Geometric path following control for an omnidirectional mobile robot

Jian Wang*, Sergey A. Chepinskiy*†, Aleksandr J. Krasnov†, Botao Zhang*
Huimin Liu†, Yifan Chen† and Denis A. Khvostov†
*Hangzhou Dianzi University
School of Automation, Xiasha Higher Education Zone
Hangzhou, Zhejiang Province 310018, P.R. China
Email: wangjian@hdu.edu.cn
†ITMO University
49 Kronverksky prospekt, Saint-Petersburg, 197101, Russia
Email: krasnov.aleksander@gmail.com, chepinsky_s@hotmail.com

***Abstract*—The paper describes an approach to the development of the geometric path following control for an omnidirectional mobile robot. Desired path of movement in the space is represented by an intersection of two implicit surfaces. Path following control problem is posed as a problem of maintaining the holonomic relationships between the system outputs. Control is synthesized using the dierential geometrical method through nonlinear transformation of initial dynamic model. The main results presented are the nonlinear control algorithms and experimental approbation result.**

I. Introduction

The paper considers the development of an omnidirectional mobile robot path control system, that is, the problem of providing a motion along a given path in the robot operating area. With the advent of unmanned vehicles the path following control problem became even more urgent, because path following is an UAV major operating mode.

A popular approach is consideration of a guidance as a tracking system controlled by a reference model is presented in [1], [2], [3]. The path is generally set by a time-dependent function, which leads to practical problems when the object motion is behind or ahead of the program due to parametric uncertainties or external disturbances. To solve this problem, the path should be parametrized by the length instead of time, and dynamics of this parameter should be introduced in the system model. In this guidance approach widely used the notion of a virtual vehicle which moves on a geometrically dened path in [1], [4], [5]. The virtual vehicle is coupled to the motion of the real one via a some abstract link. This method rather easily realizes the motion along polynomial curves, which provides better path planning and more accurate path following.

An alternative approach is based on stabilization of invariant manifolds in state space based on feedback linearization [6], [7] or passive-based control [8], [9]. Simply speaking, a transformation generating an attractor in state space is selected for the initial system. In path following context, the attractor is a desired path set in output coordinates. Then the designer should only stabilize this solution, which is much less demanding than creating a tracking system as in the first approach. As a control object, an autonomous robot is a multichannel nonlinear dynamic system. Control system of a mobile robot should generate control actions providing preset motion of the centre of mass in operating area. The one of the methods for synthesizing the control algorithms was proposed by I.V. Miroshnik [12]-[15]. It is based on the second approach and implies nonlinear transformation of robot model to the task-oriented coordinate system, which makes it possible to reduce the complex multichannel control problem to several simple problems of compensation of linear and angular deviations and then to find adequate control laws using nonlinear stabilization [16]-[18].

Differentially geometric methods of nonlinear control theory [12]-[16] are used in the analysis method for these systems and synthesis of control algorithms solving the path following problem as a stabilization problem with respect to implicit curve (Fig. 1).

This article focuses directly on the synthesis of controllers without restricting the path planning method. The main requirement is that the path should be composed of straight lines and circles. All physical limitations on the circle radius should be taken into consideration at the planning stage based on the object maneuverability, speed, and available control resources.

II. Motion control design

The position of robot body as a solid body on a plane (Fig. 1) is characterized by the Cartesian position vector $q = [x, y]^T \in \mathbb{R}^2$ of the center of mass C and the angle α of C-fixed frame with respect to the base frame XOY).

Consider the model of the plant in the following form

$$m\ddot{q} = F, \tag{1}$$

$$\dot{q} = R_O^I(\alpha)v, \tag{2}$$

This work was financially supported by the Government of the Russian Federation, Grant 074-U01.

This work was supported by the Ministry of Education and Science of Russian Federation (Project 14.Z50.31.0031).

