

# Trajectory tracking control with obstacle avoidance capability for unicycle-like mobile robot

W. KOWALCZYK, M. MICHAŁEK, and K. KOZŁOWSKI\*

Poznań University of Technology, Chair of Control and Systems Engineering, 3a Piotrowo St., 60-965 Poznań, Poland

**Abstract.** In this paper the trajectory tracking control algorithm with obstacle avoidance capability is presented. As a robot gets into a neighborhood of the obstacle, the collision avoidance behavior is turned on. It is implemented using the artificial potential function (APF) that increases to infinity as the robot approaches a boundary of the obstacle. This feature guarantees collision avoidance. As avoidance behavior is active only in the neighborhood of the obstacle it does not affect the motion when there is no risk of the collision. Authors show that trajectory of the robot converges to desired one when a robot is out of the APF area. Due to a local characteristic of the APF, the implementation of the algorithm of the robot that uses only on-board sensors is possible. The stability proof is presented for both a near obstacle and obstacle-free areas. Effectiveness of the algorithm is illustrated with experiments on a real robot in an environment with static circle-shaped obstacles.

**Key words:** mobile robot, trajectory tracking, robot control, collision avoidance.

## 1. Introduction

One of the classical motion control tasks for mobile robots is the trajectory tracking one, which has been treated by many authors during the last two decades of robotics research (selected surveys can be found in [1–3]). Usually, the motion control solutions are proposed under assumption that the robot movement takes place in an obstacle-free environment. As a consequence, the stability and convergence analysis of a closed-loop control system does not take into account possible collisions which may happen between a robot and obstacles. Furthermore, it is also commonly assumed that the a priori-planned reference trajectory is itself collision-free. In practice, however, such the prerequisites may become quite limiting, not only due to the obvious reasons but also due to the high computational power usually required for the collision-free motion planning task when performed for a cluttered robot environment [4]. Hence, the practically-motivated approaches to trajectory tracking problem should go further toward the motion control problem extension which admits existence of obstacles in the robot motion space, and toward its solution which would improve reliability to the possible and unexpected collision events.

So far, a number of different collision avoidance strategies have been proposed in the literature. The fundamental work [5] treats the avoidance problem from the formal Lyapunov perspective. Some new results can be also found in [6]. When considering obstacles they usually are surrounded by convex local zones (usually of circular shape) within which artificial potential functions (APFs) are defined [7–9]. The role of APFs is to repel the robot body when it is too close to the obstacle boundary. This approach can be used in a case of static obstacles but it can be efficient also in the case of dynamic obstacles, which can be represented by other robots

moving in a task space in the multiagent systems of mobile sensor networks [10–21]. The collision-cone concept adopted for instance in [22] and [23] is based on the geometrical approach formulated by using some azimuthal and radial distances. Fluid-flow techniques has been adapted to avoidance control and presented for example in the recent papers [24, 25]. An alternative approach utilizes the so-called navigation function [26] which leads to the solution without any local minima. In this case the control problem can be solved even for a very complex geometry of robot environment [27], but utilization of the method requires very large computational effort making the concept less popular in applications.

In the paper we propose a geometrically motivated control solution to the merged two control problems – trajectory tracking and obstacle collision avoidance. We consider the problem of asymptotic tracking of admissible and persistently exciting reference trajectories in the task space where multiple separated static obstacles are present. Our solution is dedicated to the unicycle-like robot in the form of a differentially driven kinematic vehicle of (2,0) type [28]. The characteristic properties of the concept proposed in the result from its geometrical origins (VFO) with clear physical interpretation of particular control components, and integration of an additional avoidance (repelling) control term to the original VFO tracking controller presented in details in [29]. Thus, the method is an intuitive extension of the VFO tracking controller, which adds a practical feedback (reactive) functionality of collision avoidance activated only inside the defined local zones surrounding obstacles by the artificial potential functions. This paper is an expanded version of our recent work presented in [30].

The paper is organized as follows. In Sec. 2 kinematic model of the robot and artificial potential function utilized to avoid collisions are presented. In Sec. 3 VFO control algo-

\*e-mail: krzysztof.kozlowski@put.poznan.pl

rithm expanded with collision avoidance module is described. In Sec. 4 a stability and convergence proofs of the closed-loop system are provided. Experimental results are presented and commented on in Sec. 5. Section 6 concludes the paper.

## 2. Robot kinematics and APFs

**2.1. Unicycle-like mobile robot.** The kinematic model of a robot is given by the equation:

$$\begin{bmatrix} \dot{\theta} \\ \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \cos \theta \\ 0 & \sin \theta \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad (1)$$

where  $x$ ,  $y$ , and  $\theta$  are position and orientation coordinates of the robot (Fig. 1), and  $\mathbf{u} = [u_1 \ u_2]^T$  is the control vector, which includes  $u_1$  – angular velocity control and  $u_2$  – longitudinal velocity control.

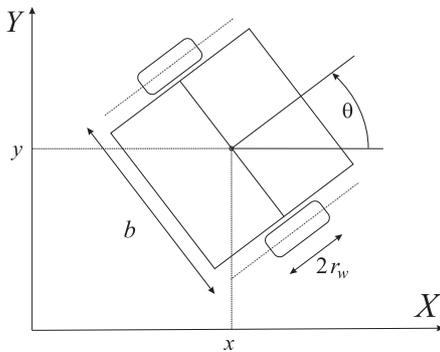


Fig. 1. Model of the mobile robot

**2.2. Artificial potential functions.** Each obstacle located at position  $\mathbf{p}_{oi} = [x_{oi} \ y_{oi}]^T$  is surrounded by the artificial potential field which exerts a repelling force on robot entering neighborhood of the obstacle. In this paper it is assumed that all obstacles are covered by the circles (Fig. 3).

Let us define the set of coordinates for the collision area of the  $i$ -th obstacle:

$$\Delta_i = \{\mathbf{p} \in \mathbb{R}^2, l_i \leq r_i\}, \quad (2)$$

where  $\mathbf{p} = [x \ y]^T$  and  $l_i = \|\mathbf{p} - \mathbf{p}_{oi}\|$ , the set of coordinates for the repel area:

$$\Gamma_i = \{\mathbf{p} \notin \Delta_i, r_i < l_i < R_i\}, \quad (3)$$

and  $D_i = \Delta_i \cup \Gamma_i$  - the set that includes both areas, where  $r_i$  is the radius of the least circle that covers the obstacle, and  $R_i$  is the radius of the area where the repel force caused by APF acts.

The artificial potential function (APF) is given by the following equation [9]:

$$B_{oi}(l_i) = \begin{cases} \text{not defined} & \text{for } l_i < r_i \\ e^{\frac{l_i - r_i}{R_i - r_i}} & \text{for } r_i \leq l_i < R_i \\ 0 & \text{for } l_i \geq R_i \end{cases}, \quad (4)$$

where  $r_i > 0$ ,  $R_i > 0$  fulfill inequality  $R_i > r_i$ , and  $l_i$  is the distance to the obstacle.

