# Path Following for Two HOG Wheels Mobile Robot

*Paweł Joniak, Robert Muszyński*

**Abstract:**

*To apply a spinning hemisphere as a mobile robot drive is an unconventional idea. Equipping a mobile robot with two such hemispheres brings to life a device with absolutely novel properties. In this paper we derive kinematics models of a mobile robot with two driving hemispheres, analyse shortly their properties, and adopt a control algorithm designed to follow a path. There are two kinematics models presented: the full model of the original system, and the model of the simplified system, equivalent to the original one. The second model is expressed in two different coordinate systems – the later allowing for the application of known control algorithms to drive the robot. The performance of the analysed algorithm is illustrated by computer simulations.*

**Keywords:** *Hemispherical Omnidirectional Gimbaled wheel, hemispheres, mobile robot, path following*

## 1. Introduction

In the late thirties of the last century there arose a concept of an entirely new system of an automobile drive resulting in a concept car, called "Hemisphere Drive Speedster" [10, 11]. This is a car which is equipped with a Hemispherical Omnidirectional Gimbaled wheel (HOG wheel) in a form of a spinning hemisphere. So far this idea has not been applied in road vehicles but a few implementations of small mobile robots equipped with such a drive system [1, 8] appeared there. These constructions with a layout of a kinematic car having attached a single spinning hemisphere exhibit quite amazing dynamical properties – constantly spinning, usually rather heavy hemisphere constitutes a perfect reservoir of kinetic energy ready to accelerate the vehicle. Moreover, a simple change of the hemisphere tilt redirects the robot course, and the actual motion speed can be changed by altering the tilt angle while keeping the constant hemisphere spinning speed.

To go further, two spinning hemispheres can be applied in a robot drive system [2, 4]. Here, the capabilities of changing the robot direction and its orientation seem to be unlimited, however to efficiently utilise this potential is a kind of art. To control the robot without slip or with a controlled slip one has to apply adequate mathematical models of the robot motion. To obtain such models one needs to consider the robot as a mobile platform being the subject to a set of velocity constraints.

In this paper we describe the structure of a mobile robot equipped with two HOG wheels. Assuming that the robot moves on a horizontal plane, we derive models of its kinematics, analyse their properties, and propose a control algorithm. The paper is composed in the following way. Basic facts on nonholonomic robots modelling are outlined in the remaining part of this section. Section 2 provides the robot kinematics models, while Section 3 explains the robot control algorithm. Results of numeric computations illustrating the robot behaviour are shown in Section 4. Section 5 concludes the paper.

We will derive models of nonholonomic robots using generalised coordinates and velocities $q(t)$, $\dot{q}(t) \in \mathbb{R}^n$, and assume that the robot motion is subject to a number $l < n$ of phase constraints. These constraints are expressed in the Pfaffian form

$$A(q)\dot{q} = 0, \qquad (1)$$

where $A(q) \in \mathbb{R}^{l \times n}$ is a full rank constraint matrix. For wheeled mobile robots the constraints reflect the motion without lateral (nonslip condition) and longitudinal (pure rolling condition) slip of the robot.

The kinematics of the robot obeying the Pfaffian constraints (1) can be described by a driftless control system

$$\dot{q} = G(q)\eta = \sum_{i=1}^{m} g_i(q)\eta_i, \qquad (2)$$

where $q \in \mathbb{R}^n$, $\eta \in \mathbb{R}^m$, $m = n - l$, denote the state variable and the control vector (auxiliary velocities) respectively, and the matrix $G(q)$ consists of vector fields $g_i(q)$ spanning the kernel of $A(q)$

$$A(q)g_i(q) = 0, \quad i = 1, \ldots, m. \qquad (3)$$

## 2. Robot Kinematics Models

The core element of the analysed construction consists of two spinning hemispheres of radius $R$, that drive the robot (see Figure 1). The hemispheres are placed at the distance of $2l$. Each hemisphere can rotate independently around 3 axes: it can be tilted about the axis parallel to the motion plane and perpendicular to the axis connecting the hemispheres centres[1], it can be tilted about the axis lying on its great circle and parallel to the axis connecting the hemisphere centres, and finally, it can spin around its axis of symmetry. Structurally, in real constructions complying with the described model, the angles describing the hemisphere tilts are limited, while spinning is unlimited. The models we shall derive do not respect these above limitations.
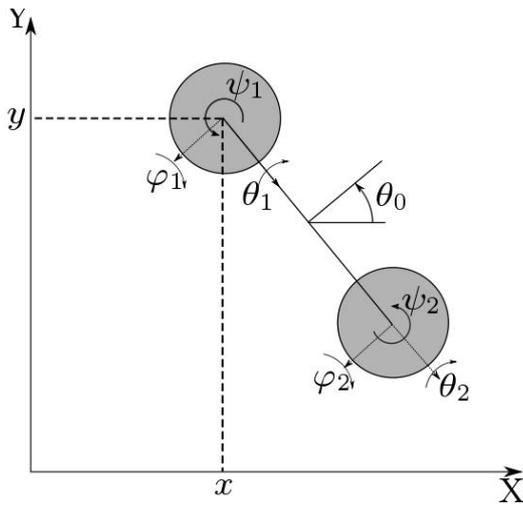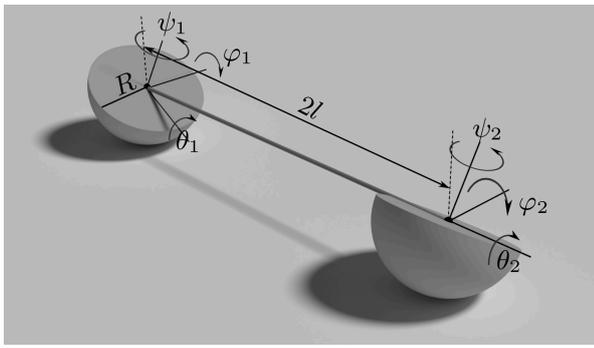
**Fig. 1. Model of the robot with two HOG wheels and its configuration coordinates**