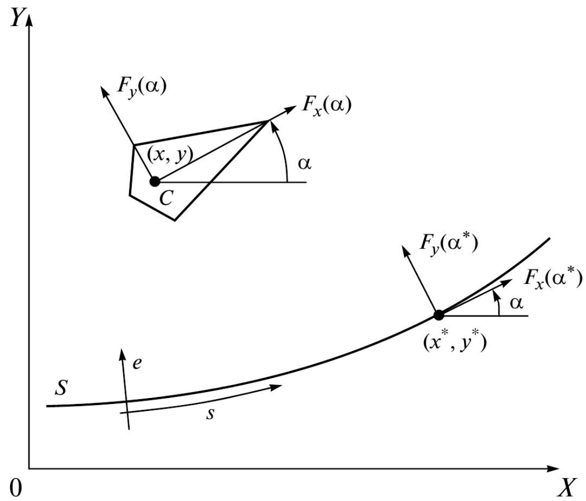

Fig. 1. Autonomous robot, path S and task-based coordinates (s, e)

$$R_O^I(\alpha) = \begin{bmatrix} cos\alpha & sin\alpha \\ -sin\alpha & cos\alpha \end{bmatrix}, \quad (3)$$

$$\dot{\alpha} = \omega, \quad (4)$$

where $q = [x, y]^T \in \mathbb{R}^2$ is the Cartesian position vector of the center of mass C in inertial frame XOY, $m \in \mathbb{R}$ is a total mass of the rigid body, $F = [F_x, F_y]^T \in \mathbb{R}^2$ is the vector of control forces in the inertial frame XOY, $v = [v_x, v_y]^T \in \mathbb{R}^2$ is the velocity vector of the center of mass C in the inertial frame XOY, $R_O^I(\alpha) \in SO(2)$ is the rotation matrix from the body-xed frame to the inertial frame.

The desired path is a smooth segment of curve S (see Fig. 1) implicitly described as

$$\varphi(q) = 0. \quad (5)$$

and relevant local coordinate s (path length) is defined as

$$s = \psi(q). \quad (6)$$

Path following control problem for an autonomous robot is posed as a problem of maintaining the holonomic relationships between the system outputs set in (6). It is augmented by the description of desired longitudinal motion of the point of the centre of mass of the rigid body along the desired path S usually set using the reference velocity of longitudinal motion $V^* = \dot{s}^*$.

Consider the errors in path following. Violation of condition (6) is characterized by orthogonal deviation

$$e(q) = \varphi(q), \quad (7)$$

zeroed at manifold S. Therefore, the path following control problem for an autonomous robot consists in determination of inputs F_x, F_y in closed loop, which provides:

(a) geometric subtask, or stabilization of robot motion with respect to the curve S, which implies asymptotic zeroing of spatial deviation e;

(b) kinematic subtask, or asymptotic zeroing of velocity error

$$\Delta V = V^* - \dot{s}. \quad (8)$$

The simple way to design the control is using of cascade approach. On the rst step we shape the inner loop with respect to velocities. Consider Lyapunov function

$$V_1 = \frac{1}{2}(\dot{q} - \bar{v})^T(\dot{q} - \bar{v}),$$

where $\bar{v}$ is the vector of desired velocities. Find the derivation of the Lyapunov function V_1.

$$\dot{V}_1 = (\dot{q} - \bar{v})^T(\ddot{q} - \dot{\bar{v}}) = (\dot{q} - \bar{v})^T(\frac{F}{m} - \dot{\bar{v}}).$$

Select the control signal in the form

$$\frac{1}{m}F = \dot{\bar{v}} - k_q(\dot{q} - \bar{v}), \quad (9)$$

where k_q is a positive constant.

