Abstract

This paper presents a time invariant kinematic motion controller for wheeled mobile robots. Actuator capability, mechanical design, and traction forces governed by terrain features provide velocity and curvature limitations that are used in the design of the controller. A novel path manifold that considers curvature limitations is introduced to provide a desired path shape and convergence to the reference posture or trajectory. Lyapunov techniques are then used to derive a control law that asymptotically converges the robot to an arbitrarily small neighborhood of the path manifold. Posture regulation, path following, and trajectory tracking capability to a similarly scaled neighborhood of the target are provided. Controller parameters are optimized and initial conditions are identified that satisfy physical constraints of the robot and provide smooth commands. Curvature boundaries and asymptotic convergence naturally limit allowable initial conditions and are resolved by driving the robot to intermediate goal points within regions of allowable initial conditions. Posture regulation is evaluated in simulation and experiment on a Compliant Framed wheeled Modular Mobile Robot (CFMMR) for two different terrain surfaces. Trajectory tracking and path following of constant curvature references are evaluated in simulation and experiment, respectively.

KEY WORDS—asymptotic convergence, kinematic motion control, physical constraint, polar coordinates, wheeled mobile robots

1. Introduction

The subject of this research is kinematic motion control algorithms for wheeled mobile robots subject to physical constraints. Limited actuators of real robots can produce only finite speeds, wheel traction forces are limited by tire–ground interaction, and achievable path curvature is limited by physical design and dynamic effects. Most kinematic motion controllers have ignored these physical effects, though, and have focused solely on control and planning considering nonholonomic constraints. It is important to include physical considerations in the kinematic motion controller since these commands are typically used as inputs to the dynamic controller of the physical system (Zhu et al. 2005). Furthermore, the primary motion control tasks of posture regulation, path following, and trajectory tracking have traditionally been treated separately. Thus, the goal of this research is to provide a single kinematic motion control law capable of all three motion control tasks while considering physical constraints. While we do not achieve this goal for all scenarios, we show that our controller can provide this capability given particular regions of initial conditions and paths and trajectories with constrained velocity and curvature.

In this research, physical constraints are based upon models of the robot, its actuators, and wheel/terrain interaction. Actuator models provide torque and speed limits that bound forward and angular velocity of the robot. Mechanical design limits achievable path curvature and further bounds angular velocity. Terrain limits wheel traction and also bounds velocities due to lateral acceleration and roll-over stability. Thus, physical constraints are considered by imposing limits on the forward velocity, angular velocity, and curvature specified by the kinematic motion controller.

In order to satisfy velocity and curvature limitations while performing all three motion control tasks, we first embed curvature constraints in the controller using a novel path manifold. The path manifold is a geometric tool that defines the shape of the path that the robot follows while converging to its target or trajectory. In this case a circular path manifold with radius satisfying curvature constraints is used. The robot is driven to the
path manifold using Lyapunov based control design. Once the robot reaches a neighborhood of the path manifold in finite time, velocity and curvature limitations are satisfied for posture regulation and for sufficiently constrained path and trajectory properties. However, controller commands during convergence to the path manifold critically depend upon initial conditions and path/trajectory properties. Thus, controller parameters are optimized to maximize the region of allowable initial conditions while satisfying curvature and velocity constraints for posture regulation. Initial conditions in immediate proximity to the target are limited by curvature constraints, but this can easily be resolved by providing intermediate target configurations within allowable regions. Controller dynamics are then extended to accommodate actual initial conditions of the robot. We then demonstrate that this time invariant controller provides smooth bounded commands for all three motion control tasks given allowable initial conditions and constrained path/trajectory properties.

As a motivating example, the control algorithm is derived based upon physical limitations of a two-axle Compliant Framed Modular Mobile Robot (CFMMR) at the University of Utah (Minor et al. 2006). The robot uses flexible frame elements to couple rigid differentially steered unicycle type axles, which provides suspension and highly controllable steering capability without any additional hardware. However, the compliant frame accentuates general issues of path curvature and velocity constraints in mobile robots. In early research (Albiset et al. 2003; Minor et al. 2006), control algorithms for the CFMMR were presented to solve posture regulation. However, their focus was axle coordination to minimize traction forces and maximize maneuverability. In contrast, this paper focuses on providing velocity and curvature commands to guide the robot to a posture, along a path, or along a trajectory while satisfying physical constraints.

Thus, major contributions of this paper are: (1) kinematic motion control considering velocity and curvature limitations based upon physical constraints; (2) the path manifold to specify the shape and properties of the path during robot convergence to its goal; (3) the resulting time invariant control law that solves all three motion control tasks; (4) specification of initial conditions that allow the controller to satisfy physical constraints and to provide smooth commands; and (5) the application of this controller to the CFMMR based upon its constraints. This controller can easily be extended to other applications by simply adjusting control parameters based upon physical constraints.

In Section 2 we compare our contributions to existing work. The general kinematics are derived in Section 3 and physical constraints are established in Section 4. The path manifold is introduced in Section 5 and the control law is developed in Section 6. The control strategy is evaluated and discussed in Section 7. Concluding remarks are given in Section 8.

2. Background

Kinematic control of mobile robotic systems has received a great deal of attention in recent years and many motion control schemes have been proposed to consider their nonholonomic constraints. Traditionally, Cartesian coordinates have been used to model and control mobile robots (Coron and d’Andrea-Novel 1993; Samson 1993, 1995; Tayebi et al. 1996), but these efforts have resulted in discontinuous or time varying control laws. This is because a smooth time invariant control law cannot be realized to stabilize nonholonomic robots in Cartesian coordinates as proven by Brockett (1983). Discontinuous velocity trajectories are not easily reproduced on real robots. Time-varying control schemes are generally slow and show oscillatory behaviors as shown in Oriolo et al. (2002). Faster convergence can be achieved by using a non-smooth time-varying controller (Morin and Samson 2000), but it is difficult to ensure bounded velocity and curvature commands since these schemes are based upon conversion to chained form.

It is important to note that Brockett’s theorem (Brockett 1983) requires a system to be continuous in a neighborhood of the equilibrium point. Thus, by introducing discontinuity in the equilibrium point with a polar coordinate system, Brockett’s theorem cannot be applied and a smooth time invariant control law is allowed (Astolfi 1994; Aicardi et al. 1995). The polar representation was introduced by Badreddin and Mansour (1993) to provide local asymptotic stability in posture regulation using time invariant control. Aicardi et al. (1995) and Astolfi (1999) subsequently applied polar coordinates to derive smooth and globally stabilizing state feedback control laws. Singularity occurs at the origin in the polar coordinate system. Singularity issues can be avoided by appropriately selecting initial conditions or intermediate points (Astolfi 1999), or by applying a simple state feedback control law to make the closed loop system nonsingular (Badreddin and Mansour 1993; Astolfi 1999). Thus, the polar representation has been commonly adopted for posture regulation.

One objective of this research is to simultaneously solve the primary motion control problems using a single smooth time invariant control law. Oriolo et al. (2002) presented a discontinuous controller to solve posture regulation and trajectory tracking using dynamic feedback linearization in Cartesian coordinates. Several researchers have provided posture regulation, path following, and/or trajectory tracking in polar coordinates by extending posture regulation results to path following problems (Aicardi et al. 1995; Kim and Minor 2005) or using a discontinuous state feedback controller (Tayebi and Rachid 1997). However, they need to use discontinuous control inputs for boundedness or path following, which may produce large errors or lead to slow convergence. Aforementioned approaches also focus on the controller development itself without resolving robot limitations.

Path curvature is generally an issue for mobile robots given steering restrictions determined by mechanical design and
traction limitations (Albiston and Minor 2003; Minor et al. 2006). As shown in Figure 1, it is assumed that permissible steering paths are restricted. Actuators likewise possess limited capabilities and realizable wheel velocity is restricted. Furthermore, excessive velocity commands may cause wheel slip, large path curvature, or excessive traction forces. In Cartesian coordinates, time-varying control laws for bounded velocity inputs have thus been derived using saturation functions (Lee et al. 2001). Likewise, velocity and curvature constraints have been considered by using saturation and designing control gains, respectively, for posture regulation in polar coordinates (Indiveri 1999; Webers and Zimmer 2000). The former paper (Lee et al. 2001) possesses limitations of time varying control whereas the latter two do not provide path following or trajectory tracking capability.

In contrast to motion control, appreciable motion planning research has considered physical constraints on velocity and path curvature to provide feasible references (Siegwart and Nourbakhsh 2004). Since tracking performance may not be guaranteed using saturation functions, Antonelli et al. (2001) present a motion planning algorithm that slows down the desired trajectory via virtual time when velocity and acceleration constraints are encountered. However, this algorithm is based upon discrete time control and limited to path tracking. Further, curvature constraints may not be satisfied by simply saturating velocities or accelerations. The dynamic window approach (Fox et al. 1997; Brock and Khatib 1999) establishes available forward and steering velocities at each time sample by considering acceleration limitations. Motion planning was then conducted in this dynamic window by using circular arcs. More recently, Spenko et al. (2006) introduced the trajectory space to provide hazard avoidance in rough terrain, which considers forward velocity and curvature limits based upon a car-like vehicle model, terrain parameters, and obstacle properties. We establish similar physical constraints based upon path curvature, forward velocity, steering velocity, and terrain interac-

tion, but we also consider the limitations created by the DC motor. Most notably, though, we consider these constraints in kinematic motion control as opposed to path planning. This helps to assure that the robot does not violate physical constraints while converging to its reference, as opposed to the latter that focuses on assuring that the path itself is mindful of constraints.

