Motion control of nonholonomic robots at low speed

Ľubica Miková, Alexander Gmiterko, Michal Kelemen, Ivan Virgala, Erik Prada, Darina Hroncová and Martin Varga

Abstract
Many applications in robotics require precise tracking of the prescribed path. The aim of this article is to develop and verify by computer simulation a control design method which ensures that the “output” of the robot will move along a prescribed path. A virtual vehicle approach algorithm was used to track a predefined vehicle path. The idea behind this algorithm is that the movement of a virtual vehicle on a predefined path is controlled by a differential equation whose input is a control deviation representing the distance between a real and a virtual vehicle. The main advantage of the path-following approach is that, based on this approach, the feedback realized is invariant to the path.

Keywords
Nonholonomic robot, algorithm, simulation

Date received: 31 October 2019; accepted: 5 January 2020

Introduction
Advances in remotely controlled robots have made their applications possible in a variety of useful areas. This is due to the fact that in the last 20 years, there has been significant activity in the field of designing control algorithms for nonlinear dynamic systems, including different classes of mechanical systems, which also represent remotely controlled nonholonomic mobile robots.

Particularly extensive literature is devoted to the problems of movement with nonholonomic constraints with application to kinematic models of mobile robots. On the other hand, there are few works dealing with control in the presence of parametric uncertainty in the kinematic models of the robot.

Traditionally, the tasks of the robots are specified by the temporal parameterization of the path in the workspace to be monitored by the robot at each time point. However, in many applications, the path time parameter is not important compared to coordination and synchronization requirements between different degrees of freedom. Shape tracking role, for example, for a robot to draw an image in a Cartesian system, belongs to this category. In the traditional approach based on time parameterization of the shape, the desired shape is parameterized as a function of time $Q(t)$. The current trajectory is $q(t)$. The goal of the trajectory tracking controller is to achieve a control deviation $\bar{Q}(t) - g(t)$ to converge to zero if $t \to \infty$. As stated by Chiu1 and also in other sources, the control deviation $\bar{Q}(t) - g(t)$ does not indicate well how the desired path shape is observed.

The basic problems of controlling mobile autonomous robots can be divided into three groups2.
• point stabilization—the aim is to stabilize the vehicle in the desired configuration;
• trajectory tracking—the aim is for the vehicle to track the parameterized setpoints over time;
• path following, maneuver regulation—the goal is for the vehicle to track and converge at the same time.

It is important to understand the difference between path following and trajectory tracking. Anguiar et al. outlined the basic difference between path following and trajectory tracking.

In the tracking problem, the main task of the controller is to follow the path if, a priori, its time parameterization is not known. In the problem of trajectory tracking, the role of the controller is to follow the path with a known time parameterization. Thus, trajectory tracking can be considered a special case of path following. One obvious advantage of path following is that there are cases, where the trajectory tracking problem is unsolvable, and there is a solution for path following. However, the main advantage of the path-following approach is that, based on this approach, the feedback realized is invariant to the path. This means that if a mobile robot is initialized on a track with a suitable orientation, track independence guarantees that it will never leave the track. The trajectory tracking controller cannot ensure the track’s invariance and thus cannot guarantee that the mobile robot will never leave the track.

Path following can even be used if the system has unstable zero dynamics. In the case of path following for a linear system with zero dynamics, better control quality can be achieved compared to trajectory tracking.

A common approach to the path-following problem is to parameterize the geometric path \( y_d \) by a path variable \( \theta \) and then select a timing law for \( \theta \).

In a path-following problem, the basic assumption is that the desired forward speed of the vehicle follows the desired speed profile, while the controller determines the direction of the vehicle speed so that the vehicle approaches the track. This control approach eliminates the trajectory tracking controller’s aggressiveness by forcing the vehicle to converge smoothly (in a mathematical sense as a smooth function) to the track. In the case of trajectory tracking, both the position and the speed of the vehicle along the desired path are controlled. Compared to trajectory tracking, path following has a smoother convergence to track and action signals are less pushed to saturation.

There are many examples of mechanical and robotic systems in which it is advisable to use controllers for path following as compared to trajectory track controllers. Examples include:

• walking robots,
• rehabilitation and exercise machines,
• teleoperators,
• bypassing obstacles,
• robotic manipulators,
• human–robot interaction.

Precise path following is useful for robots when welding, cutting, and drawing in indoor and outdoor navigation. If in these processes the controller is invariant on the track, this means that if the tool is set on the track, it will always remain on it even if it faces rough locations that can cause the tool to slow down.