In this section we compute two kinematics models of the robot: the first fully reflecting the robot real behaviour, and the second, simplified, in which the spinning hemispheres will be replaced by relevant variable radius wheels.

### 2.1. Full Kinematics Model

We will assume that the robot moves on a horizontal plane equipped with a global, Cartesian coordinate frame. The robot configuration will be described as shown in Figure 1 by the configuration vector

$$q = (x, y, \theta_0, \varphi_1, \theta_1, \psi_1, \varphi_2, \theta_2, \psi_2)^T \in \mathbb{R}^9, \quad (4)$$

where $x$, $y$, $\theta_0$, describe the robot body position and orientation, and $\varphi_i, \theta_i, \psi_i, i = 1, 2$, are two tilt and one spin angles of each of the hemispheres, respectively. These angles become zero when hemispheres great circles are parallel to the motion plane.

To derive a kinematics model it is convenient to introduce the mounting points coordinate frames, which are fixed to the robot body and originated at the hemispheres centres. Their Z axes are perpendicular to the motion plane, and the X axes are parallel to this plane and perpendicular to the axis connecting the hemispheres centres. The transformations from the global coordinate system to these frames are given as

$$A_G^{M_1} = Trans(X, x)Trans(Y, y)Trans(Z, R)$$
$$Rot(Z, \theta_0), \quad (5)$$

$$A_G^{M_2} = Trans(X, x)Trans(Y, y)Trans(Z, R)$$
$$Rot(Z, \theta_0)Trans(Y, -2l). \quad (6)$$

The contact points $P_{K_i}$, $i = 1, 2$ of the hemispheres with the ground expressed in the corresponding mounting points frames take the form

$$P_{K_i}^{M_i} = \begin{pmatrix} 0 \\ 0 \\ -R \end{pmatrix}, \quad (7)$$

and the hemispheres movement is characterised by a set of following transformations

$$A_{M_i}^{H_i} = Rot(X, \varphi_i)Rot(Y, \theta_i)Rot(Z, \psi_i). \quad (8)$$

**Nonholonomic Constraints**   To derive a robot kinematics model we assume, that the robot moves with no lateral and longitudinal slip at the contact points $P_{K_i}$, $i = 1, 2$ of the hemispheres with the ground. It is straightforward, that every change of the hemispheres configuration displaces these points on the hemispheres. To avoid slip the velocity of these displacements must be equal to the linear velocity of the hemispheres centres, which we will call the mounting points $P_{M_i}$, $i = 1, 2$.

To obtain the no slip conditions, one has to compute the velocities of the hemispheres contact points (relative to the global coordinates) expressed in their mounting points local coordinate frames. They take the form

$$\dot{P}_{K_i}^{M_i} = \begin{pmatrix} R(\dot{\theta}_i \cos \varphi_i - \dot{\psi}_i \cos \theta_i \sin \varphi_i) \\ -R(\dot{\varphi}_i + \dot{\psi}_i \sin \theta_i) \\ 0 \end{pmatrix}, \quad i = 1, 2.$$
$$(9)$$

The velocities of the hemispheres mounting points, expressed as above in the mounting points frames, are given by

$$\dot{P}_{M_1}^{M_1} = \begin{pmatrix} \dot{x} \cos \theta_0 + \dot{y} \sin \theta_0 \\ \dot{y} \cos \theta_0 - \dot{x} \sin \theta_0 \\ 0 \end{pmatrix}, \quad (10)$$

$$\dot{P}_{M_2}^{M_2} = \begin{pmatrix} \dot{x} \cos \theta_0 + \dot{y} \sin \theta_0 + 2l\dot{\theta}_0 \\ \dot{y} \cos \theta_0 - \dot{x} \sin \theta_0 \\ 0 \end{pmatrix}, \quad (11)$$

respectively. A comparison of the velocities (9) with (10) and (9) with (11) gives the equations of nonholonomic constraints resulting from the no slip assumption, which can be written in the Pfaffian form (1) as

