) and the position and orientation of the coincident reference frames. The canonical system, in which the maximum number of partial derivatives in Eq. (2) are zero, resulting in the most parsimonious description of the motion, is obtained by placing the Yy axes of the coincident fixed and moving frames along the polar line, the locus of all possible instantaneous poles. This ensures that the partials b_λ and b_μ are zero. Furthermore, the origin of the frames is located on the polar line such that at least one of the three second-order partial derivatives, $b_{\lambda\lambda}$, $b_{\lambda\mu}$ and $b_{\mu\mu}$, is zero. In the canonical system, the non-zero partial derivatives are called the instantaneous invariants of motion.

Second-order coordination requires a mapping of the instantaneous first- and second-order properties of motion. The first-order mapping is obtained by differentiating equation (2) with respect to time and evaluating the result in the zero position to determine the coordinates of the pole of the system. This yields

$$x_p = -\frac{b_\lambda + nb_\mu}{\phi_\lambda + n\phi_\mu} = 0, \tag{4}$$

$$y_p = \frac{a_\lambda + na_\mu}{\phi_\lambda + n\phi_\mu}, \tag{5}$$

where n is the first-order instantaneous speed ratio of the motion parameters defined as $n = \frac{d\mu}{d\lambda}\Big|_{\lambda=0,\mu=0}$, and (x_p, y_p) are the coordinates of the pole in the canonical system. From Eq. (5),

$$n = -\frac{a_\lambda - \phi_\lambda y_p}{a_\mu - \phi_\mu y_p}. \tag{6}$$

Therefore, the y coordinate of the pole maps the output-space path tangent into the control-space path tangent.

The mapping of the second-order properties of motion relates the radius of curvature of the output-space path to the second-order speed ratio n', which is the instantaneous rate of change of n: $n' = \frac{dn}{d\lambda}\Big|_{\lambda=0,\mu=0} = \frac{d^2\mu}{d\lambda^2}\Big|_{\lambda=0,\mu=0}$. The relationship between n' and the curvature of the output-space path is obtained using the velocity of the pole and the inflection circle. The pole velocity is found by differentiating equation (2) and retaining the first-order terms. n' appears in the expression for the velocity of the pole in the Yy direction as

$$\dot{y}_p = \frac{\dot{\lambda}[n^3(\phi_\mu a_{\mu\mu} - \phi_{\mu\mu} a_\mu) - n^2\Gamma - n\Lambda + \phi_\lambda a_{\lambda\lambda} - \phi_{\lambda\lambda} a_\lambda + n'(\phi_\lambda a_\mu - \phi_\mu a_\lambda)]}{(\phi_\lambda a_\mu - \phi_\mu a_\lambda)^2}, \tag{7}$$

where

$$\Gamma = -2\phi_\mu a_{\lambda\mu} + 3a_\mu \phi_{\lambda\mu} + \phi_{\mu\mu} a_\lambda - \phi_\lambda a_{\mu\mu},$$
$$\Lambda = -\phi_\mu a_{\lambda\lambda} + a_\mu \phi_{\lambda\lambda} + 2\phi_{\lambda\mu} a_\lambda - 2\phi_\lambda.$$

From curvature theory [7], it is known that $\dot{y}_p = (PJ)_x \dot{\phi}$, where $\dot{\phi}$ is the angular velocity of the moving frame, point P is the pole, PJ is the directed line segment P to J along the diameter of the inflection circle, and $(PJ)_x$ is the x component of this line. In general, $\dot{\phi} = \phi_\lambda \dot{\lambda} + \phi_\mu \dot{\mu}$, so Eq. (7) can be rewritten to give

$$n' = \frac{(PJ)_x(\phi_\lambda + \phi_\mu n)^3 + n^2\Gamma + n^3(-\phi_\mu a_{\mu\mu} + \phi_{\mu\mu} a_\mu) + n\Lambda - \phi_\lambda a_{\lambda\lambda} + \phi_{\lambda\lambda} a_\lambda}{\phi_\lambda a_\mu - \phi_\mu a_\lambda}. \tag{8}$$