In order to realize bounded path curvature in the primary motion control problems with a single control law, we thus present a curvature based geometric approach using path manifolds that considers physical constraints. Several papers (Sellen 1995; Shkel and Lumelsky 1997) discuss constrained path curvature for motion planning based on a geometric approach. Shkel and Lumelsky (1997) present a control strategy to develop a curvature constrained path within a limited workspace using arcs of a circle with minimum radius. However, aforementioned approaches concentrate on the motion planning to generate a purely geometry based path. In other effort (de Wit and Sordalen 1992), circles are used to transform Cartesian coordinates into new discontinuous coordinates. A mobile robot is then stabilized via a piecewise continuous control law and resulting trajectories are piecewise continuous with cusps. The path manifold is, in contrast, a smooth path that a robot posture can be regulated along while satisfying curvature constraints and providing nonsingular uniform coordinates for the primary motion control tasks.

The path manifold has similarity to the sliding surface in sliding mode control in that the system variables converge to the equilibrium point along this manifold using a controller derived using Lyapunov based techniques. However, this research proposes a smooth and continuous kinematic controller based on ideal kinematics without the switching characteristic of sliding mode control. Thus, our kinematic controller provides smooth velocity references that can be used as inputs to a dynamic controller (Zhu et al. 2005). Sliding mode controllers have been used previously in dynamic motion controllers, but these have been used to track ideal velocity commands provided by kinematic motion controllers (Chwa 2004) and to track gradient lines of a potential field for obstacle avoidance (Guldner and Utkin 1993).

3. Kinematic Model

In this section general kinematics of a unicycle type robot are derived in the polar representation. To consider the primary motion control tasks simultaneously, the reference posture is denoted using the reference frame \( \hat{R} \), Figure 2. The reference posture is described using a virtual robot that inherits kinematics of the real robot such that it provides reference paths or trajectories that the robot can follow. We first consider unicycle type kinematic models of robot posture and reference posture described by Cartesian variables, \([x, y, \phi]\) and \([x_r, y_r, \phi_r]\), respectively,
where \( x \) and \( y \) are the Cartesian coordinates of a moving coordinate frame attached to the point \( O \). A reference position \((x_r, y_r)\) is attached to the moving frame \( R \) to describe its location relative to the global frame \( G \). The variable \( v \) represents the velocity of the coordinate frame \( O \) moving in a heading \( \phi \) relative to the global frame \( G \). The subscript \( r \) denotes the reference frame. Thus, \( v_r \) and \( \phi_r \) are the reference velocity and the reference heading angle of the coordinate frame \( R \), respectively. In this research, we focus on forward motion along paths such that \( v_r \geq 0 \). Furthermore, backward motion can easily be realized by using coordinate transformation.

Using the polar representation relative to posture \( O \), the kinematics can be written in error coordinates. The error states in polar representation are defined as,

\[
\begin{align*}
e & = \sqrt{(x - x_r)^2 + (y - y_r)^2} \\
\theta & = \text{ATAN2}((-y - y_r), -(x - x_r)) - \phi_r \\
\alpha & = \theta - \phi + \phi_r.
\end{align*}
\]

(2)

To derive state equations in polar coordinates, differentiate (2),

\[
\begin{align*}
\dot{e} & = \frac{(x_r - x)(\dot{x}_r - \dot{x}) + (y_r - y)(\dot{y}_r - \dot{y})}{e} \\
\dot{\theta} & = \frac{(\dot{y}_r - \dot{y})(x_r - x) - (y_r - y)(\dot{x}_r - \dot{x})}{(x_r - x)^2 \sec^2(\theta + \phi_r)} - \dot{\phi}_r \\
\dot{\alpha} & = \dot{\theta} - \dot{\phi} + \dot{\phi}_r.
\end{align*}
\]

(3)

Substituting (1) and (2) into (3) and applying trigonometric identities, the state equations are then obtained,

\[
\begin{align*}
\dot{x} & = v \cos \phi + v_r \cos \theta \\
\dot{y} & = v \sin \phi - v_r \sin \theta \\
\dot{\theta} & = \frac{v \sin \alpha}{e} - \frac{v_r \sin \theta}{e} - \dot{\phi}_r \\
\dot{\alpha} & = \frac{v \sin \alpha}{e} - \frac{v_r \sin \theta}{e} - \dot{\phi},
\end{align*}
\]

where the angular velocity of point \( O \) can be described as a function of the path curvature, \( \kappa \), and the linear velocity, \( v \), such that \( \dot{\phi} = v \kappa \). Likewise, the reference angular velocity is expressed as \( \dot{\phi}_r = v_r \kappa_r \).

The path following and trajectory tracking problems can be solved by applying traditional nonlinear techniques to (4). However, traditional tracking controllers lack the ability to stabilize the robot to the desired posture when the reference coordinates are fixed (i.e., posture regulation). For this reason, posture regulation and reference tracking have traditionally been considered as different problems. Note that the reference frame \( R \) represents a virtual reference posture such that it may describe the primary motion control tasks simultaneously. For posture regulation, the frame \( R \) is fixed in the global frame \( G \). For path following, the frame \( R \) moves along a predefined geometric path consisting of the locus of positions and orientations. Arbitrary, but bounded, velocities can thus be chosen for the reference. Further, for trajectory tracking, both trajectories of the frame \( R \) and the desired path are identically parameterized by time such that velocities are specified at each position. Thus, by simply modifying reference velocity expressions, the trajectory tracking controller may easily solve all primary motion control tasks.

4. Physical Constraints

Path curvature and velocity are critical issues for mobile robots. In this section we establish curvature and velocity constraints,

\[
|v| \leq v_{\text{max}}, \quad |\dot{\phi}| \leq \dot{\phi}_{\text{max}}, \quad |\kappa| \leq \kappa_{\text{max}},
\]

(5)

based upon actuator limitations, mechanical design, and terrain features limiting robot dynamic motion. These constraints are then used to tune the path manifold based controller in Section 6.

In order to evaluate the effect of actuator limitation on a wheeled mobile robot, we first express forward and steering velocities as a function of wheel velocities, \( \dot{q}_i \):

\[
\begin{bmatrix}
\dot{v} \\
\dot{x}
\end{bmatrix} = \frac{r_w}{2} \begin{bmatrix}
1 & 1 \\
1/d & -1/d
\end{bmatrix} \begin{bmatrix}
\dot{q}_1 \\
\dot{q}_2
\end{bmatrix},
\]

(6)
Table 1. Parameters for the CFMMR.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{\text{max}}$ (V)</td>
<td>40</td>
<td>$n$</td>
<td>48</td>
</tr>
<tr>
<td>$K_B$ (V-sec/rad)</td>
<td>$55.2 \times 10^{-3}$</td>
<td>$m_i$ (kg)</td>
<td>2.38</td>
</tr>
<tr>
<td>$K_T$ (Nm/A)</td>
<td>$55.2 \times 10^{-3}$</td>
<td>$r_w$ (m)</td>
<td>0.073</td>
</tr>
<tr>
<td>$R$ ($\Omega$)</td>
<td>41.5</td>
<td>$d$ (m)</td>
<td>0.172</td>
</tr>
<tr>
<td>$J_{\text{eff}}$ (kg-m$^2$)</td>
<td>$2.28 \times 10^{-6}$</td>
<td>$x_{cg}$ (m)</td>
<td>0.172</td>
</tr>
<tr>
<td>$B_{\text{eff}}$ (Nm-sec/rad)</td>
<td>$2.47 \times 10^{-5}$</td>
<td>$z_{cg}$ (m)</td>
<td>0.168</td>
</tr>
</tbody>
</table>

where $d$ is the half length of an axle and $r_w$ is the wheel radius. Note that $v_{\text{max}}$ occurs when $\dot{q}_1 = \dot{q}_2 = q_{\text{max}}$ and $\phi_{\text{max}}$ corresponds to $\dot{q}_1 = -\dot{q}_2 = q_{\text{max}}$ such that,

$$v_{\text{max}} = r_w q_{\text{max}}, \quad \phi_{\text{max}} = r_w q_{\text{max}}/d. \quad (7)$$

Actuator limited wheel velocities are determined from a geared DC motor model (Craig 1989) given motor voltage, $V_i$, and torque, $\tau_{m,i}$:

$$\dot{q}_i = \frac{1}{n K_B} V_i - \frac{R}{n K_B K_T} \tau_{m,i}, \quad (8)$$

where the gear ratio, $n$, the resistance, $R$, the back EMF constant, $K_B$, and the torque constant, $K_T$, are motor parameters, Table 1. Torque, $\tau_{m,i}$, is determined by motor and robot dynamics, Figure 3:

$$\tau_{m,i} = J_{\text{eff}} \frac{\ddot{q}_i}{n} + B_{\text{eff}} \frac{\dot{q}_i}{n} + r_w \frac{F_{y,i}}{n}, \quad (9)$$

where $J_{\text{eff}}$ and $B_{\text{eff}}$ are the effective inertia and damping (Craig 1989) of the drive system and $F_{y,i}$ is the longitudinal traction force. Using (8) and (9), the wheel velocity is

$$\dot{q}_i = \frac{n K_T V_i - R (n^2 J_{\text{eff}} \dot{q}_i + r_w F_{y,i})}{n^2 (K_B K_T + R B_{\text{eff}})} \quad (10)$$

Note that $F_{y,i} = m_i r_w \dot{q}_i + F_{\text{load}}$ where $F_{\text{load}}$ is an external wheel load and $m_i$ is the mass supported by the wheel.