The collision avoidance task requires the APF to attain to infinity as the Euclidean distance to the boundary of the colliding object decreases to zero. To fulfill this condition Eq. (4) is mapped to  $\langle 0, \infty \rangle$  using the following transformation:

$$V_{oi}(l_i) = \frac{B_{oi}(l_i)}{1 - B_{oi}(l_i)}. \quad (5)$$

In Fig. 2 an example of the APF for  $r = 1$  and  $R = 2$  is presented. The function is smooth in the whole range:  $l \in \langle r, \infty \rangle$ . Other APF's can be found in the literature [12, 10]. The former is non-smooth but applicable to the kinematic algorithm presented here, the latter is smooth but it is integral-based and requires many iterations to be computed precisely.

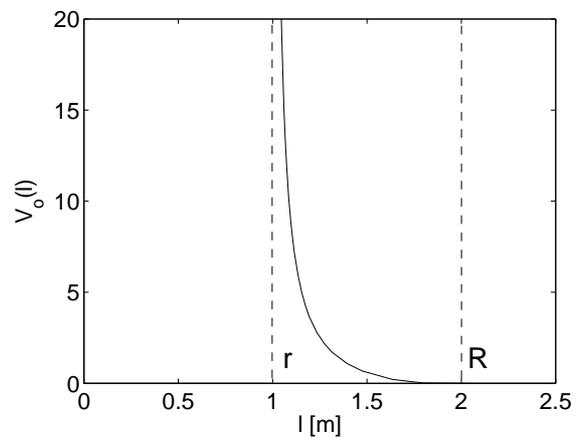


Fig. 2. Scaled APF:  $V_{oi}(l_i) \in \langle 0, \infty \rangle$

## 3. Control law formulation

**Definition 1. [Control problem]** Let us define the admissible reference trajectory  $\mathbf{q}_d(t) = [\theta_d(t) \ \mathbf{p}_d^T(t)]^T \in \mathcal{R}^3$ ,  $\mathbf{p}_d(t) = [x_d(t) \ y_d(t)]^T \in \mathcal{R}^2$  which fulfills kinematics (1) for some bounded reference input  $\mathbf{u}_d(t) = [u_{1d}(t) \ u_{2d}(t)]^T \in \mathcal{R}^2$  with the following persistent excitation condition:  $\forall t \geq 0 : u_{2d}(t) \neq 0$ . Assuming that:

- A1.  $\forall t \geq 0 \ \mathbf{p}_d(t) \notin D = \cup_i D_i$   
 A2.  $\mathbf{p}(t) \in \Gamma = \cup_i \Gamma_i \Rightarrow \mathbf{q}_d(t) = \mathbf{q}_d(t^-)$ ,  $\dot{\mathbf{q}}_d(t) = \dot{\mathbf{q}}_d(t) \equiv \mathbf{0}$   
 A3.  $D_i \cap D_j = \emptyset, \ i \neq j$

the aim is to find the bounded feedback control law  $\mathbf{u} = \mathbf{u}(\mathbf{q}_d, \mathbf{q}, \cdot)$  which guarantees that the tracking errors

$$e_\theta = f_\theta(\theta_d - \theta), \quad f_\theta : \mathbb{R} \mapsto \mathcal{S}^1 \quad (6)$$

$$\mathbf{e}(t) = \mathbf{p}_d(t) - \mathbf{p}(t) = \begin{bmatrix} e_x \\ e_y \end{bmatrix} \quad (7)$$

asymptotically converge to zero and  $\mathbf{p}(t) \notin \Delta = \cup_i \Delta_i$ .

Assumption A1 means that planned trajectory does not intersect APF regions. In the case the robot is out of the APF the task for the robot is pure trajectory tracking. As later shown that the robot's trajectory converges to the desired trajectory only out of the APF area the assumption A1 is necessary to

### Trajectory tracking control with obstacle avoidance capability for unicycle-like mobile robot

let the position error converge to zero. When the robot gets near the obstacle the tracking task is modified to avoid collision. As the robot approaches the boundary of the obstacle the tracking task become less important and convergence is more disturbed.

As collision avoidance is of a higher priority problem, because possible collision could lead to system damage, the trajectory tracking is temporarily suspended to facilitate bypassing the obstacle (assumption A2). After the robot leaves the collision avoidance region the desired trajectory is updated. This behavior causes that desired trajectory jumps from the point where it was suspended to the current desired position.

Assumption A3 means that obstacle's APFs are separated from each other. This feature guarantees that there are no local minima in the task space. Local minima can lead to a deadlock.

Now the trajectory tracking algorithm with collision avoidance will be presented. We propose to use the modified Vector Field Orientation method (VFO) [29]. It was expanded to avoid collisions by replacing tracking error from the original algorithm with modified tracking error that incorporates repel term.

We propose the following convergence vector:

$$\mathbf{h} = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} \triangleq \begin{bmatrix} k_1 e_a + \dot{\theta}_a \\ k_p E_x + \dot{x}_d \\ k_p E_y + \dot{y}_d \end{bmatrix}, \quad (8)$$

where  $k_1, k_p > 0$  are orientation and position control gains, respectively. Modified tracking error  $\mathbf{E}$  is computed as a difference between the tracking error and the sum of gradients of the APFs associated with the obstacles:

$$\mathbf{E} = \begin{bmatrix} E_x & E_y \end{bmatrix}^T \triangleq \mathbf{e} - \mathbf{w}, \quad (9)$$

where  $\mathbf{w}$  is defined as follows:

$$\mathbf{w} = \sum_{i=1}^M \left[ \frac{\partial V_{oi}(l_i)}{\partial \mathbf{p}} \right]^T. \quad (10)$$

In the above equation  $V_{oi}(l_i)$  is the APF of the  $i$ -th obstacle,  $l_i = \|\mathbf{p} - \mathbf{p}_i\|$  is the distance between the robot and the center of the  $i$ -th obstacle for  $i = 1, \dots, M$ ,  $M$  - the number of obstacles. Due to assumption A3 only one component of the sum in the right hand side of Eq. (10) is active at the moment. A graphical interpretation of the modified position error is shown in Fig. 3.

The auxiliary orientation error is defined as  $e_a = \theta_a - \theta$ . The auxiliary orientation variable  $\theta_a$  has the following form:

$$\theta_a = \text{Atan2c}(\sigma_d h_3, \sigma_d h_2) \quad (11)$$

where  $\sigma_d = \text{sgn}(u_{2d})$  denotes the signum function and  $\text{Atan2c}(\cdot, \cdot)$  is a continuous version of  $\text{Atan2}(\cdot, \cdot)$ . The definition of  $\text{Atan2c}(\cdot, \cdot)$  can be found in [29]. The term  $\sigma_d$  in (11) determines the strategy of the platform motion along the trajectory (forward or backward).

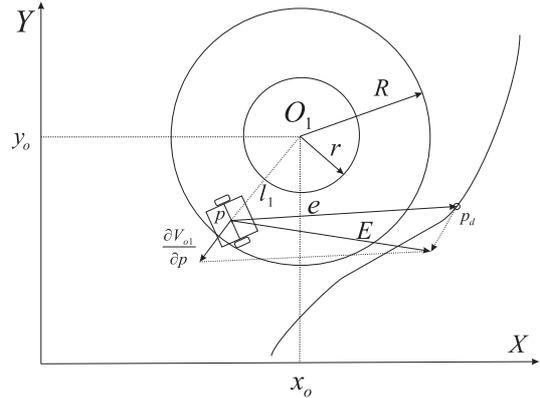


Fig. 3. Robot is in the repel area of obstacle  $O_1$ . Modified position error  $\mathbf{E}$  is computed as a difference between the tracking error and the gradient of the APF

The control vector  $\mathbf{u} = \begin{bmatrix} u_1 & u_2 \end{bmatrix}^T$  is given by the following equation:

$$\begin{aligned} u_1 &= h_1 \\ u_2 &= h_2 \cos \theta + h_3 \sin \theta \end{aligned} \quad (12)$$

#### 4. Stability and convergence analysis

In this section proofs of stability and convergence are presented. According to assumption A3 the indexes denoting the number of the obstacles are omitted in this section without loss of generality.