Then the derivation of the Lyapunov function V_1 is

$$\dot{V}_1 = -k_q(\dot{q} - \bar{v})^T(\dot{q} - \bar{v}) \leq 0.$$

It proves the asymptotic stability of the point $\dot{q} - \bar{v} = 0$. Now we can rewrite the original system (1) in the reduced form:

$$\dot{q} = \bar{v}. \quad (10)$$

Afterwards we should shape outer loop for the solution of the path following task. Lets construct the control $\bar{v}$ in the next form:

$$\bar{v} = u_e + u_s, \quad (11)$$

where u_e is the term which provides the stabilization with respect to the desired path and u_s is the feed-forward term, which maintains the desired velocity along the path. Introduce the coordinate transformation using the Jacobian matrix in the following form:

$$\Upsilon(q) = \begin{bmatrix} \frac{\partial\varphi(q)}{\partial y} & -\frac{\partial\varphi(q)}{\partial x} \\ \frac{\partial\varphi(q)}{\partial x} & \frac{\partial\varphi(q)}{\partial y} \end{bmatrix}. \quad (12)$$

Then the transformation to the task-oriented basis will be

$$\begin{bmatrix} \dot{s} \\ \dot{e} \end{bmatrix} = \Upsilon(q)\dot{q} = \Upsilon(q)R_I^O(\alpha)v. \quad (13)$$

Respectively, the inverse transformation will be described as

$$\dot{q} = R_O^I(\alpha)\Upsilon^{-1}(q)\begin{bmatrix} \dot{s} \\ \dot{e} \end{bmatrix}. \quad (14)$$

And obviously we can choose the control u_s to solve kinematic subtask (b) in the form

$$u_s = R_O^I(\alpha)\Upsilon^{-1}(q)\begin{bmatrix} V^* \\ 0 \end{bmatrix}. \quad (15)$$

The second term is the stabilizing control. Consider the Lyapunov function V_2 to obtain u_e.

$$V_2 = \frac{k_e}{2}\varphi^2(q). \quad (16)$$

Find the derivation of the Lyapunov function V_2.

$$\dot{V_2} = k_e\varphi(q)\left(\frac{\partial}{\partial q}\varphi(q)\right)^T \dot{q} = k_e\varphi(q)\left(\frac{\partial}{\partial q}\varphi(q)\right)^T u_s +$$

$$+k_e\varphi(q)\left(\frac{\partial}{\partial q}\varphi(q)\right)^T \Upsilon^{-1}(q)\begin{bmatrix}V^* \\ 0 \\ 0\end{bmatrix} = k_e\varphi(q)\left(\frac{\partial}{\partial q}\varphi(q)\right)^T u_s.$$

As you can see, the second half of the expression is identically zero due to orthogonality. Now select u_e as

$$u_e = -k_e\varphi(q)\frac{\partial}{\partial q}\varphi(q), \tag{17}$$

where k_e is a positive constant.

Then the derivation of the Lyapunov function V_2 is

$$\dot{V_2} = -u_e^2 \leq 0,$$

then the derivation of the Lyapunov function V_2 is negative definite. It proves the asymptotic stability of the initial system at the point $e(q) = 0$ which provide the solution of geometric subtask (a) and as a result, together with the solution of kinematic subtask (b) shown above, the initial path following task.

III. Control algorythms for typical paths

Modern industrial motion control systems mostly use an approach where the predetermined path is parameterized using the straight line segments and arcs of circles. This approach has a number of weak points [20], but it is rather simple and illustrative. Now consider the applications of the described method for these typical paths (Fig. 2).

Let the robot path be a straight line segment. The normalized equation of the straight line is given by

$$\varphi(q) = -\sin\alpha^* x + \cos\alpha^* y + \varphi_0 = 0, \tag{18}$$

$$\psi(q) = \cos\alpha^* x + \sin\alpha^* y + \psi_0, \tag{19}$$

where α^* is the line inclination, $\varphi_0 = const$, $\psi_0 = const$. Orthogonal Jacobian matrix takes the form

$$\Upsilon(q) = \begin{bmatrix}\cos\alpha^* & \sin\alpha^* \\ -\sin\alpha^* & \cos\alpha^*\end{bmatrix} \in SO(2). \tag{20}$$

Obviously, the path curvature is zero.

