Proceedings of the 32nd Conference
on Decision and Control
San Antonio, Texas • December 1993

Dynamic Path Following: A New Control Algorithm for Mobile Robots

Nilanjan Sarkar, Xiaoping Yun, and Vijay Kumar

General Robotics and Active Sensory Perception (GRASP) Laboratory
University of Pennsylvania
3401 Walnut Street, Room 301C
Philadelphia, PA 19104-6228

ABSTRACT

A new algorithm for the control of wheeled mobile robots called *dynamic path following* is presented. This algorithm regulates the motion of the mobile robots to desired *geometric paths* as opposed to desired trajectories which are given as time histories in conventional trajectory tracking algorithms. The desired trajectory for the *dynamic path following* algorithm is parameterized by a convenient geometrical parameter (e.g., arc length of the path). The algorithm is designed by using a nonlinear feedback for input-output linearization and decoupling. The performance of this new algorithm is compared with that of the trajectory tracking algorithm and is evaluated under different initial conditions as well as in the presence of various uncertainties using computer simulations.

1 Introduction

Wheeled mobile robots are typical examples of mechanical systems with nonholonomic constraints. Although navigation [9, 22] and planning [1, 10] of mobile robots have been investigated extensively over the past decade, the work on dynamic control of mobile robots with nonholonomic constraints is much more recent [6, 17, 7].

It is well known that systems with nonholonomic constraints cannot be stabilized to equilibrium points using a static-state feedback [3]. Bloch *et al.* showed that such systems are strongly accessible and small-time locally controllable [2]. It has also been shown [4] that these systems are controllable regardless of the structure of nonholonomic constraints. Given these results, researchers have tried to overcome the problem of stabilizing these systems to equilibrium points by either designing non-smooth feedback controllers [23, 13] or using time-varying feedback laws [14, 18]. Hybrid strategy which employs a discontinuous state feedback that combines time-invariant smooth feedback control with time-varying smooth feedback control is discussed in [15]. On the other hand, stabilization of such systems about trajectories has been studied in [25]. Steering nonholonomic systems using sinusoids [11] and use of dynamic-state feedback to linearize nonholonomic wheeled mobile robots [5] have also been studied.

In this paper, we apply an unified approach developed in [19] to the control of wheeled mobile robots subject to both holonomic and nonholonomic constraints. We first discuss two broad categories of output equations. In the first category, the output equations consists of a subset of generalized coordinates, say p. Thus the system is designed to follow a desired trajectory, $p^d(t)$ where t is the time. This is called *trajectory tracking* in this paper. In the other category, the history $p^d(t)$ is not as important as the path, $p^d(s)$. Here s is any convenient parameter (say an arc-length variable) that parameterizes the path. The trajectory tracking control scheme is shown in Figure 1 (a) for a vehicle subject to nonholonomic constraints. The desired path is a straight line and the initial position is not on the path. The locus of a reference point on the vehicle is shown in the figure for different desired forward velocities. By forward velocity we mean the component of the velocity of the reference point perpendicular to the axis of the wheels in a preferred "forward" direction. If a trajectory tracking control algorithm is employed, the path of the reference point is not a "smooth merge". It is possible that the vehicle may first go in one direction and then the opposite direction depending on the definition

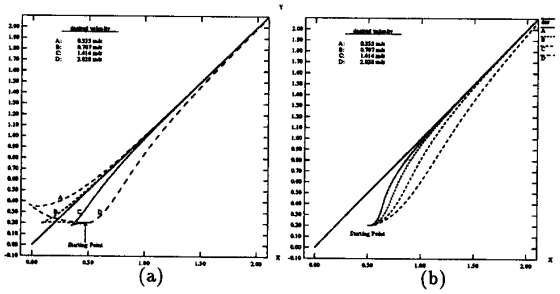

Figure 1: (a) Trajectories and (b) paths for the geometric center P_o.

of $p^d(t)$. While this may be acceptable and even desirable for some applications, for road following or path following, the paths shown in Figure 1 (b) are more appropriate. Here the desired output is specified in terms of the path $p^d(s)$ and the speed along the path, $\dot{s}$. This is termed *dynamic path following* in this paper. The adjective dynamic is used to emphasize the dynamic, closed loop nature of the control algorithm. Thus this method is different from a *geometric path following* method such as *pure pursuit* [24]. We note that a particular case of trajectory tracking (Type I output in Section 2.3) has earlier been studied by other researchers [4, 17, 7].

For each category of output equations, we develop a nonlinear feedback which realizes the input-output linearization and input-output decoupling. A computer simulation of a mobile robot is used to study both types of control schemes through numerical experiments and their performances are compared. It is concluded that a *dynamic path-following* scheme is more appropriate for vehicle control applications. We then thoroughly investigate the performance of the new algorithm for different conditions.

2 Dynamics and Control of a Mobile Platform

2.1 Constraint Equations

We consider a mobile robot similar to the LABMATE mobile platform for our analysis. It has two driving wheels on an axis which passes through the vehicle geometric center. They are powered by D.C. motors. The platform has four passive wheels (castors) on each corner.