The problem of path following deals with the design of a control algorithm that controls a controlled object (robot arm, mobile robot, ship, aircraft, etc.) to achieve following of the geometric track. The general approach to the path-following problem is based on the parameterization of the geometric path \( y_d \) by the path variable \( s \).

The intuitive interpretation of the tracking problem is as follows:

• to determine the distance of the vehicle to the desired track,
• to determine the angle between the vehicle speed vector and the path tangent, and
• to reduce both data to zero without considering the time specification.

This is done by controlling the forward speed so that it converges to the setpoint (constant in our case) and by acting on the orientation of the vehicle to get on the track. Pioneering works in this area can be considered.

Feedback control strategies are important for mobile robots to compensate for fault variables and errors in initial conditions. Path following, trajectory tracking, and stabilization around constant configuration problems are treated as separate problems for nonholonomic mobile robots. This state is due to the lack of smooth static feedback state control algorithms.

A virtual vehicle approach algorithm was used to track a predefined vehicle path.

An intuitive interpretation of this algorithm is that the robot is controlled to move to a point on a predefined path in which the virtual vehicle is currently located.

The idea of this algorithm is that the movement of a virtual vehicle on a predefined path is controlled by a differential equation, whose input is a control deviation representing the distance between a real and a virtual vehicle.

**Motion control of nonholonomic mobile robots at low speed**

Wheeled mobile robots constitute a class of mechanical systems characterized by kinematic constraints that are not integrable and cannot, therefore, be eliminated from the model equations. Automatic control of nonholonomic robots has attracted the attention of the scientific and research community since the early 1980s. Already at the outset of this activity, Brockett defined the necessary
conditions for such systems, concluding that these systems are controllable in the case of an open control system but cannot be stabilized using smooth time-invariant feedback. A number of time-varying control feedback algorithms have been developed to stabilize such systems.

The chosen path-following algorithm was tested for a four-wheel drive, where the two rear wheels were driving and the two front wheels were steerable. It is assumed that the conditions of nonautonomy are met.

The kinematic model of a four-wheeled robot was described as follows:

\[
\begin{align*}
\dot{x} &= v \cos(\psi + \theta) \\
\dot{y} &= v \sin(\psi + \theta) \\
\dot{\psi} &= v / L \sin(\theta)
\end{align*}
\]

where \(\psi\) is the orientation angle of the robot relative to the \(x\)-axis (Figure 1) and \(v\) is the forward speed of the robot, \(L\) is the distance between the axes of rotation of the front and rear wheels.

The robot was tested to follow a path in the form of a semicircle with a radius \(r = 5\) m. The initial configuration of the virtual vehicle is given in polar coordinates, namely the radius of the circle \(r\) and the angle \(\alpha\). Control objectives are set by the constant \(d_\rho\). This is the maximum permitted distance between the robot and the virtual vehicle, and according to the above theory, for \(t \rightarrow \infty\), the following applies

\[
\lim_{t \to \infty} \rho(t) \leq d_\rho
\]

The last entered data at this stage are the constants of the controller that executes its own control algorithm. In our case, we used a proportional control algorithm according to equation (3)

\[
\theta = k_p (\psi_d - \psi)
\]

where \(\theta\) is an action variable and the difference \((\psi_d - \psi)\) represents the control deviation.

The execution phase operates iteratively in a loop that ends when the virtual vehicle reaches the end of the predefined path. The first operation in this phase is to determine the actual distance between the robot and the virtual vehicle according to the following relation

\[
\rho = \sqrt{(X - X_v)^2 + (Y - Y_v)^2}
\]

where \(X, Y\) and \(X_v, Y_v\) are Cartesian coordinates of the robot and the virtual vehicle, respectively. Furthermore, the so-called smoothness parameter \(k_s\) is used to prevent the virtual vehicle speed from being changed by a jump and to determine the current reference angle \(\psi_d\) according to the relation

\[
\psi_d = \arctan \left( \frac{Y_v - Y}{X_v - X} \right)
\]

For a four-wheeled robot, the new configuration is determined by equation (6)

\[
\begin{align*}
X(t) &= X(t-1) + v \cos[\psi(t-1) + \theta(t-1)] \Delta t \\
Y(t) &= Y(t-1) + v \sin[\psi(t-1) + \theta(t-1)] \Delta t \\
\psi(t) &= \psi(t-1) + v / t \sin[\theta(t-1)] \Delta t
\end{align*}
\]

For a virtual vehicle, the new configuration is determined as follows:

\[
\begin{align*}
\alpha(t-1) &= \arctan \left( \frac{Y_v(t-1)}{X_v(t-1)} \right) \\
\alpha(t) &= \alpha(t-1) + \frac{v_r(t-1)}{r} \Delta t \\
X_v &= r \cos[\alpha(t)] \\
Y_v &= r \sin[\alpha(t)]
\end{align*}
\]

The structure of the steering system is shown in Figure 2, and the given algorithm for following the robot path is shown in a clear form in the form of a flowchart in Figure 3.