$(PJ)_x$ thus maps the output-space path curvature into the second-order speed ratio n' in the control-space of a planar path tracking system. n' is representative of the control-space path curvature. Eqs. (6) and (8) are a purely geometric mapping of the first- and second-order geometric properties of the output-space path into the control-space trajectory. Once the speed ratios are known, the control variables can be coordinated using the second-order Taylor series

$$\mu = n\lambda + \frac{n'}{2}\lambda^2. \tag{9}$$

In Eq. (9), λ is the driving variable, and motion of μ is obtained from that of λ. The inverse relation can also be readily obtained.

$$\lambda = m\mu + \frac{m'}{2}\mu^2, \tag{10}$$

where m and m' are speed ratios defined as, $m = \frac{d\lambda}{d\mu}\Big|_{\lambda=0,\mu=0}$ and $m' = \frac{d^2\lambda}{d\mu^2}\Big|_{\lambda=0,\mu=0}$. m and m' can be obtained from n and n'.

$$m = \frac{d\lambda}{d\mu} = \frac{1}{n}, \tag{11}$$

$$m' = \frac{d^2\lambda}{d\mu^2} = -\frac{n'}{n^3}. \tag{12}$$

Eq. (9) uses λ as the driving variable, and Eq. (10) uses μ as the driving variable. Switching between Eqs. (9) and (10) for system coordination amounts to switching driving variables.

3. Methods

It is possible to track a given path by incrementing the driving variable by a sufficiently small value and the other control variable by a likewise small value determined using Eq. (9) or Eq. (10), and calculating the speed ratios after each interval based on the path properties corresponding to the new position. Although this process results in high-fidelity tracking, it imposes high computational load on the tracking system. Further, a finite increment in the control variables, however small, creates a deviation from the desired path. This leads to ambiguity in the selection of path properties to be used for recalculating the speed ratios. The algorithm developed in this section allows the adjustment of the accuracy level and the computational load on the system by controlling the increment magnitude of the driving variable and permissible error limits. The algorithm also demonstrates how new path properties can be selected for recalculating the speed ratios once the system deviates from the desired path. A vision system can be used to obtain all information required to execute a tracking task with the specified accuracy. A visual image can be processed to obtain the tracking error and the local geometric properties of the desired path.

Inputs to the algorithm are the initial pose of the system specified by initial values for the variables (λ_0 and μ_0) with respect to a global reference frame, and the geometry of the system specified by various link lengths. Note that the control variables λ and μ are different from λ_0 and μ_0. At the instant of calculating the speed ratios using Eqs. (6) and (8), λ and μ assume zero values defining the zero position. The path to be tracked is called the desired path, and its properties in the vicinity of the point of interest, point C, are assumed known. The actual path followed by point C is called the generated path.

Assuming that point C lies on the desired path at the beginning of the motion, the tangent and center of curvature of the desired path at C are determined. The pole and the inflection circle are located, and the instantaneous invariants are determined. The instantaneous speed ratios n and n' or m and m' are determined from Eqs. (6) and (8) or Eqs. (11) and (12).

The driving variable is selected based on the magnitude of the speed ratios. If n and n' are large, the increment in μ calculated from Eq. (9) will be large regardless of the increment in λ. n approaches infinity as y_p approaches $\frac{a_\mu}{\phi_\mu}$, but since n' is proportional to n^3, n' increases much faster. In other words, as y_p approaches $\frac{a_\mu}{\phi_\mu}$, λ approaches a dwell in its motion, and path tracking requires motion of μ alone. For these reasons, μ is made the driving variable when the second-order speed ratio n' becomes large. In this case, the speed ratios m and m' are calculated from Eqs. (11) and (12), and coordination is achieved with Eq. (10). Note that if the ratios n and n' are large, m and m' will necessarily be small. The reverse logic applies when m' is large. In implementation, an upper limit is specified for the second-order ratios n' and m' such that when the ratio corresponding to the current driving variable exceeds the limit, the driving variable is switched.