Given Table 1, maximum velocity, $q_{\text{max}} = 11.3$ rad/sec, occurs at steady state ($\dot{q}_i = 0$) when $V_i = V_{\text{max}}$ and $F_{\text{load}} = 0$. More conservatively, $q_{\text{max}} = 10.7$ rad/s, is determined when $F_{\text{load}} = 1.95$ N at a steering angle, $\psi = 36.5^\circ$, which is the force required to deflect the compliant frame during extreme steering maneuvers (Minor et al. 2006). As a result, (7) defines actuator limited velocities, $v_{\text{act, max}} = 0.78$ m/s and $\phi_{\text{act, max}} = 4.53$ rad/s.

Path curvature, $\kappa$, and ultimately path radius, $r = 1/\kappa$, are correlated with allowable velocities per $v = r \phi$. While an ideal unicycle type robot may achieve $\kappa_{\text{max}} = \infty$, path curvature of typical mobile robots and ground vehicles is usually constrained by mechanical steering limitations. For example, wheel interference in the CFMMR imposes $\kappa_{\text{max}} = 3$ m$^{-1}$ such that $r \geq r_{\text{min}} = 0.34$ m, which significantly restricts velocities achieved by actuators as shown in Figure 4. Note that curvature limited steering rate, $\phi_{\text{cur, max}} = \kappa_{\text{max}} v_{\text{act, max}} = 2.34$ rad/s, is based upon actuator and curvature limitations whereas the forward velocity is still limited by $v_{\text{act, max}}$.

Tire-ground interaction also limits lateral acceleration forces, $m_i a_x = m_i r \dot{\phi}^2 = m_i v \dot{\phi}$, since $F_{x,i} \leq \sqrt{(\mu m g)^2 - (F_{y,i})^2}$, Figure 3, where $\mu$ represents tire-ground friction (Shoop et al. 1994) and $g$ is gravitational acceleration. Given $F_{x,i}$ and $\phi$, the velocity limits based upon lateral traction are then,

$$v_{\text{lat, max}} = \frac{F_{x,i}}{m_i \dot{\phi}}; \quad \dot{\phi} \leq \dot{\phi}_{\text{lat, max}} = \sqrt{\frac{\kappa_{\text{max}} F_{x,i}}{m_i}}. \quad (11)$$

Given $F_{x,i} = F_{\text{load}} = 1.95$ N, conservative velocity constraints based upon lateral acceleration are determined. Friction, $\mu$, bounds $F_{x,i}$ and ultimately governs velocities $v_{\text{lat, max}}$ by (11) as shown in Figure 4. Due to $v_{\text{act, max}}$ and $\dot{\phi}_{\text{lat, max}}$, only a small segment of $v_{\text{lat, max}}$ bounds velocities for $\mu = 0.12$. 

---

Fig. 3. Dynamic model for a wheeled mobile robot.

Fig. 4. Velocity limitations on typical surfaces (Shoop et al. 1994) based upon actuator, curvature, and traction limitations. Shaded area indicates allowable velocities.
While lateral acceleration is not especially limiting in this case, it is also worth noting that lateral acceleration limits (11) are directly applied to roll-over stability using \( \mu = \mu_{\text{roll}} = \frac{x_{cg}}{z_{cg}} \) where \( x_{cg} \) and \( z_{cg} \) are the center of gravity relative to the \( x \) and \( z \) axes. In this case, we have \( \mu_{\text{roll}} = 1.02 \) and Figure 4 indicates roll-over stability at maximum velocities.

In the case of the CFMMR, it is important to note that actuator and curvature limitations impose more conservative bounds. Figure 4. Thus, bounds (5) are determined by \( \kappa_{\text{max}} = 3 \text{ m}^{-1}, v_{\text{max}} = 0.5 \text{ m/s}, \) and \( \phi_{\text{max}} = 1.5 \text{ rad/s} \) with a velocity safety factor of 1.56.

5. The Path Manifold

The path manifold is introduced as a geometric tool to specify path shape during convergence to a desired posture or trajectory. In this research a circular path manifold is defined to satisfy curvature constraints. The path manifold is then applied to the kinematic equations to derive velocity expressions that assure stabilization along the manifold.

To realize a circular path manifold, Figure 5, position and angle conditions are derived first. Position error in the coordinate frame of the reference is determined by,

\[
\hat{x}_e = \hat{x}_r - \hat{x} = r \sin \theta, \quad \hat{y}_e = \hat{y}_r - \hat{y} = r - r \cos \theta,
\]

where \( r \) is the radius of the circular path manifold. Applying (12) to the error definition (2), the position \( e \) is defined as, \( e = \sqrt{(r \sin \theta)^2 + (r - r \cos \theta)^2} = r \sqrt{2(1 - \cos \theta)} \) where requirements for \( e \) can be deduced from Figure 5. Note that the velocity vector \( v \) and reference velocity vector \( v_r \) are tangent to Circles I and II. The position error vector \( e \) bisects the angle between \( v \) and \( v_r \) such that the angle \( \alpha \) is equal and opposite of \( \theta \) on the manifold. Denoting \( \eta = \sqrt{1 - \cos \theta} \geq 0 \), the circular path manifold is thus,

\[
e = \sqrt{2r} \eta, \quad \alpha = -\theta.
\]

Several features of the circular path manifold are noted. First, since the path manifold is tangential to \( \hat{X} \) axis, this results in \( \theta = \alpha = 0 \) at the reference origin. Second, recalling physical constraints (5), we also have \( r \geq r_{\text{min}} = 1/\kappa_{\text{max}} \). In particular, we choose \( r = 1/\kappa_{\text{max}} \) in this paper to demonstrate steering capability using maximum curvature. Finally, the circular path manifold selected depends upon initial orientation, \( \theta(0) \); Circle I for \( 0 < \theta(0) < 0 \) and Circle II for \( -\pi < \theta(0) < 0 \), Figure 5. Thus, paths generated by Circles I and II are symmetric with respect to the \( x \)-axis since Circles I and II are so.

We now show that velocities can be calculated to assure convergence of the error state equations (4) along the path manifold. In order to determine generalized velocity expressions, derivatives of (13) are first expressed as,

\[
\dot{\theta} = 2r \dot{\theta} U \cos \theta.
\]

(14)

where \( U \) is determined based upon the initial condition of the robot and \( \dot{\theta} \) has the opposite sign of \( \theta \) such that \( \dot{\theta} U = -|\dot{\theta}| \).

Applying (13) to (4), we then obtain the system equations along the circular path manifold:

\[
\begin{align*}
\dot{\theta} &= -(v + v_r) \sin \theta \\
\dot{\dot{\theta}} &= -\left(\frac{v + v_r}{2r}\right) \left(U - \dot{\phi}_r\right) \\
\dot{\phi} &= -\left(\frac{v + v_r}{2r}\right) \left(U - \dot{\phi}_r\right) \\
\end{align*}
\]

where path following and trajectory tracking are also considered given \( v_r \) and \( \dot{\phi}_r \). Velocity expressions are then evaluated to make (15) equivalent to (14):

\[
\begin{align*}
v &= v_r + 2r |\dot{\theta}| \\
\dot{\phi} &= 2\dot{\theta} + \dot{\phi}_r,
\end{align*}
\]

which provides stabilization along the path manifold. Thus, these velocities drive the robot to the origin along a circular path such that \( (\theta, \alpha) \) converge to zero. Once at the origin, (16) gives \( v = v_r \) and \( \dot{\phi} = \dot{\phi}_r \) such that \( \dot{\theta} = \dot{\phi} = 0 \). The origin of the error coordinates is thus an equilibrium point for trajectory tracking and path following as well as postural regulation. Also note that the singularity observed in (4) is not present in (15) given the path manifold.

6. Control Law

We now derive a motion controller that will drive the robot to the path manifold, which then steers the error coordinates to...
their origin. As a result the robot is asymptotically driven to an arbitrarily small neighborhood of the desired posture, path, or trajectory. Control parameters are then tuned to assure that physical constraints are satisfied.

6.1. Lyapunov based control design

Lyapunov based techniques are used to derive the foundation of our motion controller, as indicated in Theorem 1, to drive the robot to the path manifold. Corollaries 1 and 2 then prove that the controller drives the robot along the path manifold to the origin of the error coordinates. Unacceptable potential singularities exist, however. Based on Theorem 1 the final velocity controller is thus presented in (34) with slight modifications to remove unacceptable singularities and to introduce the perturbed path manifold to allow a and θ to be well defined at the equilibrium point. Theorem 2 proves convergence to the perturbed path manifold in posture regulation. The equilibrium point of the perturbed path manifold is presented in Theorem 3 and local stability is proven in Corollary 3. Asymptotic stability of the error coordinates is examined in Corollary 4. Finally, regions of attraction are discussed.

The discontinuity at the origin of the polar coordinates allows a time invariant controller to be derived and Brockett’s obstruction does not apply (Astolfi 1994; Aicardi et al. 1995). Due to the singularity, Lyapunov based techniques are valid everywhere except for the origin, but linearization at the origin can be used to verify stability. Thus, for practical purposes a stabilizing control law can be derived using Lyapunov functions (Aicardi et al. 1995; Astolfi 1999).