Now consider the case when the path segment is an arc of a circle with radius R centered at (x_0, y_0). Write the equation of the circle as follows

$$\varphi(q) = \frac{1}{2R}(R^2 - (x - x_0)^2 - (y - y_0)^2) = 0, \tag{21}$$

$$\psi(q) = R\arctan\frac{(y - y_0)}{(x - x_0)}. \tag{22}$$

Orthogonal Jacobian matrix takes the form

$$\Upsilon(q) = \frac{1}{R}\begin{bmatrix}-(y - y_0) & (x - x_0) \\ -(x - x_0) & -(y - y_0)\end{bmatrix} \in SO(2). \tag{23}$$

These typical paths are shown in Fig. 2.

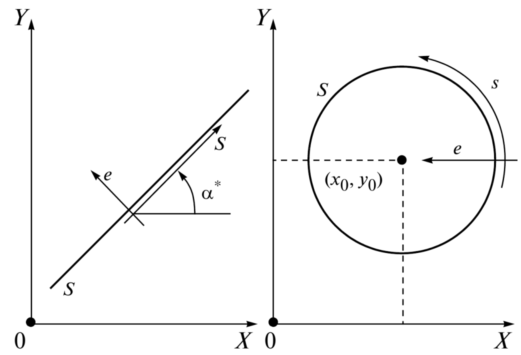

Fig. 2. Task-based coordinates for typical paths

Control methods for elementary base smooth curves have been considered above, and, obviously, these results can be combined for a complex path consisting of straight line segments and arcs of circle. Control is realized by a hybrid controller including the algorithms for motion along a straight line, along a circle and the switching logic. Using more elementary paths provides more flexible parameterization of the path, which improves the performance.

A sensor system onboard the robot is assumed to generate its current position in the base frame where the path is specified. Algorithms of switching between the points have not been deeply investigated so far, but some options can be proposed:

1. $((x - x_{wp})^2 + (y - y_{wp})^2) < \varepsilon$, where (x, y) is the robot position, (x_{wp}, y_{wp}) are the point coordinates, ε is the width of the switching area. This algorithm was used in the study because of its simplicity.

2. Operating area can be divided into sections (for example using Voronoi partition [21]), containing a part of the path and limited by preset points. It is a more powerful approach since it can determine in which sections the path following problem need not be solved for some reason (for example the robot is too far), but it is more difficult to realize as compared with the first one.

Figure 3 shows the simulation for a complex path. Consider this example in more detail. Waypoints $P_1 - P_6$ were used as initial data to set the path. It is a rather simple and popular method. At the first step we connect these points with straight lines. For smooth transition between the segments we introduce transition areas specified using the circles of fixed radius R_c determined by the physical capacity of the control object. In this example, $R_c = 0.5$ m. The radius was used to calculate the centers of the circles $C_1 - C_4$. Neighborhood ε (here, $\varepsilon = 0.01$) was introduced, where switching between the segments occurred. Clearly, the proposed method solves the considered problem even without any additional analysis, though the switching conditions and robot behavior during the switching require additional studies.

At transitions from straight lines to the circles the path is

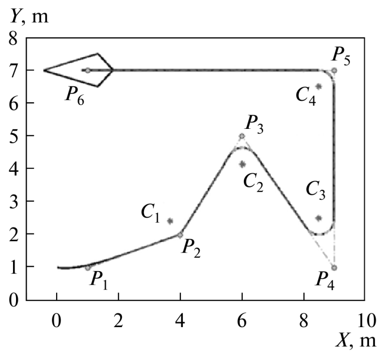

Fig. 3. Motion along a complex path simulation

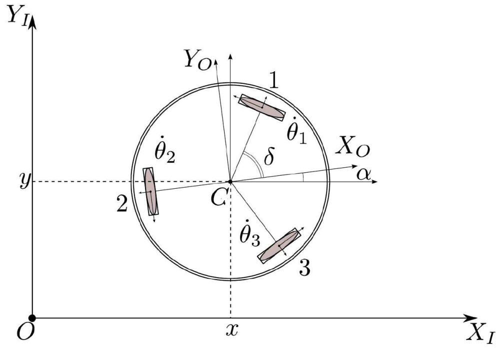

Fig. 4. Omnidirectional mobile robot Robotino

followed without any errors (Fig. 3). But this simulation does not account for the dynamics of control organs. In practice such an accurate transition would be surely impossible because physically the path curvature can not be changed in an abrupt manner.

