*Science & Technology Development Journal 2022, 25(2):2444-2452*

Open Access Full Text Article

*Research article*

# Bézier trajectory tracking control of the omnidirectional mobile robot based on a linear time-varying state feedback controller

**Trịnh Thị Khánh Ly[1,*], Hoàng Thiên[2]**

Use your smartphone to scan this QR code and download this article

**ABSTRACT**

**Introduction**: The controller design method based on linear time varying state feedback controller is proposed to enforce an omnidirectional mobile robots (OMRs) to track a given Bézier trajectory. **Methods**: Initially, the linear error model of the robot system is obtained and used in the design of the linear time varying state feedback controller. Next, the controller gains are determined according to the linear and angular velocities for the desired trajectory to providing small errors. To verify the effectiveness of the proposed control strategy, simulation is carried out with desired Bézier trajectory for the OMR to reduce sudden speed changes. **Results**: The simulation results are presented, which have verified the good performance of the proposed controller for tracking the complex motion trajectories. **Conclusion**: Therefore, it is possible to apply this result to control the OMRS in logistics applications of the modern manufacturing systems.

**Key words:** omnidirectional mobile robot, kinematic error model, linear time-varying state feedback controller, trajectory tracking, Bézier trajectory

## INTRODUCTION

The mecanum wheel was invented by Ilon in 1973 at Mecanum Company and applied to the first AGV design in 1997 [1]. Today, the mecanum wheel has been used for the OMR design that movesmove in tight spaces [2,3]. A mobile robot with omnidirectional capabilities is very attractive because it guarantees perfect mobility in tight areas, avoids obstacles and moves instantaneously in any direction from any configuration [4,5]. With those advantages, the OMR has been developed in smart logistics systems [6,7] and the modern industrial environment [8,9]. The AGVs with lifting mechanismmechanisms in industrial production plants [10,11]. Unmanned autonomous vehicles are logistics for producing super-heavy components, such as manufacturing wind turbine blades [12], in aircraft production lines [13] etc. Figure 1 below illustrates one of the above applications.

Thereby, it can see that the application potential of this type of robot in the future is enormous. Therefore, this type of robot is now the object of many domestic and foreign researchers to improve and expand its applicability to many different realistic scenarios. In the techniques related to the mobile robot, the problem of tracking motion control is one of the important problems to ensure that the robot moves according to the desired scenario [14,15]. That is confirmed through many studies on this issue, such as improving robot control performance is one of the main foci in the field of the OMR [16–22]. etc. Especially, a high-precision controller which guarantees the OMR tracking the desired trajectory accurately is required [23].

However, trajectory tracking control of OMR is difficult because it is a complex nonlinear, time-varying system [2]. For simplicity, a traditional linear controller based on the linearization model of the robot can be proposed [24,25]. However, traditional linear controllers are not suitable because the control system is not guaranteed to be stable [2,23]. To deal with this control problem, some researchers have used nonlinear control strategies for OMR such as: fuzzy control [26,27], sliding mode control [28,29], and backstepping control [30]. The above methods are all based on the complex nonlinear model of the robot, leading the complex controller structure requires the processor and the memory capacity of the hardware is large.

To overcome the above problem, this paper proposes to design a linear time varying state feedback controller based on the linear error model for OMR. Also, to avoid sudden changes in the robot velocities, we use the Bézier curve to design the simulation problem path to verify the efficiency of the controller designed by this study with the assumption that there is no longitudinal and transverse slip of the OMR during the movement.

## KINEMATIC ERROR MODEL OF THE OMNIDIRECTIONAL MOBILE ROBOT

[1]*Electric Power University (EPU)*

[2]*Hanoi University of Science and Technology (HUST)*

**Correspondence**

**Trịnh Thị Khánh Ly**, Electric Power University (EPU).

Email: lyttk@epu.edu.vn

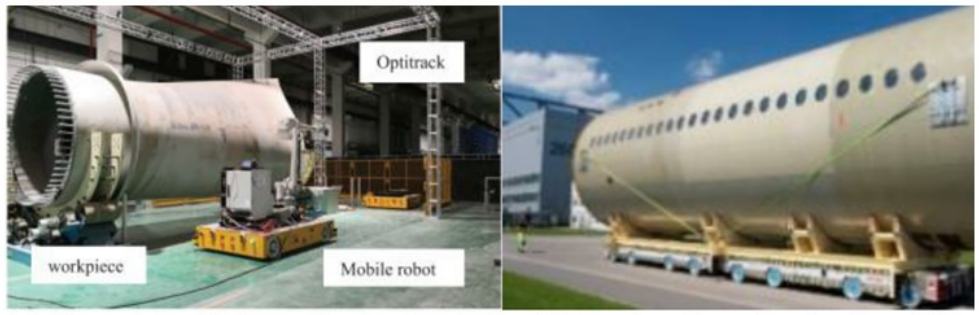
**History**

- Received: 2022-03-20
- Accepted: 2022-06-06
- Published: 2022-06-30

**DOI : 10.32508/stdj.v25i2.3914**



**Cite this article :** Ly T T K, Thiên H. **Bézier trajectory tracking control of the omnidirectional mobile robot based on a linear time-varying state feedback controller**. *Sci. Tech. Dev. J.* 2022; 25(2):2444-2452.