$$
\begin{bmatrix}
-c_{\theta_0} & -s_{\theta_0} & 0 & 0 & Rc_{\varphi_1} & -Rc_{\theta_1}s_{\varphi_1} \\
s_{\theta_0} & -c_{\theta_0} & 0 & -R & 0 & -Rs_{\theta_1} \\
-c_{\theta_0} & -s_{\theta_0} & -2l & 0 & 0 & 0 \\
s_{\theta_0} & -c_{\theta_0} & 0 & 0 & 0 & 0
\end{bmatrix}
$$

$$
\begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & Rc_{\varphi_2} & -Rc_{\theta_2}s_{\varphi_2} \\
-R & 0 & -Rs_{\theta_2}
\end{bmatrix}
\begin{pmatrix}
\dot{x} \\ \dot{y} \\ \dot{\theta}_0 \\ \dot{\varphi}_1 \\ \dot{\theta}_1 \\ \dot{\psi}_1 \\ \dot{\varphi}_2 \\ \dot{\theta}_2 \\ \dot{\psi}_2
\end{pmatrix}
= 0,
$$

(12)

with standard abbreviations $s_\alpha = \sin\alpha$ and $c_\alpha = \cos\alpha$.

**Kinematics Model** While deriving the kinematics model of the two HOG wheel mobile robot, which takes a form of a driftless control system (2), one has to choose control vector components. Since the dimension of the robot generalised coordinates vector $\dim q(t) = n = 9$, and the rank of the Pfaff matrix in (12) rank $A(q) = l = 4$, the control vector contains $m = n - l = 5$ components. Naturally, the best choice is to use controls feasible in the real robot. However, in typical constructions there are 6 such controls, directly influencing $\dot{\varphi}_1, \dot{\theta}_1, \dot{\psi}_1,$ and $\dot{\varphi}_2, \dot{\theta}_2, \dot{\psi}_2$ velocities. Thus, one has to select a velocity, which will not be controlled directly. In view of the robot construction symmetry, theoretically 3 velocity choices are possible [5]. Below, we abandon the direct control of one of the spinning velocities, choosing as control vector components $\dot{\varphi}_1, \dot{\theta}_1, \dot{\psi}_1, \dot{\varphi}_2, \dot{\theta}_2$. We will directly control all of the tilts velocities and one spinning velocity – the other spinning velocity will be computed automatically so as to avoid slip.

An analysis of the equation (3) for the constraints (12) yields a robot kinematics model in the form of a driftless control system (2) as follows

$$
\begin{cases}
\dot{x} = Rs_{\theta_0}\eta_1 + Rc_{\theta_0}c_{\varphi_1}\eta_2 + R(s_{\theta_0}s_{\theta_1} - c_{\theta_0}c_{\theta_1}s_{\varphi_1})\eta_3 \\
\dot{y} = -Rc_{\theta_0}\eta_1 + Rs_{\theta_0}c_{\varphi_1}\eta_2 - R(c_{\theta_0}s_{\theta_1} + s_{\theta_0}c_{\theta_1}s_{\varphi_1})\eta_3 \\
\dot{\theta}_0 = \frac{R}{2l}(-\cot\theta_2 s_{\varphi_2}\eta_1 - c_{\varphi_1}\eta_2 + c_{\theta_1}s_{\varphi_1}\eta_3 - \\
\qquad\quad - \cot\theta_2 s_{\theta_1}s_{\varphi_2}\eta_3 + \cot\theta_2 s_{\varphi_2}\eta_4 + c_{\varphi_2}\eta_5) \\
\dot{\varphi}_1 = \eta_1 \\
\dot{\theta}_1 = \eta_2 \\
\dot{\psi}_1 = \eta_3 \\
\dot{\varphi}_2 = \eta_4 \\
\dot{\theta}_2 = \eta_5 \\
\dot{\psi}_2 = \frac{1}{s_{\theta_2}}(\eta_1 + \sin\theta_1\eta_3 - \eta_4)
\end{cases}.
$$

(13)

It should be noticed that this model is not well defined for $s_{\theta_2} = 0$, thus one should avoid controls driving the robot through these configurations.

### 2.2. Simplified Kinematics Model

It is a straightforward observation that the spinning hemisphere behaves like a rotating, steering wheel[2] with a variable radius[3], which we will refer
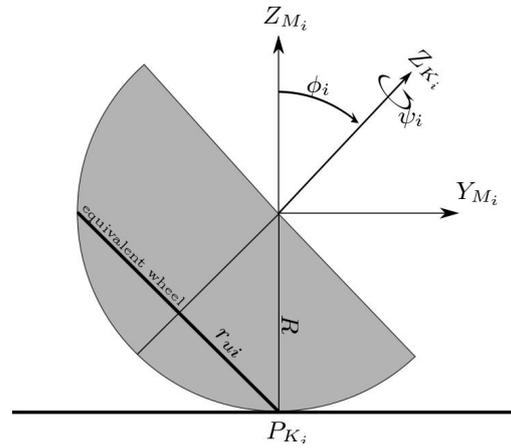