Real motion apart the straight line and the circle contains the Cornus spiral [22] where the curvature changes linearly from zero to the value inversely proportional to the circle radius (or vice versa for the circle-straight line transition). The length of this segment is determined by the circle radius, so smaller curvature should be selected to reduce the transition errors. An alternative method to completely eliminate the transition errors is to use polynomial curves for the transition segments, which provide smooth changes in the curvature.

Thus, an effective robot path following control system can be realized by combining rather simple methods.

IV. Experiment

Proposed in the previous section synthesis procedure was applied to design the path following control system for omnidirectional mobile robot Robotino (Fig. 4). At the beginning let us take a detailed look of the construction of the robot. As part of this work should pay particular attention to the motor system (platform with three DC motors with integrated gearbox and belt drive on omni wheels), which allows the robot to move in different directions. It should be noted that in order to stabilize the shaft rotational speed of each of the three engines of the robot built using proportional-integral-differential controllers with preset coefficients. Also, the robot is equipped with a rotation angle sensor for measuring rotation angles of motors axes. In addition, the robot is equipped with a navigation system "North Star" which greatly facilitates the testing of developed algorithms of path following in action.

The mathematical model of the robot Robotino will be considered in the form of (1) - (4). Since the robot three wheels evenly spaced around the circumference (Fig. 4), controls the distribution of the matrix is as follows:

$$\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\\ \dot{\theta}_3\end{bmatrix} = \frac{n}{R}\begin{bmatrix}-\sin\frac{\pi}{3} & \cos\frac{\pi}{3} & L\\ 0 & -1 & L\\ \sin\frac{\pi}{3} & \cos\frac{\pi}{3} & L\end{bmatrix}\begin{bmatrix}v\\ \dot{\alpha}\end{bmatrix}, \tag{24}$$

where n is the gear ratio of the drive gear, R is the radius of the wheel and L is the distance from the center of mass C to the wheel.

In this experiment, the aim was also to show the behavior of the closed-loop system with the switching paths. The desired trajectory is composed of two straight sections and two circles:

1. Straight-line section from the point $[-400.400]$ to the point $[400,400]$ and angle $\alpha = 0$;
2. An arc of a circle with center at $[400.0]$ and the radius $R = 400$;
3. Straight-line section from the point $[400-400]$ to a point $[-400,-400]$ and angle $\alpha = 0$;
4. An arc of a circle with center $[-400.0]$ and the radius $R = 400$.

As described in the previous section, we will generate a control action in the form of two components which are responsible for decision respectively kinematic and geometric subtasks:

$$\bar{v} = u_e + u_s,$$

$$u_s = R_O^I(\alpha)\Upsilon^{-1}(q)\begin{bmatrix}V^*\\ 0\end{bmatrix},$$

$$u_s = \frac{V^*}{\sqrt{\left(\frac{\partial\varphi(q)}{\partial x}\right)^2 + \left(\frac{\partial\varphi(q)}{\partial y}\right)^2}}\begin{bmatrix}\cos\alpha\frac{\partial\varphi(q)}{\partial y} - \sin\alpha\frac{\partial\varphi(q)}{\partial x}\\ -\sin\alpha\frac{\partial\varphi(q)}{\partial y} - \cos\alpha\frac{\partial\varphi(q)}{\partial x}\end{bmatrix},$$

$$u_e = -k_e\varphi(q)\frac{\partial}{\partial q}\varphi(q),$$

In this experiment regulator parameters were chosen as follows: $k_e = 3$, $V^* = 3$ m/c.