**Results of computer simulations**

The objective is to obtain:

- recording of the path traveled by the robot from the initial configuration compared to the predefined path for the different starting positions of the robot;
- distance dependence between the robot and the virtual vehicle for different initial positions of the robot in the workspace depending on the number of iterations;
- obtaining the dependencies of the control deviation represented by the difference of the current desired orientation of the angle \(\psi_d\) and the current orientation of the angle \(\psi\) depending on the number of iterations; and
obtaining dependencies of the action variable $\theta$ for a four-wheeled robot depending on the number of iterations.

**Simulation 1**

The objective of the simulation is to get information on how the path difference between the robot and the virtual vehicle $\rho$ in a four-wheel robot converges to the predefined value $d_\rho$.

The parameters are set as follows:

- the radius of the predefined arc of the semicircle: $r = 5$ m,
- initial virtual vehicle configuration: $r = 5$ m, $\alpha = -60^\circ$,
- distance between axes of the front and the rear wheels: $L = 0.3$ m.

Initial robot configurations:

- $X = 0$ m, $Y = 4$ m, $\psi = 0$ rad
- $X = 0$ m, $Y = 0$ m, $\psi = 0$ rad
- $X = 0$ m, $Y = -4$ m, $\psi = 0$ rad
- $X = -2$ m, $Y = -2$ m, $\psi = 0$ rad
- $X = 2$ m, $Y = -7$ m, $\psi = 0$ rad
- $X = 5$ m, $Y = -7$ m, $\psi = 0$ rad

Parameter of smoothness: $k_s = 1$

Constant of P-controller: $k_p = 0.4$

Objective of control: $d_\rho = 0.2$ m

Step of numeral integration: 0.04

**Results and discussion**

The graphical outputs obtained by the simulation are shown in Figures 4 to 6. The results obtained confirm the theoretical assumptions that the distance between the robot and the virtual vehicle converges to the predefined value $d_\rho = 0.2$ m (right top Figures 4 to 6). This is consistent with relation (2) presented in the theoretical formulation of algorithm for following of the predefined path. At the end of the followed path, the value $d_\rho$ was in all cases equal to $d_\rho = 0.203$ m. From the figures, it is clearly seen that the distance between the robot and the virtual vehicle converges to the
predefined value $d_p$. In addition, it will be noted that the control deviation $\psi_d - \psi$ shown in section (d) of Figures 4 to 6 also converges with increasing iteration. This fact satisfies the assumption

$$\lim_{t \to \infty} |\psi_d - \psi| \leq d_\psi$$

presented in the theoretical description of the path-following algorithm. When using the P-controller, the action variable is a...
copy of the control deviation $\psi - \psi'$ to scale, as shown in section (c) of Figures 4 to 6. Section (a) of Figures 4 to 6 visualizes how the robot follows (red curve) the predefined path (blue curve). As can be seen, the robot reaches a predefined path from different starting positions and stops at a distance of 0.203 m behind the virtual vehicle at the end of the orbit.
Simulation 2

The objective of the simulation is to find out the influence of parametric uncertainty in following the path of a four-wheeled robot, if the monitored parameter is the length of the robot between the axes of the front and rear wheel $L$.

Figure 6. Configuration $5_{-7.0}$ and $2_{-7.0}$.
Figure 7. Length $L = 0.20 \text{ m}$ and length $L = 1.0 \text{ m}$. 
Figure 8. Robot speed $v = 1.20 \text{ m/s}$ and $v = 0.90 \text{ m/s}$.
The parameters are set as follows:
The parameters except length $L$ had the same value as in simulation 1. Only length $L$ was changed, which was set as follows:

- $L = 0.2 \text{ m}$
- $L = 1.0 \text{ m}$

**Results and discussion**
The simulation found that if the length $L$ varies from $0.2 \text{ m}$ to $1.0 \text{ m}$, the convergence rate to value $\rho = 0.2 \text{ m}$ changes minimally, namely:

- for $L = 0.2 \text{ m}$, its value at the end of the distance traveled is $0.202 \text{ m}$,
- for $L = 1.0 \text{ m}$, it is $0.212 \text{ m}$.