Similar to sliding mode control techniques, a state feedback control law is derived to steer the system to the path manifold (13) where \( z_1 = 0 \) and \( z_2 = 0 \) by denoting,

\[
\begin{align*}
  z_1 &= e - \sqrt{r} \eta \\
  z_2 &= \theta + \alpha.
\end{align*}
\]

Note that \( z_1 \geq -\sqrt{r} \eta \) since \( e \geq 0 \). Once the robot reaches the path manifold, the path manifold guarantees stabilization of the robot to the origin of the error coordinates.

**Theorem 1.** The following control law provides asymptotic convergence of the states in \( M = \{ (e, \theta, \alpha) \in \mathbb{R}^3 \mid e > 0 \text{ and } \alpha \neq \tan^{-1} \left( \frac{\cos \theta}{\sqrt{2} \sin \theta} \right) \} \) to the path manifold, which drives \( z_1 \) and \( z_2 \) to zero,

\[
\begin{align*}
  \nu &= \frac{k_1 \eta (z_1 + \sqrt{r} \eta) \tan z_1 + v_r \eta (z_1 + \sqrt{r} \eta) \cos \theta}{v + \sqrt{r} \sin 2\theta \left( \sin \theta + \eta (z_1 + \sqrt{r} \eta) \right)} \\
  \dot{\phi} &= \kappa \nu = k_2 \tanh z_2 + 2\dot{\theta} + \dot{\phi}_r.
\end{align*}
\]

**Proof:** First, define quadratic Lyapunov candidate functions,

\[
V = V_1 + V_2: \quad V_1 = \frac{1}{2} z_1^2, \quad V_2 = \frac{1}{2} z_2^2.
\]

Applying (18) and (19), the time derivatives of \( V_1 \) and \( V_2 \) are then negative definite,

\[
\begin{align*}
  \dot{V}_1 &= z_1 \dot{z}_1 = z_1 \left( v_r \cos \theta - v \cos (z_2 - \theta) - \frac{\sqrt{r} \sin 2\theta}{\eta} \dot{\phi} \right) \\
  &= -k_1 z_1 \tanh (z_1) \leq 0 \\
  \dot{V}_2 &= z_2 \dot{z}_2 = z_2 (2\dot{\theta} + \dot{\phi}_r - \dot{\phi}) \\
  &= -k_2 z_2 \tanh (z_2) \leq 0,
\end{align*}
\]

where \( k_1 \) and \( k_2 \) are positive control gains that determine maximum convergence rates. Thus, \( \dot{V} = \dot{V}_1 + \dot{V}_2 \) is negative definite, which proves that the states asymptotically approach the path manifold where \( z_1 = 0 \) and \( z_2 = 0 \).

Note that the proposed control law is nonsingular if the denominator of (18) is nonzero since the numerator is bounded. Applying \( \eta = \sqrt{1 - \cos 2\theta} \) to the denominator of (18) and rewriting in error coordinates, we have,

\[
\text{den}(\nu) = e \cos(\alpha) \sqrt{1 - \cos 2\theta} + r \sqrt{2} \sin (2\theta) \sin(\alpha) \neq 0.
\]

Solving (23) for \( \alpha \), we then have,

\[
\alpha \neq \tan^{-1} \left( \frac{e \sqrt{1 - \cos 2\theta}}{r \sqrt{2} \sin (2\theta)} \right).
\]

Thus, \( \nu \) is nonsingular if (24) is provided. The control law, \( \nu \), is modified subsequently to resolve this singularity issue and initial conditions are further discussed.

Notice that \( \text{tanh} \) functions are implemented to make the Lyapunov function derivatives negative definite and to guarantee smooth bounded control inputs for arbitrarily large initial conditions. This resolves the common problem of excessive velocity demands given large initial conditions. While this is similar in spirit to Webers and Zimmer (2000), their controller only provides posture regulation whereas we provide all three primary motion control tasks.

**Corollary 1.** The control law (18) and (19) stabilizes the closed loop system to the origin of the polar error coordinate system along the path manifold.

**Proof:** Let \( a = (z_3 - 1)\theta \) where \( z_3 \in \mathbb{R}^1 \), we have,

\[
z_2 = z_3 \theta.
\]

Differentiating (25) and substituting into (22), we have,

\[
\dot{V}_2 = z_2 \dot{z}_3 \theta + z_3 \dot{\theta} = \theta^2 z_3 z_3 + z_3^2 \dot{\theta} = -k_2 z_2 \tanh (z_2) \leq 0.
\]
In order to provide a unique equilibrium point, we define \( S = \{ e > 0, \theta \in [-\pi, \pi], a \in [-\pi, \pi], a \neq \tan^{-1}\left(\frac{\sqrt{1 + \cos 2\theta}}{\sqrt{2} \sin(2\theta)}\right)\} \subset M \). Then, define Lyapunov functions to show the stabilization of \( \theta \) to zero in \( S \),

\[
V_3 = \frac{1}{2} z_3^2, \quad V_4 = \frac{1}{2} \theta^2. \tag{27}
\]

First, consider Case (i) where \( \theta \neq 0 \) in \( S \) where \( t \geq 0 \). According to (22)–(26), the variables \( z_3, \dot{z}_3, \) and \( \theta \) are bounded since \( z_2 \) and \( \theta \) are bounded. Note that \( z_2 \) asymptotically converges to zero by Theorem 1 for any \( \theta \) and \( a \) for \( t \geq 0 \), and by (25) \( z_3 = 0 \) if and only if \( z_2 = 0 \) for all \( \theta \neq 0 \). In order for (26) to be true for any non-zero \( \theta, z_2, \) and \( z_3 \), it must be true that \( V_3 = z_3 \dot{z}_3 < 0 \) and \( V_4 = \theta \dot{\theta} < 0 \). Further, since \( z_2 \to 0 \) as \( t \to \infty \), it is easily verified that \( V_3 \to 0 \) and \( V_4 \to 0^+ \) such that \( \theta \) and \( z_3 \) asymptotically approach zero.

Now consider two other cases where \( \theta = 0 \). In Case (ii) where \( z_3 \) is bounded (i.e. \( |z_3| < \infty \)), it must be true from (25) that \( z_2 = 0 \) and boundedness of \( \dot{z}_3 \) is shown by (26). As a result, we have \( \dot{z}_3^2 \dot{\theta} = -k_2 z_2 \tanh(z_2) \), which can be rewritten as,

\[
\dot{V}_4 = \dot{\theta} = \frac{-k_2 z_2 \tanh(z_2)}{z_3} = \frac{-k_2 \dot{z}_2^2 \tanh(z_2)}{z_2}. \tag{28}
\]

Applying \( z_2 = 0 \) and \( \theta = 0 \) to (28), by L’Hôpital’s rule, we have,

\[
\dot{V}_4 \bigg|_{\theta=0} = \left. -k_2 \theta^2 \right|_{\theta=0} \cdot \lim_{z_2 \to 0} \frac{\tanh(z_2)}{z_2} = -k_2 \theta^2 |_{\theta=0} = 0. \tag{29}
\]

Further, in Case (iii) consider \( \theta = 0 \) and unbounded \( z_3 \) (i.e. \( z_3 = \pm \infty \)), it is verified that \( \dot{z}_3^2 \dot{\theta} \) must be bounded to satisfy (26). As a result, in conjunction with Case (ii), \( \dot{V}_4 = \theta \dot{\theta} = 0 \) must always be true when \( \theta = 0 \).

Summarizing the aforementioned cases, it is shown that \( \dot{V}_4 < 0 \) for \( \theta \neq 0 \) and \( \dot{V}_4 = 0 \) for \( \theta = 0 \). This proves that \( \theta \) asymptotically converges to zero as \( t \to \infty \). Further, since \( z_1 \to 0 \) and \( z_2 \to 0 \) as \( t \to \infty \) (by Theorem 1), we thus have \( e \to \sqrt{2} r \eta \to 0 \) and \( a \to -\theta \to 0 \) as \( t \to \infty \) per (17).

The closed-loop system, (4) with (18) and (19), is then analyzed near the origin based upon linearization in Corollary 2.

**Corollary 2.** The origin of the closed loop system is locally exponentially stable by the control law (18) and (19).

**Proof:** The controller (19) drives \( \dot{\alpha} \to -\theta \) by Theorem 1 such that the system (4) becomes

\[
\begin{align*}
\dot{e} & = (\dot{v} + v_r) \cos \theta \\
\dot{\theta} & = \dot{\alpha} = -(\dot{v} + v_r) \frac{\sin \theta}{e} - \dot{\phi}_r.
\end{align*} \tag{30}
\]

The following must be true near the equilibrium point per (30),

\[
v \to v_r, \quad e \to \frac{2v_r}{\phi_r} \sin \theta. \tag{31}
\]

Applying (31) to the control law (18) and (19), and linearizing near the origin, we have,

\[
\begin{align*}
v & = k_1 e + v_r \\
\dot{\phi} & = k_2 (\theta + a) + 2k_1 a + \dot{\phi}_r. \tag{32}
\end{align*}
\]

Further, applying (32) to (4) and linearizing, we then have,

\[
\begin{align*}
\dot{e} & = -k_1 e \\
\dot{\theta} & = k_1 a = -k_1 \theta \\
\dot{\alpha} & = -k_2 \theta - (k_1 + k_2) a = -k_1 a,
\end{align*} \tag{33}
\]

which proves that the control law provides local exponential stability if \( k_1 > 0 \).

While the above shows theoretically that ideal convergence is achieved, it must be noted that \( a \) and \( \theta \) are arbitrarily defined at \( e = 0 \). To assure that they are well defined at the equilibrium of the system, we modify the path manifold by introducing an arbitrarily small \( \varepsilon > 0 \),

\[
\eta = \sqrt{1 + \varepsilon - \cos 2\theta}, \tag{34}
\]

which produces a perturbation on the controller (18) that will be discussed in the following Theorems and Corollaries.