The following notation will be used in the paper (see Figure 2).

- x–y: the world coordinate system;
- X–Y: the coordinate system fixed to the mobile platform as shown in Figure 2;
- P_o: the geometric center with coordinates (x_o, y_o) which is the intersection of the axis of symmetry with the driving wheel axis;
- P_c: the center of mass of the platform with coordinates (x_c, y_c);
- P_l: a virtual reference point attached to the platform with coordinates (x_l, y_l);

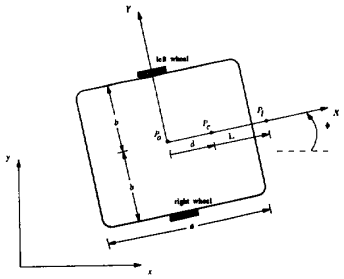

Figure 2: Notation for the geometry of the mobile platform

b: the distance between either driving wheel and the axis of symmetry;
r: the radius of each driving wheel;
m_c: the mass of the platform without the driving wheels and the rotors of the DC motors;
m_w: the mass of each driving wheel plus the rotor of its motor;
I_c: the moment of inertia of the platform without the driving wheels and the rotors of the motors about a vertical axis through P_c;
I_w: the moment of inertia of each wheel and the motor rotor about the wheel axis;
I_m: the moment of inertia of each wheel and the motor rotor about a wheel diameter;
a: the length of the platform in the direction perpendicular to the driving wheel axis;
d: the distance from P_o to P_c along the positive X-axis.

If we ignore the passive wheels, the configuration of the platform can be described by five generalized coordinates. These are the three variables that describe the position and orientation of the platform and two variables that specify the angular positions for the driving wheels. Therefore, the Lagrange coordinates are

$$q = (x_c, y_c, \phi, \theta_r, \theta_l)$$

where (x_c, y_c) is the coordinates of the center of mass P_c in the world coordinate system, and ϕ is the heading angle of the platform as shown in Figure 2. θ_r and θ_l are the angular positions of the right and left driving wheels respectively.

Assuming the driving wheels roll (and do not slip) there are three constraints. First, the velocity of the point P_o of the platform must be in the direction of the axis of symmetry, the X-axis:

$$\dot{y}_c \cos\phi - \dot{x}_c \sin\phi - \dot{\phi}\, d = 0 \tag{1}$$

Further, if the driving wheels do not slip,

$$\dot{x}_c \cos\phi + \dot{y}_c \sin\phi + b\dot{\phi} = r\dot{\theta}_r \tag{2}$$
$$\dot{x}_c \cos\phi + \dot{y}_c \sin\phi - b\dot{\phi} = r\dot{\theta}_l \tag{3}$$

The three constraints can be written in the form:

$$A(q)\dot{q} = 0 \tag{4}$$

where

$$A(q) = \begin{bmatrix} -\sin\phi & \cos\phi & -d & 0 & 0 \\ -\cos\phi & -\sin\phi & -b & r & 0 \\ -\cos\phi & -\sin\phi & b & 0 & r \end{bmatrix} \tag{5}$$

Thus the mechanical system has two degrees of freedom. It can be shown that out of these three constraints, two are nonholonomic and one is holonomic [20].

2.2 Dynamic Equations

We now derive the dynamic equation for the mobile platform. The Lagrange equations of motion of the platform with the Lagrange multipliers λ_1, λ_2, and λ_3 are given by

$$m\ddot{x}_c + 2m_w d(\ddot{\phi}\sin\phi + \dot{\phi}^2\cos\phi) - \lambda_1 \sin\phi - (\lambda_2 + \lambda_3)\cos\phi = 0 \tag{6}$$
$$m\ddot{y}_c - 2m_w d(\ddot{\phi}\cos\phi - \dot{\phi}^2\sin\phi) + \lambda_1 \cos\phi - (\lambda_2 + \lambda_3)\sin\phi = 0 \tag{7}$$
$$2m_w d(\ddot{x}_c \sin\phi - \ddot{y}_c \cos\phi) + I\ddot{\phi} - d\lambda_1 + b(\lambda_3 - \lambda_2) = 0 \tag{8}$$
$$I_w\ddot{\theta}_r + \lambda_2 r = \tau_r \tag{9}$$
$$I_w\ddot{\theta}_l + \lambda_3 r = \tau_l \tag{10}$$

where

$$m = m_c + 2m_w$$
$$I = I_c + 2m_w(d^2 + b^2) + 2I_m$$

and τ_r and τ_l are the torques acting on the wheel axis generated by the right and left motors respectively. These five equations of motion can be written in the form:

$$M(q)\ddot{q} + V(q,\dot{q}) = E(q)\tau - A^T(q)\lambda \tag{11}$$

where $M(q)$ is the 5×5 inertia matrix, $V(q,\dot{q})$ is the vector of position and velocity dependent forces, $E(q)$ is the 5×2 input transformation matrix[1], τ is the 2-dimensional input vector, $A(q)$ is the 3×5 Jacobian matrix, and λ is the vector of constraint forces. The matrices $M(q)$, $V(q,\dot{q})$, $E(q)$ and τ are given by:

$$M(q) = \begin{bmatrix} m & 0 & 2m_w d\sin\phi & 0 & 0 \\ 0 & m & -2m_w d\cos\phi & 0 & 0 \\ 2m_w d\sin\phi & -2m_w d\cos\phi & I & 0 & 0 \\ 0 & 0 & 0 & I_w & 0 \\ 0 & 0 & 0 & 0 & I_w \end{bmatrix}$$

$$V(q,\dot{q}) = \begin{bmatrix} 2m_w d\dot{\phi}^2\cos\phi \\ 2m_w d\dot{\phi}^2\sin\phi \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad E(q) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad \tau = \begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix}$$