**Fig. 2. Equivalent wheel model**
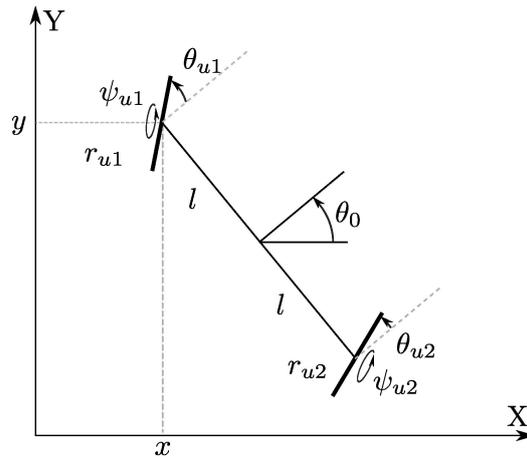


**Fig. 3. Simplified robot model – configuration coordinates**

to as an equivalent wheel (see Figure 2). Such an observation allows one to consider the robot with two HOG wheels as a class (1,2) robot [3], equipped with two steering, variable radius wheels, each described by a spinning angle $\psi_{ui}$, a steering angle $\theta_{ui}$, and an actual radius $r_{ui}$, $i = 1, 2$, although such simplified model does not reflect the robot body movement caused by the hemispheres tilting – the remaining model properties are maintained. Similarly to the full kinematics model case, the configuration vector will be defined as (see Figure 3)

$$
q = (x, y, \theta_0, \theta_{u1}, \psi_{u1}, \theta_{u2}, \psi_{u2}, r_{u1}, r_{u2})^T, \quad (14)
$$

where $x, y$ – the robot body position, $\theta_0$ – its orientation, $\theta_{u1}, \theta_{u2}$ – the wheel rotation angles, $\psi_{u1}, \psi_{u2}$ – the wheels spin angles, $r_{u1}, r_{u2}$ – the wheel radii.

**Equivalent Wheel** To derive a kinematics model of the robot equipped with two equivalent wheels one has to determine transformations between the coordinates describing a hemisphere and the coordinates describing an equivalent wheel. One can see that for fixed values of the hemisphere tilt angles $(\varphi_i, \theta_i)$ the spinning of the hemisphere will result in the motion of the contact point along one of its parallels. This parallel defines a circle which we call the equivalent wheel

of radius $r_{ui}$. According to this definition the radius $r_{ui}$, and the angles of the wheel rotation $\theta_{ui}$ and spin $\psi_{ui}$ can be computed as

$$\begin{cases} r_{ui} = R\sqrt{\sin^2 \varphi_i + \cos^2 \varphi_i \sin^2 \theta_i} \\ \theta_{ui} = \arctan\left(\frac{\sin \theta_i}{\cos \theta_i \sin \varphi_i}\right) \\ \psi_{ui} = \psi_i \end{cases}, \qquad (15)$$

where $R$ stands for the hemisphere radius, and $\varphi_i$, $\theta_i$ and $\psi_i$ describe its configuration. The inverse transformation to (15) takes the form

$$\begin{cases} \varphi_i = \pm \arccos\left(\sqrt{\frac{R^2 - r_{ui}^2}{R^2 - r_{ui}^2 \sin^2 \theta_{ui}}}\right) \\ \theta_i = \pm \arcsin\left(\frac{r_{ui}}{R} |\sin \theta_{ui}|\right) \\ \psi_i = \psi_{ui} \end{cases}. \qquad (16)$$

The sign in (16) is specified by $\theta_{ui}$ angle (plus in the first and fourth quadrants, minus otherwise). Using (16) one is able to transfer solutions of path following problem obtained for the simplified kinematics model (20) to the original system moving on hemispheres (13).

**Nonholonomic Constraints** Again, we assume that the robot moves without lateral and longitudinal slips. For the configuration vector (14), the constraints reflecting no slip of the wheel 1 are given as

$$\begin{cases} \dot{x} \sin(\theta_0 + \theta_{u1}) - \dot{y} \cos(\theta_0 + \theta_{u1}) = 0 \\ \dot{x} \cos(\theta_0 + \theta_{u1}) + \dot{y} \sin(\theta_0 + \theta_{u1}) - r_{u1}\dot{\psi}_{u1} = 0 \end{cases}. \qquad (17)$$

Similarly, for the wheel 2 one gets

$$\begin{cases} \dot{x} \sin(\theta_0 + \theta_{u2}) - \dot{y} \cos(\theta_0 + \theta_{u2}) + \\ \qquad\qquad + 2l \sin(\theta_{u2})\dot{\theta}_0 = 0 \\ \dot{x} \cos(\theta_0 + \theta_{u2}) + \dot{y} \sin(\theta_0 + \theta_{u2}) + \\ \qquad\qquad + 2l \cos(\theta_{u2})\dot{\theta}_0 - r_{u2}\dot{\psi}_{u2} = 0 \end{cases}. \qquad (18)$$

These constraints can be given the Pfaffian form (1)

$$\begin{bmatrix} s_{0u1} & -c_{0u1} & 0 & 0 & 0 & 0 \\ c_{0u1} & s_{0u1} & 0 & 0 & -r_{u1} & 0 \\ s_{0u2} & -c_{0u2} & 2ls_{u2} & 0 & 0 & 0 \\ c_{0u2} & s_{0u2} & 2lc_{u2} & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{matrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -r_{u2} & 0 & 0 \end{matrix} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta}_0 \\ \dot{\theta}_{u1} \\ \dot{\psi}_{u1} \\ \dot{\theta}_{u2} \\ \dot{\psi}_{u2} \\ \dot{r}_{u1} \\ \dot{r}_{u2} \end{pmatrix} = 0, \qquad (19)$$

where $s_{u2} = \sin \theta_{u2}$, $c_{u2} = \cos \theta_{u2}$, $s_{0w} = \sin(\theta_0 + \theta_w)$, $c_{0w} = \cos(\theta_0 + \theta_w)$, $w \in \{u1, u2\}$.