Further, note that a singularity occurs at \([a, \theta] = [\pi/2, 0]\) in the control law (18). Thus, we also modify (18) to resolve this problem by eliminating \( \cos(2z_2 - \theta) = \cos(a) \) in the denominator of (18), which leads to,

\[
\dot{V}_4 = z_1 \dot{z}_1 = -k_1 z_1 \tanh z_1 + z_1 v(1 - \cos a). \tag{35}
\]

Note that (35) is identical to (21) as \( a \) converges to zero. The final control law, \( v \), is then,

\[
\begin{pmatrix}
k_1 (z_1 + \sqrt{2} \eta) \tanh z_1 \\
+ v_r (z_1 + \sqrt{2} \eta) \cos \theta \\
+ v_r \sqrt{2} \sin 2\theta \left( \sin \theta + \kappa_r (z_1 + \sqrt{2} \eta) \right)
\end{pmatrix}
\]

\[
v = \frac{e}{\eta(z_1 + \sqrt{2} \eta) + r \sqrt{2} \sin 2\theta \sin(z_2 - \theta)}. \tag{36}
\]

These modifications provide an arbitrarily small perturbation, which is analyzed next. The controller (18) and (36) produce almost similar outputs and they become identical for small \( a \) and as the robot approaches the path manifold. Now, we analyze how these modifications affect the closed loop system. We first consider posture regulation since closed form analysis can be shown.
Theorem 2. The control law (19) and (36) converges the states defined in $D = \{(e, \theta, \alpha) \in \mathbb{R}^3 \mid e \geq \sqrt{2}r \eta \text{ (i.e., } z_1 \geq 0), |\theta| \leq \pi, |\alpha| \leq \pi\}$ to a perturbed path manifold in posture regulation.

Proof: The perturbed path manifold is defined by introducing $\varepsilon > 0$ in (34),

$$e = r \sqrt{2} \sin \theta + e - \cos \theta, \quad \theta = -\alpha.$$  \hfill (37)

Applying arbitrarily small $\varepsilon$, the perturbed path manifold (37) is arbitrarily close to the path manifold (13). It is important to note that $z_2 = \theta + \alpha$ is not affected by the modified path manifold. Thus, considering the same Lyapunov function for $V_2$ in (13), the control law (19) satisfies (22), which maintains asymptotic convergence of $z_2$ to zero as shown in Theorem 1. We now provide a state equation for $z_1$ to discuss convergence to the perturbed path manifold. By differentiating $z_1 = e - \sqrt{2} \eta r$, we have,

$$\dot{z}_1 = -k_1 \tanh(z_1) + f(v, \alpha), \quad (38)$$

where $f = f(v, \alpha) = (1 - \cos \alpha)v$ and $0 < f \leq 2v_{\text{max}}$ given $0 < v \leq v_{\text{max}}$. Note that nonvanishing $f$ perturbs the equilibrium point of $z_1$ to a finite value that depends on the relative magnitude of $-k_1 \tanh(z_1)$ in (38), which is discussed further in the next theorem.

Considering posture regulation ($v_r = 0, \dot{\phi}_r = v_r \gamma_r = 0$), (36) can be simplified,

$$v = \frac{k_1 \eta(z_1 + \sqrt{2}r \eta) \tanh(z_1)}{\eta(z_1 + \sqrt{2} \eta) + r \sqrt{2} \sin \theta \sin(z_2 - \theta)}.$$  \hfill (39)

Note that $e = z_1 + \sqrt{2}r \eta > 0$ in $M$. In order to show convergence of $z_1$, it is important to note that $z_1$ and $\theta$ are coupled such that their convergence must be considered simultaneously. By applying (39) to (38) and the $\theta$ state equation in (4), and noting $z_2 \rightarrow 0$, we have,

$$\dot{z}_1 = -\tanh(z_1) \cos \theta F \quad (40)$$

$$\dot{\theta} = -\sin \theta \tanh(z_1) G,$$  \hfill (41)

where

$$F = \frac{k_1 (\eta z_1 + \varepsilon \sqrt{2})}{\eta z_1 + \varepsilon \sqrt{2} + \sqrt{2} \sqrt{r} (1 - \cos \theta) \sin^2 \theta}$$

$$G = \frac{k_1 \eta}{\eta z_1 + \varepsilon \sqrt{2} + \sqrt{2} \sqrt{r} (1 - \cos \theta) \sin^2 \theta}. \quad (42)$$

It should be noted that the controller is applied in $D$ where $z_1 \geq 0$, or $e \geq \sqrt{2}r \eta$, to satisfy the curvature constraint, $|\alpha| \leq \kappa_{\text{max}}$, which excludes the inside of the path manifold. In particular, asymptotic convergence can guarantee that the robot stays in $D$ given proper initial conditions (i.e. $z_1(0) \geq 0$). Further, since $e > 0$ and $\eta > 0$, we can easily verify that the denominator in (39)–(41) is always positive for $z_1 \geq 0$ such that we have $v \geq 0$, $F > 0$, and $G > 0$, which is useful in Lyapunov analysis and guarantees only forward motion.

Finally, we can easily show that the derivative of $V_1$ is negative definite for $z_1 > 0$ and zero for $z_1 = 0$ by using the Lyapunov function $V_4$ in (27), which shows that,

$$\dot{V}_4 = \theta \dot{\theta} = -\tanh(z_1) (\theta \sin \theta) G,$$  \hfill (43)

indicating that $\dot{\theta}$ can only decrease while $z_1$ is finite in $D$. At the same time, the derivative of $V_1$ is negative definite for $|\theta| < \pi / 2$ and zero for $z_1 = 0$,

$$\dot{V}_1 = z_1 \dot{z}_1 = -\cos \theta (z_1 \tanh(z_1) F,$$  \hfill (44)

which indicates that $z_1$ asymptotically converges to zero if $|\theta| < \pi / 2$. Further, (43) indicates that $\dot{\theta}$ will decay such that $|\theta| < \pi / 2$ can easily be achieved given $z_1 > 0$.

Thus, in a worst case situation ($\pi / 2 < |\theta(0)| < \pi$), (44) and (43) indicate that $z_1$ increases while $\dot{\theta}$ simultaneously decreases. As soon as $|\theta|$ becomes less than $\pi / 2$, $z_1$ and $\dot{\theta}$ both asymptotically decrease until $z_1 = 0$ is achieved. Thus, convergence to the perturbed path manifold is proven in posture regulation.

In the general case where $v_r \neq 0$, the closed loop state equations for $z_1$ and $\theta$ are difficult to analyze analytically. Thus, we first numerically establish a unique equilibrium point to show the effect of perturbations on the system. We then use linearization to show local convergence to the path manifold near the equilibrium point.

Theorem 3. Using (19) and (36) an equilibrium point $(z_1^*, \theta^*)$ is created in $D$ that can be made arbitrarily close to the origin of the error coordinates by tuning $\varepsilon$.

Proof: Again, note that the controller (19) converges $z_2$ to zero by Theorem 1. We can thus rewrite (36) by,

$$v = \frac{k_1 \eta(z_1 + \sqrt{2} \eta) \tanh(z_1 + v_r \gamma)}{h}, \quad (45)$$

where

$$h = \eta z_1 + \varepsilon \sqrt{2} + 2 \sqrt{2} \sqrt{r} (1 - \cos \theta) \sin^2 \theta$$

$$g = \eta(z_1 + \sqrt{2} \eta) \cos \theta + r \sqrt{2} \sin \theta \left( \sin \theta + \kappa_r(z_1 + \sqrt{2} \eta) \right). \quad (46)$$

Expressing $e$ in terms of $z_1$ by applying the coordinate transformation $e = z_1 + \sqrt{2} \eta r$ to (4) and (38), and assuming $z_2$ has already converged to 0, the state equations for $z_1$ and $\theta$ then become,
Table 2. Numerical solution of the equilibrium point for \((z_1, \theta)\) with \(k_1 = 1\) and Reference (A) \([v_r \text{ (m/s)}, \kappa_r \text{ (m\(^{-1}\})}] = [0.5 \text{ v}_{\text{max}}, 0.33 \kappa_{\text{max}}]\), (B) \([0.75 \text{ v}_{\text{max}}, 0.75 \kappa_{\text{max}}]\), and (C) \([0, 0.33 \kappa_{\text{max}}]\) where \(v_{\text{max}} = 0.5, \kappa_{\text{max}} = 3\).