The proof consists of three steps: 1.  $\lim_{t \rightarrow \infty} (\theta(t) - \theta_a(t)) = 0$  - proof of convergence of the orientation to the auxiliary orientation variable, 2.  $\lim_{t \rightarrow \infty} (\mathbf{p}(t) - \mathbf{p}_d(t)) = \mathbf{0}$  - proof of stability and asymptotic convergence of the robot position to the reference position, 3.  $\lim_{t \rightarrow \infty} e_\theta = 0$  - proof of convergence of the auxiliary orientation variable to the desired orientation. Finally collision avoidance proof is shown.

Considering the first stem one has to substitute the first row of (8) into the first equation of (12) and using the first row of Eq. (1) we get:  $\dot{e}_a = -k_1 e_a$ , which guarantees that an auxiliary orientation error decreases exponentially to zero:

$$\lim_{t \rightarrow \infty} e_a = 0, \quad (13)$$

and the robot orientation converges exponentially to the auxiliary orientation.

In order to prove the second step the following Lyapunov function candidate is proposed:

$$V_l = \frac{1}{2} \mathbf{e}^T \mathbf{e} + V_o(l) = \frac{1}{2} (e_x^2 + e_y^2) + V_o(l), \quad (14)$$

where  $V_o$  is a potential given by (5).

Time derivative of (14) is as follows:

$$\frac{dV_l}{dt} = e_x \dot{e}_x + e_y \dot{e}_y + \frac{\partial V_o}{\partial x} \dot{x} + \frac{\partial V_o}{\partial y} \dot{y}. \quad (15)$$

Substitution of  $\theta = \theta_a - e_a$  into second row of Eq. (12) results in so called transformed pushing control:

$$u_2 = h_2 \cos(\theta_a - e_a) + h_3 \sin(\theta_a - e_a). \quad (16)$$

Using trigonometric identities in Eq. (16) gives:

$$u_2 = h_2(\cos(\theta_a) \cos(e_a) + \sin(\theta_a) \sin(e_a)) + h_3(\sin(\theta_a) \cos(e_a) - \cos(\theta_a) \sin(e_a)). \quad (17)$$

Taking into account that:

$$\sin(\theta_a) = \sigma_d \frac{h_3}{\|\mathbf{h}^*\|}, \quad \cos(\theta_a) = \sigma_d \frac{h_2}{\|\mathbf{h}^*\|}, \quad (18)$$

where  $\mathbf{h}^* = \begin{bmatrix} h_2 & h_3 \end{bmatrix}^T$ , equation (17) is simplified to:

$$u_2 = \sigma_d \|\mathbf{h}^*\| \cos(e_a). \quad (19)$$

Using above equation the second and the third row of the model (1) are transformed as follows:

$$\dot{x} = \sigma_d \|\mathbf{h}^*\| \cos(e_a) \cos(\theta) \quad (20)$$

$$\dot{y} = \sigma_d \|\mathbf{h}^*\| \cos(e_a) \sin(\theta). \quad (21)$$

Substituting  $\theta = \theta_a - e_a$  into above equations and using trigonometric identities yields:

$$\dot{x} = \sigma_d \|\mathbf{h}^*\| \cos(e_a) [\cos(\theta_a) \cos(e_a) + \sin(\theta_a) \sin(e_a)] \quad (22)$$

$$\dot{y} = \sigma_d \|\mathbf{h}^*\| \cos(e_a) [\sin(\theta_a) \cos(e_a) - \cos(\theta_a) \sin(e_a)]. \quad (23)$$

Substituting (18) into (22) and (23), taking into consideration that  $\sigma_d^2 = 1$  and simplifying results in:

$$\dot{x} = h_2 \cos^2(e_a) + h_3 \sin(e_a) \cos(e_a) \quad (24)$$

$$\dot{y} = -h_2 \sin(e_a) \cos(e_a) + h_3 \cos^2(e_a). \quad (25)$$

Time derivative of the Lyapunov function (15) can be calculated as follows:

$$\frac{dV_l}{dt} = e_x(\dot{x}_d - \dot{x}) + e_y(\dot{y}_d - \dot{y}) + \frac{\partial V_o}{\partial x} \dot{x} + \frac{\partial V_o}{\partial y} \dot{y}. \quad (26)$$

Taking into account that

$$e_x = E_x + \frac{\partial V_o}{\partial x} \quad \text{and} \quad e_y = E_y + \frac{\partial V_o}{\partial y}$$

one obtains:

$$\frac{dV_l}{dt} = e_x \dot{x}_d + e_y \dot{y}_d - E_x \dot{x} - E_y \dot{y}. \quad (27)$$

Substituting equations (24) and (25) into (27) and simplifying gives:

$$\begin{aligned} \frac{dV_l}{dt} &= e_x \dot{x}_d + e_y \dot{y}_d \\ &+ E_x(-h_2 \cos^2(e_a) - h_3 \sin(e_a) \cos(e_a)) \\ &+ E_y(h_2 \sin(e_a) \cos(e_a) - h_3 \cos^2(e_a)). \end{aligned} \quad (28)$$

The last equation can be transformed using the definition of convergence vector as follows:

$$\begin{aligned} \frac{dV_l}{dt} &= e_x \dot{x}_d + e_y \dot{y}_d - k_p E_x^2 \cos^2(e_a) \\ &- k_p E_y^2 \cos^2(e_a) - E_x \dot{x}_d \cos^2(e_a) - E_y \dot{y}_d \cos^2(e_a) \\ &- E_x \dot{y}_d \sin(e_a) \cos(e_a) + E_y \dot{x}_d \sin(e_a) \cos(e_a). \end{aligned} \quad (29)$$

Two cases are investigated separately:

C1.  $\mathbf{p} \in D^c$ ,  $D^c = \mathbb{R}^2 \setminus D$ ,

C2.  $\mathbf{p} \in \Gamma$ .