Let us define a 5×2 matrix $S(q)$ such that it satisfies $A(q)S(q) = 0$. One choice of such $S(q)$ is given by

$$S(q) = [s_1(q),\ s_2(q)] = \begin{bmatrix} c(b\cos\phi - d\sin\phi) & c(b\cos\phi + d\sin\phi) \\ c(b\sin\phi + d\cos\phi) & c(b\sin\phi - d\cos\phi) \\ c & -c \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

where the constant $c = \frac{r}{2b}$.

Noting the constraint equation (4), we find that $\dot{q}$ is in the null space of $A(q)$. Since the two columns of $S(q)$ are in the null space of $A(q)$ and are linearly independent, it is possible to express $\dot{q}$ as a linear combination of $s_1(q)$ and $s_2(q)$, *i.e.*,

$$\dot{q} = \nu_1 s_1(q) + \nu_2 s_2(q)$$

In this case, because of the choice of $S(q)$, it happens that

$$\nu = \begin{bmatrix} \nu_1 \\ \nu_2 \end{bmatrix} = \begin{bmatrix} \dot{\theta}_l \\ \dot{\theta}_r \end{bmatrix}$$

Now, following the method as described in [2, 19] we choose the following vector as the state variable:

$$x = [x_c \ \ y_c \ \ \phi \ \ \theta_l \ \ \theta_r \ \ \dot{\theta}_l \ \ \dot{\theta}_r]^T$$

and after some manipulation we represent the dynamic system of wheeled mobile robot in the following state-space form:

$$\dot{x} = \begin{bmatrix} S\nu \\ f_2 \end{bmatrix} + \begin{bmatrix} 0 \\ (S^T M S)^{-1} S^T E \end{bmatrix} \tau \tag{12}$$

where $f_2 = (S^T M S)^{-1}(-S^T M \dot{S}\nu - S^T V)$.

We further simplify the state equation to the form

$$\dot{x} = f(x) + g(x)u \tag{13}$$

[1] $E(q)$ is an identity matrix in most cases. However, if the generalized coordinates are chosen to be some variables other than the joint variables, or if there are passive joints without actuators, it is not an identity matrix.

where

$$f(x) = \begin{bmatrix} S(q)\nu \\ 0 \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 \\ I \end{bmatrix}$$

by applying the nonlinear feedback

$$\tau = S^T M S(u - f_2)$$

Here we have used the fact that $S^T E = I$.

2.3 Output Equations

While the state equations of a dynamic system are uniquely determined by its dynamic characteristics, the output variables are chosen in such a way that the tasks to be performed by the dynamic system can be *conveniently specified* and the controller design can be *easily accomplished*. For example, if a six degree-of-freedom robot manipulator is to perform pick-and-place or trajectory tracking tasks, the six-dimensional joint position vector or the six-dimensional Cartesian position and orientation vector is normally chosen as the output vector. In this section, we present a number of possible choices for output variables of the control system for the mobile platform and discuss each case.

Let P_l be the reference point on the mobile platform. We choose P_l to be a virtual point on the axis of symmetry displaced through a distance L from the center of mass P_c as shown in Figure 2 (L can be positive, negative, or zero). Its coordinates are denoted by (x_l, y_l):

$$x_l = x_c + L\cos\phi \tag{14}$$
$$y_l = y_c + L\sin\phi \tag{15}$$

Because the system has two inputs, we may choose any two output variables. We consider the following four types of output equations:

Type I: $y = h(q) = [x_l \quad y_l]^T$
Type II: $y = h(q) = [x_l \quad \phi]^T$
Type III: $y = h(q) = [y_l \quad \phi]^T$
Type IV: $y = h(x) = [h_1(q) \quad h_2(\nu)]^T$