<table>
<thead>
<tr>
<th>Ref.</th>
<th>(\epsilon)</th>
<th>(z_1^*) (m)</th>
<th>(\theta^*) (rad)</th>
<th>(\lambda_1)</th>
<th>(\lambda_2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A)</td>
<td>(1 \times 10^{-1})</td>
<td>(7.91 \times 10^{-4})</td>
<td>-0.0796</td>
<td>0.996</td>
<td>3.14</td>
</tr>
<tr>
<td>(A)</td>
<td>(1 \times 10^{-6})</td>
<td>(7.78 \times 10^{-9})</td>
<td>-2.50 \times 10^{-4}</td>
<td>1</td>
<td>999.9</td>
</tr>
<tr>
<td>(A)</td>
<td>(1 \times 10^{-12})</td>
<td>0</td>
<td>-9.16 \times 10^{-11}</td>
<td>1</td>
<td>1.06 \times 10^6</td>
</tr>
<tr>
<td>(B)</td>
<td>(1 \times 10^{-1})</td>
<td>0.0173</td>
<td>-0.305</td>
<td>0.892</td>
<td>2.81</td>
</tr>
<tr>
<td>(B)</td>
<td>(1 \times 10^{-6})</td>
<td>(1.21 \times 10^{-7})</td>
<td>-8.02 \times 10^{-4}</td>
<td>1</td>
<td>1.05 \times 10^3</td>
</tr>
<tr>
<td>(B)</td>
<td>(1 \times 10^{-12})</td>
<td>(1.42 \times 10^{-15})</td>
<td>8.72 \times 10^{-8}</td>
<td>1</td>
<td>1.75 \times 10^6</td>
</tr>
<tr>
<td>(C)</td>
<td>(1 \times 10^{-1})</td>
<td>0</td>
<td>-0.0791</td>
<td>0.996</td>
<td>0</td>
</tr>
<tr>
<td>(C)</td>
<td>(1 \times 10^{-6})</td>
<td>0</td>
<td>-2.50 \times 10^{-4}</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(C)</td>
<td>(1 \times 10^{-12})</td>
<td>0</td>
<td>-5.82 \times 10^{-10}</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

\[
\begin{align*}
\dot{z}_1 &= -k_1 \tanh(z_1) + (1 - \cos \theta)v_r \\
&= -\tanh(z_1) \cos(\theta) F + v_r (1 - \cos \theta) \frac{g}{h} \\
\dot{\theta} &= -\dot{\phi} = -(v + v_r) \frac{\sin \theta}{z_1 + \sqrt{2r \eta}} - \dot{\phi}_r \\
&= -\sin(\theta) \tanh(z_1) G \\
&\quad - v_r \left( \frac{g}{h} + 1 \right) \frac{\sin \theta}{z_1 + \sqrt{2r \eta}} - \dot{\phi}_r. 
\end{align*}
\] (47)

Note that an equilibrium point cannot be solved analytically due to nonlinearity in the closed system (47) and (45). These state equations are thus solved numerically to find the equilibrium point \((z_1^*, \theta^*)\) as shown in Table 2. Three references are shown to illustrate the affect of reference and \(\epsilon\). Reference (A) and (B) are used to demonstrate the general case where \(v_r \neq 0\). It should be noted that \(z_1^*\) remains in a small neighborhood of the origin in spite of large variation of reference velocities and \(\epsilon\). Reference (C) illustrates that posture regulation produces \(z_1^*\) remains in a small neighborhood of the origin in spite of large variation of reference velocities and \(\epsilon\), which is discussed in Sec. VI.C.

Corollary 3. The perturbed equilibrium point is locally exponentially stable for \(v_r \neq 0\), and asymptotically stable for \(v_r = 0\).

Proof: Local stability can be determined by linearizing (47) at the equilibrium point such that,
\[
\dot{z}_1 \approx -\lambda_1 (z_1 - z_1^*), \quad \dot{\theta} \approx -\lambda_2 (\theta - \theta^*),
\] (48)
where eigenvalues, \(-\lambda_1\) and \(-\lambda_2\), are,
\[
\lambda_1 = \lim_{\theta \rightarrow \theta^*} \left( \cos(\theta) F \right) \\
\lambda_2 = \lim_{\theta \rightarrow \theta^*} \left( \tanh(z_1) G + v_r \left( \frac{g}{h} + 1 \right) \frac{1}{z_1 + \sqrt{2r \eta}} \right). 
\] (49)

It is important to note that \(\lambda_1\) is always positive, and that \(\lambda_2\) is positive for \(v_r \neq 0\), which proves local exponential stability for path following and trajectory tracking. Also, note that \(\lambda_1\) converges to \(k_1\) and \(\lambda_2\) increases proportional to \(1/\eta\) as \(\epsilon\) decreases for \(v_r \neq 0\), which are both illustrated in Table 2 for \(k_1 = 1\).

For \(v_r = 0\), we have \(z_1^* = 0\) as indicated in Table 2 such that \(\lambda_2 = 0\) by (49), which is insufficient to show asymptotic stability. However, Theorem 2 proves via Lyapunov analysis that \(\theta\) asymptotically decreases until \(z_1 = 0\). Since \(z_1 = 0\) is a unique equilibrium point for (40) and (41), \(\theta\) is then in equilibrium on the path manifold, which proves local asymptotic stability in posture regulation. As a result, the states converge to the perturbed path manifold where \(z_1 = 0, z_2 = 0,\) and \(\theta = \theta^*\).

Given convergence to the perturbed path manifold, we now estimate the resulting equilibrium point \((\epsilon^*, \theta^*, \alpha^*)\). Corollary 1 is then extended to show asymptotic stability of the new equilibrium point.
Corollary 4. The system is asymptotically stabilized to a new equilibrium point \((e^*, \theta^*, \alpha^*)\) that can be made arbitrarily close to the origin by applying the control law (19) and (36).

**Proof:** As shown in Theorems 2 and 3, and Corollary 3, the equilibrium point of the closed-loop system, \((e^*, \theta^*, \alpha^*)\), is on the perturbed path manifold (37),

\[
e^* = r \sqrt{2}\sqrt{1 + e - \cos(2\theta^*)}, \quad \theta^* = -\alpha^*,
\]

which avoids the singularity issue at the origin. Using (31), we have,

\[
e^* = \frac{2v_r}{\phi_r} \sin \theta^*.
\]

Solving (50) and (51) simultaneously and linearizing, we then have,

\[
e^* = \frac{r \sqrt{2\alpha}}{\sqrt{1 - (rK_r)^2}}, \quad \alpha^* = -\theta^* = \frac{r \sqrt{2\alpha}}{\sqrt{2(1 - (rK_r)^2)}},
\]

where \(|rK_r| < 1\) since \(r \geq 1/k_{\text{max}}\) and \(|K_r| < k_{\text{max}}\). This result shows that the equilibrium point of the closed-loop system is perturbed into an arbitrarily small neighborhood of the origin given arbitrarily small \(e\).

Let \(\bar{\theta} = \theta - \theta^*\) and \(\bar{\alpha} = \alpha - \alpha^*\) to prove asymptotic convergence of the states to \((e^*, \theta^*, \alpha^*)\). We then have \(z_2 = \theta + \alpha = (\theta + \alpha) - (\theta^* + \alpha^*) = \bar{\theta} + \bar{\alpha}\) and \(\bar{\bar{z}}_2 = (\bar{\bar{z}}_2 - 1) \bar{\theta}\). Thus, Corollary 1 is applied to prove that and asymptotically converge to zero (i.e., \(\bar{\theta} \rightarrow 0\) and \(\bar{\alpha} \rightarrow 0\)). We then have \(e \rightarrow e^*\) per (17) and (34) since \(z_1 \rightarrow 0\). Finally, these results prove that the controller asymptotically stabilizes the system to the equilibrium point arbitrarily close to the origin without singularity. \(\blacksquare\)

In path following and trajectory tracking, the reference must be smooth and satisfy the physical limitations of the robot (i.e., \(v_r < v_{\text{max}}\) and \(k_r < k_{\text{max}}\)) such that the robot has sufficient authority to compensate for error. For a slower moving reference the robot has greater ability to catch up to the reference and path curvature constraints are discussed in following sub sections.

6.2. Dependence on Initial Conditions

Due to fundamental path geometry constraints, allowable initial conditions must be considered to assure that curvature
bounds are satisfied during convergence to the path manifold. Initial conditions are divided into three zones, Figure 7, based upon the path manifold using \( r = 1/\kappa_{\text{max}} = 0.34 \text{ m} \).

Dashed lines in Figure 7 based on (54) provides a locus where singularity will occur for specific values of \( \alpha \). The locus was generated by plotting \( e \) as a function of \( \theta \) where \( e = 1 \times 10^{-6} \). Note that \( e \) is largest at \( |\alpha| = \pi/2 \) and when \( \theta \approx 0 \) or \( \pm \pi \). Further, \( e \) decreases and goes to zero as the magnitude of \( \alpha \) does the same and when \( \theta \) approaches \( \pm \pi/2 \). There is also a very small region at \( \theta = 0 \) or \( \pm \pi \) where \( e = 0 \) due to \( e \) in the denominator of (54). It is important to note that these loci are contained well inside of Zone 2, where initial conditions are known to possibly cause difficulty. Most importantly, though, the controller (19) quickly forces \( \alpha = -\theta \) (i.e., \( z_2 = 0 \)), which causes the loci to contract entirely into Zone 3 as shown in Figure 7 where they are not a threat. Thus, this singularity issue is essentially resolved by the controller driving the system to the path manifold.

6.3. Boundedness by Design of \( k_1 \) and \( k_2 \)

Control gains \( k_1 \) and \( k_2 \) must be designed to assure boundedness of \( v \) and \( \phi \) during convergence to the path manifold. Once the robot reaches the path manifold (13), the control law (36) and (19) then converges to the velocities (16) such that both curvature and velocity are bounded. It is assumed in this analysis that allowable initial conditions have been specified.

Implementing a fixed \( k_1 \) in the controller, it is observed in the workspace analysis that the maximum velocity increases as \( e \) decreases. This phenomenon is not desirable considering physical limitations of performing tight steering maneuvers at higher velocities. Furthermore, since the velocity and curvature expressions are highly nonlinear, simple closed form expressions of \( k_1 \) and \( k_2 \) for bounded control inputs are not easily found.