As the iteration increases, the control deviation $\psi_a - \psi$ also converges. These results confirm the theoretical claim that the path-following algorithm under investigation is robust (Figure 7).

**Simulation 3**
The objective of the simulation is to determine the effect of parametric uncertainty in following the four-wheeled robot path on the distance value $\rho$ at the end of the traveled path, if the monitored parameter is the robot speed $v$.

The parameters are set as follows:
The parameters, except for the robot speed, had the same value as in simulation 1. Only the robot speed $v$, which was set as follows, changed:

- $v = 1.20 \text{ m/s}$
- $v = 0.90 \text{ m/s}$
- $v = 0.75 \text{ m/s}$

**Results and discussion**
The graphical outputs of these four simulations for the indicated robot speeds are shown in Figures 8 and 9. From the graph section (b) of Figures 8 and 9, it can be seen that the change in robot speed compared to the nominal speed $v = 1.0 \text{ m/s}$ has an impact on the $\rho$ value read at the end of the path. The smallest deviation was achieved for speed $v = 1.2 \text{ m/s}$ and the largest for speed $v = 0.75 \text{ m/s}$.

Keeping the speed of the robot is therefore important for the path-following algorithm under investigation. If the robot speed exceeded $1.20 \text{ m/s}$, the robot behaviour was unstable.
Conclusions

Three simulations were performed and their results presented. The first simulation focused on detecting convergence to a predefined path in a four-wheel robot. The second and third simulations concerned the determination of the effect of parametric uncertainty in the path following by a four-wheel robot. The monitored parameter was the length of the robot between the front and rear wheel axes and detecting the distance between the robot and the virtual vehicle at the end of the distance path if the uncertainty parameter is the speed of the robot.

The chosen control algorithm for path following enables tracking of a predetermined track by a four-wheeled robot, where the rear wheels are driving and the front wheels are steerable. The control algorithm converges to a predefined distance value between the robot and the virtual vehicle. This is in accordance with the relationship that results from the theoretical analysis of the algorithm. Path following by the robot is possible from any position in the robot workspace.

The control deviation of the robot angular orientation also converges to the desired angle orientation with increasing iteration, which is also in accordance with the conditions imposed on this algorithm.

From the simulations performed, it was found that the distance between the robot and the virtual vehicle after traveling a predetermined distance changes very little with the change in the distance between the axes of the front and rear wheels of the four-wheeled robot. In particular, the simulation found that if the length \( L \) varies between 0.2 m and 1.0 m, the convergence rate to \( d_\rho = 0.2 \) m changes minimally, namely:

- for \( L = 0.2 \) m, it is its value at the end of traveled path is 0.202 m;
- for \( L = 1.0 \) m, it is 0.212 m.

From the simulations performed, it was found that if the specified speed of the four-wheeled robot is not observed during path following, this has a significant effect on the distance between the robot and the virtual vehicle \( \rho \). In particular, the smallest deviation was achieved for speed \( v = 1.2 \) m/s and the largest for speed \( v = 0.75 \) m/s. Keeping the speed of the robot is therefore important for the path-following algorithm under investigation. If the robot speed exceeded 1.20 m/s, the robot behavior became unstable.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research was funded by Slovak Grant Agency VEGA 1/0389/18 “Research on kinematically redundant mechanisms.”

ORCID iD

Ľubica Miková  ⓒ https://orcid.org/0000-0003-2864-0651
Ivan Virgala  ⓒ https://orcid.org/0000-0001-8273-2384