(a) The turbine blade production line [12] (b) The aircraft production lines [13]

(c) The turbine blade production line [11] (d) The AGV pallet trucks [10]

**Figure 1**: Some applications of the OMR [10–13]

## Kinematic model

Consider a differential drive mobile robot moving along the trajectory $\xi$ with pure rolling and without lateral slip derived in the global coordinate system $\vartheta_f\{O_f x_f y_f z_f\}$, as shown in Figure 2. The relationship between the wheels' angular speeds and the linear and angular velocities of the robot are determined by[4]:

$$\omega = J\dot{q} \tag{1}$$

where $\omega(t) = \begin{bmatrix}\omega_1(t) & \omega_2(t) & \omega_3(t) & \omega_4(t)\end{bmatrix}^T$;

$$J = \begin{bmatrix}1/r & 1/r & -(L+d)/2r \\ 1/r & -1/r & (L+d)/2r \\ 1/r & 1/r & (L+d)/2r \\ 1/r & -1/r & -(L+d)/2r\end{bmatrix};$$

$\dot{q}(t) = \begin{bmatrix}\dot{x}(t) & \dot{y}(t) & \dot{\varphi}(t)\end{bmatrix}^T = [V_{xr} \;\; V_{yr} \;\; \Omega_r]^T$.

Here, r, L and d are the radius of the wheel, the distance in the $x_r$-axis and the $y_r$-axis of two wheels, while $V_{xr}$, $V_{yr}$ and $\Omega_r$ are the $x_r$-axis, $y_r$-axis velocities and angular velocity of the robot, respectively.

Thus, the mathematical model of the robot kinematics in $\vartheta_f\{O_f x_f y_f z_f\}$ is given by:

$$\dot{q}(t) = Q(\varphi)\dot{q}_r(t) \tag{2}$$

wherein

$$Q(\varphi) = \begin{bmatrix}\cos\varphi(t) & -\sin\varphi(t) & 0 \\ \sin\varphi(t) & \cos\varphi(t) & 0 \\ 0 & 0 & 1\end{bmatrix};$$

$\dot{q}_r(t) = \begin{bmatrix}\dot{x}_r(t) & \dot{y}_r(t) & \dot{\varphi}_r(t)\end{bmatrix}^T$.

## Kinematic error model

The kinematic error model describes the variation in position and and orientation of the robot while following a desired trajectory, defined by the error vector **e**[23]:

$$e = q_d(t) - q(t) = \begin{bmatrix}e_x & e_y & e_\varphi\end{bmatrix}^T \tag{3}$$

where $q_d(t) = [x_d(t) \;\; y_d(t) \;\; \varphi_d(t)]^T$ is the desired trajectory of the OMR in the global coordinate system $\vartheta_f\{O_f x_f y_f z_f\}$.

With $\vartheta_r\{G x_r y_r\}$ being the coordinate system associated with the OMR, transform e(t) from $\vartheta_f$ to $\vartheta_r$ of the OMR:

$$e_r = \begin{bmatrix}e_{xr} & e_{yr} & e_{\varphi r}\end{bmatrix}^T = Q^T e \tag{4}$$

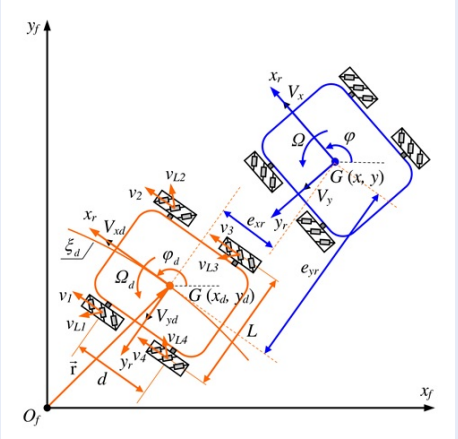



**Figure 2**: Schematic to determine the error of the robot in the mobile coordinate system

Derivative of equation (4), combining equation (2) and equation (3), we have:

$$\dot{e}_r = \begin{bmatrix} V_{xdr}(t)\cos e_\varphi - V_{ydr}(t)\sin e_\varphi - V_{xr}(t) + \Omega_r(t)\, e_{yr} \\ V_{xdr}(t)\sin e_\varphi - V_{ydr}(t)\cos e_\varphi - V_{yr}(t) + \Omega_r(t)\, e_{xr} \\ \Omega_{dr}(t) - \Omega_r(t) \end{bmatrix} \quad (5)$$

Rewrite equation (5) as a matrix form:

$$\dot{e}_r = \widetilde{\Omega}_r e_r + Q(e_\varphi)\,\dot{q}_{dr} - \dot{q}_r \quad (6)$$

wherein $\widetilde{\Omega}_r = \begin{bmatrix} 0 & \Omega_r(t) & 0 \\ -\Omega_r(t) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

Equation (6) is a nonlinear error model of the OMR. To control the robot, we linearize equation (6) around the working point with the desired values. One finally obtains $\varphi_r(t) \approx \varphi_{dr}(t)$ and $\Omega_r(t) \approx \Omega_{dr}(t)$, and one finally obtains the linear error model of the OMR:

$$\begin{cases} \cos e_\varphi \approx 1 \\ \sin e_\varphi \approx e_\varphi \end{cases} \quad (7)$$

Substituting equation (7) into equation (6), we have the linear error model of the robot:

$$\dot{e}_r = Be_r + I_{3\times 3}u \quad (8)$$

where $B = \begin{bmatrix} 0 & \Omega_{dr}(t) & -V_{ydr}(t) \\ -\Omega_{dr}(t) & 0 & V_{ydr}(t) \\ 0 & 0 & 0 \end{bmatrix}$ and $u = [V_{xdr} \;\; -V_{xr} \;\; V_{ydr} \;\; -V_{yr} \;\; \Omega_{dr} \;\; -\Omega_r]^T$ is the input vector.