The necessary and sufficient condition for input-output linearization is that the decoupling matrix has full rank [12]. If the output equations are functions of position state variable q only (which covers the first three types of outputs as shown above; Type IV output is discussed separately in the next section), *i.e.* if

$$y = h(q) = [h_1(q) \quad h_2(q)]^T \tag{16}$$

then decoupling matrix $\Phi(q)$ for the system is the 2×2 matrix

$$\Phi(q) = J_h(q)S(q) \tag{17}$$

where $J_h = \frac{\partial h}{\partial q}$ is the 2×5 Jacobian matrix. $\Phi(q)$ is nonsingular if the rows of J_h are independent of the rows of $A(q)$. The Type I output equation results in a trajectory tracking control system which has been studied in [6, 17]. The corresponding decoupling matrix for this output is

$$\Phi(q) = J_h(q)S(q) = \begin{bmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{21} & \Phi_{22} \end{bmatrix} \tag{18}$$

where

$$\Phi_{11} = c(b\cos\phi - (d+L)\sin\phi) \tag{19}$$
$$\Phi_{12} = c(b\cos\phi + (d+L)\sin\phi) \tag{20}$$
$$\Phi_{21} = c(b\sin\phi + (d+L)\cos\phi) \tag{21}$$
$$\Phi_{21} = c(b\sin\phi - (d+L)\cos\phi) \tag{22}$$

Since the determinant of the decoupling matrix is $\det(\Phi) = -\frac{r^2(d+L)}{2b}$, it is singular if and only if $L = -d$, that is, if point P_l coincides with point P_o. Therefore, trajectory tracking of the point P_o is not possible as pointed out in [17]. This is clearly due to the presence of nonholonomic constraints. Choosing L not equal to $-d$, we may decouple and linearize the system as follows. The derivative of the decoupling matrix is (noting that Φ is a function of ϕ only)

$$\dot{\Phi}(q) = \frac{d\Phi}{d\phi}\dot{\phi} = \frac{d\Phi}{d\phi}c(\nu_1 - \nu_2)$$

Now, differentiating the output equation (16) twice and noting that $\dot{\nu} = u$, we identify the following nonlinear feedback

$$u = \Phi^{-1}(q)(v - \dot{\Phi}(q)\nu) \tag{23}$$

which upon application results in the following linearized and decoupled subsystems:

$$\ddot{y}_1 = v_1 \tag{24}$$
$$\ddot{y}_2 = v_2 \tag{25}$$

Type II and Type III outputs are similar. For Type II output equation, the decoupling matrix is

$$\Phi_{II}(q) = J_h(q)S(q) = \begin{bmatrix} \Phi_{11} & \Phi_{12} \\ c & -c \end{bmatrix} \tag{26}$$

where Φ_{11} and Φ_{12} are defined by Equations (19) and (20). Its determinant is $\det\Phi_{11}(q)) = -2c^2 b\cos\phi$. Φ_{II} is nonsingular if $\cos\phi \neq 0$. Similarly, the decoupling matrix for Type III output is nonsingular if $\sin\phi \neq 0$. Thus, it is possible to decouple and linearize the system with Type II and Type III outputs in a large region of the state space. However, it is not convenient to specify control tasks with these two types of outputs. We discuss Type IV output equations and the *dynamic path following problem* in the next section.

2.4 Dynamic Path Following

If we analyze automobile manoeuvring, the two most important requirements are to follow the road (or path) by staying as close to the path as possible and to maintain the desired forward velocity. With this in mind, we would like to choose an output equation with two variables: the shortest distance of a reference point on the mobile platform from the desired path and the forward velocity. By doing so, we formulate the control problem as a dynamic path following problem [21] instead of as a trajectory tracking problem. In a trajectory tracking problem, the desired time history of the output variables is specified. Therefore, in this case, the task is not only to reach a point but also to reach it at a specified time instant. In a path following problem, however, the geometry of the path is specified. In this case, it is more important to follow the path closely than to reach points on the path at specified time instants. By specifying the desired forward velocity, we (indirectly) ensure that the vehicle reaches desired points on the path.

In this section, we achieve path following by appropriately choosing h_1 and h_2. h_1 is defined as the shortest distance from the point P_l on the mobile platform to the desired path. The formulation is quite general since P_l can be anywhere on the vehicle, although we prefer to choose P_l on the X-axis. However, note that for an arbitrary path, there is no closed form expression for the shortest distance from P_l to the path. We define h_2 to be the component of the velocity of P_l along the X-axis. We call this the forward velocity.

We first consider two basic paths: a straight line path and a circular path. A closed form expression for the distance from a point to the path can be easily obtained in either case.

We first consider a circular path. Let P_f be the center of the circular path whose coordinates are denoted by (x_f, y_f) in the world coordinate system. Let R be the radius of the circular path. We choose h_1 as follows:

$$h_1(q) = h_1(x_c, y_c, \phi) = ((x_l - x_f)^2 + (y_l - y_f)^2)^{1/2} - R \tag{27}$$

Note that the shortest distance from point P_l to the circular path is the absolute value of $h_1(q)$. Here (x_f, y_f) and R are constants and x_l and y_l are related to the state variables, x_c, y_c, and ϕ, by Equations (14) and (15). The forward velocity of the platform is given by

$$h_2(\nu) = \dot{x}_c\cos\phi + \dot{y}_c\sin\phi = \frac{r}{2}(\nu_1 + \nu_2) \tag{28}$$

It is clear that we have a Type IV output. The decoupling matrix for this output equation is computed as follows.