3.1. Driving-variable increment sign and magnitude

The sign of the driving variable increment is determined such that point C advances in the desired direction as a result of the increment. The increment magnitude is determined based on the curvature properties of the desired path. The driving variable increment magnitude defines a *step* along the continuously generated path of point C, such as points C_1 and C_2 in Fig. 2. A step corresponds to an instant when the tracking system senses the error in the tracking. It is desirable to maximize the increment magnitude so that the system traverses the desired path in fewer steps, reducing the sensory and computational demands on the tracking system. However, if the desired path has high curvature or rate of change of curvature near the current position, a large increment will create large position error. The general logic followed is to have large increment magnitudes where the tracked path has small curvature properties and to reduce the magnitude while approaching portions of the path with high curvature and/or rate of change of curvature. The example presented in Section 4 uses a simple binary approach wherein the increment magnitude switches between a high and a low value. The lower value is selected if the curvature properties of the desired path are greater than predefined limits. Otherwise, the higher value is selected.

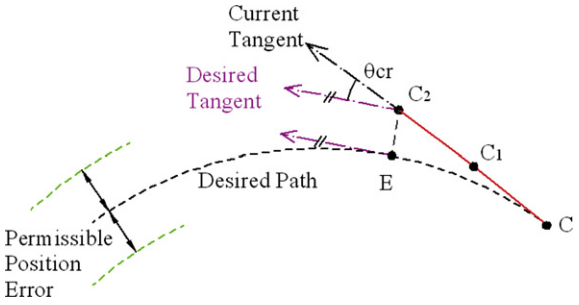

Fig. 2. Defining errors; C_2E is the position error and θ_{cr} is the tangential error.

After selecting the driving variable and its increment magnitude, the increment in the other variable is calculated from the appropriate coordination equation, i.e. Eq. (9) or Eq. (10). The motion variables are incremented through the computed values causing point C to move from its current position to a neighboring position C_1. To move from C_1 to C_2, the coordinating equation is reused, this time with twice the driving-variable increment magnitude, to obtain the increment in the other variable. The motion variables are incremented to the new computed values, making point C assume position C_2.

3.2. Error parameters

The error sensed by the system at every step is characterized by defining *position error* and *tangential error* as shown in Fig. 2. The position error at C_2 is defined as the distance C_2E, where E is the point on the desired path for which the distance C_2E is minimal. The tangent to the desired path at E represents the desired direction of motion. The angle between the current and desired tangents, θ_{cr}, defines the tangential error. Limiting values are specified for the two error parameters. When either error limit is exceeded, the speed ratios are recalculated. The recalculation of the speed ratios is called *correction*, so the position at which this occurs is termed a *correction point*. The speed ratios are based on local curvature characteristics, and there may be several correction points while tracking a desired path.

Limits are specified for the two errors based on the following considerations. Specifying a low value for the permissible position error will make the tracking more accurate. On the other hand, it will increase the number of correction points, consequently increasing the computational load on the system. Path tracking can be carried out without considering the tangential error at all. However, including a tangential error limit will smooth the generated path considerably. This is because the tangential error is predictive, such that corrections can occur when the position error is still within the specified limit, but tending to get worse. Small values for the permissible tangential error will be quite restrictive and trigger corrections more often, increasing computational requirements.

The accuracy of tracking does not directly correspond to either position or tangential error. The tangential error limit typically affects the smoothness of the generated path. Although the position error limit has greater control over the position error, if the tangential error is within the prescribed limit, correction calculations are made *after* the position error limit is exceeded. There will always be a position error overshoot, the magnitude of which depends on the speed ratios and the joint angle increment magnitude. The overshoot can be reduced, as discussed later, but cannot be removed using this methodology. However, a variation in which the position error at the end of the next step is calculated at the current position would eliminate the overshoot by enabling correction calculations *before* the position error is exceeded.