## SIMULATION SETUP AND DETERMINE CONTROLLER COEFFICIENTS

### Design of the moving trajectory of the robot by the Bézier curve

According to [31], the Bézier curve is a special case of the NURBS curve defined by trajectory interpolation points and given by the equation:

$$P(t) = \sum_{i=0}^{n} B_i J_{n,i}(t),\ 0 \le t \le 1 \quad (10)$$

where $B_i$ is the coordinates of interpolated points, i is the ordinal number of the interpolation points, and $J_{n,i}(t)$ is a Bernstein polynomial of degree n and given by the formula:

$$J_{n,i}(t) = \binom{n}{i} t^i (1-t)^{n-i} \quad (11)$$

With Pascal coefficient $\binom{n}{i} = \frac{n!}{i!(n-i)!}$, n is the degree of the Bernstein polynomial and the degree of the curve.

The Model (8) is controllable by the matrix [**I**, **AI**] with a maximum rank of 3. Therefore, selecting the state feedback control law with three state variables $x(t)$, $y(t)$, $j(t)$ and three control inputs $u_1$, $u_2$, $u_3$ given by [23,32]:

$$u = -Ke_r \quad (9)$$

where $K = \begin{bmatrix} K_1 & 0 & 0 \\ 0 & K_2 & 0 \\ 0 & 0 & K_3 \end{bmatrix}$ is the parameter matrix of the linear state feedback controller. Figure 3 shows the schematic diagram of the closed-loop system composed of the state space controller, the desired trajectory and the OMR.

From the above Bézier curve design method with interpolation points given by Table 1, we have the desired Bézier curve trajectory shown in Figure 4.

### The OMR dimensions setup

Length x Widthwidth x Heightheight (380 mm x 260mm 260 mm x 165 mm), $L$ = 316 (mm), $d$ = 270 (mm), radius of wheels $r$ = 30 (mm).

### Determine the linear and angular velocities of the robot

The desired velocity $V_d(t)$ of the robot is determined to the Bézier motion trajectory $x$ as follows:

$$V_d(t) = \frac{\triangle S}{\triangle t} = \frac{\sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}}{t_i - t_{i-1}} \quad (12)$$

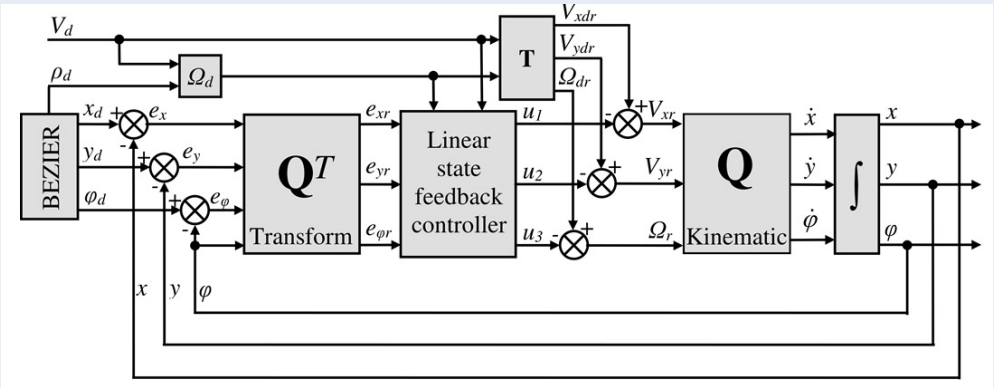



**Figure 3**: Bézier trajectory tracking control for the OMR

**Table 1: Interpolation points data of the Bézier curve trajectory**

| | | | | |
|---|---|---|---|---|
| $A_1$ | $A_2$ | $A_3$ | $A_4$ | $A_5$ |
| $\begin{bmatrix}0\\0\end{bmatrix}$ | $\begin{bmatrix}6.5\\0\end{bmatrix}$ | $\begin{bmatrix}6.5\\5.0\end{bmatrix}$ | $\begin{bmatrix}3.0\\5.0\end{bmatrix}$ | $\begin{bmatrix}3.0\\8.0\end{bmatrix}$ |
| $A_6$ | $A_7$ | $A_8$ | $A_9$ | $A_{10}$ |
| $\begin{bmatrix}6.5\\8.0\end{bmatrix}$ | $\begin{bmatrix}6.5\\13.0\end{bmatrix}$ | $\begin{bmatrix}-6.5\\13.0\end{bmatrix}$ | $\begin{bmatrix}-6.5\\8.0\end{bmatrix}$ | $\begin{bmatrix}-3.0\\8.0\end{bmatrix}$ |
| $A_{11}$ | $A_{12}$ | $A_{13}$ | $A_{14}$ | |
| $\begin{bmatrix}-3.0\\5.0\end{bmatrix}$ | $\begin{bmatrix}-6.5\\5.0\end{bmatrix}$ | $\begin{bmatrix}-6.5\\0\end{bmatrix}$ | $\begin{bmatrix}0\\0\end{bmatrix}$ | |