In case C1  $E_x = e_x$  and  $E_y = e_y$ . The time derivative of the Lyapunov function is as follows:

$$\begin{aligned} \frac{dV_l}{dt} &= e_x \dot{x}_d + e_y \dot{y}_d - k_p e_x^2 \cos^2(e_a) - k_p e_y^2 \cos^2(e_a) \\ &\quad - e_x \dot{x}_d \cos^2(e_a) - e_y \dot{y}_d \cos^2(e_a) \\ &\quad - e_x \dot{y}_d \sin(e_a) \cos(e_a) + e_y \dot{x}_d \sin(e_a) \cos(e_a). \end{aligned} \quad (30)$$

Using identity  $\sin^2(\cdot) = 1 - \cos^2(\cdot)$  and substituting  $e_x \dot{y}_d - e_y \dot{x}_d = \|\mathbf{e} \times \dot{\mathbf{q}}_d\|$  into above equation one can write:

$$\begin{aligned} \frac{dV_l}{dt} &= e_x \dot{x}_d \sin^2(e_a) + e_y \dot{y}_d \sin^2(e_a) - k_p e_x^2 \cos^2(e_a) \\ &\quad - k_p e_y^2 \cos^2(e_a) - \|\mathbf{e} \times \dot{\mathbf{q}}_d\| \sin(e_a) \cos(e_a). \end{aligned} \quad (31)$$

Taking into account that  $\|\mathbf{e} \times \dot{\mathbf{q}}_d\| = \|\mathbf{e}\| \|\dot{\mathbf{q}}_d\| \sin \alpha$ , where  $\alpha = \angle(\mathbf{e}, \dot{\mathbf{q}}_d)$  and introducing matrix  $\mathbf{Q} = \text{diag}\{k_p \cos^2(e_a), k_p \cos^2(e_a)\}$  time derivative of the Lyapunov function fulfills the following inequality:

$$\begin{aligned} \frac{dV_l}{dt} &\leq -\mathbf{e}^T \mathbf{Q} \mathbf{e} + \|\mathbf{e}\| \|\dot{\mathbf{p}}_d\| [\sin^2(e_a) \\ &\quad - \sin \alpha \sin(e_a) \cos(e_a)] = -W. \end{aligned} \quad (32)$$

Using (32) one can easily show that function  $W$  is positive definite if:

$$\|\mathbf{e}\| > \frac{1}{\lambda_{\min}(\mathbf{Q})} [\|\dot{\mathbf{p}}_d\| \sin e_a (\sin e_a - \sin \alpha \cos e_a)], \quad (33)$$

where  $\lambda_{\min}(\mathbf{Q})$  is the smallest eigenvalue of matrix  $\mathbf{Q}$ . Taking into account that  $e_a$  converges exponentially to zero according to Eq. (13) one can conclude that

$$\lim_{t \rightarrow \infty} e(t) = 0. \quad (34)$$

In case C2 the time derivative of the Lyapunov function is as follows:

$$\frac{dV_l}{dt} = -k_p E_x^2 \cos^2(e_a) - k_p E_y^2 \cos^2(e_a). \quad (35)$$

It always fulfills the condition  $\dot{V}_l \leq 0$  and the system is stable. To show the convergence of  $e_\theta$ , in the third step note that for  $e \rightarrow 0$  the auxiliary orientation variable (11) takes the form  $\theta_a = \text{Atan2c}(\sigma_d \dot{y}_d, \sigma_d \dot{x}_d)$ , that is equivalent to the reference orientation generated according to model (1). The above result together with (13) leads to:

$$\lim_{e \rightarrow 0} (\theta_a(e) - \theta_d) = 0 \stackrel{(13,34)}{\Rightarrow} \lim_{t \rightarrow \infty} e_\theta(t) = 0 \pm 2k\pi \quad (36)$$

where  $k = 0, 1, \dots$

In [17] methodology for designing control laws that guarantees collision avoidance in multiagent systems is presented. These concepts can be applied also to a single robot control system in the environment with static obstacles. Taking into account that  $V_l$  is nonincreasing in  $\Gamma$  as shown above (case C2) and designed APF fulfills condition  $\lim_{l \rightarrow r^+} V_o(l) = +\infty$  utilizing results from [17] we conclude that the robot

### Trajectory tracking control with obstacle avoidance capability for unicycle-like mobile robot

will never enter the set  $\Delta$ . Thus, collision avoidance is guaranteed.

**Remark 4.1.** When the robot reaches a saddle point, the reference trajectory is disturbed to drive the robot out of a local equilibrium point.

A saddle point is non-attracting, however, robot can get stuck in it. The disturbance on the reference trajectory is necessary to push it way. This technique was discussed in detail in [7]. As shown in [26] there is one saddle point associated with each circle-shaped obstacle (Fig. 4).

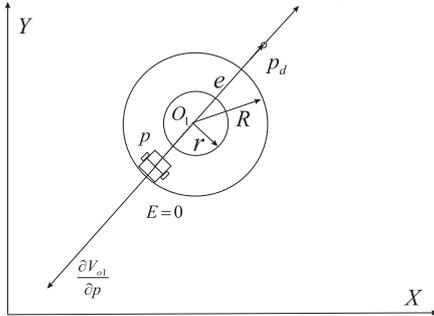


Fig. 4. Saddle point near the obstacle

**Remark 4.2.** In some configurations the following condition may occur:  $\|\mathbf{h}^*\| = 0$ . In this case  $\theta_a(t)$  cannot be computed by Eq. (11) and the previous value of the auxiliary orientation variable is taken:  $\theta_a(t) = \theta_a(t^-)$ .

The situation described in Remark 4.2 occurs at a saddle point. Equality  $\|\mathbf{h}^*\| = 0$ , may also occur outside  $\Gamma_i$ , however, this state is non-attracting and the same strategy as shown in Remark 4.2 is applicable.

## 5. Experimental results

Before the control values are applied to the motion controllers they are transformed to the wheel velocities:

$$\omega_R = \frac{u_2 + \frac{1}{2}bu_1}{r_w}, \quad \omega_L = \frac{u_2 - \frac{1}{2}bu_1}{r_w}, \quad (37)$$

where  $r_w$  is the radius of the robot wheels and  $b$  is the distance between robot wheels.

Practical realization of the control algorithm requires control input consideration. In the case of a differentially-driven wheeled vehicle the kinematic limitation is imposed on the maximum angular velocity  $\omega_{w \max} > 0$  which can be realized by the vehicle wheel. In order to take into account this limit the following control scaling procedure is proposed. Denoting by  $\boldsymbol{\omega} = [\omega_R \ \omega_L]^T$  the computed and unlimited control input vector from (37), the scaled and physically realizable input  $\boldsymbol{\omega}_d = [\omega_{Rd} \ \omega_{Ld}]^T$  can be obtained as follows:

$$\boldsymbol{\omega}_d(\tau) = \frac{\boldsymbol{\omega}(\tau)}{s(\tau)}, \quad (38)$$

where

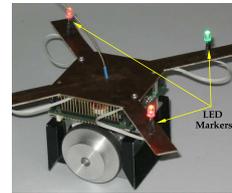
$$s(\tau) = \max \left\{ 1, \frac{|\omega_R(\tau)|}{\omega_{w \max}}, \frac{|\omega_L(\tau)|}{\omega_{w \max}} \right\} \geq 1. \quad (39)$$

The above scaling procedure keeps the direction of the rescaled (limited) control vector  $\boldsymbol{\omega}_d$  equal to the previously computed control vector  $\boldsymbol{\omega}$  preserving in turn the instantaneous vehicle motion curvature  $\kappa = \boldsymbol{\omega}/v$ , where  $v$  is linear velocity of the platform.

As Eq. (10) produces infinitely large values for  $l = r$  and the procedure given by (38) is used to make control feasible, the condition  $R - r \gg 0$  must be fulfilled to avoid collisions.