**Kinematics Model** As in the case of full kinematics model, the dimension of the robot generalised coordinates vector $\dim q(t) = n = 9$, the rank of the Pfaff

matrix rank $A(q) = l = 4$, and thus the control vector contains $m = n - l = 5$ components. Again, as previously, the deficiency of one control input in relation to the number of controls feasible in the real robot ($\dot{\theta}_{u1}$, $\dot{\theta}_{u2}$, $\dot{\psi}_{u1}$, $\dot{\psi}_{u2}$, $\dot{r}_{u1}$, $\dot{r}_{u2}$) can be observed. To include the radii derivatives $\dot{r}_{u1}$, $\dot{r}_{u2}$ in the control vector is an obvious choice. The analysis of the remaining coordinates relationship described by (3) leads to a conclusion, that in this case the only possible solution is to control directly two rotation angles $\theta_{u1}, \theta_{u2}$ and one spinning $\psi_{u1}$, while the other spinning angle $\psi_{u2}$ will be computed automatically. Consequently, for constraints (19) the equation (3) allows one to determine the matrix $G(q)$ defining the control system (2) in the form

$$\begin{cases} \dot{x} = \cos(\theta_0 + \theta_{u1})r_{u1}\eta_2 \\ \dot{y} = \sin(\theta_0 + \theta_{u1})r_{u1}\eta_2 \\ \dot{\theta}_0 = \frac{\sin(\theta_{u1} - \theta_{u2})}{\sin \theta_{u2}} \frac{r_{u1}}{2l} \eta_2 \\ \dot{\theta}_{u1} = \eta_1 \\ \dot{\psi}_{u1} = \eta_2 \\ \dot{\theta}_{u2} = \eta_3 \\ \dot{\psi}_{u2} = \frac{\sin \theta_{u1}}{\sin \theta_{u2}} \frac{r_{u1}}{r_{u2}} \eta_2 \\ \dot{r}_{u1} = \eta_4 \\ \dot{r}_{u2} = \eta_5 \end{cases}. \qquad (20)$$

As in (13), the model is not well defined for $s_{\theta_2} = 0$, so we assume that controls driving the robot through these configurations are restrained.

Now, having the system (20), instead of solving the path following problem for the original system (13), one can solve the problem for this simplified system, and transform the obtained solution via (16) to the original system solution. This procedure, called the control transfer, is described in Section 3.

It is worth noticing that, according to the relations defined in the model (20), a change of the robot body linear speed can be caused by both, a change of the wheels spinning speed, as well as a change of the wheels radii – the body linear speed is a product of the spinning speed and the radius. In consequence, identical robot trajectories can be achieved by the spinning speed change at a constant wheel radius, and by the wheel radius variation at a constant spinning speed. While the first case seems to be more intuitive, typically the second should be applied, since the equivalent wheel spinning speed transforms via (16) directly to the HOG wheel spinning speed. It is recommended to keep it constant – robot accelerating and decelerating should result from the radius changes, which explicitly reflects the HOG wheel tilt.

## 3. Control Algorithm

To control the two HOG wheels mobile robot we shall adopt the method designed for controlling a two steering wheels robot [7]. In this method rather than writing the system equations with respect to a fixed reference frame, the robot state is parametrised to the followed path, in terms of distance and orientation (see Figure 4 – position of the point $M$ is described with coordinates $(y_c, s_c)^T$). With $\theta_c$ describing the angle of the line tangent to the path at point $P(s_c)$ and
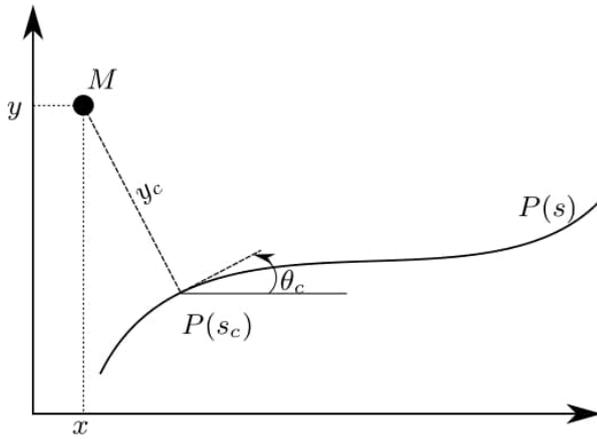
**Fig. 4. Point position description**

$c_c(s)$ denoting the path curvature, we have

$$\dot{\theta}_c(s_c(t)) = c_c(s_c(t))\dot{s}_c(t). \qquad (21)$$

The derivative of the curvature with respect to $s$ is given as $g_c(s)$, so

$$\dot{c}_c(s(t)) = g_c(s(t))\dot{s}(t). \qquad (22)$$

Conditions for which such a representation is well defined without ambiguity and limitations resulting from its local nature are given in [9].