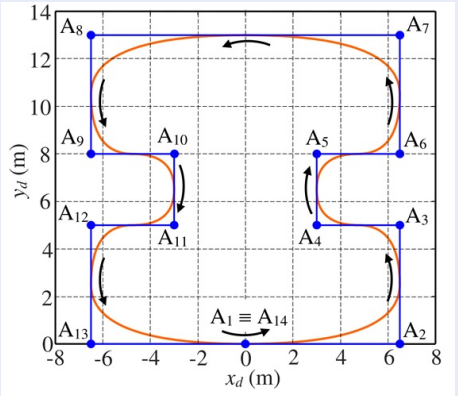



**Figure 4**: The desired Bézier trajectory of the OMR

The angular velocity $\Omega_d$ at point $G$ on the robot is determined as follows:

$$\Omega_d(t) = V_d(t)/\rho(t) \tag{13}$$

where $\rho_i \in [\rho_{min}, \rho_{max}]$ is the radius of the $\xi$ trajectory with i = 1,2.

$$\rho_i = \left| \frac{\left(\dot{x}_i^2 + \dot{y}_i^2\right)^{3/2}}{\dot{x}_i\ddot{y}_i - \dot{y}_i\ddot{x}_i} \right| \tag{14}$$

Here:

$$\begin{cases} \dot{x}_i = \frac{\triangle x}{\triangle t} = \frac{x_i - x_{i-1}}{t_i - t_{i-1}} \\ \dot{y}_i = \frac{\triangle y}{\triangle t} = \frac{y_i - y_{i-1}}{t_i - t_{i-1}} \end{cases};$$

$$\begin{cases} \ddot{x}_i = \frac{\triangle \dot{x}}{\triangle t} = \frac{\dot{x}_i - \dot{x}_{i-1}}{t_i - t_{i-1}} \\ \ddot{y}_i = \frac{\triangle \dot{y}}{\triangle t} = \frac{\dot{y}_i - \dot{y}_{i-1}}{t_i - t_{i-1}} \end{cases}.$$

From formulas (12, 13, 14) and the data of the trajectory interpolation points, we have the desired linear velocity $V_d$ and angular velocity $\Omega_d$ of the OMR described in Figure 5 when the robot moves along the Bézier trajectory $\xi_d$ with $\Delta t = 1$ s and maximal allowed speed $V_{dmax} = 0.6$ (m/s).

Therefore, the desired velocities of the OMR are given by:

$$\left[V_{xdr}\ V_{ydr}\ \Omega_{dr}\right]^T = T \begin{bmatrix} V_d \\ \Omega_d \end{bmatrix} \tag{15}$$

where $T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T$ (see Figure 3).

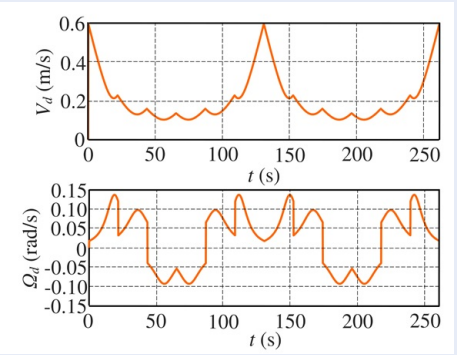



**Figure 5**: The desired linear and angular velocities of the OMR

## Determine the coefficients of the controller

By substituting Eq. (9) into Eq. (8) is obtained:

$$\dot{e} = (A - I_{3\times3}K)\, e_r \tag{16}$$

From that, we have the characteristic polynomial of the closed system:

$$\chi(s) = det\left[sI_{3\times3} - (A - I_{3\times3}K)\right] = \left| \begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & s \end{bmatrix} - \begin{bmatrix} 0 & \Omega_d & -V_{yd} \\ -\Omega_d & 0 & V_{yd} \\ 0 & 0 & s \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} K_1 & 0 & 0 \\ 0 & K_2 & 0 \\ 0 & 0 & K_3 \end{bmatrix} \right| \tag{17}$$

The controller gains of the matrix **K** is determined by comparing the real characteristic equation (16) with the desired closed-loop characteristic equation:

$$(s + 2\omega_n\zeta)\left(s^2 + 2\omega_n s\zeta + \omega_n^2\right) = 0 \tag{18}$$

with constant eigenvalues, natural angular frequency $\omega_n = \left(\Omega_d^2 + gV_d^2\right)^{0.5}$ and damping coefficient $0 < \zeta < 1$.

Finally, we obtain the controller gains as:

$$\begin{cases} K_1 = \zeta\left(\Omega_d^2 + gV_d^2\right)^{0.5} + \left(\zeta^2\Omega_d^2 + gV_d^2\left(\zeta^2 - 1\right)\right)^{0.5} \\ K_2 = \zeta\left(\Omega_d^2 + gV_d^2\right)^{0.5} - \left(\zeta^2\Omega_d^2 + gV_d^2\left(\zeta^2 - 1\right)\right)^{0.5} \\ K_3 = 2\zeta\left(\Omega_d^2 + gV_d^2\right)^{0.5} \end{cases} \tag{19}$$

where in $g \le \zeta^2 / \left(\left(1 - \zeta^2\right)\rho_{max}^2\right)$

From the above formulas, we choose the control coefficients $g$ and $\zeta$ by the simulation method presented below.