**5.1. Experimental set-up.** Experimental tests were carried out using the MTV3 differentially-driven mobile robot presented in Fig. 5. Three LED markers were mounted on the vehicle top for use with the vision localization system employed in the control scheme as illustrated in Fig. 5. All the control blocks of the proposed algorithm were implemented on an external PC computer (platform control level). The desired angular wheel velocities  $\boldsymbol{\omega}_d$  computed on the platform control level were transmitted through the radio link to the PI regulatory loops working on the vehicle board (drive control level). The feedback rate of the platform control level with vision feedback was equal to 45 Hz; the drive control level worked at the rate of 512 Hz. In the theoretical analysis the time delay of the control loop was not taken into account. By the use of faster control and transmission hardware the disturbances caused by the delay can be reduced. The upper bound of the delay is 22ms in case there are no frame losses (99% frames was transmitted correctly in practice).

a)



b)

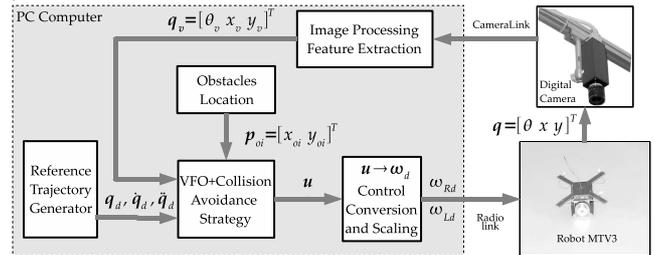


Fig. 5. Mobile robot MTV3 used in experiments (a) and a block schema of the control system (b)

**5.2. Experiment 1 – collision avoidance with a single obstacle.** In Figs. 6–10 experimental results for robot in an environment with a single obstacle are shown. The values of the control parameters are as follows:  $k_1 = 2$ ,  $k_p = 1$ ,  $\omega_{w \max} = 10$  rad/s. An obstacle with radius  $r = 0.1$  m is placed at position  $P_{O1} = (0.48 \text{ m}, -0.3 \text{ m})$  and the radius of the APF is  $R = 0.3 \text{ m}$ . The desired trajectory is a Lissajous curve with the center at the origin. In Fig. 7 time graphs of the position error components, and in Fig. 8 orientation error

and auxiliary orientation error are presented. The discontinuity at  $t \cong 5s$  is caused by updating the desired trajectory when the robot leaves the APF area of the obstacle (when desired trajectory is unfrozen desired position 'jumps' to perform imposed task). In Fig. 9 time evolution of controls are shown. Fig. 6 presents the desired and real path of the robot. In this figure it also can be observed that after the robot leaves the APF, the desired point on the trajectory jumps some distance. In Fig. 10 time evolution of collision avoidance term  $\|w\|$  is shown. It is increased as the robot approaches the obstacle. The dashed line represents a logical variable that indicates whether the robot is in  $\Gamma$ : it is 0 when the robot is in the area of the obstacle's APF and 1 otherwise. This value indicates also when desired trajectory is frozen (0) or not (1).

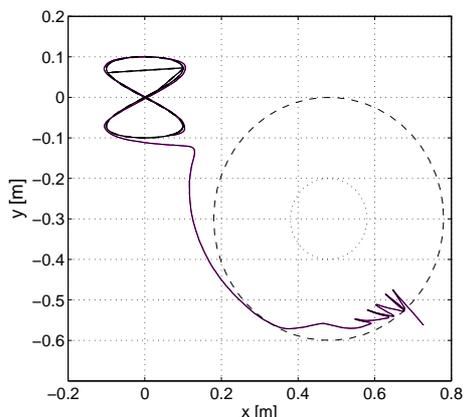


Fig. 6. Experiment 1: trajectory of the robot and desired trajectory on the  $(x, y)$  plane

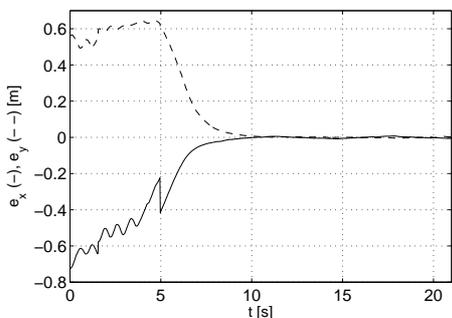


Fig. 7. Experiment 1: position errors as a function of time

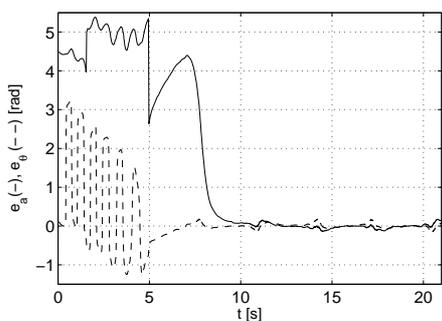


Fig. 8. Experiment 1: angle errors as a function of time

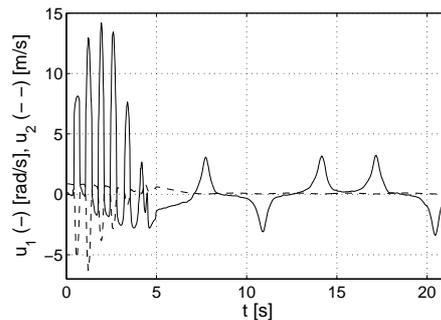


Fig. 9. Experiment 1: longitudinal and angular controls

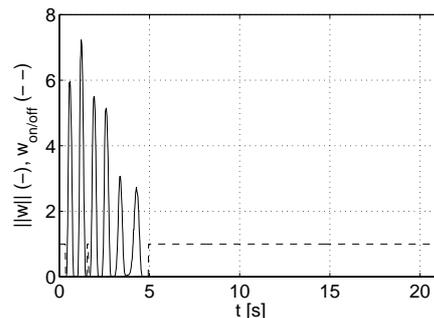


Fig. 10. Experiment 1: module of the collision avoidance component of the control ( $\|w\|$ ) and logical value that indicates collision avoidance is on (value 0) off (value 1)

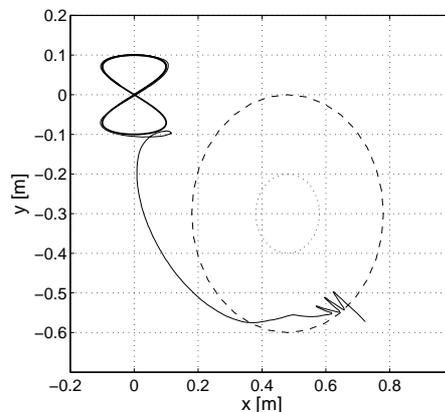


Fig. 11. Experiment 2: trajectory of the robot and desired trajectory on the  $(x, y)$  plane

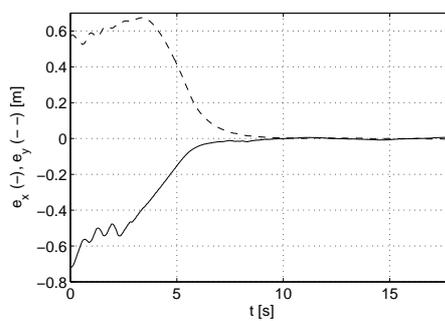


Fig. 12. Experiment 2: position errors as a function of time

*Trajectory tracking control with obstacle avoidance capability for unicycle-like mobile robot*