3.3. Corrected tangent and curvature

At a correction point, the new speed ratios are calculated based on a corrected tangent and a corrected curvature, chosen such that further motion will reduce position error. In Fig. 3, the current tangent direction tends to drive point C away from the desired path. A corrected tangent, at an angle θ_{new} to the desired tangent,

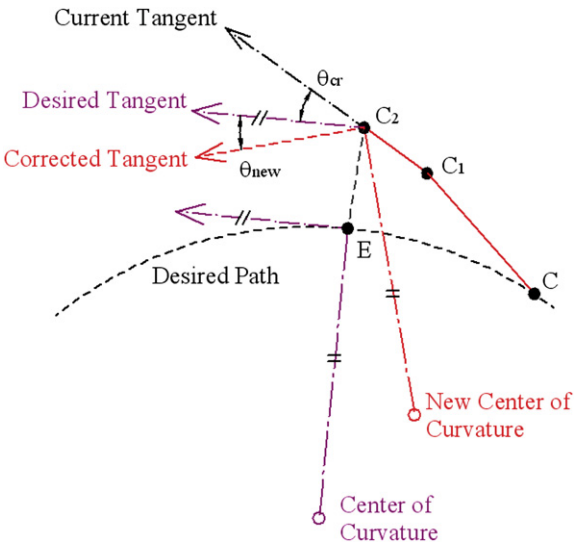

Fig. 3. Defining curvature properties for the generated path; if $\theta_{cr} >$ permissible tangential error or if $C_2E >$ permissible position error, $\theta_{new} = f(\theta_{cr})$.

is chosen such that further motion will drive point C toward the desired path. A large θ_{new} would cause a large change in the direction of motion, resulting in large tangential error on the other side of the desired path, and ultimately oscillations, as point C progresses. If θ_{new} is too small, the correction will not be effective since C will travel a long distance before the correction has any effect on the position error magnitude. A simple approach implemented in the example in Section 4 is to make θ_{new} proportional to θ_{cr}. If the current tangent direction indicates that further motion of C is toward the desired path, the tangent direction is not corrected even if the position error is exceeded.

The corrected curvature is chosen based on the curvature of the desired path at point E. Work with the algorithm has shown that choosing the corrected curvature to be the same as the curvature of the desired path at point E yields reasonably smooth generated paths.

The corrected tangent and curvature define a center of curvature for the path that C will follow. Fig. 3 shows the curvature properties of the desired path and the generated path at the corresponding correction point. Once the corrected tangent and curvature are determined, the pole and the inflection circle can be located, and the speed ratios can be calculated.

This process of sensing error and correcting the motion of point C by recalculating the speed ratios continues until either the workspace boundary or the end of the desired path is reached.

3.4. Algorithm summary and discussion

The algorithm described in this section is summarized below.

(1) Assuming that point C lies on the desired path, the tangent and center of curvature of the desired curve at C are determined. The pole and the inflection circle are located. If point C does not lie on the desired path, curvature properties of the path to be generated are determined based on the properties of the desired path at the corresponding error point.
(2) The speed ratios associated with the current driving variable are calculated. If the second-order speed ratio is greater than the limiting value, the driving variable is switched and the ratios corresponding to the other control variable are calculated.
(3) The increment magnitude and sign for the driving variable are determined based on the curvature properties of the desired path and the direction of the desired motion. The increment in the other control variable is obtained using the appropriate coordination equation. The control variables are incremented

through the computed values, causing point C to move from the current position to a neighboring position.

(4) The position and tangential errors are sensed by the system. If either error limit is exceeded, the system repeats steps 1–4. If not, the tracking system repeats steps 3 and 4. This process continues until either the workspace boundary or the end of the desired path is reached.

The following observations regarding the tracking algorithm are relevant. First, the effectiveness of the correction strategy is limited by the frequency of sensing of the tracking system and the maximum curvature of the desired path. If the local curvature at a section of the desired path is too great, the error has to be sensed more frequently. If the required sensing frequency cannot be handled by the system, the system will produce large error overshoot. Second, to arrive at acceptable values for the joint angle increment magnitude and the speed ratio limit, trial runs must be made wherein the parameters are adjusted until a satisfactory result is obtained. This process can be compared to the tuning of the gains in a conventional PD controller. Lastly, the stability analysis of the correction scheme outlined here is not included in this introductory paper. It is a subject of future work.