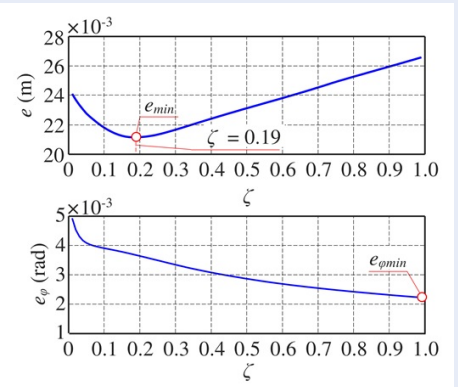



**Figure 6**: The position and posture errors of the OMR according to the coefficient $\zeta$

## Select the coefficient $\zeta$

We perform simulations for $\zeta$ is between [0.01 to 0.99], with a factor $g$ given by:

$$g \le \frac{\zeta_{min}^2}{\left(1 - \zeta_{min}^2\right)\rho_{max}^2},\ (\rho_{max} = 33.82) \tag{20}$$

so that condition (19) is always satisfied when the coefficient $\zeta$ varies. Figure 6 shows the robot errors according to the coefficient $\zeta$. From Figure 6, we choose $\zeta = 0.19$ because at that position, the error is the smallest.

## Select the coefficient $g$

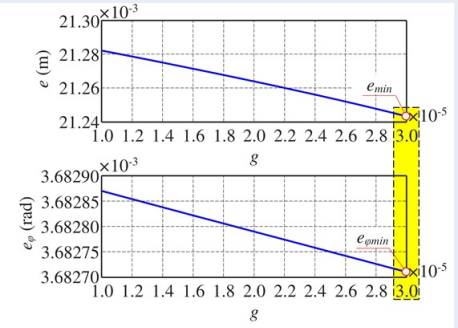



**Figure 7**: The position and posture errors of the OMR according to the coefficient $g$

With the coefficient $\zeta$ = 0.19, we carry out the same simulation with $g$ is between [$10^{-5}$ to $3\times10^{-5}$]. The curve in Figure 7 depicts the error of the robot according to $g$. From Figure 7, we choose $g = 3\times10^{-5}$ because at that position, the error is the smallest.

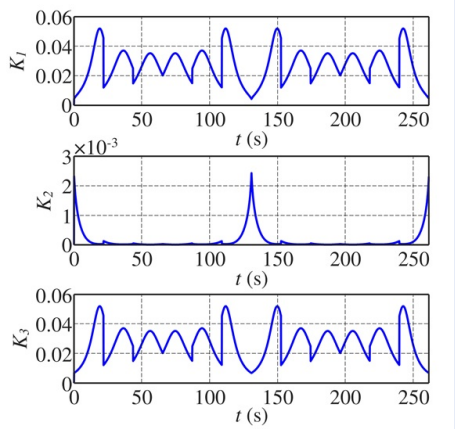



**Figure 8**: The change ofin $K_1$, $K_2$ and $K_3$

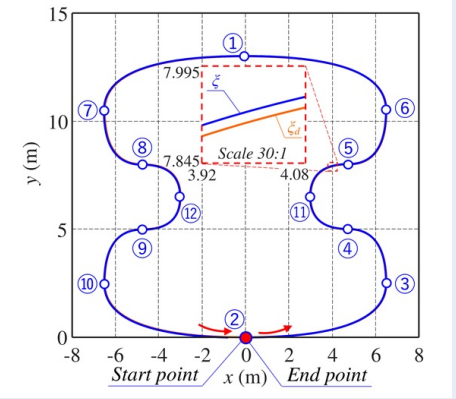



**Figure 9**: The motion trajectory of the OMR

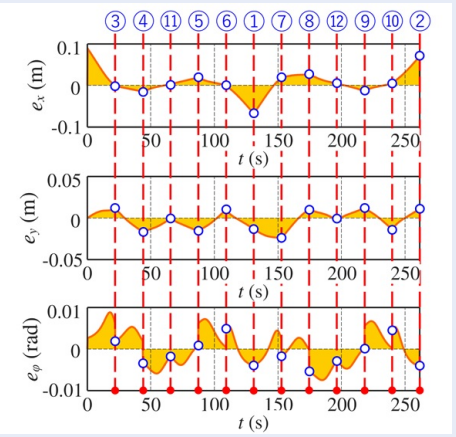



**Figure 10**: Position error of the G-point and posture error of the OMR.

With the coefficients $\zeta$ and g chosen as above, we have the variation of control parameters ($K_1$, $K_2$, $K_3$) described in Figure 8.

Figure 8 shows the variation of the controller gains $K_1$, $K_2$ and $K_3$ for the OMR tracks of the complex trajectory with small error (position and posture).

## SIMULATION RESULTS

With the above setting parameters including: (1) The complex motion trajectory has been modeled by Bézier curve, (2) The demension of the robot and (3) the varying-time gains $K_1(t)$, $K_2(t)$, $K_3(t)$ are selected so that the position and posture errors are the smallest. Figure 9 shows the Bézier trajectory tracking of the OMR when the robot is controlled by the linear state controller designed in Figure 3. Where the blue line $x$ is the motion trajectory, while the orange line is the desired Bézier trajectory.