Now, the first three equations of the simplified model (20), describing the state of the robot body, can be written as

$$\begin{cases} \dot{s}_c = \frac{c_{\theta_0-\theta_c+\theta_{u1}}}{1-c_c y_c}\dot{\psi}_{u1} \\ \dot{y}_c = s_{\theta_0-\theta_c+\theta_{u1}}\dot{\psi}_{u1} \\ \dot{\theta}_0 = \sigma\dot{\psi}_{u1} \end{cases} \qquad (23)$$

where $\sigma = \frac{1}{2l}(\sin\theta_{u1} - \tan\theta_{u2}\cos\theta_{u1})$ is the inverse of the distance to the robot instant centre of rotation. According to [7], we assume a constant speed of the first robot wheel $\dot{\psi}_{u1} = v$, and take into account no slip conditions. With these we can complement the equations (23) to obtain the complete control system

$$\begin{cases} \dot{s}_c = v\frac{c_{\theta_0-\theta_c+\theta_{u1}}}{1-c_c y_c} \\ \dot{y}_c = vs_{\theta_0-\theta_c+\theta_{u1}} \\ \dot{\theta}_0 = v\sigma \\ \dot{\theta}_{u1} = \eta \\ \dot{\theta}_{u2d} = \frac{\dot{\theta}_{u1}(1-2l\sigma\sin\theta_{u1})-2l\dot{\sigma}\cos\theta_{u1}}{(2l\sigma-\sin\theta u1)^2+\cos^2\theta_{u1}} \\ \dot{\theta}_{u2} = \frac{\dot{\theta}_{u1}(1-2l\sigma\sin\theta_{u1})-2l\dot{\sigma}\cos\theta_{u1}}{(2l\sigma-\sin\theta u1)^2+\cos^2\theta_{u1}} - k_{\theta_{u2}}(\theta_{u2}-\theta_{u2d}) \\ \dot{\psi}_{u1} = v \\ \dot{\psi}_{u2} = v\sqrt{(2l\sigma-\sin\theta_{u1})^2+\cos^2\theta_{u1}} \end{cases} , \quad (24)$$

where $\eta$ and $\dot{\sigma}$ play the role of control inputs, and the configuration vector contains $(s_c, y_c, \theta_0, \theta_{u1}, \theta_{u2d}, \theta_{u2}, \psi_{u1}, \psi_{u2})^T$ with $\theta_{u2d}$ being an auxiliary variable; $k_{\theta_{u2}}$ is a non-negative gain. Since the algorithm proposed in [7] does not allow the varying radii of wheels we do not include them into the robot configuration vector, and the model.

To control the system one has to determine a control function, such that the distance $y_c$ goes to zero, and

the angle $\theta = \theta_0 - \theta_c$, being the difference between the robot body orientation and the path tangent angle, approaches a desired value $\theta_d$. For the robot with two steering wheels this task can by solved with the control algorithm proposed in [7]. This method utilises the feedback linearisation, which for the system (24) can be written as[4]

$$\begin{cases} \dot{s}_c = v\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c} \\ \dot{y}_c = vs_{\theta+\theta_{u1}} \\ \dot{\theta} = v(\sigma - c_c\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c}) \\ \dot{\theta}_{u1} = v\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c}[y_c\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c}(g_c s_{\theta+\theta_{u1}}-k_{p_y})+ \\ \qquad + s_{\theta+\theta_{u1}}(c_c s_{\theta+\theta_{u1}}- \\ \qquad - k_{v_y}c_{\theta+\theta_{u1}}sign(\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c}))+c_c]-v\sigma \\ \dot{\theta}_{u2d} = \frac{\dot{\theta}_{u1}(1-2l\sigma\sin\theta_{u1})-2l\dot{\sigma}\cos\theta_{u1}}{(2l\sigma-\sin\theta u1)^2+\cos^2\theta_{u1}} \\ \dot{\theta}_{u2} = \frac{\dot{\theta}_{u1}(1-2l\sigma\sin\theta_{u1})-2l\dot{\sigma}\cos\theta_{u1}}{(2l\sigma-\sin\theta u1)^2+\cos^2\theta_{u1}} - k_{\theta_{u2}}(\theta_{u2}-\theta_{u2d}), \\ \dot{\psi}_{u1} = v \\ \dot{\psi}_{u2} = v\sqrt{(2l\sigma-\sin\theta_{u1})^2+\cos^2\theta_{u1}} \\ \dot{\sigma} = v\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c}\left\{\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c}\left[-k_{p_\theta}\tilde{\theta}+g_c\right]+ \\ \qquad +\sigma\left\{\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c}y_c\left(g_c c_{\theta+\theta_{u1}}+k_{p_y}s_{\theta+\theta_{u1}}\right)+ \\ \qquad +s_{\theta+\theta_{u1}}\left[c_c c_{\theta+\theta_{u1}}+k_{v_y}s_{\theta+\theta_{u1}}sign\left(\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c}\right)\right]\right\}- \\ \qquad -k_{v_\theta}\left[\sigma-c_c\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c}\right]sign\left(\frac{c_{\theta+\theta_{u1}}}{1-c_c y_c}\right)\right\} \end{cases}$$
$$(25)$$