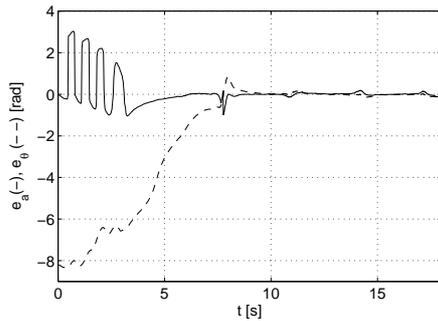


Fig. 13. Experiment 2: angle errors as a function of time

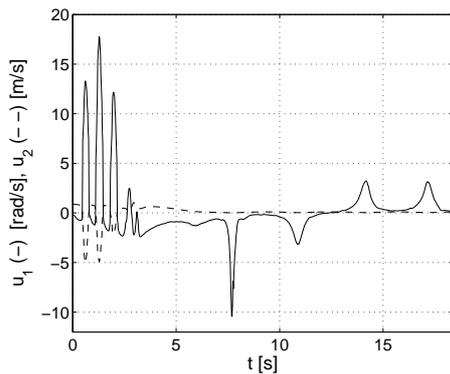


Fig. 14. Experiment 2: longitudinal and angular controls

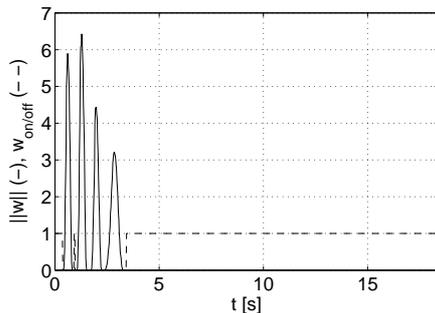


Fig. 15. Experiment 2: module of the collision avoidance component of the control ( $\|w\|$ ) and logical value that indicates collision avoidance is on (value 0) off (value 1)

To make the experiment more difficult the initial position of the robot was chosen near the saddle point (Fig. 4). The chattering observed in the transient state is caused by the delays in the control loop. Oscillations of this kind were not observed in simulations. The most time consuming operations are: acquisition of data from the vision system, computation of the robot's position and sending the controls to the low level motion controllers through the serial radio link.

**5.3. Experiment 2 – collision avoidance with a single obstacle without freezing the desired trajectory.** Experiment 2 was conducted in the same environment as in case of experiment 1. The control parameters are also the same, however, assumption A1 from definition 1 is removed. There is no stability proof for this case but the presented experiments show

that the system works well. In Fig. 12 the position errors and in Fig. 13 orientation errors are shown. Compared to the previous experiment fewer oscillations are observed when robot gets into the APF area. The graphs of control signals are similar (Fig. 14), but robot leaves the set  $\Gamma$  of the obstacle 2 seconds earlier. In Fig. 15 time evolution of the module of collision avoidance component of the control  $\|w\|$  is shown. The value of collision avoidance component of the control reaches similar values to these shown in previous case, however, the number of enterings into the repel area is reduced significantly. Figure 11 presents the desired and real paths of the robot.

**5.4. Experiment 3 – collision avoidance with two obstacles.**

In Figs. 16–20 the results for a more complex environment are presented. The desired trajectory is frozen as in experiment 1. In Fig. 17 the time graphs of position errors and in Fig. 18 orientation errors are shown. A discontinuity caused by switching on the desired trajectory can be easily observed at  $t \cong 4s$ . The robot gets to the APF area of both obstacles. Fig. 16 presents the position on the  $(x, y)$  plane. Control signals reach higher values (Fig. 19) as compared to these in experiments 1 and 2. Collision avoidance  $\|w\|$  of control is presented in Fig. 20.

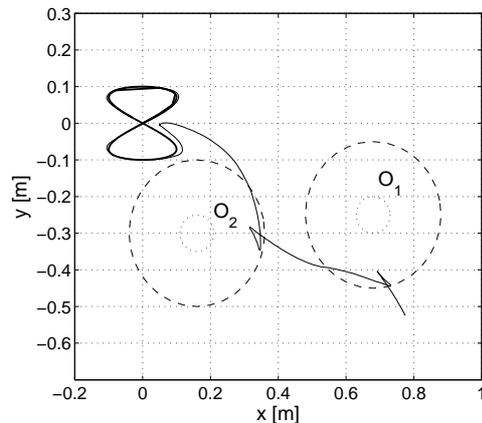


Fig. 16. Experiment 3: trajectory of the robot and desired trajectory on the  $(x, y)$  plane

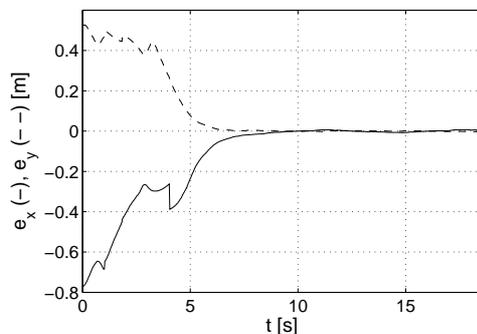


Fig. 17. Experiment 3: position errors as a function of time

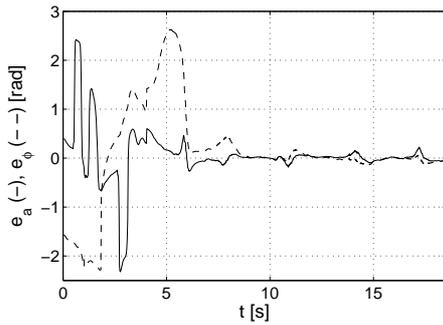


Fig. 18. Experiment 3: angle errors as a function of time

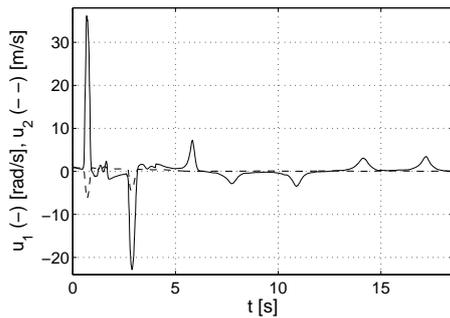


Fig. 19. Experiment 3: longitudinal and angular controls

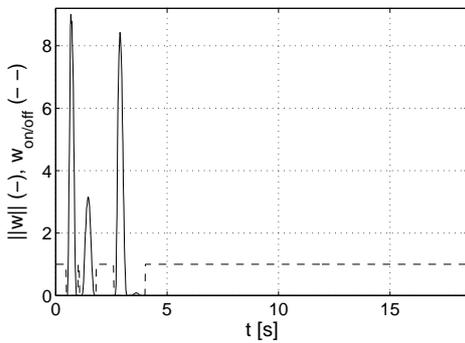


Fig. 20. Experiment 3: module of the collision avoidance component of the control ( $\|w\|$ ) and logical value that indicates collision avoidance is on (value 0) off (value 1)