Optimization techniques based upon worst case analyses are used to determine \( k_1 \) and \( k_2 \) in order to provide bounded curvature and velocities. Workspace mapping is thus conducted to find worst cases where maximum velocity and curvature commands are produced, respectively, given \( e \) applying constant \( k_1 \) and \( k_2 \). Worst cases for \( k_1 \) and \( k_2 \) are observed at \( (\theta, \alpha) \approx (0.2557, -1.6047) \) and \( \theta(0) = \alpha(0) = \pi \), where respective maximum velocity is commanded and large orientation correction is required. Table 3 shows a selected optimized set of \( k_1 \) and the posture of the robot when maximum velocity occurs as a function of \( e \). These optimization results illustrate that \( k_1 \) lies in approximately \( 0.2 = k_1 \leq 0.5 \) and is proportional to the inverse of \( e \). Thus, \( k_1 \) is determined as a function of states,

\[
k_1 = (k_{1\text{max}} - k_{1\text{min}}) \left(1 - \tanh \left( \frac{g_1}{e} \right) \right) + k_{1\text{min}},
\]

where \( g_1 = 1.3, k_{1\text{max}} = 0.5, \) and \( k_{1\text{min}} = 0.2 \) are selected such that \( k_1 \) correlates to the optimized results in Table 3. The
value of $k_{1\text{min}}$ determines the minimum convergence rate of $e$ and $g_1$ establishes a boundary beyond which $k_{1\text{max}}$ will dominate.

Gain $k_2$ is a parameter governing the angular velocity and curvature commands. Similar to $k_1$, the parameter $k_2$ is optimized based upon worst case conditions, Table 4. These results show that $k_2$ is significantly dependent upon $e$. An expression for $k_2$ is then determined as a function of states, $e$ and $\alpha$, and initial error distance, $e(0)$, in order to match this data and provide bounded curvature such that,

$$k_2 = 0.3 \tanh\left(\frac{1}{e(0)}\right) \tan\left(\frac{1}{2 |\alpha|}\right) + 0.3 \tanh\left(\frac{1}{e}\right). \quad (56)$$

These gains improve boundedness as shown in Figure 8(b) and (d) compared to the cases where constant gains are applied, Figure 8(a) and (c). Also note that $k_1$ and $k_2$ are always positive, and the previously discussed stability proof still applies.

### 6.4. Dynamic Extension

Initial values of $v$ and $\kappa$ specified by (36) and (19) rarely match those of the robot. These problems may be resolved by extending the controller dynamics in a cascade fashion as per Bacciotti Theorem 19.2 (Bacciotti 1991) by introducing new states,

$$\dot{v} = -k_v (v - v_D) + \dot{v}_D; \quad k_v > 0, \quad k_c > 0$$

$$\dot{\kappa} = -k_c (\kappa - \kappa_D) + \dot{\kappa}_D. \quad (57)$$

where $v$ and $\kappa$ are the extended velocity and curvature states used to command the robot, and $v_D$ and $\kappa_D$ are the desired control velocity and curvature established by (36) and (19). Since the dynamic extensions add additional servo-loops to the original system (4), the eigenvalues $k_v$ and $k_c$ of the dynamic extension should be faster than those of the original system. Since $k_1$ and $k_2$ are both small (recall $k_1 \leq 0.5$ and $k_2 \leq 0.3$), we simply use $k_v = k_c = 1$.

### 7. Controller Evaluation

#### 7.1. Methods and Procedures

The proposed controller is evaluated in simulations and experiments. Simulations are used to validate the controller’s capability to satisfy curvature and velocity constraints from a wide variety of initial postures. Posture regulation, path following, and trajectory tracking are evaluated in simulation. Zones of valid initial conditions are more precisely defined via workspace sweeps.

Experimental results are shown to demonstrate application of the controller to an actual robot considering physical constraints. Both posture regulation and path following are considered in experiment. A high traction carpet surface is used to illustrate performance under ideal circumstances. Both simulation and experimental results are based upon the ideal unicycle kinematic model (4). In experiments the controller (19) and (36) combined with (57) is applied to the two-axle CFMMR, Figure 9, using matlab with Real-Time Workshop.
and a dSPACE™ 1103 DSP tethered to the robot. The complex
kinematics of the CFMMR, Figure 10, are reduced to (4) in
unicycle equivalent coordinates $O$ by applying curvature based
steering where
\[
\frac{1}{r} = \frac{2\psi}{L \cos \psi},
\]
given the frame length, $L$, and curvature command, $\kappa$. The
linear and angular velocities of each axle, $v_i$ and $\phi_i$, and
wheel angular velocities, $\dot{q}_{i,j}$, can then be found using $v$ and $\phi$
of the center posture, $O$,
\[
v_i = \frac{v}{\cos \psi} + \frac{(-1)^i}{6} L \psi \frac{\sin \psi}{\psi}
\]
\[
\dot{\phi}_i = \dot{\phi} + (-1)^{i-1} \frac{\psi}{\psi}
\]
\[
\dot{q}_{i,j} = \frac{v_i + (-1)^{i-1} \dot{\phi}_i d}{r_w}
\]
where
\begin{align*}
  i &= 1 \quad \text{for front axle} \nonumber \\
  i &= 2 \quad \text{for rear axle} \nonumber \\
  j &= 1 \quad \text{for right wheel} \nonumber \\
  j &= 2 \quad \text{for left wheel} \nonumber
\end{align*}

Since this paper focuses on ideal kinematic motion control algorithms that can produce references for the dynamic controller (Zhu et al. 2005, Zhu et al. 2007), robot dynamics and disturbances are not considered here. In this paper, traditional servo-type wheel controllers based upon filtered wheel encoder odometry are used to drive the robot. In a worst case scenario, wheel odometry is fed directly to the kinematic controller (Zhu et al. 2005; Zhu et al. 2007) instead of using ideal kinematic models. Final robot positions are obtained relative to grid work suspended above the robot to illustrate actual performance independently of wheel odometry. Measurements are taken with a tape measure and the estimated accuracy of these measurements is ±1 mm. These experiments ultimately demonstrate that the kinematic controller is also robust to disturbances and can also be used in traditional servo loop configurations.

Since reference path or trajectory generation is not the focus of this research, we implement typical simple paths with constant curvature (Shkel and Lumelsky 1997). In path following, the reference path is
\[
x_r = v_{des} \tanh(0.1/e), \quad \kappa_r = \kappa_{des}
\]
where $v_{des}$ and $\kappa_{des}$ represent desired path segment velocity and curvature, respectively. In trajectory tracking, the reference trajectory is
\[
\dot{x}_r = v_{des} \cos(\phi_r(t)), \quad \dot{y}_r = v_{des} \sin(\phi_r(t)),
\]
and $\dot{\phi}_r(t) = \kappa_{des} v_{des}$, where $x_r(0) = y_r(0) = 0$ m.

7.2. Results and Discussion

Figure 11 shows the posture regulation response of the controller given different initial orientations with the same initial error distance of $e(0) = 2$ m. Paths, states, velocities, and curvatures of Point $O$ are shown in Figure 11(a)–(d), respectively. These results confirm that the paths have smooth bounded curvature and require only forward motion as designed. In the worst case, curvature is 2.9 m⁻¹ and velocity is 0.37 m/s,
which is less than $\kappa_{\text{max}} = 3 \text{ m}^{-1}$ and $v_{\text{max}} = 0.5 \text{ m/s}$, respectively. For symmetric positive and negative initial orientations, curvature profiles and robot paths are symmetric with respect to the $x$-axis whereas velocity profiles are identical. Further, the error states approach a small neighborhood of the origin in finite time and converge to the perturbed equilibrium point asymptotically. Note that in all cases $z_1$ and $z_2$ converge faster than the error states as designed, Figure 11(b)–(c). This verifies that the robot approaches the perturbed path manifold first and then error states are stabilized along the path manifold, which is proven in the earlier theorems and corollaries of Section 6.1.

Table 5. Simulation results of posture regulation: $\theta(0) = a(0) = \pi$.

<table>
<thead>
<tr>
<th>$e$ (m)</th>
<th>Runtime, $t_{\text{final}}$ (s)</th>
<th>$\max(v)$ (m/s)</th>
<th>$\max(\kappa)$ (m$^{-1}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>50</td>
<td>0.3735</td>
<td>1.5218</td>
</tr>
<tr>
<td>5</td>
<td>100</td>
<td>0.4811</td>
<td>0.7987</td>
</tr>
<tr>
<td>20</td>
<td>290</td>
<td>0.4960</td>
<td>0.2040</td>
</tr>
<tr>
<td>200</td>
<td>2700</td>
<td>0.4996</td>
<td>0.02</td>
</tr>
</tbody>
</table>

Path following demonstrates similar smooth paths and convergence of error states. Boundedness of path curvature and velocity are demonstrated in Figure 12 for a circular reference path with a 1 m radius. Velocity and curvature profiles vary according to their initial orientation angles similar to posture regulation. In both posture regulation and path following, larger velocity and path curvature are commanded for large initial orientation errors. Likewise, maximum velocity is detected at $a(0) = -\pi$ while maximum curvature is observed at $a(0) = -3\pi/4$.

Figure 13 illustrates that velocities and curvatures are bounded even with the most awkward initial posture of $a(0) = \pi$; regardless of the magnitude of $e(0)$. Variations in curvature are considerable, though, as $e(0)$ increases. Thus, Figure 13 is non-dimensionalized based upon data shown in Table 5. As designed, the maximum velocity approaches 0.5 m/sec with larger $e(0)$ and is maintained for a longer period. For smaller $e(0)$, the maximum velocity decreases $\sim 25\%$ and is maintained for a shorter period, which is desirable for such tight maneuvers.