where $c_{\theta+\theta_{u1}} = \cos(\theta+\theta_{u1})$, $s_{\theta+\theta_{u1}} = \sin(\theta+\theta_{u1})$, $\tilde{\theta} = \theta - \theta_d$ – the orientation error, $k_{\theta_{u2}}, k_{p_y}, k_{v_y}, k_{p_\theta}, k_{v_\theta}$ – non-negative gains, and $v$ a constant, desired spinning velocity of the first wheel. It is shown in [7] that with the algorithm (25) the distance $y_c$ and the orientation error $\tilde{\theta}$ asymptotically converge to zero.

To determine the control function for the original model (13), the solution obtained with (25) has to be transformed to the one for (13). To this end, after applying control algorithm (25) one obtains the motion trajectory of the simplified model (20). Now, with the relationship (16) one transforms this trajectory to the trajectory of the original system (13). Finally, derivatives computation of suitable components of the original system trajectory gives the controls for the original model. Such a procedure will be referred to as "offline" mode control transfer. Admittedly, this procedure is simple, but unfortunately it does not allow to correct the control values in a closed-loop form while applying them to a real system.

## 4. Computer Simulations

To illustrate the performance of the proposed control method we shall provide an application example. As an example desired path $(x_d(s), y_d(s))^T$ we have chosen a Lissajous curve

$$\begin{cases} x_d(s) = \sin\frac{s}{2} \\ y_d(s) = \cos\left(s+\frac{\pi}{4}\right) \end{cases} , \qquad (26)$$

which forms a lemniscate line. In simulations we have assumed $l = 0.1$, and $R = 0.03$.

The algorithm (25) ensures the convergence of the error $(y_c, \tilde{\theta})$ to zero, which causes the simplified model to follow the path. Here, to illustrate the algorithm

**Fig. 5. Control algorithm results**
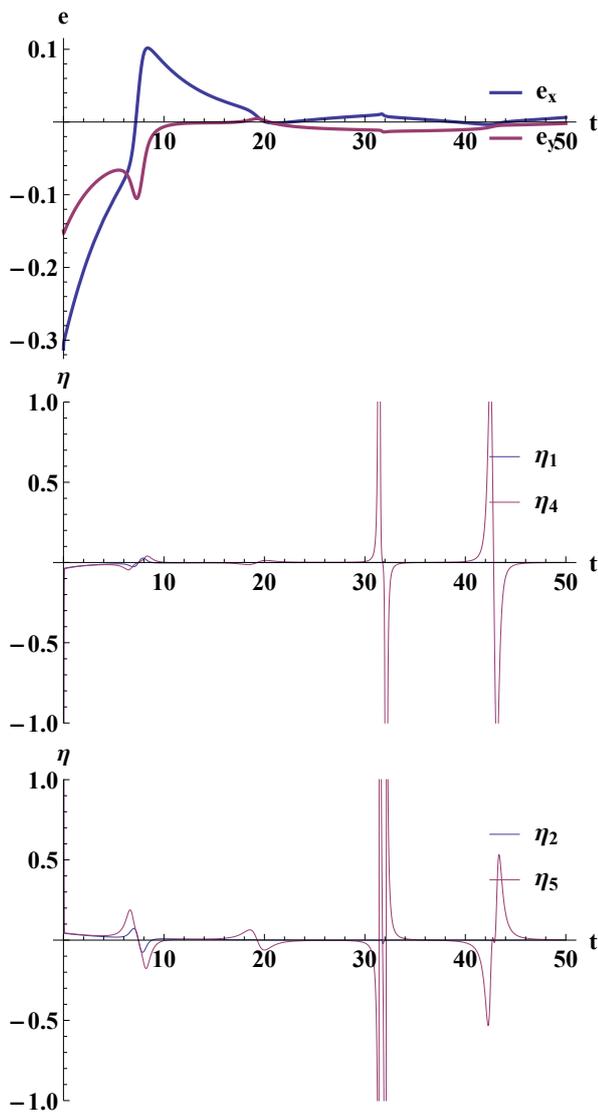


**Fig. 6. Control algorithm results (continued)**

performance for the full model (including the controls transfer procedure), as a path tracking error the difference between the actual robot position and desired path will be shown ($e_x = x - x_d$, $e_y = y - y_d$). All the simulations were performed with use of Mathematica system [6].