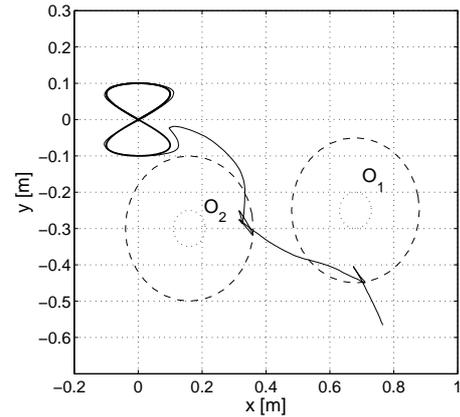


Fig. 21. Experiment 4: trajectory of the robot and desired trajectory on the  $(x, y)$  plane

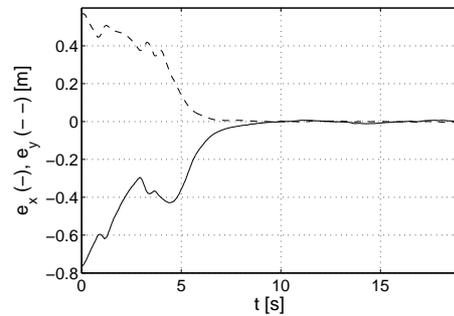


Fig. 22. Experiment 4: graph of  $e_x, e_y$  as a function of time

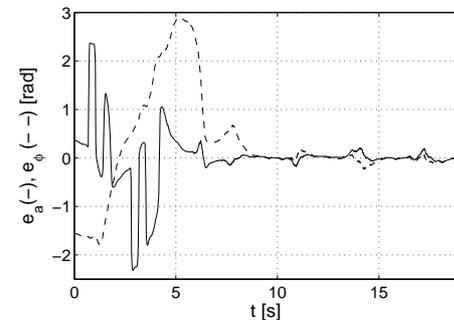


Fig. 23. Experiment 4: angle errors as a function of time

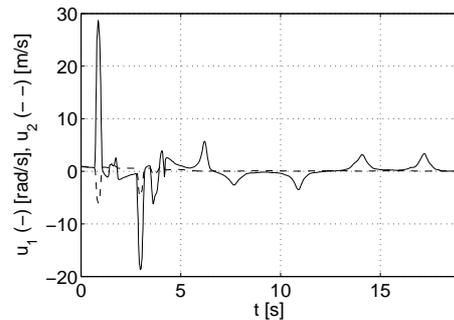


Fig. 24. Experiment 4: longitudinal and angular controls

**5.5. Experiment 4 – collision avoidance with two obstacles without freezing the desired trajectory.** In Figs. 21–25 the results for an environment with two obstacles are shown. The desired trajectory is not frozen when robot gets into the APF. In this case a second collision caused multiple changes of the motion direction (Fig. 21). In this case position error (Fig. 22) and orientation error (Fig. 23) graphs are similar to these presented in the previous case. Also controls (Fig. 24) are similar, however leaving the repel area of the obstacle takes about 0.5s longer as compared to the previous case. This can be observed in the time graph of repulsive control component  $\|w\|$  (Fig. 25).

### Trajectory tracking control with obstacle avoidance capability for unicycle-like mobile robot

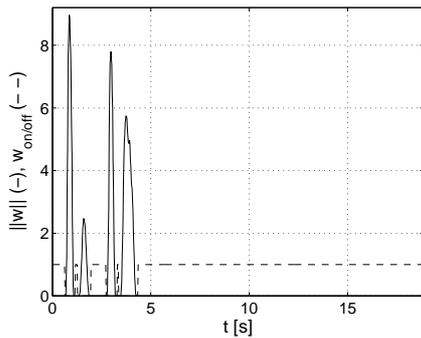


Fig. 25. Experiment 4: module of the collision avoidance component of the control ( $\|w\|$ ) and logical value that indicates collision avoidance is on (value 0) off (value 1)

**5.6. Experiment 5 – collision avoidance with two obstacles – high value of  $\omega_{w \max}$ .** In this case it is presented how a value of  $\omega_{w \max}$  parameter influences task performance. In the experiment limit of 40 rad/s was set on robots wheels. This parameter can be used to tune the control algorithm according to practical abilities of the robot. In the present case the robot's path approaches closely the boundary of the obstacle (Fig. 26). It is caused by delays that exist in real systems due to communication between sensors, main control module, low-level motor controllers and inertia of the actuators. These delays play more important role as velocity of the robot increases because the platform passes longer distance during control-loop time delay. This delay causes that robot

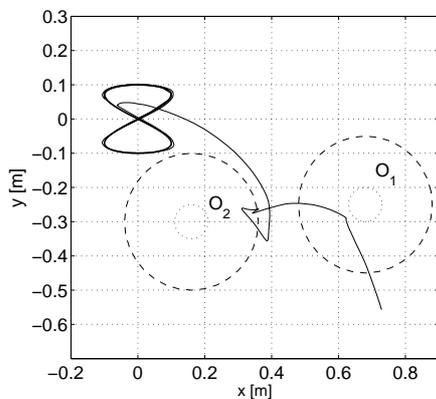


Fig. 26. Experiment 5: trajectory of the robot and desired trajectory on the  $(x, y)$  plane

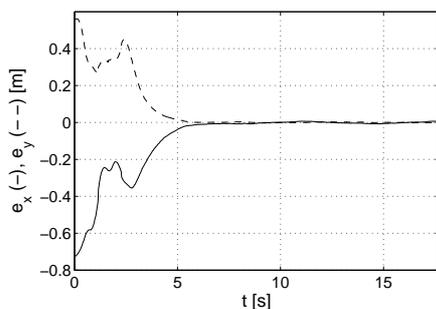


Fig. 27. Experiment 5: graph of  $e_x, e_y$  as a function of time

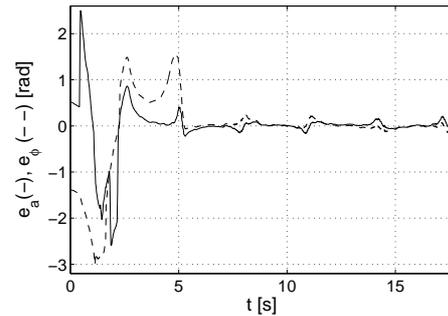


Fig. 28. Experiment 5: angle errors as a function of time

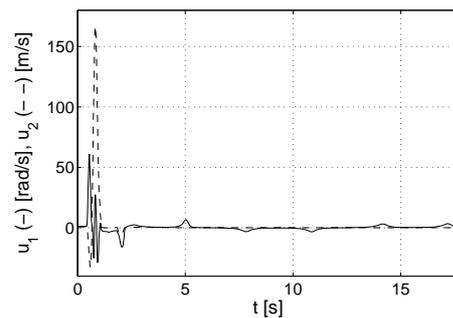


Fig. 29. Experiment 5: longitudinal and angular controls

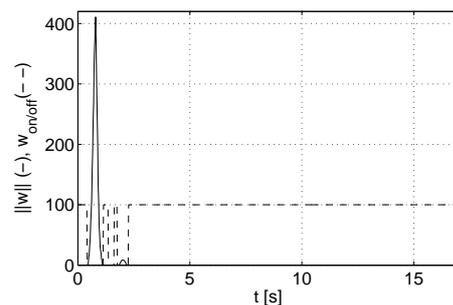


Fig. 30. Experiment 5: module of the collision avoidance component of the control ( $\|w\|$ ) and logical value that indicates collision avoidance is on (value 0) off (value 100)

approaches the obstacle closer. The value of the repulsion component of the control is significantly larger in comparison to earlier experiments. The controls (Fig. 29) and  $\|w\|$  (Fig. 30) reach high values. In Fig. 27 the time graphs of position errors and in Fig. 28 orientation errors are shown.