$$\dot{y}_1 = \frac{\partial h_1}{\partial x}\dot{x} = J_{h_1}(q)S(q)\nu$$
$$\ddot{y}_1 = \frac{\partial(J_{h_1}S\nu)}{\partial q}S(q)\nu + J_{h_1}(q)S(q)u$$
$$\dot{y}_2 = J_{h_2}u$$

where

$$J_{h_1}(q) = \frac{\partial h_1}{\partial q} = \frac{1}{d_c}\begin{bmatrix} x_c - x_f + L\cos\phi \\ y_c - y_f + L\sin\phi \\ L\cos\phi(y_c - y_l) - L\sin\phi(x_c - x_f) \\ 0 \\ 0 \end{bmatrix}^T$$

$$J_{h_2} = [\frac{r}{2} \quad \frac{r}{2}]$$

$$d_c = \sqrt{(x_l - x_f)^2 + (y_l - y_f)^2}$$

Therefore, the decoupling matrix is

$$\Phi = \begin{bmatrix} J_{h_1}(q)S(q) \\ J_{h_2} \end{bmatrix} \tag{29}$$

and the determinant of Φ is

$$\det(\Phi) = \frac{rc(d+L)}{d_c}((y_c - y_f)\cos\phi - (x_c - x_f)\sin\phi)$$

It follows that the decoupling matrix is singular if (1) $L = -d$ (point P_l coincides with point P_o), or (2) $(y_c - y_f)\cos\phi = (x_c - x_f)\sin\phi$ (the heading direction of the platform or the X-axis is normal to the circular path). While the first condition is due to the nonholonomic constraint of the platform, the second condition is due to the fact that the direction along the path is not explicitly specified in the function $h_1(q)$. Thus when the X-axis is normal to the path, specifying the forward velocity does not uniquely specify the path direction. We also note that in order to avoid this type of singularity it is beneficial to have $L > -d$ if the forward velocity is positive. Motion in the reverse direction can be accomplished with $L < -d$ and a negative forward velocity.

We now consider a straight line path. Let the path be described by $Ax + By + C = 0$.

$$h_1(x_c, y_c, \phi) = \frac{Ax_l + By_l + C}{\sqrt{A^2 + B^2}} \tag{30}$$

Once again, the shortest distance from point P_l to the path is the absolute value of h_1. The second component of the output equation, h_2, is the same as for the circular path. The decoupling matrix Φ has the same form except that J_{h_1} is now replaced by

$$J_{h_1} = \frac{1}{\sqrt{A^2 + B^2}}\begin{bmatrix} A \\ B \\ BL\cos\phi - AL\sin\phi \\ 0 \\ 0 \end{bmatrix}^T$$

The determinant of Φ is

$$\det\Phi = \frac{rc(d+L)}{\sqrt{A^2 + B^2}}(B\cos\phi - A\sin\phi)$$

Once again, the decoupling matrix is singular if $L = -d$ or the X-axis is perpendicular to the straight line path.

More generally, if $f(x, y) = 0$ is an arbitrary path, solving the shortest distance from point P_l to the path involves solving the extremization problem

$$\min_{x,y} \left\{\frac{1}{2}((x - x_l)^2 + (y - y_l)^2) + \lambda^T f(x, y)\right\}$$

A closed form solution is impossible in a general case. However, an approximate expression for h_1 may be used instead:

$$h_1 = f(x_l, y_l) \tag{31}$$

Although h_1 in this case is not the true distance to the path, it is a measure of the closeness to the given path. In the two basic paths discussed above, it is noted that the distance representation differs from the path description only by a constant. Finally, we note that Dubins [8] and later Reeds and Shepp [16] showed that given any initial and final position and orientation of a car, there exists a family of paths composed of only straight line and circular arc segments between any two points. In fact, Dubins proved that this family contains the R-geodesic[2] between the two points. We use this result to argue that any path can be suitably broken down into straight line and circular arc segments. Therefore if we are able to control the mobile platform on such basic paths and on piecewise continuous paths composed of these two, we can effectively move from any position and orientation to any other position and orientation.

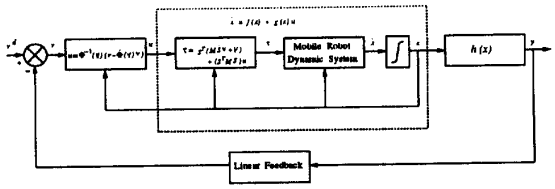

Figure 3: Schematic of the control algorithms

For either of the two basic paths or arbitrary path, by applying the nonlinear feedback, Equation (23), we obtain a linearized and decoupled system in the form

$$\ddot{y}_1 = v_1 \tag{32}$$

$$\dot{y}_2 = v_2 \tag{33}$$

A linear feedback can be designed to make each subsystem stable and to meet the performance specifications.

2.5 Design of the Control Algorithms

We presented two types of control algorithms for mobile robots: (a) trajectory tracking; (b) path following. While they differ in the selection of output equations, the basic scheme is the same as shown in Figure 3. In the figure, v^d is the reference (desired) values for the outputs, h_1 and h_2. The nonlinear feedback (Equation (2.2)) cancels the nonlinearity in the dynamics so that the state equation is simplified into the form of Equation (13). This is represented by the dotted block in Figure 3. Note that the nonlinearity in the kinematics remains in the simplified state equation. A second nonlinear feedback (Equation (20)) linearizes and decouples the input-output map. The overall system is thus decoupled into two linear subsystems. For trajectory tracking, both subsystems are of second order. In the case of path following, the distance control subsystem is of second order, and the velocity control subsystem is of first order. To stabilize these subsystems and to achieve the desired performance, an outer linear feedback loop is designed to place the poles of the system.