Figure 10 depicts the position and posture errors of the OMR between the controlled and desired values during the OMR moves along the desired trajectory.

The points numbered from 1 to 10 in Figure 10 correspond to the special points in Figure 9, which are the inflection points of the motion trajectory. At these points there is the direction change and the linear velocity varies with time. Figure 11 illustrates the variation of the linear and angular robot velocities (between the controlled and the desired values). Where the blue line is the controlled value, and the orange line is the calculated value according to equation (12, 13) of the desired trajectory $x_d$ (Bezier curve see Figure 4 and Figure 5).

Figure 12 compares the angular velocities of the four mecanum wheels (between the calculated value from the inverse kinematics problem is calculated from equation (1) and the controlled value). In addition, the angular velocity on Figure 12 is the value that is controlled so that the OMR moves along the trajectory in Figure 9 and achieves linear and angular velocity, as depicted in Figure 11.

## DISCUSSION

From the simulation results, it can be seen that the controller established in this paper can track the complex trajectory with small errors as shown in Figure 9, Figure 10 and tend to right deviation along the x-axis. Thereby it shows that the OMR moves from the inside to outside of the desired trajectory around the poles with a small error. In which, the position error varies from 0 cm to 1.7 cm (at positions 1 and 2 in Figure 9 corresponds to t = 131 s and t = 261.5 s), these are the two locations with the largest curvature radius and linear velocity variation. While the postu-

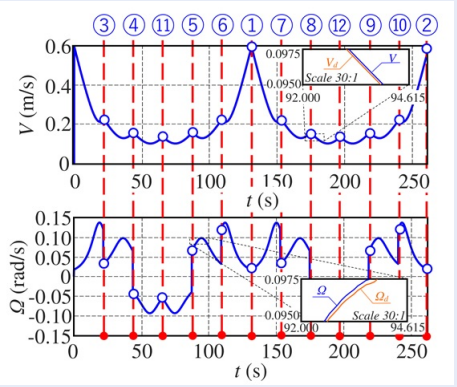



**Figure 11**: Linear and angular velocities of the OMR

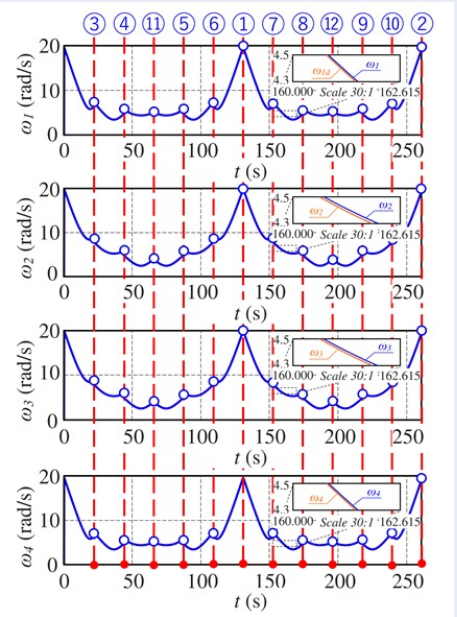



**Figure 12**: The angular velocities of four wheels

ral error varies from $0^o$ to $0.3^o$ and these are locations where the trajectory has a small curvature radius and the OMR changes direction suddenly from clockwise to counterclockwise along the tangent direction of the trajectory.

In addition, Figure 9 to Figure 12 show the motion law with symmetry in the y direction by two arcs: (1) the arc $\tau_1$ goes from position 2-> 3-> ..-> 1 and (2) the arc $\tau_2$ backs from position 1-> 7 -> ....-> 10 ->2. Therefore, we only need to discuss with the arc $\tau_1$. In the arc $\tau_1$ when observing three arcs, three arcs are observed (see Figure 9): (1) the first arc from position 2 -> 3->4, (2) the second arc from position 4->5 and (3) the third arc from position 5->6 - >1. We see that the linear velocity $V_G$ and the angular velocity Ω of the OMR are automatically adjusted according to the curvature radius $r$ of the motion trajectory so as not to exceed $V_{dmax}$ = 0.6 (m/s) with an error (see Figure 10): (a) The linear velocity error varies from 0 m/s to 0.00048 m/s (at positions 4 and 5, respectively, t = 43.7 s and 87.2 s); (b) The angular velocity error varies from 0 rad/s to 0.000056 rad/s (at position 11 corresponds, corresponding to t = 65.4 ss).

From the above discussion, it can be seen that: (1) the advantage of the proposed controller is that it has a simple structure while still ensuring that the OMR follows a complex trajectory with a small error. While some other studies have to use the used modern controllers [20,22,26] that require large hardware structures to satisfy the real-time processing speed. That makes the data processing part will be less and the hardware structure of the OMR does not need to be large, which is the advantage of this study compared to other studies; (2) The control gains are determined as functions depending on the curvature radius and error, which can respond to the change of linear and angular velocities of the OMR with small error. Therefore, it will reduce the position and posture error of the OMR.

In addition, the phenomenon of wheel slip versus the road surface hasn't been considered when modeling of robot in this study also the causes of the robot's error of position and posture.