## 6. Conclusions

In this paper the VFO control method expanded with collision avoidance module was presented. Collision avoidance behavior was implemented utilizing local artificial potential functions. The main advantages of this approach are low computational power required to perform the task in comparison to global solutions, like the navigation function approach [26], [27] and possibility of implementation of the robot equipped only with on-board sensors (with limited sensing range). On the other hand, the proposed solution can be applied only in

the environment where obstacles can be covered with separated circle-shaped areas. This can be seen as significant weakness in some applications. The proof of stability and convergence of the algorithm are given. The effectiveness of the method was illustrated by a number of experiments. In the near future results for the proposed approach applied to many robots will be published.

**Acknowledgements.** This work was supported by the Polish Ministry of Science and Higher Education grant No. N N514 406236.

REFERENCES

[1] I. Kolmanovsky and N.H. McClamroch, “Developments in nonholonomic control problems”, *IEEE Control Systems Magazine* 16 (6), 20–36 (1995).

[2] P. Morin and C. Samson, “Trajectory tracking for non-holonomic vehicles: overview and case study”, *4th Int. Workshop on Robot Motion and Control* 1, 139–153 (2004).

[3] P. Morin and C. Samson, “Motion control of wheeled mobile robots”, *Springer Handbook of Robotics* 1, 799–826 (2008).

[4] J. Minguez, F. Lamiroux, and J.P. Laumond, “Motion planning and obstacle avoidance”, *Springer Handbook of Robotics* 1, 827–852 (2008).

[5] G. Leitmann, “Guaranteed avoidance strategies”, *J. Optim. Theory and Applications* 32 (4), 569–576 (1980).

[6] D.M. Stipanovic, “A survey and some new results in avoidance control”, *15th Int. Workshop on Dynamics and Control IWDC* 1, 166–173 (2009).

[7] T. Urakubo, K. Okuma, and Y. Tada, “Feedback control of a two wheeled mobile robot with obstacle avoidance using potential functions”, *Proc. 2004 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* 1, 2428–2433 (2004).

[8] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L.E. Kavraki, and S. Thurn, *Principles of Robot Motion. Theory, Algorithms, and Implementations*, The MIT Press, London, 2005.

[9] W. Kowalczyk, “Control algorithms for formation of mobile robots”, *PhD Thesis*, Poznan University of Technology, Poznań, 2008.

[10] K.D. Do, “Formation tracking control of unicycle-type mobile robots with limited sensing range”, *IEEE Trans. on Control Systems Technology* 16 (3), 527–538 (2008).

[11] W. Kowalczyk, K. Kozłowski, and J. Tar, “Trajectory tracking for formation of mobile robots”, *Robot Motion and Control*, vol. 396, pp. 57–66, Springer, Berlin, 2009.

[12] S. Mastellone, D.M. Stipanovic, C.R. Graunke, K.A. Intlekofer, and M.W. Spong, “Formation control and collision avoidance for multi-agent nonholonomic systems: Theory and experiments”, *Int. J. Robotics Research* 27 (1), 107–126 (2008).

[13] C. De La Cruz and R. Carelli, “Dynamic model based formation control and obstacle avoidance of multi-robot systems”, *Robotica* 26, 345–356 (2008).

[14] D.E. Chang, S.C. Shadden, J.E. Marsden, and R. Olfati-Saber, “Collision avoidance for multiple agent systems”, *42nd IEEE Conf. on Decision and Control* 1, 539–543 (2003).

[15] I.I. Hussein and D.M. Stipanovic, “Effective coverage control for mobile sensor networks with guaranteed collision avoidance”, *IEEE Trans. on Control Systems Technology* 15 (4), 642–657 (2007).

[16] K. Kozłowski and W. Kowalczyk, “Motion control for formation of mobile robots in environment with obstacles”, in *Towards Intelligent Engineering and Information Technology*, vol. 243, pp. 203–219, Springer, Berlin, 2009.

[17] D.M. Stipanovic, P.F. Hokayem, M.W. Spong, and D.D. Siljak, “Avoidance control for multiagent systems”, *ASME J. Dynamic Systems, Measurement and Control* 129, 699–707 (2007).

[18] E.J. Rodriguez-Seda, D.M. Stipanovic, and M.W. Spong, “Collision avoidance control with sensing uncertainties”, *2011 American Control Conf.* 1, 3363–3368 (2011).

[19] G. Klancar, D. Matko, and S. Blazic, “Wheeled mobile robots control in a linear platoon”, *J. Intelligent and Robotic Systems* 54, 709–731 (2009).

[20] D.V. Dimarogonas, S.G. Loizou, K.J. Kyriakopoulou, and M.M. Zavlanos, “A feedback stabilization and collision avoidance scheme for multiple independent non-point agents”, *Automatica* 42, 229–243 (2006).

[21] D.V. Dimarogonas and E. Frazzoli, “Analysis of decentralized potential field based multi-agent navigation via primal-dual lyapunov theory”, *49th IEEE Conf. on Decision and Control* 1, 1215–1220 (2010).

[22] A. Chakravarthy and D. Ghose, “Obstacle avoidance in a dynamic environment: a collision cone approach”, *IEEE Trans. on Systems, Man, And Cybernetics – Part A: Systems and Humans* 28 (5), 562–574 (1998).

[23] E. Lalish, K.A. Morgansen, and T. Tsukamaki, “Decentralized reactive collision avoidance for multiple unicycle-type vehicles”, *American Control Conf.* 1, 5055–5061 (2008).

[24] R.M. Murray S. Waydo, “Vehicle motion planning using stream functions”, *Proc. IEEE International Conf. on Robotics and Automation* 1, 2484–2491 (2003).

[25] P. Szulczynski, D. Pazderski, and K. Kozłowski, “Obstacle avoidance and trajectory tracking using fluid-based approach in 2d space”, in *Robot Motion and Control*, pp. 15–17, Springer, Berlin, 2012.

[26] E. Rimon and D. Koditschek, “Exact robot navigation using artificial potential function”, *IEEE Trans. on Robotics and Automation* 8 (5), 501–518 (1992).

[27] E. Rimon and D. Koditschek, “The construction of analytic diffeomorphisms for exact robot navigation of star worlds”, *Trans. American Mathematical Society* 327 (1), 71–116 (1991).

[28] G. Campion, G. Bastin, and B. d’Andrea Novel, “Structural properties and classification of kinematic and dynamic models of wheeled mobile robots”, *IEEE Trans. on Robotics and Automation* 12, 47–62 (1996).

[29] M. Michałek and K. Kozłowski, “Vector-field-orientation feedback control method for a differentially driven vehicle”, *IEEE Trans. on Control Systems Technology* 18 (1), 45–65 (2010).

[30] W. Kowalczyk, M. Michałek, and K. Kozłowski, “Trajectory tracking control and obstacle avoidance for a differentially driven mobile robot”, *Proc. 18th IFAC World Congress* 1, 1058–1063 (2011).