The results of the experiment are presented on figures (5)-(7). It is obvious that proposed algorythm demonstrated very high efficiency dyring the experiment and provided fairly accurate motion along the compex path (see Fig. 5 and 7) with small spatial deviation (see Fig. 6).

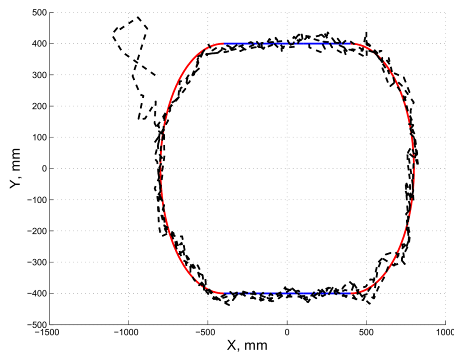

Fig. 5. Experiment on motion along the complex path

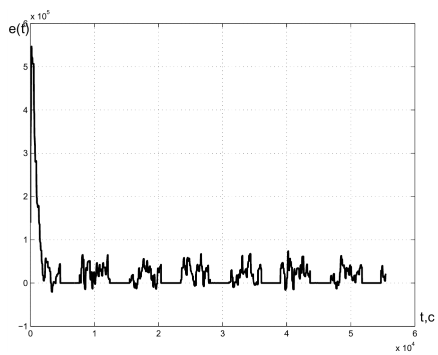

Fig. 6. Spatial deviation $e(q)$

V. CONCLUSION

The proposed controller and control algorithms can be helpful in development of path following control systems for mobile robots (underwater or airborne robots). The approach is rather flexible: adding new types of base paths improves the system efficiency, which is the primary goal of the future research. To illustrate efficiency of this approach testing was performed on the basis of omnidirectional robot Robotino from the company Festo Didactics, which showed a high efficiency of the proposed trajectory control algorithms. The performance of proposed controllers in the presence of parametric uncertainties and external disturbances should be the subject to further analysis.

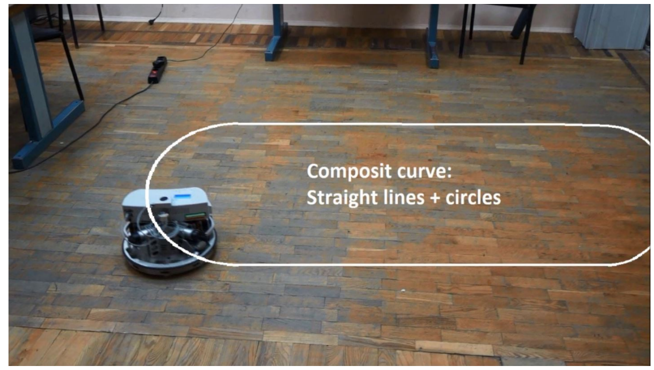

Fig. 7. Photo of the experiment with Robotino

REFERENCES

[1] Aguiar A.P., Hespanha J.P., Kokotovic P.V. Path-following for nonminimum phase systems removes performance limitations // IEEE Transactions on Automatic Control. 2005. V. 50. P. 234239.

[2] Breivik, M. and Fossen, T. (2005). Principles of guidance-based path following in 2d and 3d. In Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC 05. 44th IEEE Conference on, 627634. doi:10.1109/CDC.2005.1582226.

[3] Lee T., Leok M., McClamroch N.H. Geometric tracking control of a quadrotor UAV on SE(3) // Proceedings of the IEEE Conference on Decision and Control. 2010. P. 54205425.

[4] Lapierre, L., Soetanto, D., and Pascoal, A. (2003). Nonlinear path following with applications to the control of autonomous underwater vehicles. In Decision and Control, 2003. Proceedings. 42nd IEEE Conference on, volume 2, 12561261. IEEE.