4. Revolute–revolute mechanism example

This section applies the results obtained for general two-DOF motion to the RR planar mechanism shown in Fig. 4. The control variables λ and μ are the joint angles of the two revolute joints of the mechanism located at points A^* and A, respectively. Angles λ_0 and μ_0 define the initial pose of the mechanism in a global frame, as seen in Fig. 4. The polar line is the line joining the centers of the two revolutes, A and A^*. The origin of the canonical system is located at point A, with the Yy axes along the polar line. The instantaneous invariants for the mechanism are $a_\lambda = -l_1, \phi_\lambda = 1, \phi_\mu = 1$, and $b_{\lambda\lambda} = -l_1$. Substituting these values into Eqs. (6) and (8) gives the speed ratios for the RR mechanism

$$n = -\frac{l_1 + y_p}{y_p}, \tag{13}$$

$$n' = \frac{(1+n)^3 (PJ)_x}{l_1}. \tag{14}$$

For the RR mechanism, Fig. 5 shows the geometric quantities y_p and $(PJ)_x$ that can be obtained using graphical methods, perhaps with a processed image of the output space [4], given the path tangent and radius of curvature at point C and the geometry of the mechanism. Point P is the pole for the desired path tangent, and the inflection circle is obtained with the Euler–Savary equation [7]. The lengths y_p and $(PJ)_x$ are used with Eqs. (13) and (14) to obtain the speed ratios.

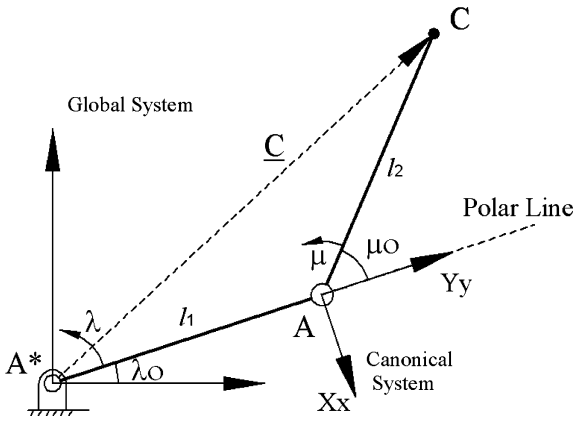

Fig. 4. Canonical system for the RR mechanism.

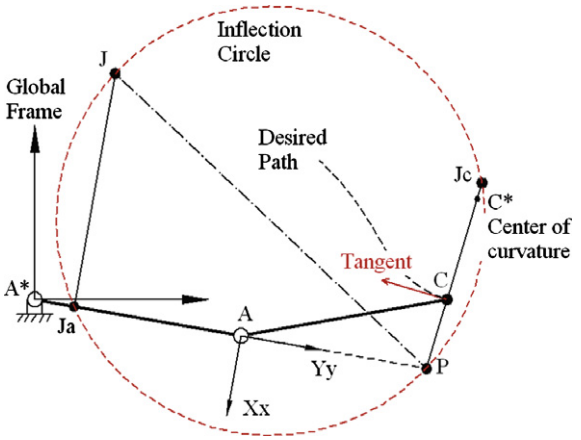

Fig. 5. Obtaining y_p and $(PJ)_x$ for a given path and linkage geometry. $AP = y_p$ and $JJ_a = (PJ)_x$.

The linkage geometry is defined by the link lengths $l_1 = l_2 = 500$ mm. The initial pose of the mechanism is described by the angles $\lambda_0 = -10^\circ$ and $\mu_0 = 20^\circ$ in the global frame. The curve to be tracked, parameterized as a function of x in the global frame is specified as $y = -10^{-6}x^3 + 10^{-5}x^2 - 10^{-3}x + 946.4$. The mechanism follows the path until point C reaches the work-space boundary, defining a 1.7 m long desired path. Note that the entire path is defined here only to enable the demonstration of the technique using computer software. For real systems employing machine vision, only local path information is necessary for executing path tracking. The parameters for tracking the specified path are listed in Table 1. The permissible position error is large so that the desired path and the generated path can be distinguished in Fig. 6. The limiting values for the curvature and the rate of change of curvature are selected to be below the maximum values of the parameters along the desired path. The driving variable increment magnitude switches between 1.5° and 0.5°. While tracking the path, curvature properties are calculated at each step, and if the absolute value of either parameter exceeds its limit, the drive angle increment is reduced to 0.5°.