## CONCLUSIONS

From the above calculated result and simulation tests, we evaluate and discuss giving some the main results as follows:

1) The nonlinear kinematic control problem of the OMR tracks a complex trajectory with small error and has been developed by a linear state feedback controller. Its advantage is a simple controller structure to increase processing speed and reduce hardware cost. Therefore, this is the an advantage of this study compared to previous studies.

2) The controller gains are determined from the position error and the motion trajectory in response to the small error of the linear and angular robot velocities. That made the complex trajectory tracking error of the OMR is small.

Therefore, we believe that the results of this study can be applied to control the OMRs in the logistics application of the modern manufacturing systems.

Additionally, the problems of longitudinal slip, lateral slip and Coulomb friction during the interaction between the wheel and the road surface, as well as the

application of the mecanum wheel for the development of borehole drilling robots in tunnelling construction[33] working in a tight environment, will be considered part of our future research goals.

## CONFLICT OF INTERESTS

The authors declared no potential conflicts of interest with respect to the research, authorship and publication of this article.

## AUTHOR CONTRIBUTION

Trinh Thi Khanh Ly made the paper's initiative idea, theoretical modelling, modeling, and implementation plan and corrected it. Author Trinh Thi Khanh Ly wrote the paper's manuscript; The author Hoang thien programmatically and simulates. All authors discussed the results, and reviewed and approved the final version of the manuscript.

## ACKNOWLEDGMENTS

This research was funded by the Ministry of Industry and Trade in a ministeriallevel scientific and technological research project, conducted in 2020, code: DTKHCN.076/20.

## REFERENCES

1. Doroftei I, Grosu V, Spinu V. Omnidirectional mobile robot - design and implementation. INTECH Open Access Publisher; 2007. p. 511-28;Available from: https://doi.org/10.5772/5518.
2. Yuan Z, et al. Trajectory tracking control of a four mecanum wheeled mobile platform: an extended state observer-based sliding mode approach. IET Control Theor Appl. 2020;Available from: https://doi.org/10.1049/iet-cta.2018.6127.
3. Qian J, Zi B, Wang D, Ma Y, Zhang D. The design and development of an omni-directional mobile robot oriented to an intelligent manufacturing system. Sensors (Basel). 2017;17(9):2073;PMID: 28891964. Available from: https://doi.org/10.3390/s17092073.
4. Ly TTK, Thai NH, Dzung LQ, Thanh NT. Determination of kinematic control parameters of omnidirectional AGV robot with mecanum wheels track the reference trajectory and velocity. Lecture Notes in Networks and Systems. International Conference on Engineering Research and Applications. 2021;178:319-28;Available from: https://doi.org/10.1007/978-3-030-64719-3_36.
5. Barreto S. JCL, Conceicao AGS, Dorea CET, Martinez L, de Pieri ER S, et al. Design and implementation of model-predictive control with friction compensation on an omnidirectional mobile robot. IEEE ASME Trans Mechatron;19(2):467-76;Available from: https://doi.org/10.1109/TMECH.2013.2243161.
6. Kirsch C, Röhrig C. Global Localization and Position T racking of an Automated Guided Vehicle. IFAC Proceedings Volumes. Proceedings of the 18th world congress the International Federation of Automatic Control Milano Italy. 2011;44(1):14036-41;Available from: https://doi.org/10.3182/20110828-6-IT-1002.01245.
7. Bačík J, Durovsky F, Biros M, Kyslan K, Perdukova D, Padmanaban S. Pathfinder development of automated Guided-Vehicle for hospital logistics. IEEE Access. 2017;5(1):26892-900;Available from: https://doi.org/10.1109/ACCESS.2017.2767899.
8. Thai NH, Ly TTK, Dzung LQ. Road map, routing and obstacle avoidance of AGV robot in the static environment of the flexible manufacturing system with matrix devices layout. Science and technology Development [journal]. 2021;24(3):2091-9;.
9. Bechtsis D, Tsolakis N, Vlachos D, Iakovou E. Sustainable supply chain management in the digitalisation era: the impact of Automated Guided Vehicles. J Cleaner Prod. 2017;142:3970-84;Available from: https://doi.org/10.1016/j.jclepro.2016.10.057.
10. Adăscăliței F, Doroftei I. Practical applications for mobile robots based on mecanum wheels a systematic survey. Rom Rev Precis Mech Opt Mechatron. 2011;40:21-9;.
11. Bayar G, Ozturk S. Investigation of the effects of contact forces acting on rollers of a Mecanum wheeled robot. Mechatronics. 2020;72:102467;Available from: https://doi.org/10.1016/j.mechatronics.2020.102467.
12. Tao B, Zhao X, Yan S, Ding H. Kinematic modeling and control of mobile robot for large-scale workpiece machining. Proc Inst Mech Eng B. 2022;236(1-2):29-38;Available from: https://doi.org/10.1177/0954405420933708.
13. Li Y, Dai S, Zhao L, Yan X, Shi Y. Topological design methods for Mecanum wheel configurations of an omnidirectional mobile robot. Symmetry. 2019;11(10):1268;Available from: https://doi.org/10.3390/sym11101268.
14. Thai NH, Thien H, Ly TTK. NURBS Curve Trajectory Tracking control for Differential-Drive Mobile Robot by a Linear State Feedback Dynamic controller. AUN/SEED-net joint regional conference in transportation, energy and mechanical manufacturing engineering. Hanoi, Vietnam. 2022;Available from: https://doi.org/10.1007/978-981-19-1968-8_50.
15. Thai NH, Ly TTK. Path tracking control for car-like robots by PID controller with time-varying parameters, science and technology development [journal]. 2022;.
16. Leena N, Saju KK. ModellingModeling and trajectory tracking of wheeled mobile robots. Procedia Technol. 2016;24:538-45;Available from: https://doi.org/10.1016/j.protcy.2016.05.094.
17. Vis IFA. Survey of research in the design and control of automated guided vehicle systems. Eur J Oper Res. 2006;170(3):677-709;Available from: https://doi.org/10.1016/j.ejor.2004.09.020.
18. Thai NH, Ly TTK. NURBS curve trajectory tracking control for differential-drive mobile robot by a linear state feedback controller. Lecture Notes in Networks and Systems. International Conference on Engineering Research and Applications. 2022;366:685-96;Available from: https://doi.org/10.1007/978-3-030-92574-1_71.
19. Thai NH, Ly TTK, Thien H, Dzung LQ. Trajectory tracking control for differential-drive mobile robot by a variable parameter PID controller. IJMERR. 2022:614-21;Available from: https://doi.org/10.18178/ijmerr.11.8.614-621.
20. Hassan N, Saleem A, Saleem A. Neural network-based adaptive controller for trajectory tracking of wheeled mobile robots. IEEE Access. 2022;10:13582-97;Available from: https://doi.org/10.1109/ACCESS.2022.3146970.
21. AmirReza Haqshenas M, Fateh MM, Ahmadi SM. Adaptive control of electrically- driven nonholonomic wheeled mobile robots: Taylor series-based approach with guaranteed asymptotic stability. IEEE Trans Syst Man Cybern Syst. 2020;51(11):6874-84;Available from: https://doi.org/10.1002/acs.3104.
22. Ahmadi SM, Behnam Taghadosi M, Haqshenas M A. A state augmented adaptive backstepping control of wheeled mobile robots. Trans Inst Meas Control. 2021;43(2):434-50;Available from: https://doi.org/10.1177/0142331220961700.
23. Thai NH, Ly TTK, Long NT, Dzung LQ. Trajectory tracking using linear state feedback controller for a Mecanum wheel omnidirectional. Mechanisms and Machine Science. IFToMM Asian conference on Mechanism and Machine Science. 2022;113:411-21;Available from: https://doi.org/10.1007/978-3-030-91892-7_39.
24. Aydogdu O, Korkmaz M. A simple approach to design of variable parameter nonlinear PID controller. IPCSIT. Interna-