For the controller parameters we have chosen $k_{\theta_{u2}} = 1$, $k_{p_y} = k_{v_y} = k_{p_\theta} = k_{v_\theta} = 500$, the desired angle $\theta_d = \frac{\pi}{2} - 0.4$, the desired velocity $v = 0.2$, and the initial conditions $q(0) = (x(0), y(0), \theta_0(0), \theta_{u1}(0), \phi_{u1}(0), \theta_{u2}(0), \phi_{u2}(0)) = (0, 0, -\frac{\pi}{2}, \frac{\pi}{2} - 0.4, 0, \frac{\pi}{2} - 0.4, 0)^T$. Simulating the system (20)-(25), and transferring the controls in the "offline" mode we obtain the results shown in Fiugres 5 and 6. In these figures the system desired path is drawn in black, while the robot real position $(x, y)^T$ in red. As one could expect, the path tracking errors converge to a vicinity of zero, however it never reaches it. Such behaviour occurs since the simplified model does not reflect the properties of the original robot model utterly, causing differences in resultant trajectories, and the control transfer is performed "offline", consequently preventing any

"online" corrections to these differences. Increasing the control gains makes the errors convergence faster, nevertheless they still do not become zero. To

deal with this problem one should think of designing an "online" methodology for the controls transfer. Though, from the practical point of view the obtained steady state errors for many applications stay at an acceptable level.

## 5. Conclusion

In this paper two kinematics models of the mobile robot equipped with two HOG wheels are derived, under the assumption that the robot moves without slip on a horizontal plane. The first model fully reflects the robot kinematic properties, while the second exploits the HOG wheels robot similarity to a robot with two steering, variable radius wheels. The relations between HOG wheels coordinates and the equivalent wheels coordinates are studied. The models similarity prompts an adaptation of the class (1,2) mobile robot control algorithm to the considered case.

The performance of the control algorithm has been tested in computer simulations. The results agree with expectations. Additionally, both kinematics models behave as it was anticipated, demonstrating easily predicted differences. What is important the differences do not prevent on adaptation of the usual wheeled robot control algorithm. Nonetheless, the additional analysis is desirable. First of all, the possibility of "online" controls transfer should be investigated. Further, the applicability of alternative control algorithms should be examined.

## ACKNOWLEDGEMENTS

## Notes

[1]By the hemisphere centre we will understand the centre of its great circle.

[2]To unify the terminology hereinafter the wheel rotating will be called the spinning, like in the case of the hemisphere.

[3]In this simplified case we neglect the tilting of a wheel defined this way.

[4]Please notice, there are some minor differences between the model (20) and this introduced in [7] – in [7] the wheels radii are constant, and initial values of the robot body and its wheels orientations are set differently.

## AUTHORS

**Paweł Joniak** – Faculty of Mathematics and Computer Science, University of Wrocław, ul. Fryderyka Joliot-Curie, 51-137 Wrocław, Poland, e-mail: p.m.joniak@gmail.com.

**Robert Muszyński**[*] – Department of Cybernetics and Robotics, Wrocław University of Science and Technology, ul. Janiszewskiego 11/17, 50-372 Wrocław, Poland, e-mail: robert.muszynski@pwr.edu.pl, www: kcir.pwr.edu.pl.

[*]Corresponding author

## REFERENCES

[1] E. Ackerman. "You've never seen a robot drive system like this before". IEEE Spectrum, http://spectrum.ieee.org/automaton/robotics/diy/youve-never-seen-a-drive-system-like-this-before, Jul 2011.

[2] T. Center. "Hemispherical gimbaled wheel drive system (HOG wheel)". http://www.techcenter.in/2011/07/hemispherical-gimbaled-wheel-drive.html.

[3] C. C. de Wit, G. Bastin, and B. Siciliano, eds., *Theory of Robot Control*, Springer-Verlag New York, Inc.: Secaucus, NJ, USA, 1996.

[4] P. Joniak. "Analysis of two HOG wheel mobile robot behaviour". Bachelor's thesis, Wroclaw University of Science and Technology, 2015. (in Polish).

[5] P. Joniak and R. Muszyński. "Behaviour of two HOG wheels mobile robot". In: K. Tchoń and C. Zieliński, eds., *Progress in Robotics*, volume I, 103–114. Oficyna Wydawnicza Politechniki Warszawskiej, 2016. (in Polish).

[6] "Wolfram Mathematica". http://www.wolfram.com/mathematica/.

[7] A. Micaelli and C. Samson. "Trajectory tracking for unicycle-type and two-steering-wheels mobile robots". Technical Report 2097, Institut National De Recherche En Informatique Et En Automatique, 1993.

[8] M. Rybczyński. "Model of mobile robot with HOG wheel". Bachelor's thesis, Wroclaw University of Science and Technology, 2011. (in Polish).

[9] C. Samson, "Path following and time-varying feedback stabilization of a wheeled mobile robot". In: *International Conference on Advanced Robotics and Computer Vision*, vol. 13, 1992, 1.1–1.5.

[10] Unknown, "Hemisphere drive speedster", *Mechanics and Handicraft*, vol. 5, no. 9, 1938, 23,73, Provided by Modern Mechanix, http://blog.modernmechanix.com/hemisphere-drive-speedster/.

[11] Wikipedia. "Hog wheel". http://en.wikipedia.org/wiki/Hemispherical_omnidirectional_gimbaled_wheel.