3 Evaluation of Control Schemes

3.1 The Dynamic Simulation

We developed a computer simulation in order to verify the validity of the dynamic model and the effectiveness of the control algorithm discussed in the previous sections for a mobile platform that is kinematically similar to the LABMATE. The dimensions and the inertial parameters are representative of the LABMATE platform. According to the notation introduced early, we have: b = 0.75m; d = 0.30m; a = 2.00m; m_c = 30.00kg; m_w = 1.00kg; I_c = 15.625kg-m^2; I_w = 0.005kg-m^2; I_m = 0.0025kg-m^2.

The virtual reference point P_l was chosen to be coincident with P_c. The gains for the linear outer loop were designed in such a way that we got an over-damped system for the decoupled position control subsystem. This is appropriate if we want to follow a wall or a path with a median or a divider. In such a case the platform should not overshoot its desired path. On the other hand, we can choose a critically-damped system if we want to follow a curve on the middle of the road.

[2] An R-geodesic is the minimal length path between two points having an average curvature everywhere less than or equal to R^{-1}, where R is a fixed positive number.

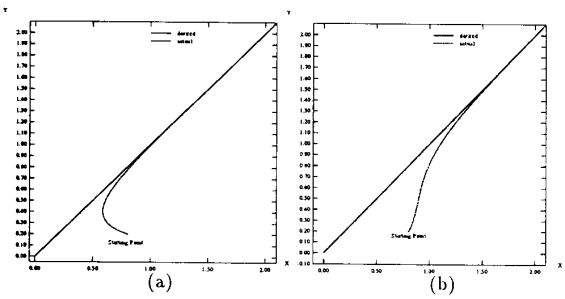

Figure 4: The motion of a reference point on a wheeled vehicle in (a) Trajectory tracking and (b) path following.

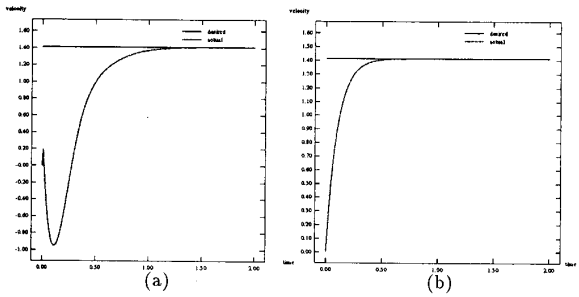

Figure 5: Forward velocity of the geometric center P_o in (a) trajectory tracking ; (b) path following.

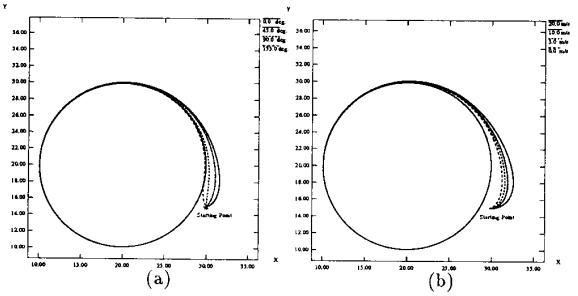

Figure 6: Path followed with : (a) different initial heading angles (in degrees) for an initial forward velocity of 5 meters/sec; (b) different initial forward velocities in (meters/sec) for an initial heading angle of 0 degrees.

3.2 Trajectory tracking versus dynamic path following

Consider a straight line path, $y = x$, as shown in Figures 1 and 4. The reference point is defined so that $L = 0.0$ meters. The initial position is such that

$$(x_c, y_c) = (x_l, y_l) = (0.8, 0.2)$$

and the initial velocity is zero. The desired forward velocity is 1.414 m/sec. For the trajectory tracking algorithm,