References

1. Chiu TC. Coordination control of multiple axes mechanical systems: theory and experiments. PhD Dissertation, Department of Mechanical Engineering, University of California, Berkeley, 1994.
2. El-Hawwary MI and Maggiore M. Global path following for the unicycle and other results. In: American control conference, Seattle, DC, USA, 11–13 June, 2008, pp. 3500–3505.
3. Anguiar AP, Hespanha JP, and Kokotović PV. Performance limitations in reference tracking and path following for nonlinear systems. Automatica 2008; 44(3): 598–610.
4. Nielsen C. Maneuver regulation, transverse feedback linearization and zero dynamics. Master’s Thesis, Department of Electrical and Computer Engineering, University of Toronto, Canada, 2004.
5. Hauser J and Hindman R. Maneuver regulation from trajectory tracking: feedback linearizable systems. In: Proc. IFAC Symp. Nonlinear Control Systems Design, Lake Tahoe, CA, USA, 25-28 June, 1995, pp. 595–600.
6. Al-Hiddabi SA and McClamroch NH. Tracking and maneuver regulation control for nonlinear nonminimum phase systems: application to flight control. IEEE Trans Contr Syst Tech 2005; 10(6): 780–792.
7. Aguiar AP and Hespanha JP. Logic-based switching control for trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty. In: Proceedings of the 2004 American control conference, Boston, MA, USA, 30 June-2 July, 2004.
8. Aguiar AP, Dačić DB, Hespanha JP, et al. Path-following or reference-tracking? An answer relaxing the limits to performance. In: Proceedings of IAV2004 -5th IFAC/EURON symposium on intelligent autonomous vehicles, Lisbon, Portugal, 5–7 July, 2004.
9. Aguiar AP, Hespanha JP, and Kokotović P. Path-following for non-minimum phase systems removes performance limitations. IEEE Trans Automat Contr 2005; 50(2): 234–239.
10. Han DP, WEI Q, and Li ZX. Path following of mobile robots using a virtual vehicle approach. In: Proceedings of the 25th Chinese control conference, Harbin, Heilongjiang, 7–11 August, 2006, pp. 1533–1537.
11. Egerstedt M, Hu X, and Stotsky A. Control of a mobile platforms using a virtual vehicle approach. IEEE Trans Automat Contr 2001; 46(11): 1777–1782.
12. Brockett RW. Asymptotic stability and feedback stabilization. In: Millmann RS (ed) Differential geometric control theory. Boston: Birkhauser, 1983, pp. 181–191.
13. Guivant J and Nebot E. Optimization of the simultaneous localization and map building algorithm for real time implementation. IEEE Trans Robot Automat 2001; 17(3): 242–257.
14. Guy C, Georges B, and Brigitte D. Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. *IEEE Trans Robot Automat* 1992; 12(1): 47–62.
15. Freund E and Mayr R. Nonlinear path control in automated vehicle guidance. *IEEE Trans Robot Automat* 1997; 13: 49–60.
16. Fredslund J and Matarić M. A general algorithm for robot formations using local sensing and minimal communications. *IEEE Trans Robot Automat* 2002; 18: 837–846.
17. Graf B, Wandosell J, and Schaeffer C. Flexible path planning for nonholonomic mobile robots. In: *Fourth European workshop on advanced mobile robots*, Lund, Sweden, 19–21 September 2001, pp. 199–205.
18. Kumar U and Sukavanam N. Dynamic modeling and tracking control of a four wheeled nonholonomic mobile robot. In: *13th national conference on mechanics and machines*, Bangalore, India, 12–13 December, 2007.
19. Petrov P. Modeling and adaptive path control of a differential drive mobile robot. In: *Proceedings of the 12th WSEAS international conference on automatic control, modelling and simulation* (eds V Niola, J Quartieri and F Neri), Stevens Point, WI, USA, 29–31 May 2010, pp. 403–408. Belize: WSEAS Ltd.
20. Tang CH. Differential flatness-based kinematic and dynamic control of a differentially driven wheeled mobile robot. In: *IEEE international conference on robotics and biomimetics*, Guilin, China, 19–23 December 2009, pp. 2267–2272.
21. Zhihua Q, Wang J, and Clinton E. A new analytical solution to mobile robot trajectory generation in the presence of moving obstacles. *IEEE Trans Robot* 2004; 20(6): 978–993.
22. Tsai PS, Wang LS, and Chang FR. Trajectory tracking control of tri-wheeled mobile robots in skew chained form system. In: *Proceeding of CACS automatic control conference St. John’s university*, Tamsui, Taiwan, 10–11 November 2006, pp. 197–202. New York: St. John’s University.
23. Turygin Y, Bozek P, Nikitin Y, et al. Enhancing the reliability of mobile robots control process via reverse validation. *Int J Adv Robot Syst* 2016; 13(6). DOI: 10.1177/1729881416680521.
24. Pivarčiová E, Božek P, Turygin Y, et al. Analysis of control and correction options of mobile robot trajectory by an inertial navigation system. *Int J Adv Robot Syst*; 15(1). DOI: 10.1177/1729881418755165.
25. Sapietová A, Sága M, Kuric I, et al. Application of optimization algorithms for robot systems designing. *Int J Adv Robot Syst* 2018; 15(1). DOI: 10.1177/1729881417754152.