Fig. 6 shows the tracking result for the selected parameters. The joint trajectories and position error are plotted against the number of increments in the drive angle in Fig. 7. Point C reaches the workspace boundary in 115 increments of the driving variable. Of these, the speed ratios are calculated at eight steps: two with μ as the driving variable and six with λ as the driving variable.

λ is near a dwell in its trajectory at the beginning of the motion, as seen in Fig. 7a, so μ is the driving variable for this portion of the path. Fig. 7a also shows a dwell in the trajectory of μ between increments 50 and 70. λ is the driving variable while tracking this part of the trajectory.

Fig. 7b shows that the maximum position error is 12.3 mm, a 20% overshoot at one point. The overshoot could be reduced by lowering the limit for the curvature and the rate of change of curvature. Doing so would reduce the driving variable increment magnitude in the region where the maximum error occurs. The increased frequency of sensing would allow the detection of excess position error earlier, thus restricting the overshoot. For example, with a limit magnitude of 0.0009 for both parameters, compared with the original value of

Table 1
Algorithm parameters for tracking example path

Parameter	Value
Permissible position error (mm)	10
Permissible tangential error (°)	5
Second-order speed ratio limit	25
Drive angle increment (°)	1.5 or 0.5
Curvature limit (mm^{-1})	0.0015
Rate of change of curvature limit (mm^{-2})	0.0015
θ_{new}	$\theta_{cr}/2$

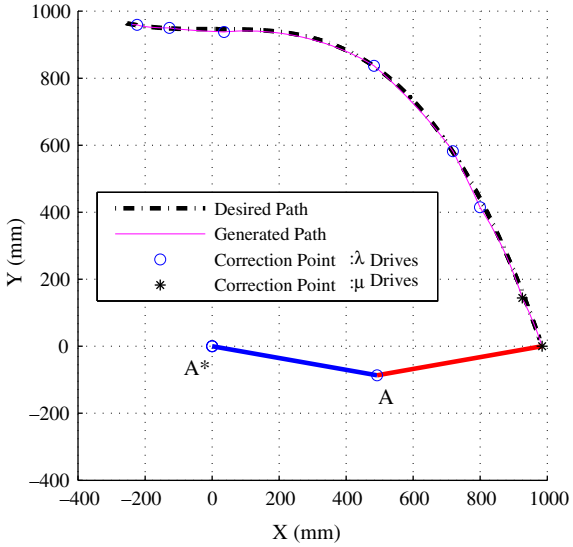

Fig. 6. Tracking a polynomial path.

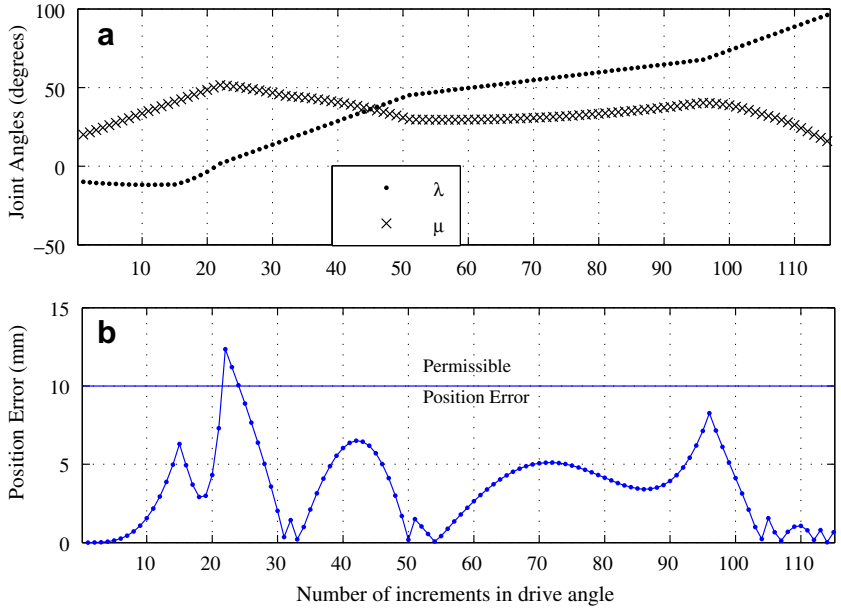

Fig. 7. Parameters for tracking a polynomial.