$$h_1 = x_l, \quad h_2 = y_l$$

$$v_1^d = t, \quad v_2^d = t$$

For the path following algorithm,

$$h_1 = \frac{1}{\sqrt{2}}(y_l - x_l), \quad h_2 = \frac{r}{2}(\nu_1 + \nu_2)$$

$$v_1^d = 0, \quad v_2^d = 1.414$$

In both cases, as shown in Figure 4 the reference point is able to reach the path and stay on the path. Note that the gains for the position variables are same for both cases. The path following algorithm seems to exhibit a gradual merge while the trajectory tracking reacts more quickly and, depending on the gains, it even forces the reference point in the wrong direction. Depending on the point of interest, the trajectory tracking algorithm also results in cusps in the trajectory. For example, consider the locus of the geometric center, P_o, in both cases for different desired forward velocities as shown in Figures 1. With a Type IV output equation the actual path followed is smooth but the Type I output equation often produces a discontinuity in the slope of the trajectory. This can also be seen from the velocity history shown in Figure 5. In this figure we have shown the forward velocity corresponding to case C of Figure 1. With the Type IV output equation the forward velocity exhibits a smooth exponential response that is typical of a first order system as expected (Figure 5 (b)). The trajectory tracking algorithm may result in discontinuities in velocities. As shown in Figure 5 (a), the center of the vehicle is accelerated and then decelerated to a stop twice before monotonically increasing to the desired velocity. If the objective is to follow a desired path, as is the case in autonomous navigation [21], it is clear that the path following algorithm is more appropriate. It is possible that trajectory tracking may be desirable in applications in which time is a critical parameter. It appears that for a wide range of applications in robotics that path following is the more appropriate strategy. For this reason, from this point on, we concentrate on Type IV output equations. The results of numerical experiments with path following algorithms are presented in the remainder of this section.

3.3 Performance of dynamic path following algorithms

3.3.1 Effect of initial conditions

The initial condition that most affects the trajectory is the initial velocity. Hence the two most important parameters are the initial heading angle (ϕ) and the magnitude of the initial forward velocity. Figure 6 (a) shows how the mobile platform follows a circular path when it starts with a forward velocity of 5m/s but with different initial heading angles (ϕ). Here the initial reference point position is

$$(x_l, y_l) = (30.0, 15.0)$$

and

$$h_1(q) = ((x_l - 20.0)^2 + (y_l - 20.0)^2)^{1/2} - 10.0, \quad h_2 = \frac{r}{2}(\nu_1 + \nu_2)$$

$$v_1^d = 0, \quad v_2^d = 1.414$$

We note that the algorithm is singular if the vehicle is oriented along the shortest line joining the reference point and the path. In this case an orientation with $\phi = -26.6$ deg or 153.4 deg leads to problems. For heading angles other than these, the response is satisfactory as seen from the Figure 6 (a).

Figure 6 (b) depicts the system response for a desired circular path for different initial forward velocities but with a constant heading angle (in this case it was 0 degrees). As expected, if the initial heading is away from the desired path (as in this case), the system exhibits better performance when the initial speed is less.

3.3.2 Effect of modeling uncertainties

Modeling errors can result due to the difficulty in measuring or estimating the geometric, kinematic or inertial parameters or from the lack of a complete knowledge of the components of the system. We simulated modeling errors in the inertial parameters namely m_c and m_w and investigated the performance on a straight line path as well as on a circular path. For as much as a 100 percent modeling error in m_c and m_w, the control scheme follows the path quite well. Although a theoretical robustness analysis was not performed, it appears from extensive simulations that the scheme is quite robust. Figures 7 (a) and (b) shows the response of the system for a straight line and circular path respectively. It takes a longer time in comparison to the case when there is no modelling error (as seen from the path C of Figure 1 (b)), to converge to the desired path for a straight line path whereas it follows the circular path but only with a small constant offset.

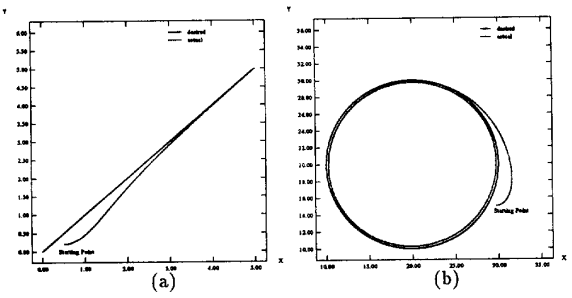

Figure 7: Desired and actual path of the mobile platform when there is a modeling error in m_c and m_w by 100% for the case of a : (a) straight line ; (b) circle.

4 Conclusion

We have presented a general, unified method of controlling wheeled mobile platforms subject to both holonomic as well as nonholonomic constraints. We have derived a nonlinear feedback that guarantees input-ouput stability and Lagrange stability for the overall system. We have investigated the importance of the choice of the output variables in the dynamical model of the system to be controlled and have shown the system response for two types of control algorithms: trajectory tracking and path following. While trajectory tracking may be essential in situations where the vehicle must follow a curve in space-time coordinates, that is, it must reach a particular point in space at a particular instant of time, the dynamic path following is a more realistic control problem where the vehicle needs to follow a geometric path with a desired speed. The dynamic path following problem has been presented here for the first time. Computer simulation results are presented to illustrate and compare the performance of each algorithm. Based on these, it has been concluded that a dynamic path following algorithm is more appropriate for vehicle control applications. Finally, the effects of modelling errors and sensor noise are investigated through numerical simulations.

Acknowledgment

This work is in part supported by NSF grants BCS-92-16691, CDA-92-22732, CISE/CDA-90-2253, CISE/CDA 88-22719, and IRI-92-09880, ONR/DARPA grants N0014-92-J-1647 and N0014-88-K-0630, Army/DAAL grant 03-89-C-0031PRI, the Whitaker Foundation, and the University of Pennsylvania Research Foundation.