The non-dimensional curvatures, Figure 13(b), are nearly identical until dimensionless time, $t/t_{\text{final}} = 0.3$, after which they approach different steady state values according to the radii of their approaching paths. This phenomenon is caused by the controller directing the path to asymptotically converge...
to the path manifold. For larger $e(0)$, maximum curvature decreases and resulting paths become longer and require more space. As a result, for large $e(0)$ it would be pertinent to use the controller to track paths optimized for a particular workspace. The point here is that regardless of initial conditions, bounds on velocity and curvature are well established even in the most awkward initial conditions.

Table 6 presents final postures in posture regulation as a function of $e$ for two different initial positions to compare numerical simulation and analytic estimates. These results verify several important controller properties: (i) the proposed control law (19) and (36) asymptotically converge $z_1$ and $z_2$ to zero independently of $e$ as indicated in Theorem 3; (ii) the equilibrium point $(e^*, \theta^*, \alpha^*)$ changes as a function of $e$; and (iii) simulated final postures are identical to the equilibrium point (52) estimated by solving the closed loop system in Corollary 4. Based upon analytical, numerical, and simulation results shown in Table 2 and Table 6, $e = 1 \times 10^{-6}$ is sufficiently small for the CFMMR such that this value is implemented in experiment.

The long and circuitous paths that result from awkward initial conditions can easily be eliminated by allowing backward motion. As indicated in Section 3, backward motion is generated by using coordinate transformation in Quadrants 1 and 4 (e.g. when $x(0) > 0$). A workspace sweep with forward and backward motion indicates the allowable initial conditions, Zones 1 and 2, as discussed in Section 6.2. Due to symmetry, only the zones for $y \geq 0$ are shown. Forward motion results are shown in Figure 14(a)–(b) whereas Figure 14(c)–(d) indicates results when backward motion is allowed.

In both forward and backward motion, results indicate that Zone 1 is actually much larger than initially estimated. Initial conditions again play a role in determining how large these zones are. If the robot is always initially pointed towards the origin where $\phi(0) = \theta(0)$, allowable initial conditions increase since Zone 2 is significantly reduced as illustrated in Figure 14(b) and (d). If backward motion is allowed, Zone 2 is almost eliminated and Zone 1 is nearly maximized, Fig.
Fig. 15. Allowable initial conditions in simulated trajectory tracking with $\phi(0) = 0$ rad, $v_{\text{max}} = 0.5$ m/s, and $\kappa_{\text{max}} = 3$ m$^{-1}$.

Thus, the actual range of allowable initial conditions is much larger than initially estimated in Figure 7. Due to the fact that the controller asymptotically converges to the path manifold, any initial conditions in Zone 3 will violate curvature constraints. If the robot starts within Zone 3, it is necessary to pick an intermediate goal point in Zone 1.

Allowable initial conditions that satisfy velocity and curvature constraints are more complex when considering trajectory tracking with a moving reference frame. A workspace mapping is used to illustrate how initial conditions are affected by a reference, Figure 15. Linear and circular references were used with the same initial heading angle, $\phi(0) = 0$ rad. These results verify that for a slower moving reference with smaller path curvature, the robot has greater ability to catch up to the reference and the range of initial conditions allowing convergence is greater as discussed in Section 6.1. Note that a reasonable neighborhood of the origin (where the trajectories start) and positions behind the target are allowable initial conditions, which is sufficient for most existing tracking problems.

Fig. 16. Trajectory tracking for a linear reference path with $v_{\text{des}} = 0.1$ m/s using well defined initial conditions $[e(0), \theta(0), \alpha(0)]$, $Q_1 = [\sim 0, 0, 0]$, $Q_2 = [1 \text{ m}, -\pi/4 \text{ rad}, -\pi/4 \text{ rad}]$, $Q_3 = [2 \text{ m}, -\pi/4 \text{ rad}, -\pi/4 \text{ rad}]$.
Fig. 17. Trajectory tracking for a linear reference path with $v_{des} = 0.1 \text{ m/sec}$ with awkward initial conditions at fixed orientations $(\theta(0) = -3\pi/4 \text{ rad}, \alpha(0) = \pi/2 \text{ rad})$: P1 ($e(0) = 1 \text{ m}$), P2 ($e(0) = 3 \text{ m}$), P3 ($e(0) = 5 \text{ m}$).

Fig. 18. Trajectory tracking by modifying the reference trajectory with the same initial conditions (P1, P2, P3) used in Figure 17.

Fig. 19. Experimental results based upon wheel odometry with initial $(e, \theta, \alpha) = (2, -\pi/4, -\pi/4)$.

To resolve initial condition problems found in trajectory tracking, reference trajectories may be designed to move slowly until the robot approaches a small neighborhood of the reference. As a result, initial trajectories during this approach phase become similar to those of posture regulation where initial conditions in Zone 1 are required to assure bounded control inputs. The approach phase can be included in the trajectory tracking algorithm by simply modifying the reference trajectory to

$$\dot{x}_r = v_{des} \tanh(t/30) \cos(\phi(t)), \quad \dot{y}_r = v_{des} \tanh(t/30) \sin(\phi(t)).$$

Thus, this trajectory tracking algorithm provides bounded control inputs in P1, P2 and P3 as shown in Figure 18. Since we have already discussed how to avoid the initial conditions outside of Zone 1, initial condition problems can be resolved for the primary motion control tasks.
Experimental results in Figure 19 were obtained by using wheel odometry on a high traction carpeted surface and correspond well to simulation results. Table 7 shows final posture errors on carpet and sand, which was obtained by using wheel odometry and actual measurements. Velocity and curvature are bounded as designed. As expected, better performance is also obtained on higher traction surface (carpet) compared to lower traction surface (sand) according to actual measurements. Whereas odometry data provide consistently small error, actual final position errors increase on lower traction surface. These results confirm that larger wheel slip occurs on the lower traction sand surface such that larger odometry errors cause larger actual errors, Figure 20. Further, Tests 2 and 3 were conducted by using the proposed controller whereas Tests 4* and 5* were obtained by using a different kinematic controller with the same initial conditions as reported in Minor et al. (2006). These results show that the path manifold controller decreases actual distance errors, \( e \), by \(~28\%\) on carpet, and \(~44\%\) on sand, respectively. The improved error is attributed to reduced wheel slip and traction forces. Thus, the proposed controller provides better performance and efficient motion control compared to our past research. Further, the controller is robust near the origin whereas previous research (Tayebi and Rachid 1997; Albiston and Minor 2003) needed to switch controllers to guarantee bounded commands near the origin.

It is also important to note that all error states in experiments converge to zero asymptotically, which means that the robot converges to the origin or tracks the reference along smooth path trajectories. These results verify that the algorithm performs efficiently for both posture regulation and path following, especially considering that the dynamics of the robot were ignored. For posture regulation and path following, the linear velocity and the path curvature are well bounded by \( v_{\text{max}} \) and \( \kappa_{\text{max}} \), respectively, although more noise is apparent due to wheel backlash and servo-loop dynamics. Ultimately, this kinematic controller can be used to provide an ideal reference input to a robust dynamic controller to reduce errors in experiment (Zhu et al. 2005). Most importantly, though, it must be noted that the proposed kinematic motion controller

<table>
<thead>
<tr>
<th>No.</th>
<th>Surface</th>
<th>Wheel Odometry</th>
<th>Actual measurement</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>( e \pm \sigma_e )</td>
<td>( \theta \pm \sigma_{\theta} )</td>
</tr>
<tr>
<td>1</td>
<td>Simulation</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>Carpet</td>
<td>2.4 \pm 0.7</td>
<td>0.4 \pm 0.2</td>
</tr>
<tr>
<td>3</td>
<td>Sand</td>
<td>1.0 \pm 0.6</td>
<td>-2.7 \pm 4.0</td>
</tr>
<tr>
<td>4*</td>
<td>Carpet</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>5*</td>
<td>Sand</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

Fig. 20. Sequential snap shots of experiment for posture regulation on sand, initial \((e, \theta, \alpha) = (1.9, 40^\circ, 40^\circ)\).
provides curvature and velocity commands satisfying physical constraints even with non-ideal sensor systems and basic servo type dynamic controllers.

Even though the control algorithms presented here respect physical constraints for a wide variety of initial conditions while performing posture regulation, path following, and trajectory tracking, several practical issues must be addressed in future work. Given large $\epsilon(0)$, the resulting paths will require large regions to maneuver the robot to its target. In practical application, these spaces may not be available and/or obstacles will likely be present. Thus, future work should focus on planning reference paths and trajectories considering obstacles and physical constraints. Constraints should include nonholonomy, curvature, and velocity. The controller presented here is ideal for providing realistic smooth and bounded robot commands in order to track these references.

8. Conclusions

Simulation and experimental results prove viability of the proposed path manifold based controller to accommodate physical constraints. Simulation results prove that velocity and curvature commands are bounded as specified by physical constraints. Posture regulation, path following, and trajectory tracking have been demonstrated. Workspace mappings confirm zones of allowable initial conditions that allow asymptotic convergence to the path manifold and target without violating physical constraints. Methods of accommodating non-ideal initial conditions have also been discussed. Experimental results demonstrate that the established physical limitations and their implementation via the path manifold based controller provide improved performance relative to previous work.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable suggestions and comments that have helped us to improve this paper. This material is based upon work supported by the National Science Foundation under Grant No. IIS-0308056.

References