0.0015, the path is traversed in 188 steps, with the overshoot reduced to 4.8%. The larger number of steps required to traverse the same path indicates the increased computational load imposed on the system to obtain lower overshoot.

The accuracy of tracking could also be improved by reducing the position error limit. For example, if the permissible position error is reduced to 3 mm, the same path is tracked in 145 steps, with 12 corrections and 3.8 mm maximum position error. Again, the increased accuracy comes at the cost of increased computational load. Tracking accuracy can also be controlled by adjusting the driving angle increment magnitude, the tangential error limit and/or the second-order speed ratio. For this mechanism, good results are generally obtained with the driving angle increment less than 2° and the tangential error limit less that 10°.

In regions where the driving variable approaches a dwell, tracking becomes erratic, and switching the driving variable is a practical solution to this problem. The idea is implemented using the second-order speed ratio limit. The limit has a large range and is usually selected for different desired paths separately by running the algorithm several times and choosing the limit such that the error overshoot is minimal.

5. Conclusions

This paper builds upon work to coordinate a two-DOF planar system via curvature theory. The developed algorithm extends the capabilities of the existing curvature-theory approach to path tracking, making it possible to track paths of any length within the planar system's workspace. Since only the local path geometry is used to specify the instantaneous coordination of the motion variables, error is produced in the tracking away from the reference point. The algorithm controls the error by recalculating the speed ratios when predefined limits are exceeded. Further, when a driving variable approaches a dwell in its trajectory, tracking is achieved by switching the driving variable. An example is presented wherein an output path described by a polynomial is tracked using a planar RR mechanism.

The methodology described in this paper is generally applicable to any two-DOF planar system and could be implemented with machine vision. The geometry of the desired path is obtained from a processed video image, and the position and tangential error are sensed. The algorithm is flexible, and a desired level of accuracy or computational effort for a particular situation can be obtained by controlling the frequency of sensing and specifying permissible error limits. The time-invariant nature of the mapping between the instantaneous desired path geometry and the control space is the advantage of the curvature theory approach to path tracking. The speed at which a desired path is traversed depends on the choice of the velocity and acceleration of the driving variable. This implies that these quantities can be chosen as desired, and the coordinating Taylor series ensures that the system follows the same desired path.

Acknowledgment

This work is supported by NSF Grant IIS-0546456 to James Schmiedeler. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] J.M. Hollerbach, Dynamic scaling of manipulator trajectories, Journal of Dynamic Systems, Measurement, and Control; Transactions of the ASME 106 (1984) 102–106.
[2] J.E. Bobrow, S. Dubowsky, J.S. Gibson, Time optimal control of robotic manipulators along specified paths, International Journal of Robotic Research 4 (1985) 3–17.
[3] O. Bottema, B. Roth, Theoretical Kinematics, North Holland Publishing Company, Amsterdam, 1979.
[4] S.J. Lorenc, M.M. Stanisic, A.S. Hall Jr., Application of instantaneous invariants to the path tracking control problem of planar two degree-of-freedom systems: a singularity-free mapping of trajectory geometry, Mechanisms and Machine Theory 30 (6) (1995) 883–896.
[5] B. Roth, Time-invariant properties of planar motions, in: J. Lenarcic, C. Galletti (Eds.), On Advances in Robotic Kinematics, Kluwer Academic Publishers, Dordrecht, 2004, pp. 79–88.
[6] M.M. Stanisic, K. Lodi, G.R. Pennock, The application of curvature theory to the trajectory generation problem of robot manipulators, ASME Journal of Mechanical Design 114 (1992) 677–680.
[7] A.S. Hall, Kinematics and Linkage Design, Prentice Hall, New York, 1961.