References

[1] J. Barraquand and Jean-Claude Latombe. Nonholonomic multibody mobile robots: controllability and motion planning in the presence of obstacles. In *Proceedings of 1991 International Conference on Robotics and Automation*, pages 2328–2335, Sacramento, CA, April 1991.

[2] A. Bloch, M. Reyhanoglu, and N. McClamroch. Control and stabilization of nonholonomic dynamic systems. *IEEE Transactions on Automatic Control*, 37(11):1746–1757, November 1992.

[3] Anthony Bloch and N. H. McClamroch. Control of mechanical systems with classical nonholonomic constraints. In *Proceedings of 28th IEEE Conference on Decision and Control*, pages 201–205, Tampa, Florida, December 1989.

[4] G. Campion, B. d'Andrea-Novel, and G. Bastin. Controllability and state feedback stabilization of non holonomic mechanical systems. In C. Canudas de Wit, editor, *Lecture Notes in Control and Information Science*, pages 106–24, Springer-Verlag, 1991.

[5] B. d'Andrea-Novel, G. Bastin, and G. Campion. Dynamic feedback linearization of nonholonomic wheeled mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2527–2532, Nice, France, May 1992.

[6] B. d'Andrea-Novel, G. Bastin, and G. Campion. Modelling and control of non holonomic wheeled mobile robots. In *Proceedings of 1991 International Conference on Robotics and Automation*, pages 1130–1135, Sacramento, CA, April 1991.

[7] C. Canudas de Wit and R. Roskam. Path following of a 2-DOF wheeled mobile robot under path and input torque constraints. In *Proceedings of 1991 International Conference on Robotics and Automation*, pages 1142–1147, Sacramento, CA, April 1991.

[8] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79:497–516, 1957.

[9] Y. Koren and J. Borenstein. Potential field methods and their inherent limitations for mobile robot navigation. In *Proceedings of 1991 International Conference on Robotics and Automation*, pages 1398–1404, Sacramento, CA, April 1991.

[10] J. P. Laumond. Finding collision-free smooth trajectories for a non-holonomic mobile robot. In *10th International Joint Conference on Artificial Intelligence*, pages 1120–1123, Milano, Italy, 1987.

[11] R. Murray and S. Sastry. Steering nonholonomic systems using sinusoids. In *Proceedings of the 31st IEEE Conference on Control and Decision*, pages 2097–2101, 1990.

[12] H. Nijmeijer and A. J. van der Schaft. *Nonlinear Dynamic Control Systems*. Springer-Verlag, New York, 1990.

[13] G. Pappus and K. Kyriakopoulos. Modeling and feedback control of nonholonomic mobile vehicles. In *Proceedings of the 31st IEEE Conference on Control and Decision*, pages 2680–2685, Tuscon, Arizona, December 1992.

[14] J. Pomet. Explicit design of time-varying stabilizing control laws for a class of controllable systems without drift. *Systems and Control Letters*, 18(2):147–158, 1992.

[15] J. Pomet, B. Thuilot, G. Bastin, and G. Campion. A hybrid strategy for the feedback stabilization of nonholonomic mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 129–134, Nice, France, May 1992.

[16] J. A. Reeds and L. A. Shepp. Optimal paths for a car that goes both forwards and backwards. *Pacific Journal of Mathematics*, 145(2):367–393, 1990.

[17] C. Samson and K. Ait-Abderrahim. Feedback control of a nonholonomic wheeled cart in cartesian space. In *Proceedings of 1991 International Conference on Robotics and Automation*, pages 1136–1141, Sacramento, CA, April 1991.

[18] C. Samson and K. Ait-Abderrahim. Feedback stabilization of a nonholonomic wheeled mobile robot. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, 1991.

[19] N. Sarkar, X. Yun, and V. Kumar. Dynamic control of 3-D rolling in multi-arm manipulation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Atlanta, Georgia, May 1993.

[20] Nilanjan Sarkar, Xiaoping Yun, and Vijay Kumar. *Control of Mechanical Systems with Rolling Constraints: Application to Dynamic Control of Mobile Robots*. Technical Report MS-CIS-92-44, GRASP LAB 320, Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104, 1992.

[21] S. Singh, D. Feng, P. Keller, G. Shaffer, W. F. Shi, D. H. Shin, J. West, and B. X. Wu. *A System for Fast Navigation of Autonomous Vehicles*. Technical Report CMU-RI-TR-91-20, The Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, September 1991.

[22] S. Singh and P. Keller. Obstacle detection for high speed autonomous navigation. In *Proceedings of 1991 International Conference on Robotics and Automation*, pages 2798–2805, Sacramento, CA, April 1991.

[23] O. Sordalen and C. Canudas de Wit. Exponential control law for a mobile robot: extension to path following. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2158–2163, Nice, France, May 1992.

[24] R. Wallace, A. Stentz, H. Moravec, W. Whittaker, C. Thorpe, and T. Kanade. First results in robot road-following. In *International Joint Conference on Artificial Intelligence*, August 1985.

[25] G. Walsh, D. Tilbury, S. Sastry, R. Murray, and J. Laumond. Stabilization of trajectories for systems with nonholonomic constraints. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1999–2004, Nice, France, May 1992.