tional Conference on Advancements in Information Technology. Vol. 20; 2011. p. 81-5;.
25. Iswanto A, Ma'arif A, Maharani Raharja N, Supangkat G, Arofiati F, Sekhar R, et al. PID-based with odometry for trajectory tracking control on four-wheel omnidirectional Covid-19 aromatherapy robot. Emerg Sci J. 2021;5:157-81;Available from: https://doi.org/10.28991/esj-2021-SPER-13.
26. Malayjerdi E, Kalani H, Malayjerdi M. Self-tuning fuzzy PID control of a four-Mecanum wheel omni-directional mobile platform. In: Electrical engineering (ICEE), Iranian conference on. IEEE Publications; 2018, 43(2). p. 816-20;Available from: https://doi.org/10.1109/ICEE.2018.8472568.
27. Jia Q, Liu S, Zhang L, Zhang S. Motion control of omnidirectional mobile robot based on fuzzy PID. Available from: https://doi.org/doiorg/doi:10.1109/CCDC.2019.8833047. In: Chinese control and decision conference (CCDC). IEEE; 2019. p. 5149-54;Available from: https://doi.org/10.1109/CCDC.2019.8833047.
28. Sun Z, Xie H, Zheng J, Man Z, He D. Path-following control of Mecanum-wheels omnidirectional mobile robots using nonsingular terminal sliding mode. Mech Syst Signal Process. 2021;147:107128;Available from: https://doi.org/10.1016/j.ymssp.2020.107128.
29. Lu X, Zhang X, Zhang G, Fan J, Jia S. Neural network adaptive sliding mode control for omnidirectional vehicle with uncertainties. ISA Trans. 2019;86:201-14;PMID: 30420141. Available from: https://doi.org/10.1016/j.isatra.2018.10.043.
30. Cui QZ, Li X, Wang XK, Zhang M. Backstepping control design on the dynamics of the omni-directional mobile robot. Appl Mech Mater. 2012;203:51-6;Available from: https://doi.org/10.4028/www.scientific.net/AMM.203.51.
31. Rogers DF. An Introductron to NURBS. Morgan kaufmann Publishers; 2001;.
32. Tzafestas SG. Introduction to mobile robot control. Elsevier Inc; 2013;Available from: https://doi.org/10.1016/B978-0-12-417049-0.00005-5.
33. Thai NH, Thai NQ. A kinematic control algorithm for blast-hole drilling robotic arm in tunneling. Science and technology Development [journal]. 2017;20(K5):13-22;Available from: https://doi.org/10.32508/stdj.v20iK5.1153.