[5] Lapierre, L. and Soetanto, D. (2007). Nonlinear path following control of an auv. Ocean engineering, 34(11), 17341744.

[6] Nielsen, C., Fulford, C., and Maggiore, M. (2009). Path following using transverse feedback linearization: Application to a maglev positioning system. In American Control Conference, 2009. ACC 09., 30453050.

[7] Hladio, A., Nielsen, C., and Wang, D. (2013). Path following for a class of mechanical systems. Control Systems Technology, IEEE Transactions on, 21(6), 23802390.

[8] El-Hawwary, M.I. and Maggiore, M. (2011). Case studies on passivity-based stabilisation of closed sets. International Journal of Control, 84(2), 336350.

[9] El-Hawwary, M.I. and Maggiore, M. (2013). Reduction theorems for stability of closed sets with application to backstepping control design. Automatica, 49(1), 214222.

[10] Anton A. Pyrkin, Alexey A. Bobtsov, Sergey A. Chepinskiy, and Yuriy A. Kapitanyuk Compensation of Unknown Multiharmonic Disturbance for Nonlinear Plant with Delay in Control // PREPRINTS 8th IFAC Symposium on Nonlinear Control Systems, Bologna, Italy (NOLCOS 2010) 1-3 september 2010 pp. 481 486

[11] Bobtsov A., Kolyubin S., Pyrkin A., Shavetov V., Chepinskiy S., Kapitanyuk Y., Kapitonov A., Bardov V., Titov A., Surov M., Using of LEGO Mindstorms NXT Technology for Teaching of Basics of Adaptive Control Theory // 18th IFAC World Congress. Milan. Italy. 2011.

[12] Miroshnik, I. and Nikiforov, V. (1996). Trajectory motion control and coordination of multilink robots. Prepr. 13th IFAC World Congress, A, 361366.

[13] Miroshnik, I.V., Fradkov, A.L., and Nikiforov, V.O., Nelineinoe upravlenie slozhnymi dinamicheskimi sistemami (Nonlinear and Adaptive Control of Complex Dynamic Systems), St. Petersburg: Nauka, 2000.

[14] Miroshnik I.V., Chepinskiy S.A. Trajectory control of underactuated mechanisms. 2nd IFAC Conference on Mechatronic Systems. Berkeley, 2002 P. 959-1004.

[15] Miroshnik I.V., Chepinskiy S.A. Trajectory motion control of underactuated manipulators. PREPRINTS 7th IFAC Symposium on Robot Control. September 1-3, Wroclaw, Poland, 2003 P. 105-110.

[16] Fradkov, A., Miroshnik, I., and Nikiforov, V. (1999). Nonlinear and Adaptive Control of Complex Systems. Mathematics and Its Applications. Springer.

[17] Kapitanyuk, Y. and Chepinskiy, S. (2013). Control of mobile robot following a piecewise-smooth path. Gyroscopy and Navigation, 4(4), 198203.
[18] Kapitanyuk, Y., Chepinskiy, S., and Kapitonov, A. (2014). Geometric path following control of a rigid body based on the stabilization of sets. In 19th IFAC World Congress.
[19] Jian Wang, Kapitanyuk Y.A., Chepinskiy S.A., Dongliang Liu, Krasnov A.J. (2015). Geometric path following control in a moving frame. IFAC-PapersOnLine 48-11 (2015). P. 150-155.
[20] Dovgobrod, G.M., Klyachko, L.M., and Rogozhnikov, A.V., Using Parametric Approximation in Planning the Vehicle Motion Trajectories, Izvestia Vuzov. Priborostroenie, 2009, vol. 52, no. 9, pp. 1117.
[21] LaValle, S.M., Planning Algorithms, Cambridge University Press, Cambridge, 2006.
[22] Girbes, V., Armesto, L., and Tornero, J., Continuous-Curvature Control of Mobile Robots with Constrained Kinematics, Proceedings of the 18th IFAC World Congress, 2011, pp. 35033508.